

제11주: 그래프

11.1. R에서의 그래프

- R은 다양한 그래픽 기능을 제공합니다. 우선 감을 잡기 위하여, `demo(graphics)` 또는 `demo(persp)`를 타입해 보시기 바랍니다.
 > `demo(graphics)`
 > `demo(persp)`
- 그래픽 함수(graphical function)들의 수행 결과는 어떤 객체에 저장되는 것이 아니라 *그래픽 디바이스*(graphical device)로 보내집니다. 그래픽 디바이스는 그래프가 출력되는 윈도우나 파일 등을 말합니다.
- 그래픽 함수는 두 종류로 나뉩니다.
 - 고수준 그래픽 함수(high-level plotting function):
 새로운 그래프를 생성하는 함수
 - 저수준 그래픽 함수(low-level plotting function):
 기존 그래프에 요소들을 추가하는 함수
 - 인터랙티브 그래픽 함수(interactive graphics functions):
 마우스 같은 장치를 이용하여 기존 그래프에 인터랙티브하게 정보를 추가하거나 추출할 수 있음
- 그래픽 파라미터(graphical parameters)는 그래프를 그리는 세부적인 방식을 결정합니다. 이것들은 함수 `par()`를 통해 살펴보거나 수정될 수 있습니다.

11.2. 그래픽 디바이스

그래픽 함수를 수행한 후 만일 그래픽 디바이스가 열려있지 않다면 R은 그래픽 디바이스를 열고 그래프를 표시합니다. 그 이외에, 사용자가 직접 그래픽 디바이스를 열 수도 있습니다.

① 현재 열려있는 그래픽 디바이스의 목록 확인하기

`dev.list()`함수는 현재 열려있는 그래픽 디바이스 목록을 보여줍니다.

```
> dev.list()
windows
  2
```

② 새로운 그래픽 디바이스 열기

- 유닉스나 리눅스에서는 `X11()` 함수를 이용하고, 윈도우에서는 `windows()` 함수를 이용하면 그래픽 윈도우를 열 수 있습니다.
- 그래픽 디바이스는 파일이 될 수도 있습니다. `pdf()`, `png()`, `postscript()`와 같은 함수를 이용하면 파일에 그림을 저장할 수 있습니다.

```
> windows(); windows(); pdf()
> dev.list()
windows windows pdf
  2         3     4
```

③ 액티브 디바이스의 번호 확인하거나 액티브 디바이스를 변경하기

`dev.cur()` 함수는 액티브 디바이스의 번호를 보여줍니다. 액티브 디바이스를 변경하려면 `dev.set(번호)`를 입력합니다.

```
> dev.cur()
pdf
  4
> dev.set(3)
windows
  3
```

④ 열려있는 디바이스 닫기

`dev.off()` 함수는 열려있는 그래픽 디바이스를 닫아줍니다. 기본적으로는 액티브 디바이스를 닫습니다.

```
> dev.off(2)
> dev.list()
windows pdf
  3      4
> dev.cur()
```

```

windows
  3
> dev.off()
> dev.list()
pdf
  4

```

11.3. 고수준 그래픽 함수들

각 함수들을 사용하기 위한 구체적인 방법 및 전달인자(옵션)들은 R의 온라인도움말(?함수)을 찾아보기 바랍니다. 아래와 같은 몇 가지 전달인자들은 많은 함수들 안에서 공통적으로 발견 됩니다.

