

Kyun-Seop Bae, Sungpil Han

R Programming - Lecture Notes



Contents

List of Tables	v
List of Figures	vii
Preface	ix
I Lecture Notes	1
1 Graphics	3
1.1 Introduction	3
1.2 상위수준 그림 함수	3
1.2.1 상위수준 그림 함수의 주요 인자 (arguments) . .	3
1.2.2 scatter plot	4
1.2.3 Histogram	7
1.2.4 Box-Whisker Plot	11
1.2.5 Bar Plot	14
1.2.6 pie chart	18
1.2.7 matplotlib 함수	19
1.2.8 Scatter plot matrices (pairs plots)	20
1.3 하위수준 그림 함수	22
1.3.1 점, 선, 설명 추가 하기 {add}	22
1.3.2 polygon 함수	24
1.4 그림 출력하기	25
1.4.1 pdf graphics devices	25
1.4.2 PNG graphics devices	26
2 Data Import / Export	27
2.1 Read.csv	27
2.2 Theoph 데이터	27
2.3 lattice	31
2.4 Subsetting and write.csv	34

II	Part II. Basics	43
3	Let's start!	45
3.1	Exploring data	48
3.2	Descriptive statistics	49
3.3	Plotting	50
3.4	Hypothesis tests	54
	Appendix	57
A	As-is R Files	57
A.1	Lecture 3	57
A.2	Lecture 4	65
B	Using Coursera	67
C	R Tips	69
D	Acknowledgement	71
	Bibliography	73
	Index	75

List of Tables



1	Creative Commons License	x
3.1	Despite what you might find at family friendly waterparks – this is NOT how real pirate swimming lessons look.	47



Preface

안녕하십니까?

2017년 1학기 울산대학교 의학과 대학원 수업 R Programming 과목 담당교수 배균섭입니다.

R은 <http://cran.r-project.org> 에서 다운로드받아 설치할 수 있습니다. 역시 같은 사이트에서 Manual이 나와 있으니 참고하시기 바랍니다. 구글에서 'R Programming pdf'와 같은 키워드로 검색하시면 많은 자료를 보실 수 있습니다.

첨부한 R.stx¹ 파일은 AcroEdit이라는 editor에서 사용할 syntax highlighting용 구문 파일입니다. <http://www.acrosoft.pe.kr> 에서 다운로드 받아 설치하시기 바랍니다. AcroEdit대신 notepad++를 선호하시는 분은 그대로 사용하셔도 됩니다.

저는 RStudio, tinnR 등을 이용해서 강의하지 않습니다만, 필요하신 분은 쓰셔도 괜찮습니다. 향후 R package 작성을 위해서는 MiKTeX와 Rtools를 설치하십시오.

추가로 말씀드리자면, <http://www.coursera.org> 에 많은 R 강좌가 개설되어 있습니다. Specialization course로 들어가면 유료이지만, (Specialization course는 여러 개의 과목이 합쳐져 있는 것입니다.) 개별 과목을 검색해서 들어가면, 무료로도 볼 수 있습니다. (대신 시험을 칠 수 없거나, certificate를 받을 수 없습니다.)

좋은 강좌가 많으니 많이 활용하시기 바랍니다.

강의 장소에 불편함이 많은 것으로 생각되어, 다음과 같이 Skype 모임을 개설하였습니다. 사정상 원거리에서 오시기 불편한 분들은 활용하시기

¹<https://groups.google.com/a/acr.kr/group/r/attach/409db97bf453a/R.stx?part=0.1&authuser=0>

바랍니다. 출석은 화면을 캡처하거나 휴대폰으로 찍은 뒤 sec@acp.kr², shan@acp.kr³ 보내주시면 출석으로 인정해 드립니다.

Skype 모임 참가 <https://meet.lync.com/uucp-acp/ksbae/SKGJ3BNQ>

2017년 3월, 배균섭 배상

The online version of this book is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License⁴.



FIGURE 1: Creative Commons License

Teaching Assistant

안녕하십니까? 서울아산병원 임상약리학과 전공의 한성필입니다. 수업과 관련된 여러 제반 업무를 담당하고 있습니다. 언제든지 의문사항 있으면 r@acr.kr⁵ 로 전체 메일 보내시거나 교수님 k@acr.kr⁶ 혹은 제 개인 메일 shan@acp.kr⁷ 로 연락해 주십시오.

교수님께서 세우신 방침에 따라 수업시간에 출석을 부르지 않을 예정입니다. 수강하시는 화면 (Skype) 을 휴대폰으로 사진 찍으시거나 강의실의 스크린을 사진으로 촬영하셔서 sec@acp.kr⁸ / shan@acp.kr⁹ 로 동시에 보내주시면 됩니다. 가급적 “2017-03-31 한성필 출석” 과 같은 식의 제목을 유지해 주시면 처리하는데 큰 도움이 될 것 같습니다.

출석 체크를 위해 전체메일을 사용하지 말아주십시오!

²mailto:sec@acp.kr

³mailto:shan@acp.kr

⁴<http://creativecommons.org/licenses/by-nc-sa/4.0/>

⁵mailto:r@acr.kr

⁶mailto:k@acr.kr

⁷mailto:shan@acp.kr

⁸mailto:sec@acp.kr

⁹mailto:shan@acp.kr

아울러 수업 중에 사용한 코드/스크립트를 사용하여 R의 패키지인 bookdown을 사용해 웹북을 제작 중에 있습니다. (Xie, 2016) 여러분이 읽고 있는 이 책 자체가 R 코드의 일종인 Rmarkdown의 결과물이라고 보시면 됩니다. Github 저장소¹⁰가 있으니 소스 코드를 보실 수 있습니다. 누구나 소스를 편집하여 Pull Request를 요청할 수 있으므로 혹시 Github를 사용하셔서 웹북의 질을 높이하고자 하시는 수강생 선생님들께서는 도움을 주십시오. 혹은 웹북의 각 페이지 아래쪽에 Disqus 창을 달아놓았으므로, 궁금한 점을 메모로 남겨주셔도 좋습니다.

감사합니다.

2017년 3월, 한성필 올림

FAQ

Q. 미국학회 참석으로 수업시간이 귀국행 비행기 기내에 있을거같아 출석이 안될것 같습니다. 방법이 있을까요?

결석 사유서를 제출해 주시면 출석 처리 하겠습니다. 대학원 홈페이지 참고 바랍니다.¹¹ 이 링크로 들어가시면 가장 위에 있습니다. (결석사유서.hwp) 참고로 수업 영상은 녹화하여 Youtube에 비공개 링크를 만들 예정이라서 추후에 관련 영상을 시청할 수 있을 것 같습니다. 결석사유서를 제출한다고 100% 출석이 인정되는 것은 아닙니다. 이것이 기본적으로는 offline 강의이기 때문에 강의시간에 강의실에 있든지, 또는 온라인으로 접속해 있어야 합니다. 출석사유서를 제출하거나, 추후 동영상 시청을 해서 그 증거(사진)을 제출하는 경우에 감점을 줄어드릴 수 있습니다. 예를 들어, 결석시에는 2점 감점인데, 결석사유서를 제출하면 1점만 감점한다는지, 동영상을 보면 0.5점만 감점한다는지 하는 것입니다. 결석 사유서 제출 시 출석 처리 원칙에 대한 설명을 드리오니, 참고하시길 바랍니다.

¹⁰<https://github.com/asancpt/Rprogramming>

¹¹<http://www.medulsan.ac.kr/graduate/?mid=72&curpage=files>

Q. 스카이프를 한번도 안써봐서 이참에 사용법을 배우고있는데, 수업시작시에 상대방을 어떻게 검색해서 들어가면 될지 알려주시면 감사하겠습니다.

Q. 온라인 수강시 접속하는 스카이프 주소는 무엇인지요?

<https://meet.lync.com/uucp-acp/ksbae/SKGJ3BNQ>

Chrome 등 웹브라우저에서 위 주소를 입력하면 직접 대화방으로 연결됩니다. (검색할 필요 없습니다.) 처음 설치시에는 Add-on이 설치될 수 있습니다. MacOS Sierra, Win7, Win10에서 Chrome, Internet Explorer 등을 사용하여 테스트해 보았고 모두 잘 동작하였습니다. 대부분의 경우 Skype For Business 계정이 없을 것으로 생각되는데 따로 로그인할 필요 없습니다.

수업 시작 30분 전부터 대화방을 개설해 놓도록 하겠습니다.

https://groups.google.com/a/acr.kr/d/msg/r/nUkrE37W2kQ/waG-FkM_BgAJ 교수님께서 처음 보낸 메일을 참고해 주십시오.

Q. 앞으로 수업은 지난 첫수업처럼 계속 온라인 수강이 가능한 것인가요?

네, 계속 온라인으로 가능합니다.

Q. 저도 웹캠을 설치하여야 하여야 하나요?

설치할 필요 없습니다. 오히려 수강자의 웹캠의 전원을 꺼두시길 권고드립니다.

Q. 수강전 온라인 강의 테스트 해볼 수 있나요?

수업 시작 30분 전부터 대화방을 개설하여 농도록 하겠습니다.

Q. 과제물이 있다고 들었는데 언제 assign 하게 되나요?

과제물은 빨라야 5주차 이후에 나갑니다.

Q. Coursera 강의를 듣고 증명서를 내면 출석을 얼마나 커버할 수 있을까요?

Coursera는 출석 커버보다는 grade를 올려 주기 위한 것입니다. 출석은 Skype로 커버해야 합니다. 출석의 성적 반영비율은 25%이지만, 규정상 4회 이상 결석이면 성적이 나갈 수 없습니다.

Q. 첫 수업 때, certification 관련 말씀을 하셨는데, 정확히 coursera 사이트에서 어떤 것을 듣고, 제출을 해야하는지 궁금합니다. (비슷한 내용이 많아, 어떤것을 들어야하는지 헷갈립니다.)

Coursera는 꼭 어느 것을 들어야 하는 것은 아니고, R programming과 관련된 것이라면 자유로이 골라서 들으면 됩니다. 대표적인 두 가지만 들자면 다음과 같습니다.

- <https://www.coursera.org/learn/r-programming>
- <https://www.coursera.org/learn/r-programming-environment>

Part I

Lecture Notes



1

Graphics

2017-03-22 임형석 교수님 강의

R을 사용해 그림 그리는 방법에 대해 알아보겠습니다.

1.1 Introduction

- 상위수준 그림 함수는 그림을 생성한다.
 - 하위수준 그림 함수는 기존의 그림에 그림을 추가한다.
-

1.2 상위수준 그림 함수

1.2.1 상위수준 그림 함수의 주요 인자 (arguments)

- main : 제목
- xlab/ylab : x축 및 y축 레이블
- xlim/ylim : x축 및 y축 범위
- col : 색깔
- lty : 선 모양
- pch : 점 모양

- `cex`: 그림 성분의 크기
- `lwd`: 선 굵기
- `type`: 그림 타입

```
dta <- read.csv("PK.csv")  
head(dta)
```

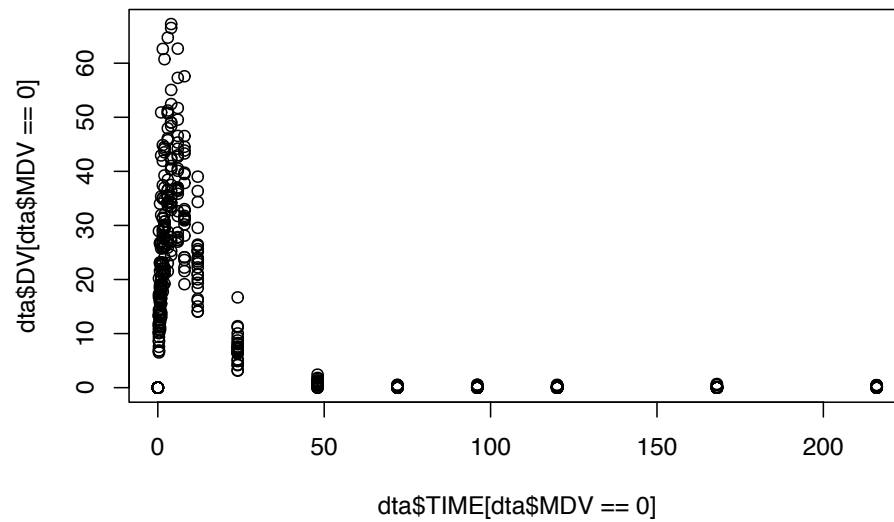
```
##   ID TIME AMT    DV MDV  
## 1  1 0.00   0  0.00   0  
## 2  1 0.00   4  0.00   1  
## 3  1 0.33   0  9.40   0  
## 4  1 0.66   0 13.71   0  
## 5  1 1.00   0 16.52   0  
## 6  1 1.50   0 29.36   0
```

```
str(dta)
```

```
## 'data.frame':    456 obs. of  5 variables:  
##  $ ID   : num  1 1 1 1 1 1 1 1 1 1 ...  
##  $ TIME: num  0 0 0.33 0.66 1 1.5 2 3 4 6 ...  
##  $ AMT  : num  0 4 0 0 0 0 0 0 0 0 ...  
##  $ DV   : num  0 0 9.4 13.7 16.5 ...  
##  $ MDV  : num  0 1 0 0 0 0 0 0 0 0 ...
```

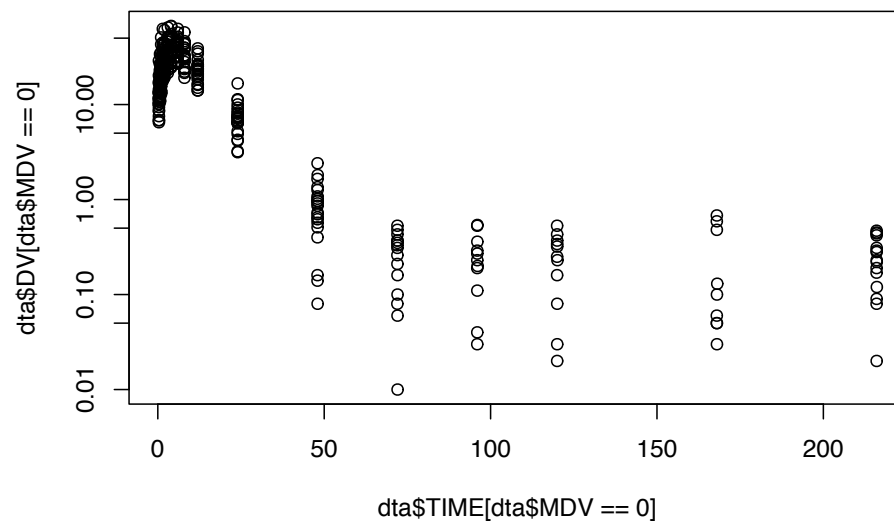
1.2.2 scatter plot

```
plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0])
```

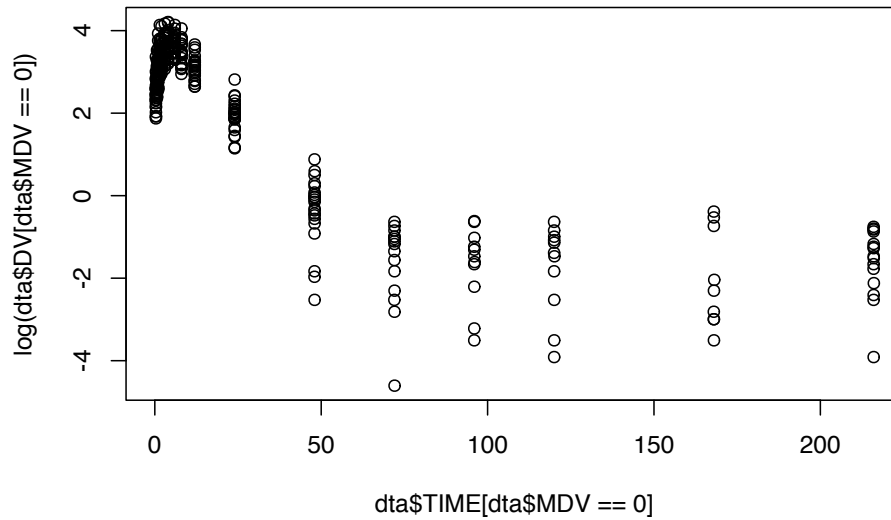


```
plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], log="y")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 86 y
## values <= 0 omitted from logarithmic plot
```

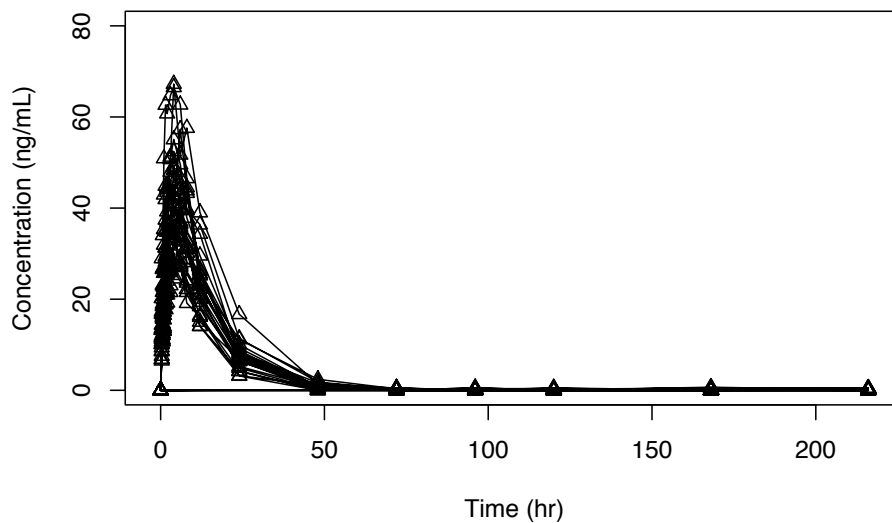


```
plot(dta$TIME[dta$MDV==0], log(dta$DV[dta$MDV==0]))
```



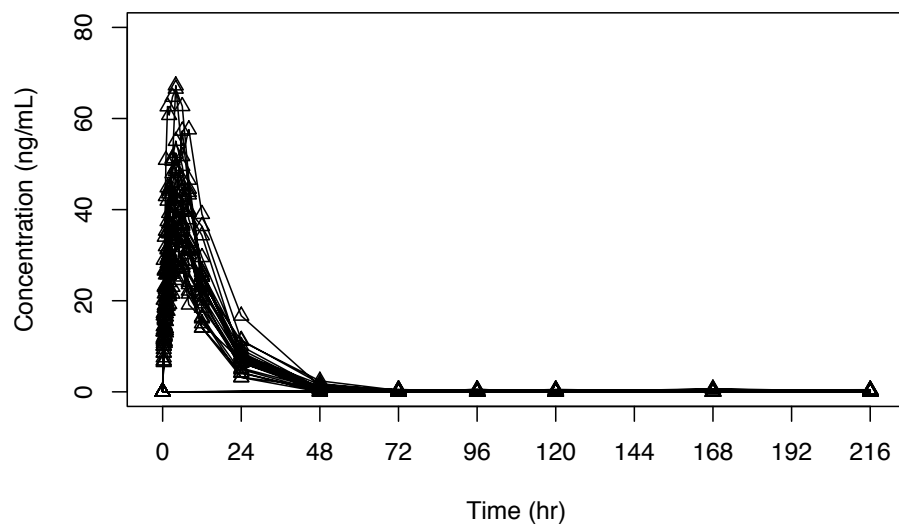
```
plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0],
     , xlab="Time (hr)", ylab="Concentration (ng/mL)"
     , type="o", pch=2, col=1, main="PK time-course of Drug X"
     , xlim =c(-2,218), ylim=c(0,80))
```

PK time-course of Drug X



```
plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], axes=F,
     , xlab="Time (hr)", ylab="Concentration (ng/mL)"
     , type="o", pch=2, col=1, main="PK time-course of Drug X"
     , xlim=c(-2,218), ylim=c(0,80))
axis(1, at=seq(0, 218, 24))
axis(2)
box()
```

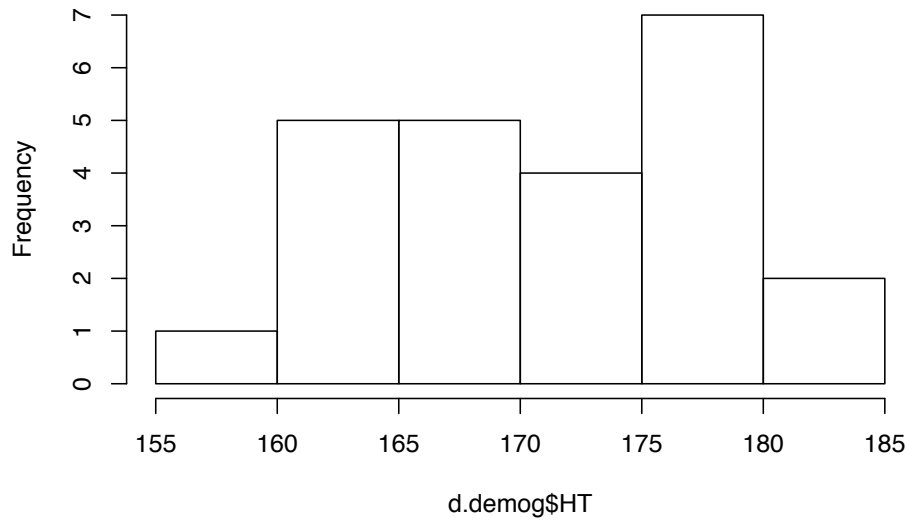
PK time-course of Drug X



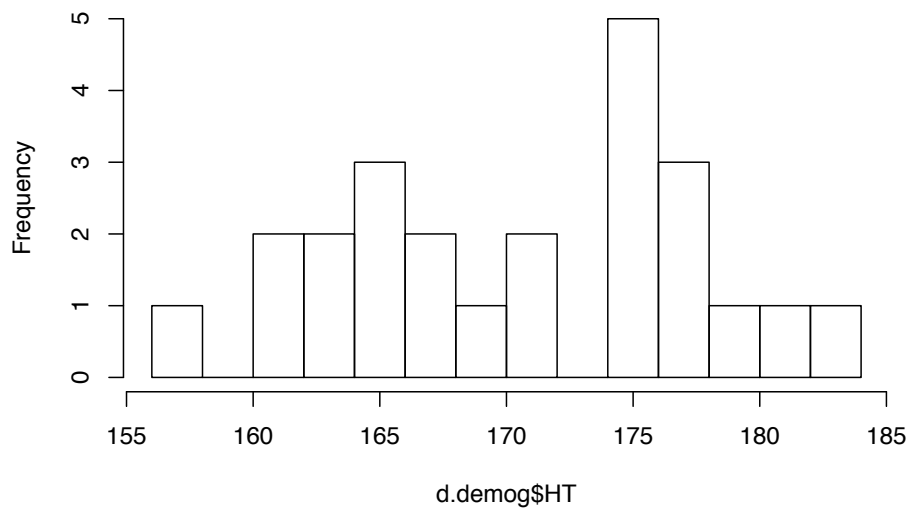
1.2.3 Histogram

```
d.demog <- read.csv("DEMOG.csv")

hist(d.demog$HT)
```

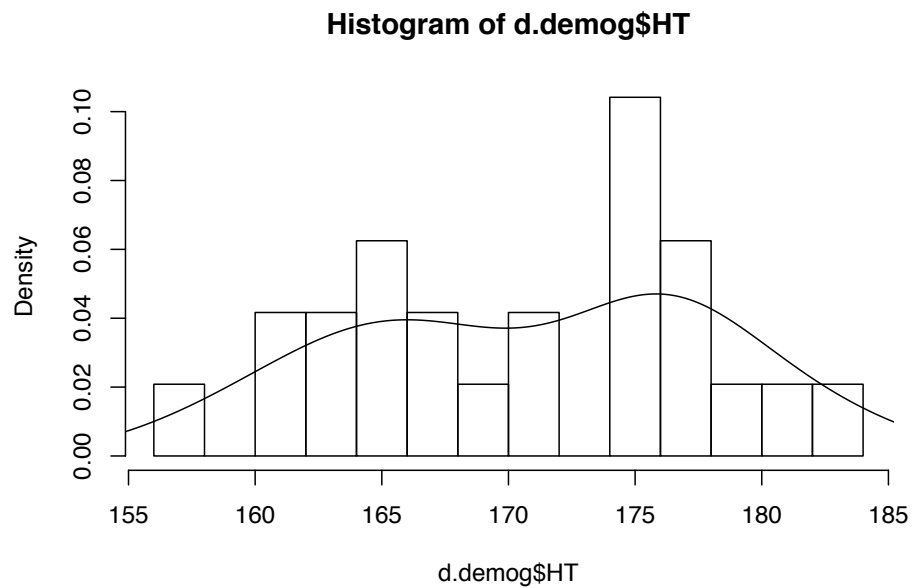
Histogram of d.demog\$HT

```
hist(d.demog$HT, breaks=10)  
hist(d.demog$HT, nclass=10)
```

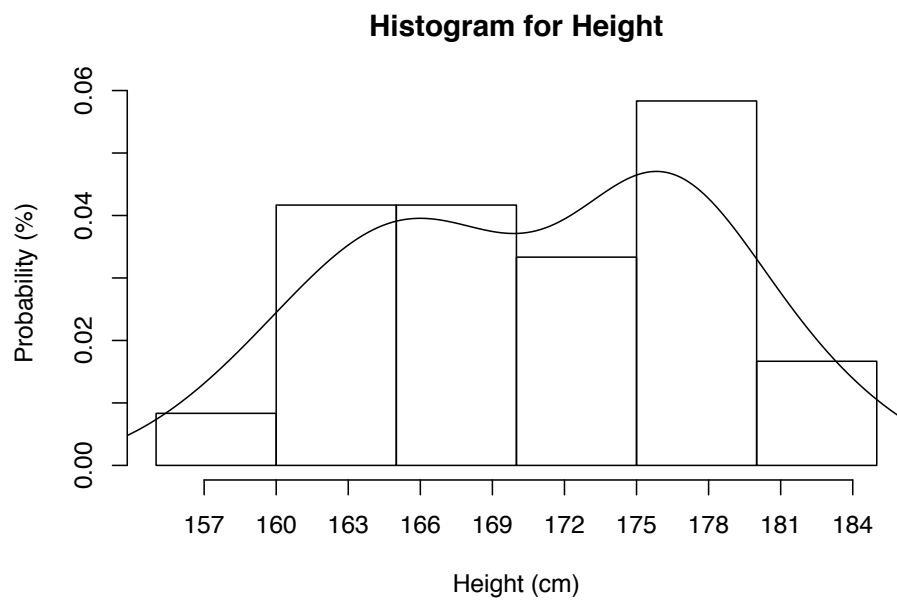
Histogram of d.demog\$HT

1.2.3.1 with density line

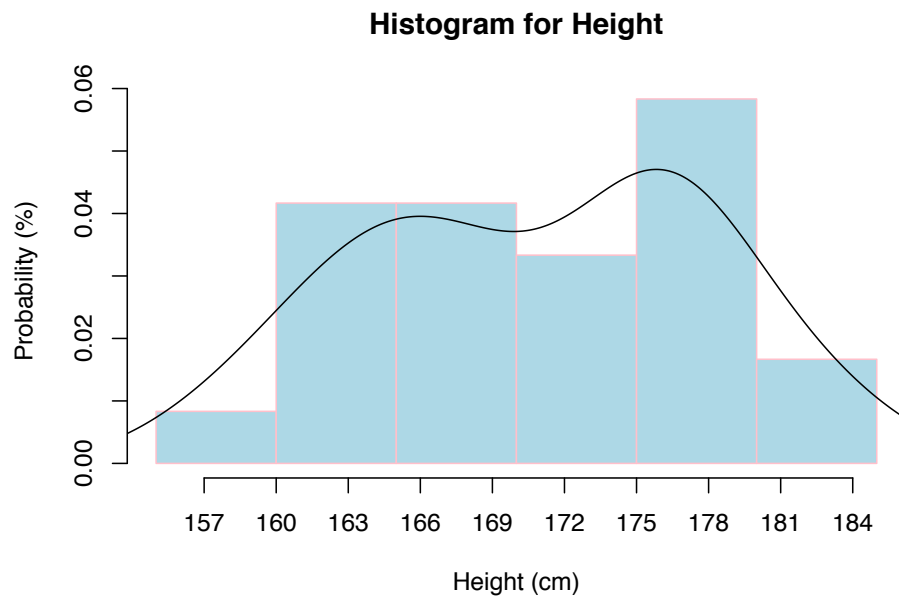
```
hist(d.demog$HT, probability=TRUE, breaks=10)  
lines(density(d.demog$HT))
```



```
hist(d.demog$HT, probability=TRUE, breaks=9, xaxt="n",  
      , main="Histogram for Height", xlab="Height (cm)", ylab="Probability (%)")  
axis(1, at=seq(min(d.demog$HT), max(d.demog$HT), 3))  
lines(density(d.demog$HT))
```

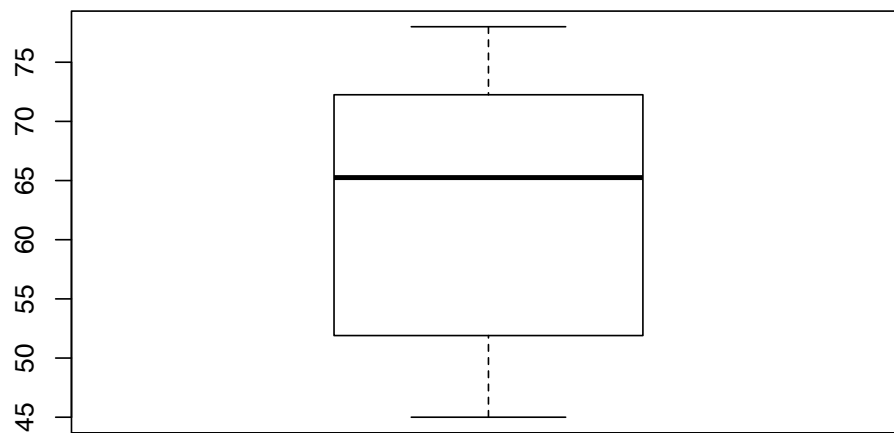


```
hist(d.demog$HT, probability=TRUE, breaks=9, xaxt="n",  
      , main="Histogram for Height", xlab="Height (cm)", ylab="Probability (%)",  
      , col = "lightblue", border = "pink")  
axis(1, at=seq(min(d.demog$HT), max(d.demog$HT), 3))  
lines(density(d.demog$HT))
```