표. 많은 그래픽 함수들 안에서 공통적으로 발견되는 전달인자들

전달인자(옵션)	설명
add=FALSE	만일 TRUE일 경우, 기존의 그림 위에 그림을 그린다.
axes=TRUE	만일 FALSE일 경우, 축들과 box를 그리지 않는다.
type='p'	그림의 타입을 정한다. "p"는 점, "l"은 선, "b"는 점과 선 둘 다, "o"는 점과 선 둘 다 그리되 선이 점 위에 그려짐, "h"는 세로수직선, "s"는 계단식 그림으로서 자료는 세로수직선의 꼭대기에 그림, "S"는 계단식 그림으로서 자료는 세로수직선의 바닥에 그림
xlim=, ylim=	x축과 y축의 눈금의 하한과 상한을 지정함. 예를 들어 xlim=c(1,10) 또는 xlim=range(x)
xlab=, ylab=	축 라벨
main=	주 제목
sub=	부 제목 (더 작은 폰트크기)
cex=, cex.axis=, cex.lab=, cex.main=, cex.sub=	기준 크기(1)에 대비한 크기 조절. 텍스트와 심볼의 크기(cex), 축의 눈금(cex.axis), x와 y축의 라벨 크기(cex.lab), 주 제목 크기(cex.main), 부 제목 크기(cex.sub)

그림 1 고수준 그래픽 함수들

<code>plot(x)</code>	plot of the values of x (on the <i>y</i> -axis) ordered on the <i>x</i> -axis
<code>plot(x, y)</code>	bivariate plot of x (on the <i>x</i> -axis) and y (on the <i>y</i> -axis)
<code>sunflowerplot(x, y)</code>	id. but the points with similar coordinates are drawn as a flower which petal number represents the number of points
<code>pie(x)</code>	circular pie-chart
<code>boxplot(x)</code>	“box-and-whiskers” plot
<code>stripchart(x)</code>	plot of the values of x on a line (an alternative to <code>boxplot()</code> for small sample sizes)
<code>coplot(x~y z)</code>	bivariate plot of x and y for each value (or interval of values) of z
<code>interaction.plot(f1, f2, y)</code>	if f1 and f2 are factors, plots the means of y (on the <i>y</i> -axis) with respect to the values of f1 (on the <i>x</i> -axis) and of f2 (different curves); the option fun allows to choose the summary statistic of y (by default fun=mean)
<code>matplot(x,y)</code>	bivariate plot of the first column of x <i>vs.</i> the first one of y , the second one of x <i>vs.</i> the second one of y , etc.
<code>dotchart(x)</code>	if x is a data frame, plots a Cleveland dot plot (stacked plots line-by-line and column-by-column)
<code>fourfoldplot(x)</code>	visualizes, with quarters of circles, the association between two dichotomous variables for different populations (x must be an array with dim=c(2, 2, k) , or a matrix with dim=c(2, 2) if $k = 1$)
<code>assocplot(x)</code>	Cohen-Friendly graph showing the deviations from independence of rows and columns in a two dimensional contingency table
<code>mosaicplot(x)</code>	‘mosaic’ graph of the residuals from a log-linear regression of a contingency table
<code>pairs(x)</code>	if x is a matrix or a data frame, draws all possible bivariate plots between the columns of x
<code>plot.ts(x)</code>	if x is an object of class “ ts ”, plot of x with respect to time, x may be multivariate but the series must have the same frequency and dates
<code>ts.plot(x)</code>	id. but if x is multivariate the series may have different dates and must have the same frequency
<code>hist(x)</code>	histogram of the frequencies of x
<code>barplot(x)</code>	histogram of the values of x
<code>qqnorm(x)</code>	quantiles of x with respect to the values expected under a normal law
<code>qqplot(x, y)</code>	quantiles of y with respect to the quantiles of x
<code>contour(x, y, z)</code>	contour plot (data are interpolated to draw the curves), x and y must be vectors and z must be a matrix so that dim(z)=c(length(x), length(y)) (x and y may be omitted)
<code>filled.contour(x, y, z)</code>	id. but the areas between the contours are coloured, and a legend of the colours is drawn as well
<code>image(x, y, z)</code>	id. but the actual data are represented with colours
<code>persp(x, y, z)</code>	id. but in perspective
<code>stars(x)</code>	if x is a matrix or a data frame, draws a graph with segments or a star where each row of x is represented by a star and the columns are the lengths of the segments
<code>symbols(x, y, ...)</code>	draws, at the coordinates given by x and y , symbols (circles, squares, rectangles, stars, thermometres or “boxplots”) which sizes, colours, etc. are specified by supplementary arguments
<code>termplot(mod.obj)</code>	plot of the (partial) effects of a regression model (mod.obj)

11.4. plot() 함수 예제

R의 내장 함수인 plot()은 산점도를 출력하는 가장 기본적인 그래프 함수입니다.