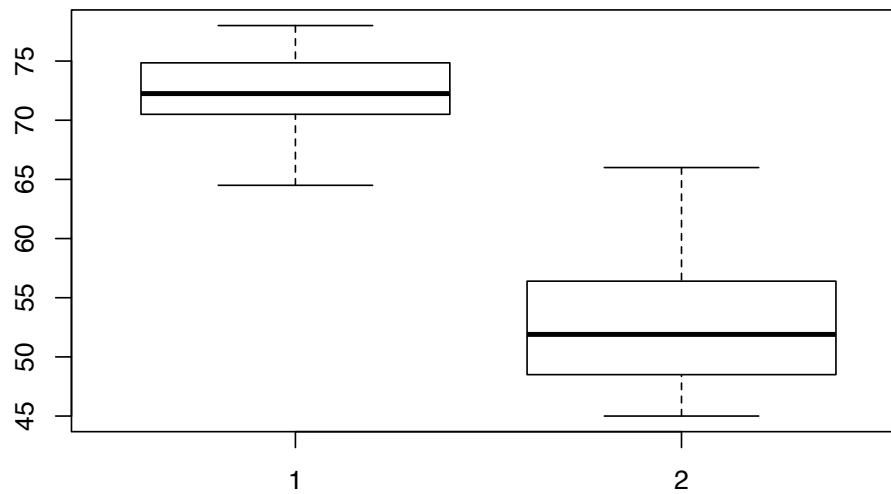
1.2.4 Box-Whisker Plot

```
boxplot(d.demog$WT)
```



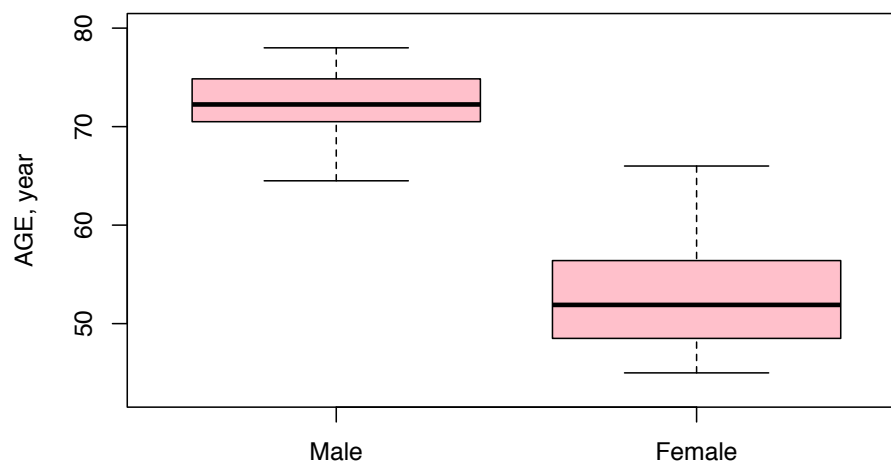
```
boxplot(d.demog$WT ~ d.demog$SEX)
```

```
boxplot(split(d.demog$WT, d.demog$SEX))
```

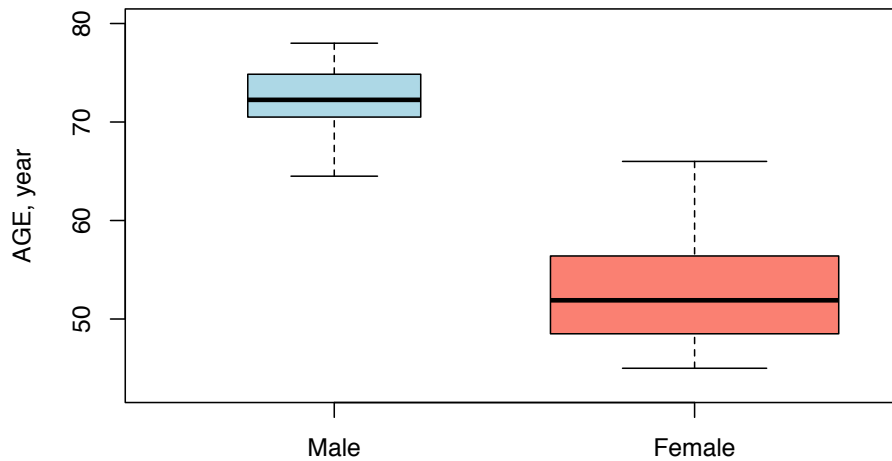


```
boxplot(WT ~ SEX, data=d.demog)
```

```
boxplot(d.demog$WT ~ d.demog$SEX  
        , names=c("Male", "Female"), ylab="AGE, year", ylim=c(min(d.demog$WT)-2, max(d.demog$WT)+2),  
        , col="pink")
```

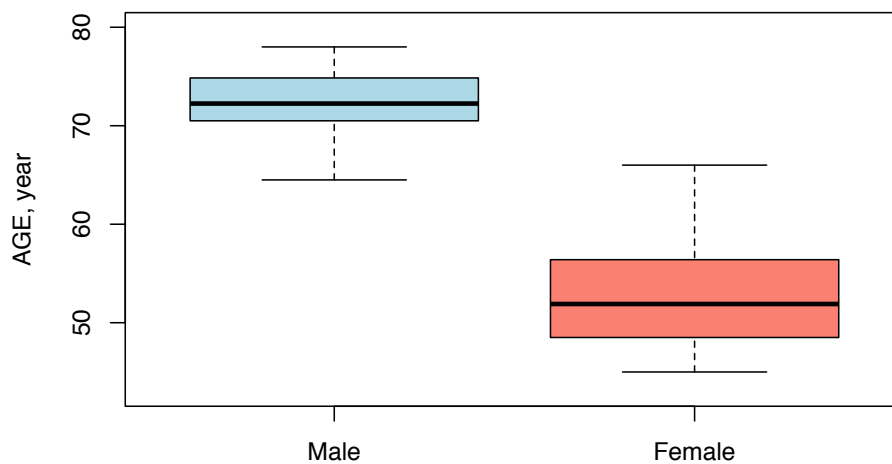


```
boxplot(d.demog$WT ~ d.demog$SEX
        , names=c("Male", "Female"), ylab="AGE, year", ylim=c(min(d.demog$WT)-2, max(d.demog$WT)+2)
        , col=c("lightblue", "salmon"), width=c(0.6, 1))
```



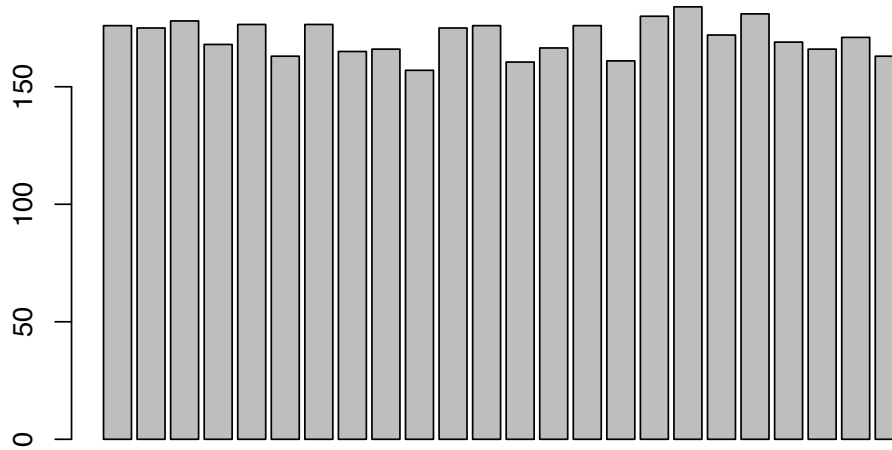
-varwidth: if varwidth is TRUE, the boxes are drawn with widths proportional to the square-roots of the number of observations in the groups.

```
boxplot(d.demog$WT ~ d.demog$SEX
        , names=c("Male", "Female"), ylab="AGE, year", ylim=c(min(d.demog$WT)-2, max(d.demog$WT)+2)
        , col=c("lightblue", "salmon")
        , varwidth=TRUE)
```



1.2.5 Bar Plot

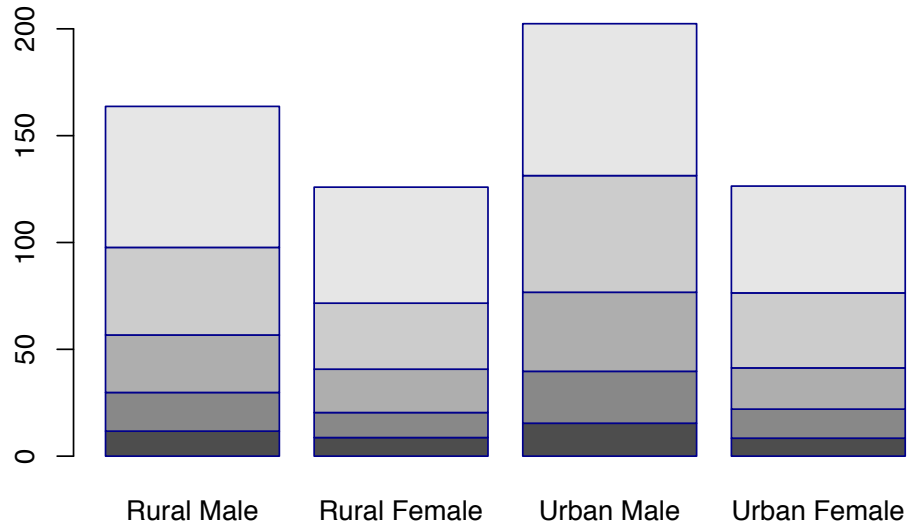
```
barplot(d.demog$HT)
```



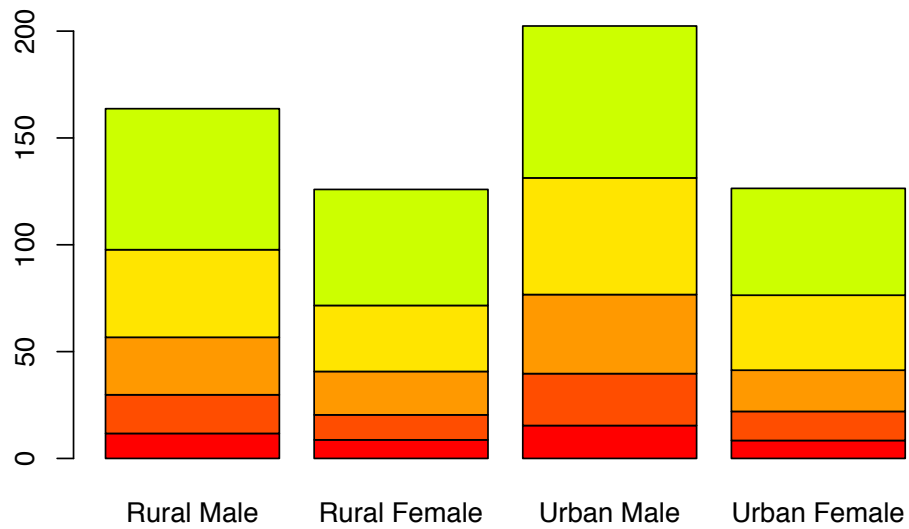
VADeaths

##	Rural Male	Rural Female	Urban Male	Urban Female
## 50-54	11.7	8.7	15.4	8.4
## 55-59	18.1	11.7	24.3	13.6
## 60-64	26.9	20.3	37.0	19.3
## 65-69	41.0	30.9	54.6	35.1
## 70-74	66.0	54.3	71.1	50.0

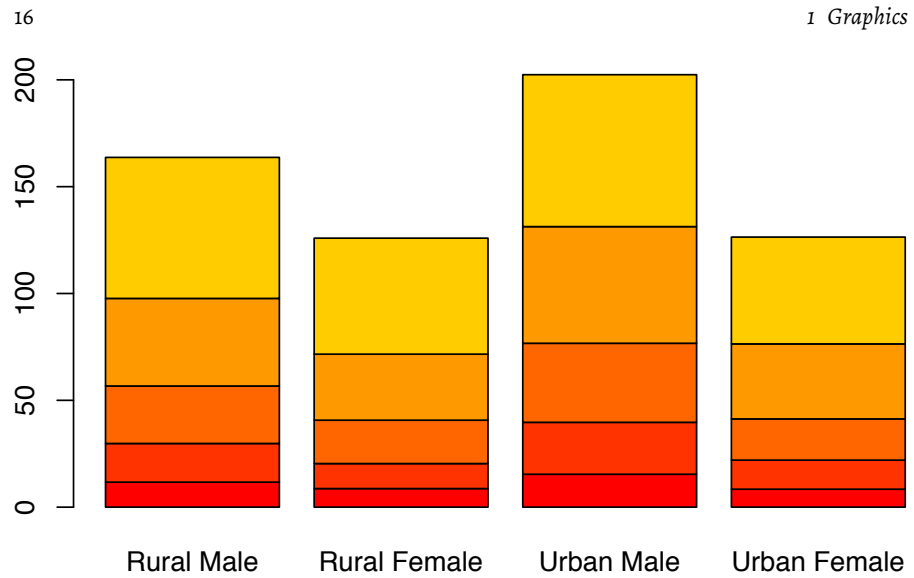
```
barplot(VADeaths, border = "dark blue")
```



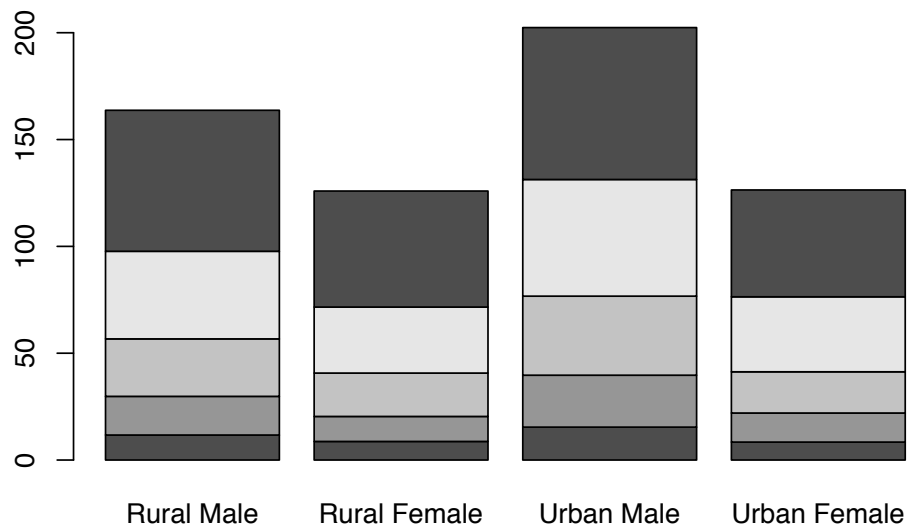
```
barplot(VADeaths, col = rainbow(20))
```



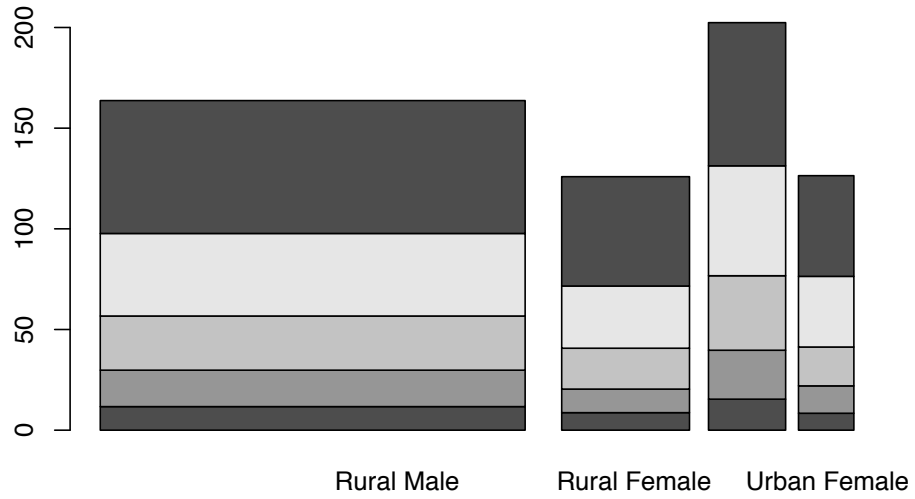
```
barplot(VADeaths, col = heat.colors(8))
```



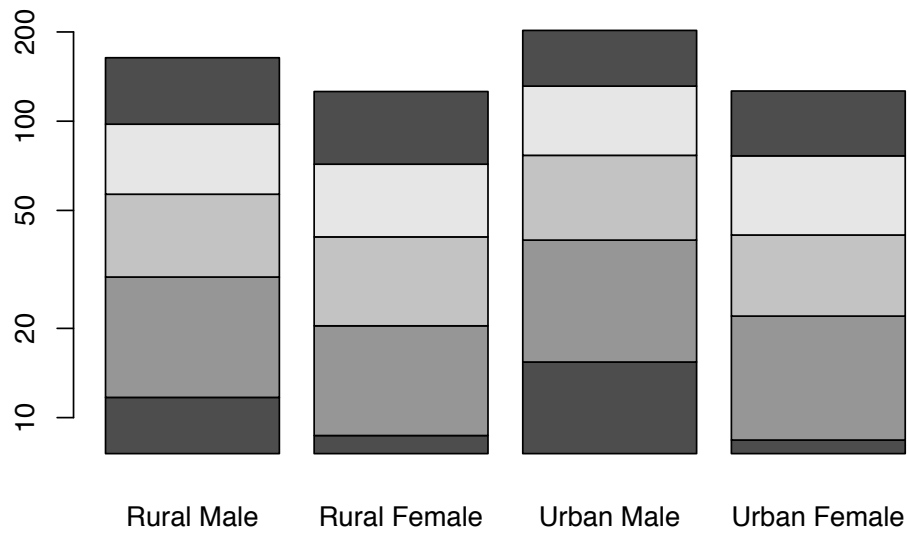
```
barplot(VADeaths, col = gray.colors(4))
```



```
barplot(VADeaths, col = gray.colors(4), log="x")
```



```
barplot(VADeaths, col = gray.colors(4), log="y")
```

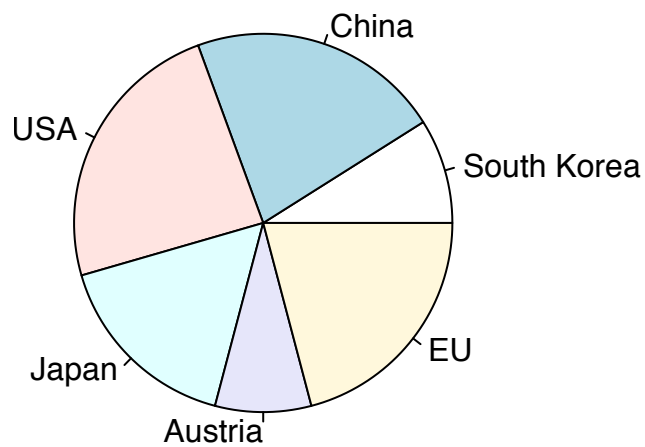


```
barplot(VADeaths, col = gray.colors(4), log="xy")
```



1.2.6 pie chart

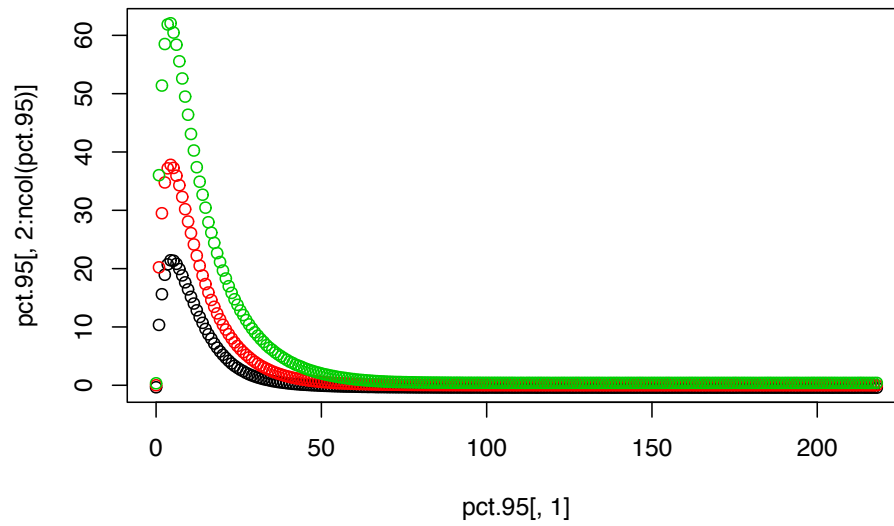
```
drug.X.market <- c(0.12, 0.29, 0.32, 0.22, 0.11, 0.28)
names(drug.X.market) <- c("South Korea", "China", "USA", "Japan", "Austria", "EU")
pie(drug.X.market)
```



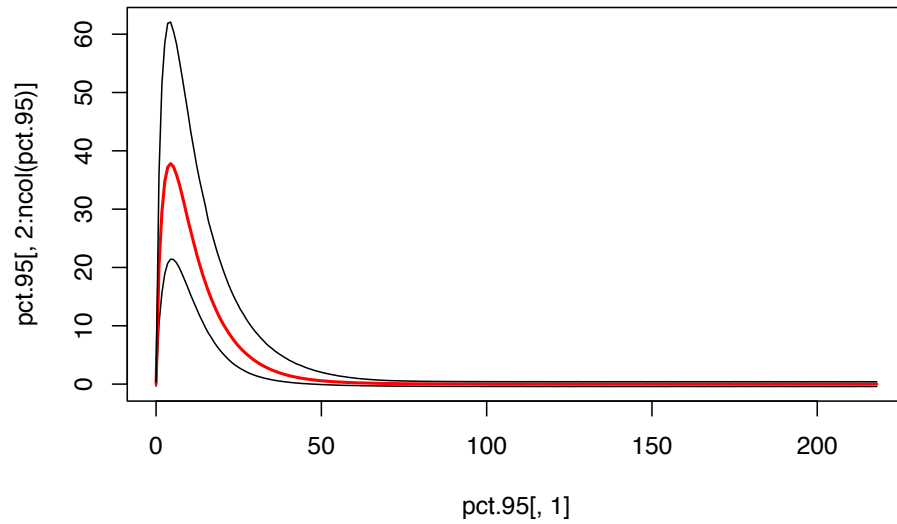
1.2.7 matplot 함수

1.2.7.1 matrix와 column 사이의 그림

```
pct.95 <- read.csv("pct95.csv")  
matplot(pct.95[,1], pct.95[,2:ncol(pct.95)], pch=1)
```

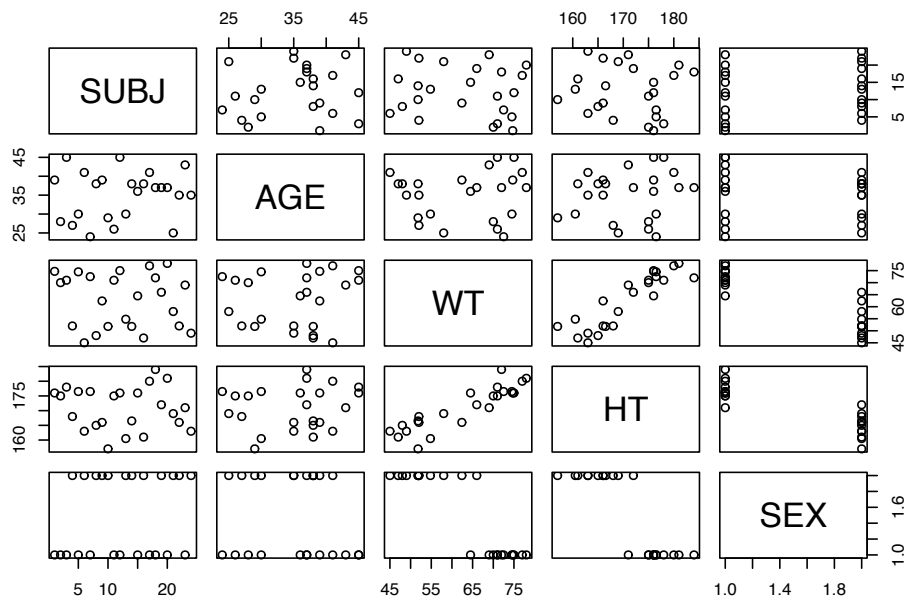


```
matplot(pct.95[,1], pct.95[,2:ncol(pct.95)], pch=1, col=c(1,2,1), type="l", lty=1, lwd=c(1,2,1))
```



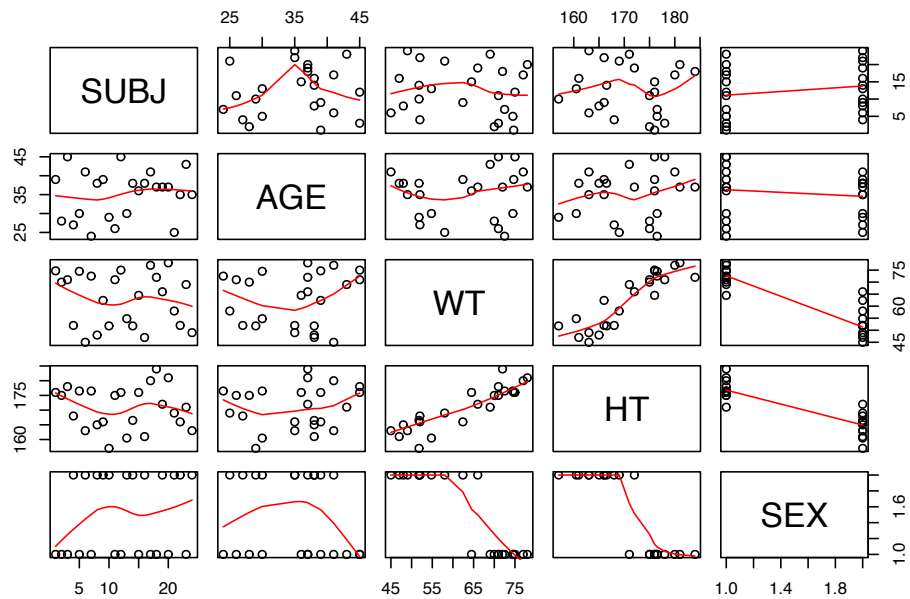
1.2.8 Scatter plot matrices (pairs plots)

```
pairs(d.demog)
```



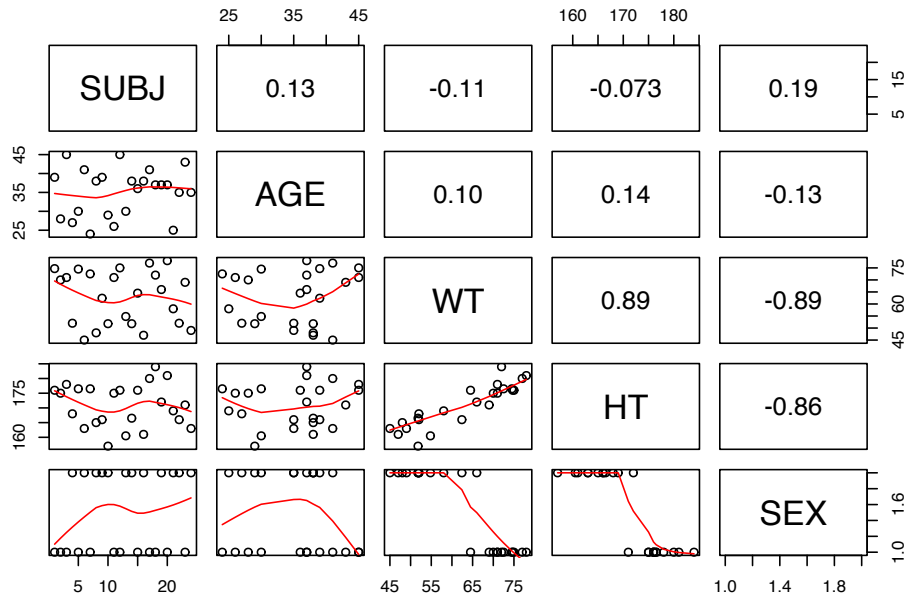
1.2.8.1 add a loess smoother, type

```
pairs(d.demog, panel = panel.smooth)
```



```
panel.cor <- function(x, y, digits=2, prefix="", cex.cor)
{
  usr <- par("usr"); on.exit(par(usr))
  par(usr = c(0, 1, 0, 1))
  r = (cor(x, y))
  txt <- format(c(r, 0.123456789), digits=digits)[1]
  txt <- paste(prefix, txt, sep="")
  if(missing(cex.cor)) cex <- 1.5
  text(0.5, 0.5, txt, cex = 1.5)
}

pairs(d.demog, lower.panel=panel.smooth, upper.panel=panel.cor)
```