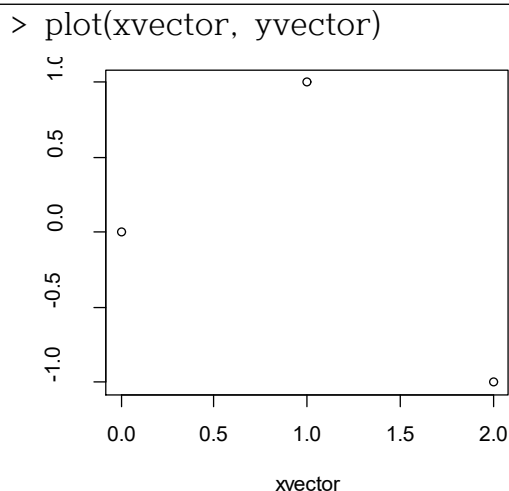
```
plot(x, y)
- x: 자료점의 x좌표 벡터
- y: 자료점의 y좌표 벡터
```

예제 1. 세 점 (0, 0), (1, 1), (2, -1)을 x-y 평면에 그리기

① 세 점의 x좌표 값들을 모아 R 벡터로 만드시오. 세 점의 y좌표 값들을 모아 R 벡터로 만드시오.

```
> xvector <-
> yvector <-
```

② 위 ①을 이용해 세 점의 산점도를 그리시오.



③ x축 라벨을 “x”로 설정하시오.

: plot()함수의 옵션 xlab=을 이용한다.

```
plot(..., xlab = “설정할 x축 라벨”)
```

④ y축 라벨을 “y”로 설정하시오.

: plot()함수의 옵션 ylab=을 이용한다.

```
plot(..., ylab = “설정할 y축 라벨”)
```

⑤ 그림 제목을 “Three Points”로 설정하시오.

: plot()함수의 옵션 main=을 이용한다.

```
plot(..., main = “설정할 제목”)
```

⑥ 그림 타입을 다음과 같이 변경하시오:

i) 선 그래프

```
plot(..., type = "l")
```

ii) 점 그래프(기본 설정)

```
plot(..., type = "p")
```

iii) 점과 선 그래프 ‘둘다(both)’

```
plot(..., type = "b")
```

iv) 점과 선 그래프, 점 위에’(over)’ 선이 그려지도록.

```
plot(..., type = "o")
```

v) 계단식’(step)’ 그래프

```
plot(..., type = "s")
```

⑦ x 축 눈금의 범위를 하한 -3부터 상한 3으로 변경하시오.

```
plot(..., xlim = c(-3, 3))
```

예제 2. Puromycin 데이터

이 예제에서 사용할 데이터는 R의 datasets패키지에 포함되어 있는 Puromycin 데이터입니다.

```
> ?Puromycin
Puromycin Data는 conc, rate, state를 변수로 갖는 데이터 프레임
자료로 Puromycin이 처리된 셀과 그렇지 않은 셀을 포함하는 요소
반응에서의 기질 농도 대비 반응 속도를 담고 있다.
```

```
> class(Puromycin)
> dim(Puromycin)
> names(Puromycin)
> str(Puromycin)
```

```
> str(Puromycin)
'data.frame': 23 obs. of 3 variables:
 $ conc : num 0.02 0.02 0.06 0.06 0.11 0.11 0.22 0.22 0.56 0.56 ...
 $ rate : num 76 47 97 107 123 139 159 152 191 201 ...
 $ state: Factor w/ 2 levels "treated","untreated": 1 1 1 1 1 1 1 1 ...
 - attr(*, "reference")= chr "A1.3, p. 269"
```