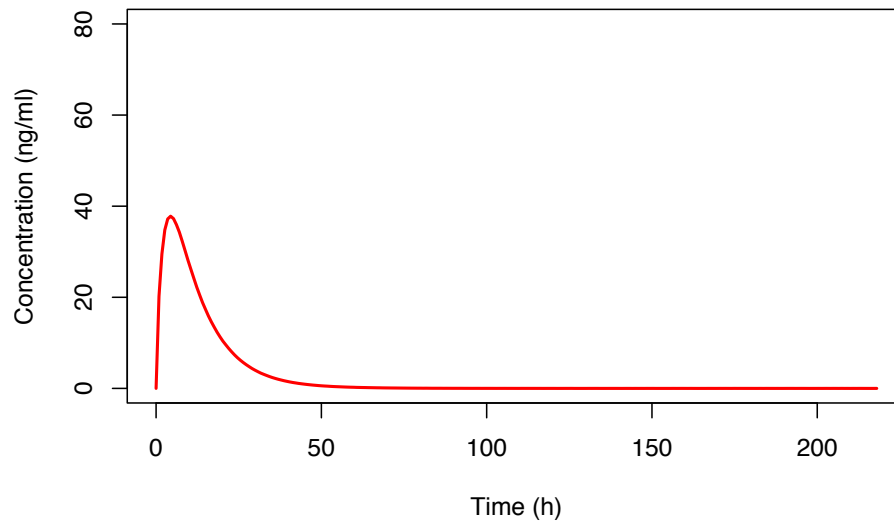
1.3 하위수준 그림 함수

- points : 점 추가
- lines : 선 추가
- abline : 기준선 추가
- mtext : 텍스트 추가
- legend : 설명 (legend) 추가
- polygon : polygon 추가

1.3.1 점, 선, 설명 추가 하기 {add}

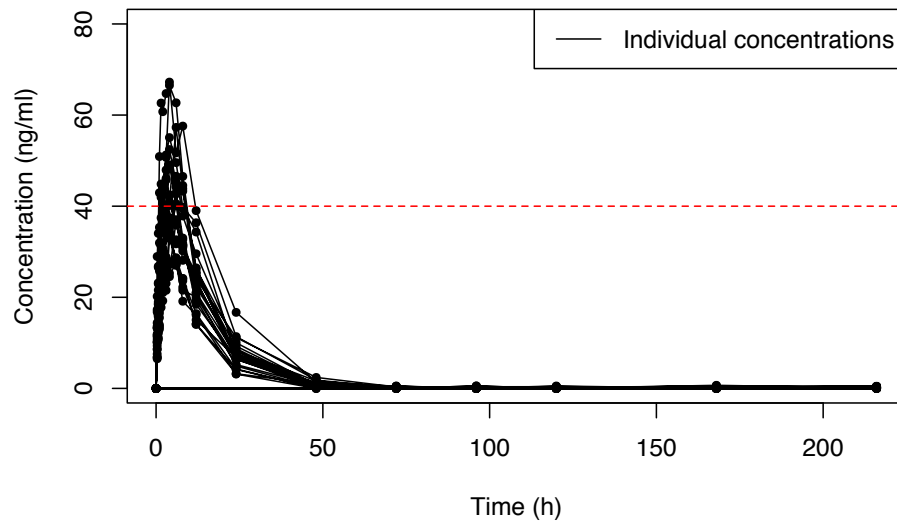
```
plot(pct.95$TIME, pct.95$PCT50, main="PK of Drug X"
     , type="l", xlab="Time (h)", ylab="Concentration (ng/ml)"
     , ylim=range(0,80), lty=1, col="red", lwd=2)
```

PK of Drug X



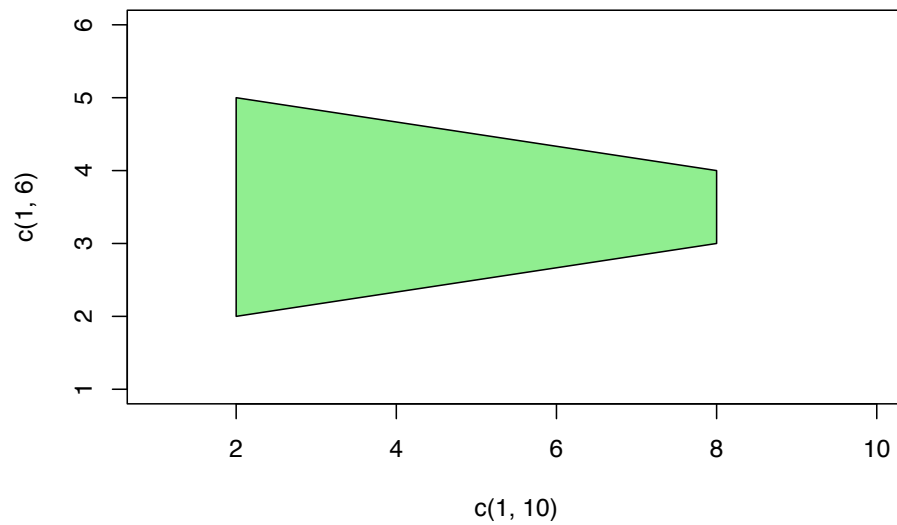
```
plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], main="PK of Drug X"
     , type="n", xlab="Time (h)", ylab="Concentration (ng/ml)"
     , ylim=range(0,80))
points(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], pch = 16, cex=0.8)
lines(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], col="black", lwd=1)
abline(40, 0, col="red", lty=2) # abline(a,b): y=a+b*x
legend("topright", legend=c("Individual concentrations")
      , lty=1, col="black")
```

PK of Drug X

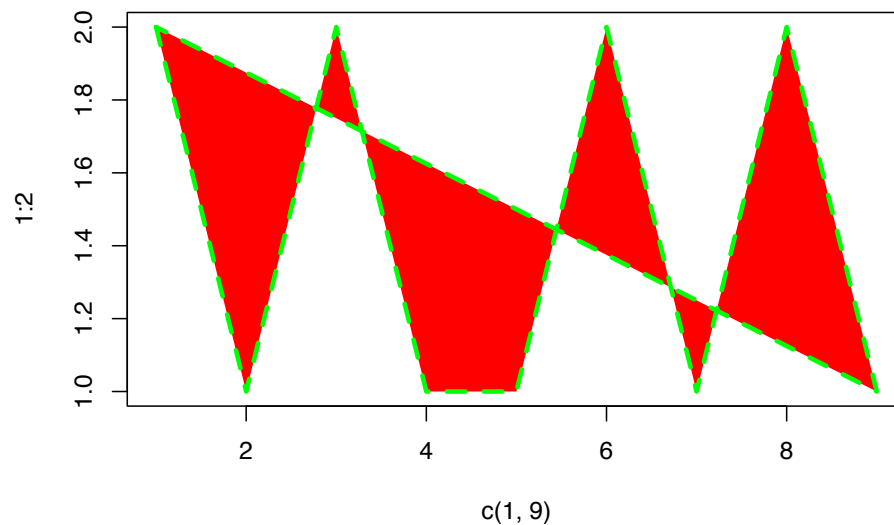


1.3.2 polygon 함수

```
plot(c(1, 10), c(1, 6), type = "n")
polygon(c(2,8,8,2), c(5,4,3,2), col="lightgreen")
```



```
plot(c(1, 9), 1:2, type = "n")
polygon(1:9, c(2,1,2,1,1,2,1,2,1),
       col = c("red", "blue"),
       border = c("green", "yellow"),
       lwd = 3, lty = c("dashed", "solid"))
```



1.4 그림 출력하기

1.4.1 pdf graphics devices

```
pdf("PK_of_Drug_X.pdf")

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], main="PK of Drug X"
     , type="n", xlab="Time (h)", ylab="Concentration (ng/ml)"
     , ylim=range(0,80))
points(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], pch = 16, cex=0.8)
lines(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], col="black", lwd=1)
```

```
abline(40, 0, col="red", lty=2) #abline(a,b): y=a+b*x
legend("topright", legend=c("Individual concentrations")
      , lty=1, col="black")

dev.off()
```

```
## cairo_pdf
##          2
```

1.4.2 PNG graphics devices

```
png("PK_of_Drug_X.png")

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], main="PK of Drug X"
     , type="n", xlab="Time (h)", ylab="Concentration (ng/ml)"
     , ylim=range(0,80))
points(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], pch = 16, cex=0.8)
lines(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], col="black", lwd=1)
abline(40, 0, col="red", lty=2) #abline(a,b): y=a+b*x
legend("topright", legend=c("Individual concentrations")
      , lty=1, col="black")

dev.off()
```

```
## cairo_pdf
##          2
```


2

Data Import / Export

2017-03-29 배균섭 교수님 강의

이번 시간에는 자료를 불러오고 조작을 가한 뒤 저장하는 방법에 대해 알아보겠습니다.

2.1 Read.csv

```
setwd("D:/Rt")  
dir()  
mydata = read.csv("MyData2017.csv", as.is=TRUE)
```

2.2 Theoph 데이터

R에 기본적으로 들어있는 약동학 자료에 대해 살펴보겠습니다.

Theoph

##	Subject	Wt	Dose	Time	conc
## 1	1	79.6	4.02	0.00	0.74
## 2	1	79.6	4.02	0.25	2.84
## 3	1	79.6	4.02	0.57	6.57
## 4	1	79.6	4.02	1.12	10.50
## 5	1	79.6	4.02	2.02	9.66
## 6	1	79.6	4.02	3.82	8.58
## 7	1	79.6	4.02	5.10	8.36
## 8	1	79.6	4.02	7.03	7.47
## 9	1	79.6	4.02	9.05	6.89
## 10	1	79.6	4.02	12.12	5.94
## 11	1	79.6	4.02	24.37	3.28
## 12	2	72.4	4.40	0.00	0.00
## 13	2	72.4	4.40	0.27	1.72
## 14	2	72.4	4.40	0.52	7.91
## 15	2	72.4	4.40	1.00	8.31
## 16	2	72.4	4.40	1.92	8.33
## 17	2	72.4	4.40	3.50	6.85
## 18	2	72.4	4.40	5.02	6.08
## 19	2	72.4	4.40	7.03	5.40
## 20	2	72.4	4.40	9.00	4.55
## 21	2	72.4	4.40	12.00	3.01
## 22	2	72.4	4.40	24.30	0.90
## 23	3	70.5	4.53	0.00	0.00
## 24	3	70.5	4.53	0.27	4.40
## 25	3	70.5	4.53	0.58	6.90
## 26	3	70.5	4.53	1.02	8.20
## 27	3	70.5	4.53	2.02	7.80
## 28	3	70.5	4.53	3.62	7.50
## 29	3	70.5	4.53	5.08	6.20
## 30	3	70.5	4.53	7.07	5.30
## 31	3	70.5	4.53	9.00	4.90
## 32	3	70.5	4.53	12.15	3.70
## 33	3	70.5	4.53	24.17	1.05
## 34	4	72.7	4.40	0.00	0.00
## 35	4	72.7	4.40	0.35	1.89
## 36	4	72.7	4.40	0.60	4.60

## 37	4	72.7	4.40	1.07	8.60
## 38	4	72.7	4.40	2.13	8.38
## 39	4	72.7	4.40	3.50	7.54
## 40	4	72.7	4.40	5.02	6.88
## 41	4	72.7	4.40	7.02	5.78
## 42	4	72.7	4.40	9.02	5.33
## 43	4	72.7	4.40	11.98	4.19
## 44	4	72.7	4.40	24.65	1.15
## 45	5	54.6	5.86	0.00	0.00
## 46	5	54.6	5.86	0.30	2.02
## 47	5	54.6	5.86	0.52	5.63
## 48	5	54.6	5.86	1.00	11.40
## 49	5	54.6	5.86	2.02	9.33
## 50	5	54.6	5.86	3.50	8.74
## 51	5	54.6	5.86	5.02	7.56
## 52	5	54.6	5.86	7.02	7.09
## 53	5	54.6	5.86	9.10	5.90
## 54	5	54.6	5.86	12.00	4.37
## 55	5	54.6	5.86	24.35	1.57
## 56	6	80.0	4.00	0.00	0.00
## 57	6	80.0	4.00	0.27	1.29
## 58	6	80.0	4.00	0.58	3.08
## 59	6	80.0	4.00	1.15	6.44
## 60	6	80.0	4.00	2.03	6.32
## 61	6	80.0	4.00	3.57	5.53
## 62	6	80.0	4.00	5.00	4.94
## 63	6	80.0	4.00	7.00	4.02
## 64	6	80.0	4.00	9.22	3.46
## 65	6	80.0	4.00	12.10	2.78
## 66	6	80.0	4.00	23.85	0.92
## 67	7	64.6	4.95	0.00	0.15
## 68	7	64.6	4.95	0.25	0.85
## 69	7	64.6	4.95	0.50	2.35
## 70	7	64.6	4.95	1.02	5.02
## 71	7	64.6	4.95	2.02	6.58
## 72	7	64.6	4.95	3.48	7.09
## 73	7	64.6	4.95	5.00	6.66

## 74	7	64.6	4.95	6.98	5.25
## 75	7	64.6	4.95	9.00	4.39
## 76	7	64.6	4.95	12.05	3.53
## 77	7	64.6	4.95	24.22	1.15
## 78	8	70.5	4.53	0.00	0.00
## 79	8	70.5	4.53	0.25	3.05
## 80	8	70.5	4.53	0.52	3.05
## 81	8	70.5	4.53	0.98	7.31
## 82	8	70.5	4.53	2.02	7.56
## 83	8	70.5	4.53	3.53	6.59
## 84	8	70.5	4.53	5.05	5.88
## 85	8	70.5	4.53	7.15	4.73
## 86	8	70.5	4.53	9.07	4.57
## 87	8	70.5	4.53	12.10	3.00
## 88	8	70.5	4.53	24.12	1.25
## 89	9	86.4	3.10	0.00	0.00
## 90	9	86.4	3.10	0.30	7.37
## 91	9	86.4	3.10	0.63	9.03
## 92	9	86.4	3.10	1.05	7.14
## 93	9	86.4	3.10	2.02	6.33
## 94	9	86.4	3.10	3.53	5.66
## 95	9	86.4	3.10	5.02	5.67
## 96	9	86.4	3.10	7.17	4.24
## 97	9	86.4	3.10	8.80	4.11
## 98	9	86.4	3.10	11.60	3.16
## 99	9	86.4	3.10	24.43	1.12
## 100	10	58.2	5.50	0.00	0.24
## 101	10	58.2	5.50	0.37	2.89
## 102	10	58.2	5.50	0.77	5.22
## 103	10	58.2	5.50	1.02	6.41
## 104	10	58.2	5.50	2.05	7.83
## 105	10	58.2	5.50	3.55	10.21
## 106	10	58.2	5.50	5.05	9.18
## 107	10	58.2	5.50	7.08	8.02
## 108	10	58.2	5.50	9.38	7.14
## 109	10	58.2	5.50	12.10	5.68
## 110	10	58.2	5.50	23.70	2.42

```
## 111      11 65.0 4.92  0.00  0.00
## 112      11 65.0 4.92  0.25  4.86
## 113      11 65.0 4.92  0.50  7.24
## 114      11 65.0 4.92  0.98  8.00
## 115      11 65.0 4.92  1.98  6.81
## 116      11 65.0 4.92  3.60  5.87
## 117      11 65.0 4.92  5.02  5.22
## 118      11 65.0 4.92  7.03  4.45
## 119      11 65.0 4.92  9.03  3.62
## 120      11 65.0 4.92 12.12  2.69
## 121      11 65.0 4.92 24.08  0.86
## 122      12 60.5 5.30  0.00  0.00
## 123      12 60.5 5.30  0.25  1.25
## 124      12 60.5 5.30  0.50  3.96
## 125      12 60.5 5.30  1.00  7.82
## 126      12 60.5 5.30  2.00  9.72
## 127      12 60.5 5.30  3.52  9.75
## 128      12 60.5 5.30  5.07  8.57
## 129      12 60.5 5.30  7.07  6.59
## 130      12 60.5 5.30  9.03  6.11
## 131      12 60.5 5.30 12.05  4.57
## 132      12 60.5 5.30 24.15  1.17
```

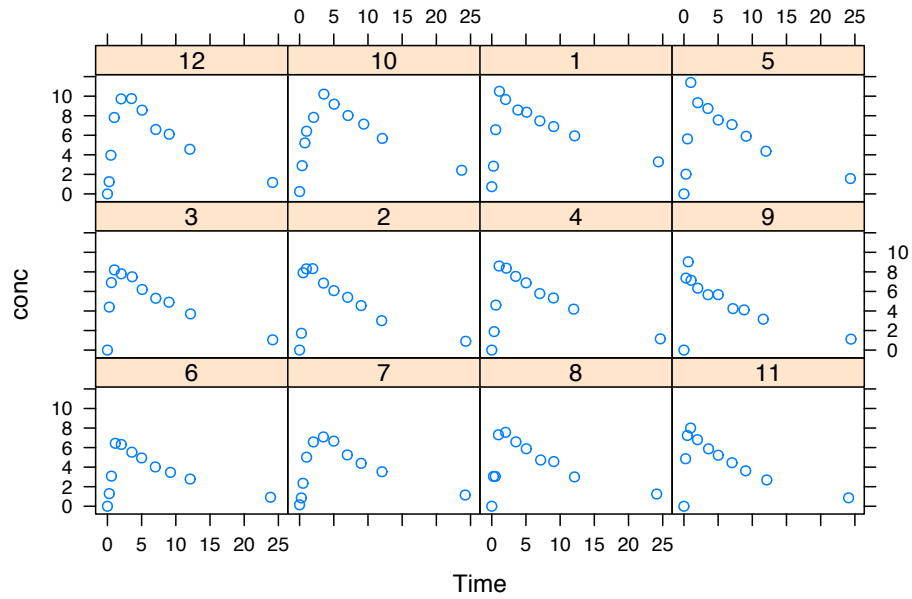
R console에서 ?Theoph를 타이핑 치면 좀 더 자세한 정보를 얻을 수 있습니다.

2.3 *lattice*

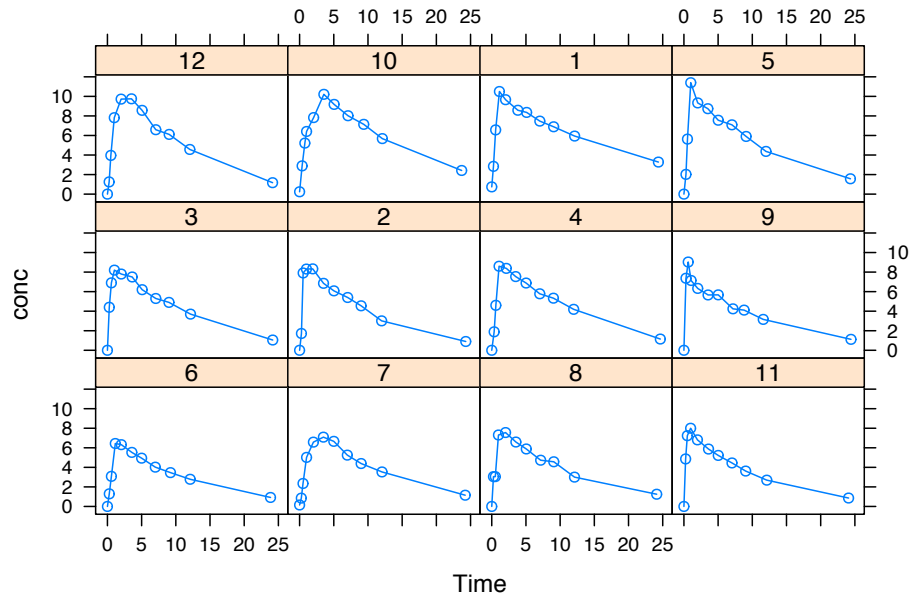
lattice 패키지를 불러온 뒤 그림을 그려보겠습니다. (Sarkar, 2017)

```
library(lattice) # trellis

xyplot(conc ~ Time | Subject, data=Theoph)
```

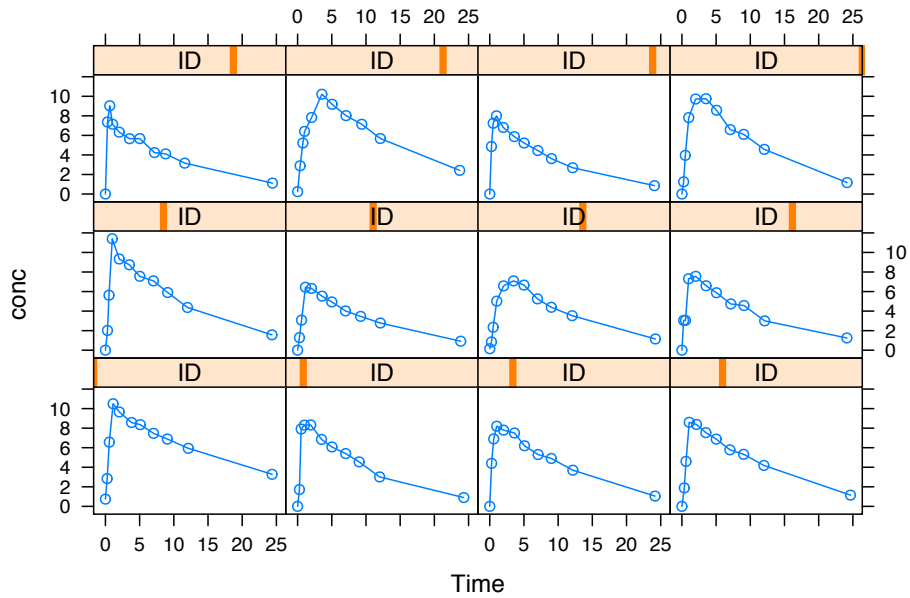


```
xyplot(conc ~ Time | Subject, data=Theoph, type="b")
```

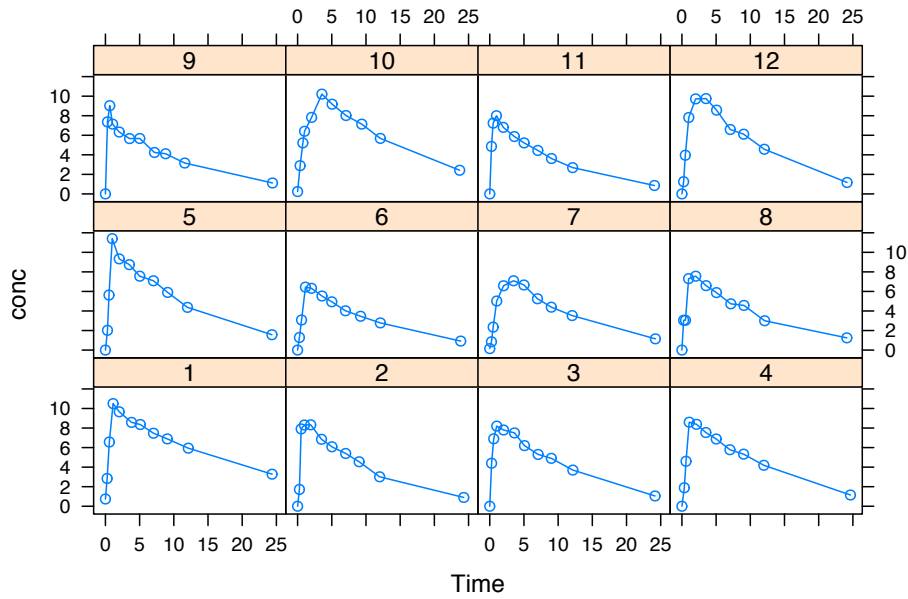


```
Theoph[, "ID"] = as.numeric(as.character(Theoph[, "Subject"]))
```

```
xyplot(conc ~ Time | ID, data=Theoph, type="b")
```



```
xyplot(conc ~ Time | as.factor(ID), data=Theoph, type="b")
```



```
write.csv(Theoph, "Theoph.csv", row.names=FALSE, quote=FALSE, na="")
```

2.4 Subsetting and write.csv

자료를 편집하고, subset을 만들고 각각을 파일로 저장하는 방법에 대해 알아보겠습니다.

```
IDs = sort(unique(Theoph[, "ID"])) ; IDs
```

```
## [1] 1 2 3 4 5 6 7 8 9 10 11 12
```

```
nID = length(IDs) ; nID
```

```
## [1] 12
```



```

demog = unique(Theoph[,c("ID","Wt")])
colnames(demog) = c("ID", "BWT")
write.csv(demog, "1-demog.csv", row.names=FALSE, quote=FALSE, na="")

DV = Theoph[,c("ID","Time", "conc")]
colnames(DV) = c("ID", "TIME", "DV")
write.csv(DV, "3-DV.csv", row.names=FALSE, quote=FALSE, na="")

adm = cbind(IDs, rep(0, nID), rep(320, nID))
colnames(adm) = c("ID", "TIME", "AMT")
write.csv(adm, "2-adm.csv", row.names=FALSE, quote=FALSE, na="")

demog = read.csv("1-demog.csv", as.is=TRUE)
adm = read.csv("2-adm.csv", as.is=TRUE)
dv = read.csv("3-dv.csv", as.is=TRUE)

AdmDv = merge(adm, dv, by=intersect(colnames(adm), colnames(dv)), all=TRUE)
AdmDv

```

```

##      ID  TIME AMT   DV
## 1    1  0.00 320  0.74
## 2    1  0.25  NA  2.84
## 3    1  0.57  NA  6.57
## 4    1  1.12  NA 10.50
## 5    1  2.02  NA  9.66
## 6    1  3.82  NA  8.58
## 7    1  5.10  NA  8.36
## 8    1  7.03  NA  7.47
## 9    1  9.05  NA  6.89
## 10   1 12.12  NA  5.94
## 11   1 24.37  NA  3.28
## 12   2  0.00 320  0.00
## 13   2  0.27  NA  1.72
## 14   2  0.52  NA  7.91
## 15   2  1.00  NA  8.31
## 16   2  1.92  NA  8.33

```

## 17	2	3.50	NA	6.85
## 18	2	5.02	NA	6.08
## 19	2	7.03	NA	5.40
## 20	2	9.00	NA	4.55
## 21	2	12.00	NA	3.01
## 22	2	24.30	NA	0.90
## 23	3	0.00	320	0.00
## 24	3	0.27	NA	4.40
## 25	3	0.58	NA	6.90
## 26	3	1.02	NA	8.20
## 27	3	2.02	NA	7.80
## 28	3	3.62	NA	7.50
## 29	3	5.08	NA	6.20
## 30	3	7.07	NA	5.30
## 31	3	9.00	NA	4.90
## 32	3	12.15	NA	3.70
## 33	3	24.17	NA	1.05
## 34	4	0.00	320	0.00
## 35	4	0.35	NA	1.89
## 36	4	0.60	NA	4.60
## 37	4	1.07	NA	8.60
## 38	4	2.13	NA	8.38
## 39	4	3.50	NA	7.54
## 40	4	5.02	NA	6.88
## 41	4	7.02	NA	5.78
## 42	4	9.02	NA	5.33
## 43	4	11.98	NA	4.19
## 44	4	24.65	NA	1.15
## 45	5	0.00	320	0.00
## 46	5	0.30	NA	2.02
## 47	5	0.52	NA	5.63
## 48	5	1.00	NA	11.40
## 49	5	2.02	NA	9.33
## 50	5	3.50	NA	8.74
## 51	5	5.02	NA	7.56
## 52	5	7.02	NA	7.09
## 53	5	9.10	NA	5.90

```
## 54  5 12.00  NA  4.37
## 55  5 24.35  NA  1.57
## 56  6  0.00 320  0.00
## 57  6  0.27  NA  1.29
## 58  6  0.58  NA  3.08
## 59  6  1.15  NA  6.44
## 60  6  2.03  NA  6.32
## 61  6  3.57  NA  5.53
## 62  6  5.00  NA  4.94
## 63  6  7.00  NA  4.02
## 64  6  9.22  NA  3.46
## 65  6 12.10  NA  2.78
## 66  6 23.85  NA  0.92
## 67  7  0.00 320  0.15
## 68  7  0.25  NA  0.85
## 69  7  0.50  NA  2.35
## 70  7  1.02  NA  5.02
## 71  7  2.02  NA  6.58
## 72  7  3.48  NA  7.09
## 73  7  5.00  NA  6.66
## 74  7  6.98  NA  5.25
## 75  7  9.00  NA  4.39
## 76  7 12.05  NA  3.53
## 77  7 24.22  NA  1.15
## 78  8  0.00 320  0.00
## 79  8  0.25  NA  3.05
## 80  8  0.52  NA  3.05
## 81  8  0.98  NA  7.31
## 82  8  2.02  NA  7.56
## 83  8  3.53  NA  6.59
## 84  8  5.05  NA  5.88
## 85  8  7.15  NA  4.73
## 86  8  9.07  NA  4.57
## 87  8 12.10  NA  3.00
## 88  8 24.12  NA  1.25
## 89  9  0.00 320  0.00
## 90  9  0.30  NA  7.37
```

```
## 91  9  0.63  NA  9.03
## 92  9  1.05  NA  7.14
## 93  9  2.02  NA  6.33
## 94  9  3.53  NA  5.66
## 95  9  5.02  NA  5.67
## 96  9  7.17  NA  4.24
## 97  9  8.80  NA  4.11
## 98  9 11.60  NA  3.16
## 99  9 24.43  NA  1.12
## 100 10  0.00 320  0.24
## 101 10  0.37  NA  2.89
## 102 10  0.77  NA  5.22
## 103 10  1.02  NA  6.41
## 104 10  2.05  NA  7.83
## 105 10  3.55  NA 10.21
## 106 10  5.05  NA  9.18
## 107 10  7.08  NA  8.02
## 108 10  9.38  NA  7.14
## 109 10 12.10  NA  5.68
## 110 10 23.70  NA  2.42
## 111 11  0.00 320  0.00
## 112 11  0.25  NA  4.86
## 113 11  0.50  NA  7.24
## 114 11  0.98  NA  8.00
## 115 11  1.98  NA  6.81
## 116 11  3.60  NA  5.87
## 117 11  5.02  NA  5.22
## 118 11  7.03  NA  4.45
## 119 11  9.03  NA  3.62
## 120 11 12.12  NA  2.69
## 121 11 24.08  NA  0.86
## 122 12  0.00 320  0.00
## 123 12  0.25  NA  1.25
## 124 12  0.50  NA  3.96
## 125 12  1.00  NA  7.82
## 126 12  2.00  NA  9.72
## 127 12  3.52  NA  9.75
```

```
## 128 12 5.07 NA 8.57
## 129 12 7.07 NA 6.59
## 130 12 9.03 NA 6.11
## 131 12 12.05 NA 4.57
## 132 12 24.15 NA 1.17
```