자료 준비를 위해 Puromycin 데이터를 처리군과 비처리군으로 나누겠습니다.

```
> PuroTrt <- subset(Puromycin, state=="treated")
> PuroUnTrt <- subset(Puromycin, state=="untreated")
```

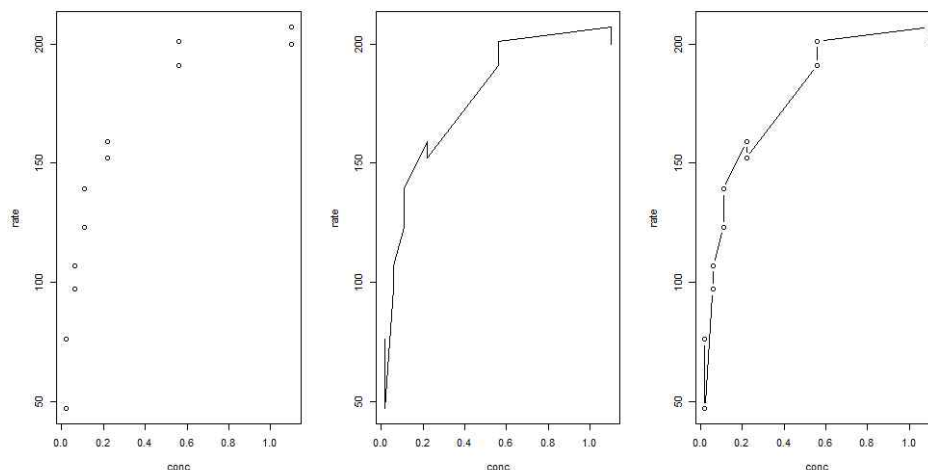
산점도를 그려보자.

- ① PuroTrt 데이터의 두 변수 중 conc를 가로축으로, rate를 세로축으로 하여 산점도를 그리시오. 고수준 그래픽 함수 plot()을 이용.

```
> plot(rate ~ conc, data = PuroTrt)
==> y축을 rate로, conc를 x축으로 하여 산점도를 작성한다.
```

- ② type이라는 전달인자를 이용하여 점들을 연결하는 선을 그린다.

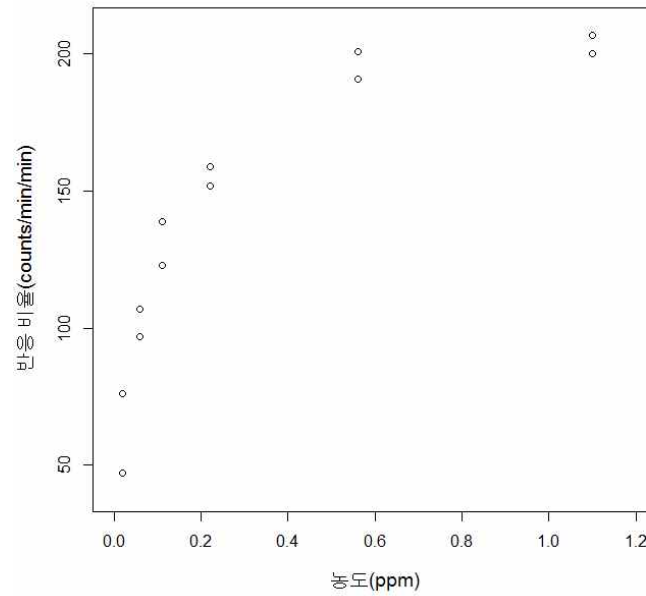
```
> par(mfrow=c(1, 3))
➔ 하나의 그래픽 장치에 1행 3열로 그래프를 배치하도록 하는 명령
> plot(rate ~ conc, data=PuroTrt, type="p")
> plot(rate ~ conc, data=PuroTrt, type="l")
> plot(rate ~ conc, data=PuroTrt, type="b")
```