자료를 병합(merge)해 보겠습니다.

```
DataAll = merge(demog, AdmDv, by=c("ID"), all=TRUE)
DataAll
```

```
##      ID  BWT  TIME AMT    DV
## 1    1 79.6  0.00 320  0.74
## 2    1 79.6  0.25  NA  2.84
## 3    1 79.6  0.57  NA  6.57
## 4    1 79.6  1.12  NA 10.50
## 5    1 79.6  2.02  NA  9.66
## 6    1 79.6  3.82  NA  8.58
## 7    1 79.6  5.10  NA  8.36
## 8    1 79.6  7.03  NA  7.47
## 9    1 79.6  9.05  NA  6.89
## 10   1 79.6 12.12  NA  5.94
## 11   1 79.6 24.37  NA  3.28
## 12   2 72.4  0.00 320  0.00
## 13   2 72.4  0.27  NA  1.72
## 14   2 72.4  0.52  NA  7.91
## 15   2 72.4  1.00  NA  8.31
## 16   2 72.4  1.92  NA  8.33
## 17   2 72.4  3.50  NA  6.85
## 18   2 72.4  5.02  NA  6.08
## 19   2 72.4  7.03  NA  5.40
## 20   2 72.4  9.00  NA  4.55
## 21   2 72.4 12.00  NA  3.01
## 22   2 72.4 24.30  NA  0.90
## 23   3 70.5  0.00 320  0.00
## 24   3 70.5  0.27  NA  4.40
## 25   3 70.5  0.58  NA  6.90
## 26   3 70.5  1.02  NA  8.20
```

## 27	3	70.5	2.02	NA	7.80
## 28	3	70.5	3.62	NA	7.50
## 29	3	70.5	5.08	NA	6.20
## 30	3	70.5	7.07	NA	5.30
## 31	3	70.5	9.00	NA	4.90
## 32	3	70.5	12.15	NA	3.70
## 33	3	70.5	24.17	NA	1.05
## 34	4	72.7	0.00	320	0.00
## 35	4	72.7	0.35	NA	1.89
## 36	4	72.7	0.60	NA	4.60
## 37	4	72.7	1.07	NA	8.60
## 38	4	72.7	2.13	NA	8.38
## 39	4	72.7	3.50	NA	7.54
## 40	4	72.7	5.02	NA	6.88
## 41	4	72.7	7.02	NA	5.78
## 42	4	72.7	9.02	NA	5.33
## 43	4	72.7	11.98	NA	4.19
## 44	4	72.7	24.65	NA	1.15
## 45	5	54.6	0.00	320	0.00
## 46	5	54.6	0.30	NA	2.02
## 47	5	54.6	0.52	NA	5.63
## 48	5	54.6	1.00	NA	11.40
## 49	5	54.6	2.02	NA	9.33
## 50	5	54.6	3.50	NA	8.74
## 51	5	54.6	5.02	NA	7.56
## 52	5	54.6	7.02	NA	7.09
## 53	5	54.6	9.10	NA	5.90
## 54	5	54.6	12.00	NA	4.37
## 55	5	54.6	24.35	NA	1.57
## 56	6	80.0	0.00	320	0.00
## 57	6	80.0	0.27	NA	1.29
## 58	6	80.0	0.58	NA	3.08
## 59	6	80.0	1.15	NA	6.44
## 60	6	80.0	2.03	NA	6.32
## 61	6	80.0	3.57	NA	5.53
## 62	6	80.0	5.00	NA	4.94
## 63	6	80.0	7.00	NA	4.02

```
## 64  6 80.0  9.22  NA  3.46
## 65  6 80.0 12.10  NA  2.78
## 66  6 80.0 23.85  NA  0.92
## 67  7 64.6  0.00 320  0.15
## 68  7 64.6  0.25  NA  0.85
## 69  7 64.6  0.50  NA  2.35
## 70  7 64.6  1.02  NA  5.02
## 71  7 64.6  2.02  NA  6.58
## 72  7 64.6  3.48  NA  7.09
## 73  7 64.6  5.00  NA  6.66
## 74  7 64.6  6.98  NA  5.25
## 75  7 64.6  9.00  NA  4.39
## 76  7 64.6 12.05  NA  3.53
## 77  7 64.6 24.22  NA  1.15
## 78  8 70.5  0.00 320  0.00
## 79  8 70.5  0.25  NA  3.05
## 80  8 70.5  0.52  NA  3.05
## 81  8 70.5  0.98  NA  7.31
## 82  8 70.5  2.02  NA  7.56
## 83  8 70.5  3.53  NA  6.59
## 84  8 70.5  5.05  NA  5.88
## 85  8 70.5  7.15  NA  4.73
## 86  8 70.5  9.07  NA  4.57
## 87  8 70.5 12.10  NA  3.00
## 88  8 70.5 24.12  NA  1.25
## 89  9 86.4  0.00 320  0.00
## 90  9 86.4  0.30  NA  7.37
## 91  9 86.4  0.63  NA  9.03
## 92  9 86.4  1.05  NA  7.14
## 93  9 86.4  2.02  NA  6.33
## 94  9 86.4  3.53  NA  5.66
## 95  9 86.4  5.02  NA  5.67
## 96  9 86.4  7.17  NA  4.24
## 97  9 86.4  8.80  NA  4.11
## 98  9 86.4 11.60  NA  3.16
## 99  9 86.4 24.43  NA  1.12
## 100 10 58.2  0.00 320  0.24
```

```
## 101 10 58.2 0.37 NA 2.89
## 102 10 58.2 0.77 NA 5.22
## 103 10 58.2 1.02 NA 6.41
## 104 10 58.2 2.05 NA 7.83
## 105 10 58.2 3.55 NA 10.21
## 106 10 58.2 5.05 NA 9.18
## 107 10 58.2 7.08 NA 8.02
## 108 10 58.2 9.38 NA 7.14
## 109 10 58.2 12.10 NA 5.68
## 110 10 58.2 23.70 NA 2.42
## 111 11 65.0 0.00 320 0.00
## 112 11 65.0 0.25 NA 4.86
## 113 11 65.0 0.50 NA 7.24
## 114 11 65.0 0.98 NA 8.00
## 115 11 65.0 1.98 NA 6.81
## 116 11 65.0 3.60 NA 5.87
## 117 11 65.0 5.02 NA 5.22
## 118 11 65.0 7.03 NA 4.45
## 119 11 65.0 9.03 NA 3.62
## 120 11 65.0 12.12 NA 2.69
## 121 11 65.0 24.08 NA 0.86
## 122 12 60.5 0.00 320 0.00
## 123 12 60.5 0.25 NA 1.25
## 124 12 60.5 0.50 NA 3.96
## 125 12 60.5 1.00 NA 7.82
## 126 12 60.5 2.00 NA 9.72
## 127 12 60.5 3.52 NA 9.75
## 128 12 60.5 5.07 NA 8.57
## 129 12 60.5 7.07 NA 6.59
## 130 12 60.5 9.03 NA 6.11
## 131 12 60.5 12.05 NA 4.57
## 132 12 60.5 24.15 NA 1.17
```


Part II

Part II. Basics



3

Let's start!

기본적인 R의 사용법에 대해 알아보도록 하겠습니다. 우선 가장 중요한 것은 적절한 `dataset`을 설정하는 것입니다. 약동학(pharmacokinetics)과 관련된 두개의 유명한 `dataset`이 R 자체적으로 내장되어 있습니다. Theoph와 Indometh가 그 예입니다.

그러나 여기서는 좀더 다양한 분석을 하기 위해서 다음과 같은 `dataset`이 들어있는 패키지를 사용하도록 하겠습니다.

```
# 패키지 설치
install.packages("devtools")
devtools::install_github("dpastoor/PKPDdatasets")

# Install the yarrrr package
install.packages('yarrrr')

# Load the package
library(yarrrr)
```

이제 패키지를 불러옵니다.

```
library(PKPDdatasets)
library(yarrrr)

## Loading required package: jpeg
## Loading required package: BayesFactor
## Loading required package: coda
## Loading required package: Matrix
## *****
```

```
## Welcome to BayesFactor 0.9.12-2. If you have questions, please contact Richard Morey (richard@rice.edu)
##
## Type BFManual() to open the manual.
## *****

## yarr v0.1.4. Citation info at citation('yarr'). Package guide at yarr.guide()

## Email me at Nathaniel.D.Phillips.is@gmail.com
```

우리가 사용할 데이터셋은 이것입니다.

IV and oral pharmacokinetic data for daptomycin

Description

A dataset containing simulated dapagliflozin PK. A single IV dose followed by 3 escalating oral doses

ID. Numerical ID (1-24)

TIME. Nominal Time after first dose (hrs)

TAD. Time After Dose (hrs)

COBS. Observed Concentration (ug/L)

AMT_IV. IV dose amount when given (ug)

AMT_ORAL. Oral dose amount when given (ug)

OCC. Occasion, associated with each dosing event

AGE. Age (years)

WEIGHT Weight (kg)

GENDER. Gender flag (female=1, male=0)

FORMULATION Formulation associated with dose (IV or oral)

Usage

data(dapa_IV_oral) Format

A data frame with 1536 rows and 11 variables

What's the first exercise on the first day of pirate swimming lessons? While it would be cute if they all had little inflatable pirate ships to swim around in

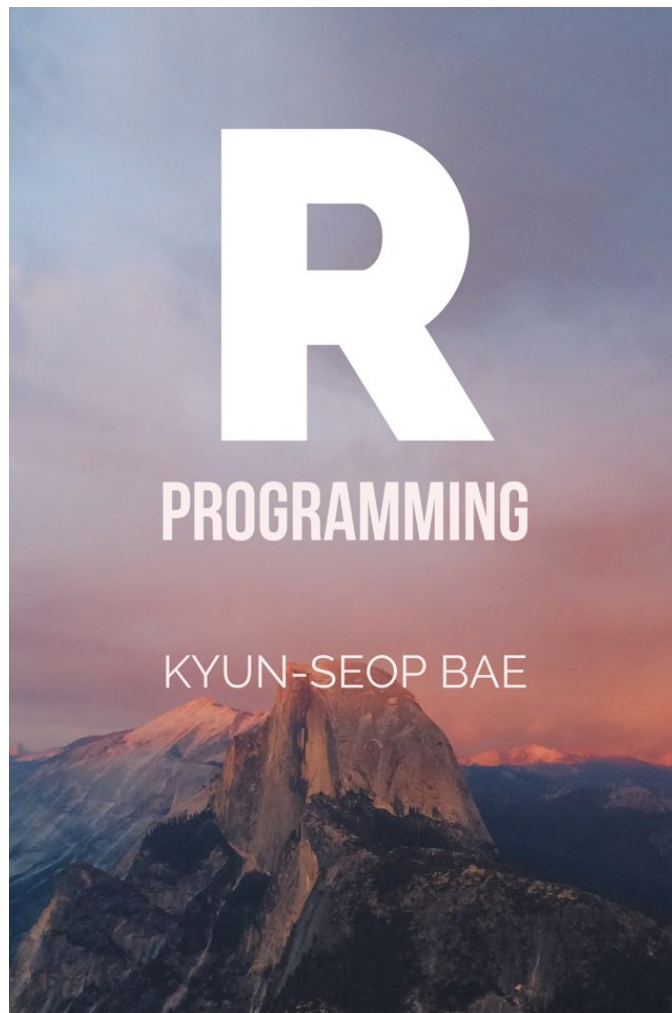


FIGURE 3.1: Despite what you might find at family friendly waterparks – this is NOT how real pirate swimming lessons look.

– unfortunately this is not the case. Instead, those baby `aht_trial2` take a walk off their baby planks so they can get a taste of what they're in for. Turns out, learning R is the same way. Let's jump in. In this chapter, you'll see how easy it is to calculate basic statistics and create plots in R. Don't worry if the code you're running doesn't make immediate sense – just marvel at how easy it is to do this in R!

In this section, we'll analyze a dataset called...wait for it...`aht_trial2`! The dataset contains data from a survey of 1,000 `aht_trial2`. The data is contained in the `yarr` package, so make sure you've installed and loaded the package:

3.1 Exploring data

Next, we'll look at the help menu for the `aht_trial2` dataset using the question mark `?aht_trial2`. When you run this, you should see a small help window open up in RStudio that gives you some information about the dataset.

```
?aht_trial2
```

First, let's take a look at the first few rows of the dataset using the `head()` function. This will show you the first few rows of the data.

```
# Look at the first few rows of the data
head(aht_trial2)
##   ID AGE WEIGHT SCREATININE SERUMALT GENDER  RACE
## 1  1 44.2  63.62         1.12    16.22   Male Caucasian
## 2  2 47.2  67.13         1.10    16.03  Female  Hispanic
## 3  3 44.5  58.14         1.03    15.64  Female    Black
## 4  4 50.8  68.80         0.92    15.44   Male Caucasian
## 5  5 44.0  63.66         1.02    18.55  Female    Other
## 6  6 47.6  71.62         1.02    18.13  Female Caucasian
##  DELDBP DELSBP      TRT DOSE
## 1   4.04 -28.48 Placebo    0
## 2 -16.79 -25.32 Placebo    0
```

```
## 3 -10.01 -49.50 Placebo    0
## 4 -13.61   2.33 Placebo    0
## 5 -15.79 -24.01 Placebo    0
## 6  -4.15 -45.69 Placebo    0
```

You can look at the names of the columns in the dataset with the `names()` function

```
# What are the names of the columns?
names(aht_trial2)
## [1] "ID"          "AGE"          "WEIGHT"
## [4] "SCREATININE" "SERUMALT"     "GENDER"
## [7] "RACE"        "DELDDBP"     "DELSBP"
## [10] "TRT"         "DOSE"
```

Finally, you can also view the entire dataset in a separate window using the `View()` function:

```
# View the entire dataset in a new window
View(aht_trial2)
```

3.2 Descriptive statistics

Now let's calculate some basic statistics on the entire dataset. We'll calculate the mean age, maximum height, and number of `aht_trial2` of each sex:

```
# What is the mean age?
mean(aht_trial2$AGE)
## [1] 49.72

# What was the tallest pirate?
max(aht_trial2$WEIGHT)
```

```
## [1] 87.05

# How many aht_trial2 are there of each sex?
table(aht_trial2$RACE)
##
##      Asian      Black Caucasian  Hispanic      Other
##         11         23         47          8         11
```

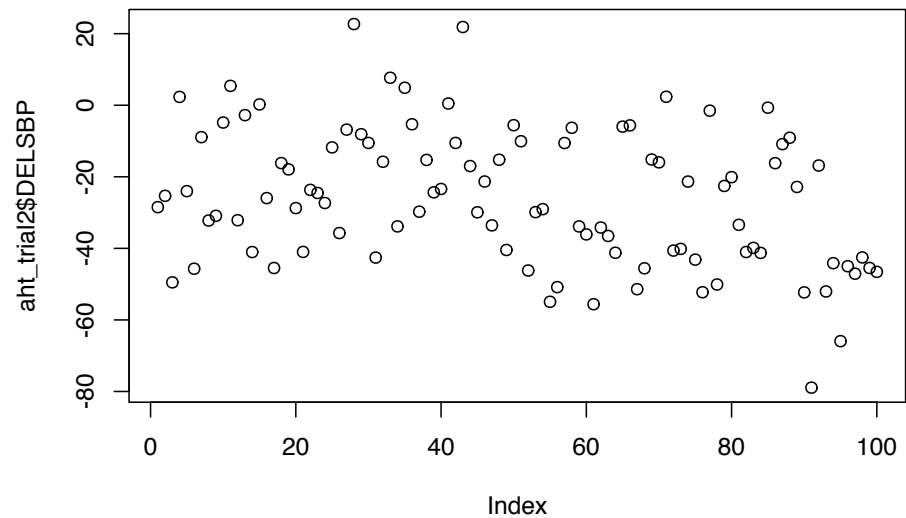
Now, let's calculate statistics for different groups of aht_trial2. For example, the following code will use the `aggregate()` function to calculate the mean age of aht_trial2, separately for each sex.

```
# Calculate the mean age, separately for each sex
aggregate(formula = DBP ~ RACE,
           data = aht_trial2,
           FUN = mean)
## Error in eval(expr, envir, enclos): 객체 'DBP'를 찾을 수 없습니다
```