- ③ 그래프의 각 축의 제목, 크기, 범위를 변경하시오.

```
> par(mfrow=c(1,1))
> plot(rate ~ conc, data=PuroTrt,
       xlim=c(0, 1.2), ylim=c(40, 210),
```

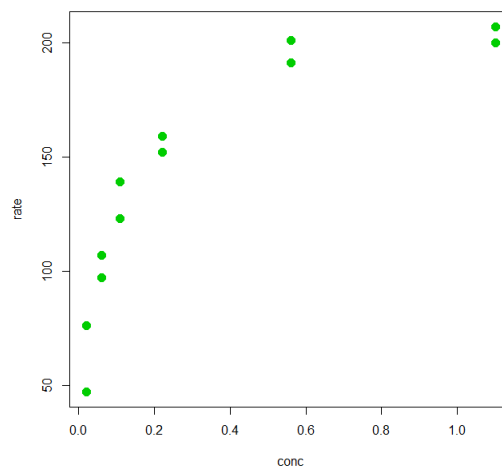
```
xlab="농도 (ppm) ", ylab="반응비율 (counts/min/min) ",
cex.lab=1.2)
```



④ 산점도로 찍히는 점의 모양, 색깔, 크기를 변경해 보겠습니다.

- 점의 모양 변경: `pch=`
- 점의 색깔 변경: `col=`
- 점의 크기 변경: `cex=`

```
> plot(rate ~ conc, data=PuroTrt,
      pch=16, col=3, cex=1.5)
```



다음은 pch=와 col=에 주어지는 값에 해당하는 모양 및 색깔입니다.

```

□ 0 ▽ 25 2 50 K 75 d100 }125 150 175 200 225 250
○ 1 26 3 51 L 76 e101 ~126 151 176 201 226 251
△ 2 27 4 52 M 77 f102 0127 152 177 202 227 252
+ 3 28 5 53 N 78 g103 128 153 178 203 228 253
× 4 29 6 54 O 79 h104 129 154 179 204 229
◇ 5 30 7 55 P 80 i105 130 155 180 205 230
▽ 6 31 8 56 Q 81 j106 131 156 181 206 231
▣ 7 32 9 57 R 82 k107 132 157 182 207 232
* 8 ! 33 : 58 S 83 l108 133 158 183 208 233
⊕ 9 " 34 ; 59 T 84 m109 134 159 184 209 234
⊙ 10 # 35 < 60 U 85 n110 135 160 185 210 235
⌘ 11 $ 36 = 61 V 86 o111 136 161 186 211 236
⊞ 12 % 37 > 62 W 87 p112 137 162 187 212 237
⊗ 13 & 38 ? 63 X 88 q113 138 163 188 213 238
⊠ 14 , 39 @ 64 Y 89 r114 139 164 189 214 239
■ 15 ( 40 A 65 Z 90 s115 140 165 190 215 240
● 16 ) 41 B 86 [ 91 t116 141 166 191 216 241
▲ 17 * 42 C 67 \ 92 u117 142 167 192 217 242
◆ 18 + 43 D 68 ] 93 v118 143 168 193 218 243
● 19 , 44 E 69 ^ 94 w119 144 169 194 219 244
● 20 - 45 F 70 - 95 x120 145 170 195 220 245
○ 21 ~ 46 G 71 ~ 96 y121 146 171 196 221 246
□ 22 / 47 H 72 a 97 z122 147 172 197 222 247
◇ 23 0 48 I 73 b 98 {123 148 173 198 223 248
△ 24 1 49 J 74 c 99 |124 149 174 199 224 249

```

```

0 10 20 30 40 50 60 70 80 90
1 11 21 31 41 51 61 71 81 91
2 12 22 32 42 52 62 72 82 92
3 13 23 33 43 53 63 73 83 93
4 14 24 34 44 54 64 74 84 94
5 15 25 35 45 55 65 75 85 95
6 16 26 36 46 56 66 76 86 96
7 17 27 37 47 57 67 77 87 97
8 18 28 38 48 58 68 78 88 98
9 19 29 39 49 59 69 79 89 99

```

11.5. 저수준 그래픽 함수들

저수준 그래픽 함수들은 이미 존재하는 그래프에 영향을 줍니다. 아래는 주요 저수준 그래픽 함수들입니다.

<code>points(x, y)</code>	adds points (the option <code>type=</code> can be used)
<code>lines(x, y)</code>	id. but with lines
<code>text(x, y, labels, ...)</code>	adds text given by <code>labels</code> at coordinates <code>(x,y)</code> ; a typical use is: <code>plot(x, y, type="n"); text(x, y, names)</code>
<code>mtext(text, side=3, line=0, ...)</code>	adds text given by <code>text</code> in the margin specified by <code>side</code> (see <code>axis()</code> below); <code>line</code> specifies the line from the plotting area
<code>segments(x0, y0, x1, y1)</code>	draws lines from points <code>(x0,y0)</code> to points <code>(x1,y1)</code>
<code>arrows(x0, y0, x1, y1, angle= 30, code=2)</code>	id. with arrows at points <code>(x0,y0)</code> if <code>code=2</code> , at points <code>(x1,y1)</code> if <code>code=1</code> , or both if <code>code=3</code> ; <code>angle</code> controls the angle from the shaft of the arrow to the edge of the arrow head
<code>abline(a,b)</code>	draws a line of slope <code>b</code> and intercept <code>a</code>
<code>abline(h=y)</code>	draws a horizontal line at ordinate <code>y</code>
<code>abline(v=x)</code>	draws a vertical line at abscissa <code>x</code>
<code>abline(lm.obj)</code>	draws the regression line given by <code>lm.obj</code> (see section 5)
<code>rect(x1, y1, x2, y2)</code>	draws a rectangle which left, right, bottom, and top limits are <code>x1</code> , <code>x2</code> , <code>y1</code> , and <code>y2</code> , respectively
<code>polygon(x, y)</code>	draws a polygon linking the points with coordinates given by <code>x</code> and <code>y</code>
<code>legend(x, y, legend)</code>	adds the legend at the point <code>(x,y)</code> with the symbols given by <code>legend</code>
<code>title()</code>	adds a title and optionally a sub-title
<code>axis(side, vect)</code>	adds an axis at the bottom (<code>side=1</code>), on the left (2), at the top (3), or on the right (4); <code>vect</code> (optional) gives the abscissa (or ordinates) where tick-marks are drawn
<code>box()</code>	adds a box around the current plot
<code>rug(x)</code>	draws the data <code>x</code> on the <code>x</code> -axis as small vertical lines
<code>locator(n, type="n", ...)</code>	returns the coordinates <code>(x,y)</code> after the user has clicked <code>n</code> times on the plot with the mouse; also draws symbols (<code>type="p"</code>) or lines (<code>type="l"</code>) with respect to optional graphic parameters (...); by default nothing is drawn (<code>type="n"</code>)

11.6. legend 함수 사용 예

```
> PuroTrt <- subset(Puromycin, state == "treated")
> PuroUnTrt <- subset(Puromycin, state == "untreated")
> plot(rate ~ conc, data = PuroTrt, type = "b")
> points(rate ~ conc, data = PuroUnTrt, type = "b", col = "red", pch
= 8)
> legend("bottomright", legend = c("treated", "untreated"),
```

```
col = c("black", "red"),  
pch = c(1, 8), lty = c(1,1))
```