3.3 Plotting

Cool stuff, now let's make a plot! We'll plot the relationship between pirate's height and weight using the `plot()` function

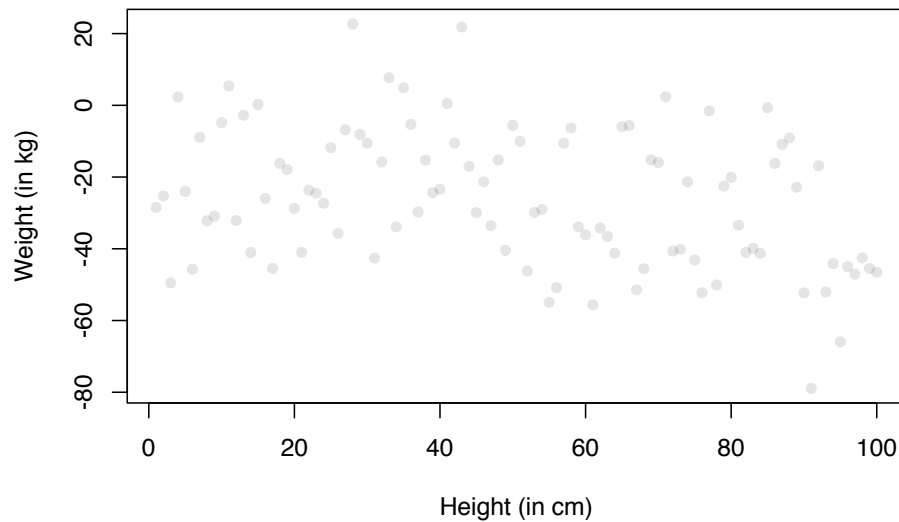
```
# Create scatterplot
plot(x = aht_trial2$DELSBP,      # X coordinates
     y = aht_trial2$DBP)        # y-coordinates
```

Now let's make a fancier version of the same plot by adding some customization

```
# Create scatterplot
plot(x = aht_trial2$DELSBP,      # X coordinates
     y = aht_trial2$DBP,        # y-coordinates
     main = 'My first scatterplot of pirate data!',
     xlab = 'Height (in cm)',    # x-axis label
     ylab = 'Weight (in kg)',    # y-axis label
     pch = 16,                  # Filled circles
     col = gray(.0, .1))        # Transparent gray
```

My first scatterplot of pirate data!

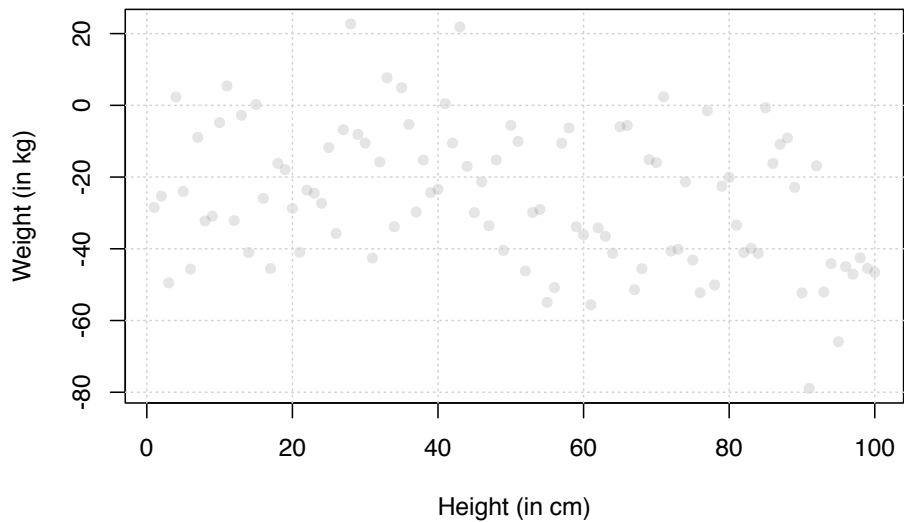


Now let's make it even better by adding gridlines and a blue regression line to measure the strength of the relationship.

```
# Create scatterplot
plot(x = aht_trial2$DELSBP,      # X coordinates
     y = aht_trial2$DBP,        # y-coordinates
     main = 'My first scatterplot of pirate data!',
     xlab = 'Height (in cm)',    # x-axis label
     ylab = 'Weight (in kg)',    # y-axis label
     pch = 16,                  # Filled circles
     col = gray(.0, .1))        # Transparent gray

grid()                          # Add gridlines
```

My first scatterplot of pirate data!



```
# Create a linear regression model
model <- lm(formula = DBP ~ DELSBP, # 순서에 조심!
            data = aht_trial2)
## Error in eval(expr, envir, enclos): 객체 'DBP'를 찾을 수 없습니다

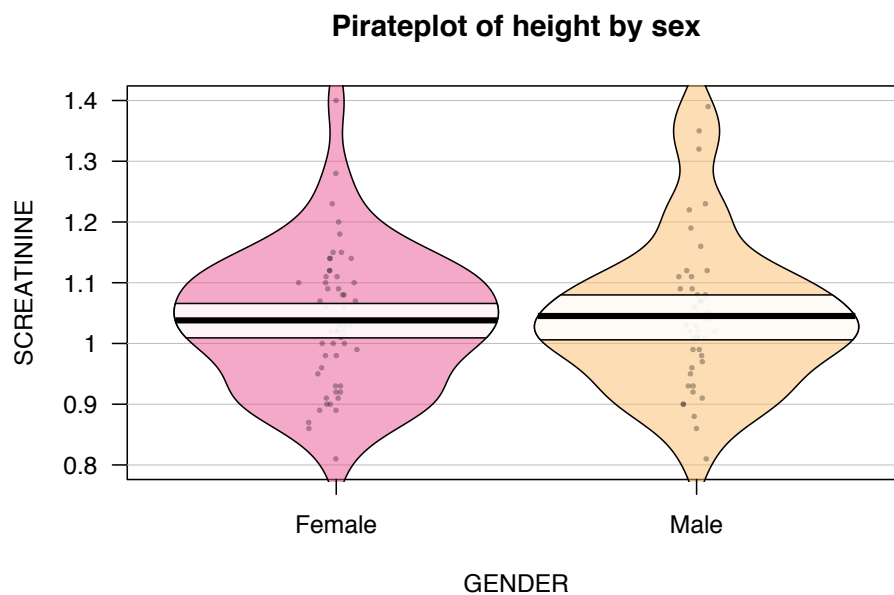
abline(model, col = 'blue') # Add regression to plot
## Error in abline(model, col = "blue"): 객체 'model'를 찾을 수 없습니다
```

Scatterplots are great for showing the relationship between two continuous variables, but what if your independent variable is not continuous? In this case, pirateplots are a good option. Let's create a pirateplot using the `pirateplot()` function to show the distribution of pirate's age based on their favorite sword:

```
pirateplot(formula = DBP ~ RACE,
            data = aht_trial2,
            main = "Pirateplot of ages by favorite sword")
## Error in eval(expr, envir, enclos): 객체 'DBP'를 찾을 수 없습니다
```

Now let's make another pirateplot showing the relationship between sex and height using a different plotting theme and the "pony" color palette:

```
pirateplot(formula = SREATININE ~ GENDER, # Plot weight as a function of sex
            data = aht_trial2,
            main = "Pirateplot of height by sex",
            pal = "pony", # Use the info color palette
            theme = 3) # Use theme 3
```



3.4 Hypothesis tests

Now, let's do some basic hypothesis tests. First, let's conduct a two-sample t-test to see if there is a significant difference between the ages of aht_trial2 who do wear a headband, and those who do not:

```
# Age by headband t-test
t.test(formula = DELSBP ~ GENDER,
```

```

data = aht_trial2,
alternative = 'two.sided')
##
## Welch Two Sample t-test
##
## data:  DELSBP by GENDER
## t = 0.83, df = 87, p-value = 0.4
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -4.557 11.056
## sample estimates:
## mean in group Female    mean in group Male
##                -24.61                -27.86

```

With a p-value of 0.7259, we don't have sufficient evidence say there is a difference in the men age of aht_trial2 who wear headbands and those that do not.

Next, let's test if there a significant correlation between a pirate's height and weight using the `cor.test()` function:

```

cor.test(formula = ~ DELSBP + DELDBP,
data = aht_trial2)
##
## Pearson's product-moment correlation
##
## data:  DELSBP and DELDBP
## t = 0.48, df = 98, p-value = 0.6
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.1489 0.2430
## sample estimates:
##      cor
## 0.04891

```

We got a p-value of $p < 2.2e-16$, that's scientific notation for $p < .000000000000000016$ – which is pretty much 0. Thus, we'd conclude that

there is a significant (positive) relationship between a pirate's height and weight.

Now, let's do an ANOVA testing if there is a difference between the number of tattoos aht_trial2 have based on their favorite sword

```
# Create tattoos model
tat.sword.lm <- lm(formula = DELDBP ~ RACE,
                   data = aht_trial2)

# Get ANOVA table
anova(tat.sword.lm)
## Analysis of Variance Table
##
## Response: DELDBP
##          Df Sum Sq Mean Sq F value Pr(>F)
## RACE      4    810      202    1.05  0.39
## Residuals 95  18401      194
```

Sure enough, we see another very small p-value of $p < 2.2e-16$, suggesting that the number of tattoos pirate's have are different based on their favorite sword.

A

As-is R Files

교수님께서 주신 원본 R 파일입니다.

A.1 Lecture 3

```
#####  
##-----##  
##                Graphics                ##  
##-----##  
#####  
  
# 상위수준 그림 함수는 그림을 생성한다.  
# 하위수준 그림 함수는 기존의 그림에 그림을 추가한다.  
  
## 상위수준 그림 함수의 주요 인자 (arguments) ###  
  
# main : 제목  
# xlab/ylab : x축 및 y축 레이블  
# xlim/ylim : x축 및 y축 범위  
# col : 색깔  
# lty : 선 모양  
# pch : 점 모양  
# cex : 그림 성분의 크기  
# lwd : 선 굵기
```

```
# type : 그림 타입

#####
#####      상위수준 그림 함수      #####
#####

WD <- "D:\\AMC\\Education\\UU\\2017\\R\\Graphics\\"

setwd(WD)

dta <- read.csv("PK.csv")
head(dta)
str(dta)

##### scatter plot #####

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0])

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], log="y")

plot(dta$TIME[dta$MDV==0], log(dta$DV[dta$MDV==0]))

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0]
      , xlab="Time (hr)", ylab="Concentration (ng/mL)"
      , type="o", pch=2, col=1, main="PK time-course of Drug X"
      , xlim =c(-2,218), ylim=c(0,80))

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], axes=F,
      , xlab="Time (hr)", ylab="Concentration (ng/mL)"
      , type="o", pch=2, col=1, main="PK time-course of Drug X"
      , xlim =c(-2,218), ylim=c(0,80))
axis(1, at=seq(0, 218, 24))
axis(2)
box()
```



```
##### Histogram #####

d.demog <- read.csv("DEMOG.csv")

# histogram
hist(d.demog$HT)

hist(d.demog$HT, breaks=10)
hist(d.demog$HT, nclass=10)

# with density line
hist(d.demog$HT, probability=TRUE, breaks=10)
lines(density(d.demog$HT))

hist(d.demog$HT, probability=TRUE, breaks=9, xaxt="n",
      , main="Histogram for Height", xlab="Height (cm)", ylab="Probability (%)")
axis(1, at=seq(min(d.demog$HT), max(d.demog$HT), 3))
lines(density(d.demog$HT))

hist(d.demog$HT, probability=TRUE, breaks=9, xaxt="n",
      , main="Histogram for Height", xlab="Height (cm)", ylab="Probability (%)",
      , col = "lightblue", border = "pink")
axis(1, at=seq(min(d.demog$HT), max(d.demog$HT), 3))
lines(density(d.demog$HT))

##### Box-Whisker Plot #####

# Box-and-Whisker Plot

boxplot(d.demog$WT)
```

```

boxplot(d.demog$WT ~ d.demog$SEX)

boxplot(split(d.demog$WT, d.demog$SEX))

boxplot(WT ~ SEX, data=d.demog)

boxplot(d.demog$WT ~ d.demog$SEX
        , names=c("Male", "Female"), ylab="AGE, year", ylim=c(min(d.demog$WT)-2, max(d.demog$WT)+2),
        , col="pink")

boxplot(d.demog$WT ~ d.demog$SEX
        , names=c("Male", "Female"), ylab="AGE, year", ylim=c(min(d.demog$WT)-2, max(d.demog$WT)+2),
        , col=c("lightblue", "salmon"), width=c(0.6, 1))

#varwidth: if varwidth is TRUE, the boxes are drawn with widths proportional
#to the square-roots of the number of observations in the groups.

boxplot(d.demog$WT ~ d.demog$SEX
        , names=c("Male", "Female"), ylab="AGE, year", ylim=c(min(d.demog$WT)-2, max(d.demog$WT)+2),
        , col=c("lightblue", "salmon")
        , varwidth=TRUE)

##### Bar Plot #####

barplot(d.demog$HT)

VADeaths

barplot(VADeaths, border = "dark blue")

barplot(VADeaths, col = rainbow(20))

barplot(VADeaths, col = heat.colors(8))

```

```

barplot(VADeaths, col = gray.colors(4))

barplot(VADeaths, col = gray.colors(4), log="x")
barplot(VADeaths, col = gray.colors(4), log="y")
barplot(VADeaths, col = gray.colors(4), log="xy")

##### pie chart #####

drug.X.market <- c(0.12, 0.29, 0.32, 0.22, 0.11, 0.28)
names(drug.X.market) <- c("South Korea", "China", "USA", "Japan", "Austria", "EU")
pie(drug.X.market)

##### matplot 함수 #####

# matrix와 column 사이의 그림

pct.95 <- read.csv("pct95.csv")
matplot(pct.95[,1], pct.95[,2:ncol(pct.95)], pch=1)

matplot(pct.95[,1], pct.95[,2:ncol(pct.95)], pch=1, col=c(1,2,1), type="l", lty=1, lwd=c(1,2,1))

##### Scatter plot matrices (pairs plots) #####

pairs(d.demog)

#add a loess smoother, type:
pairs(d.demog, panel = panel.smooth)

panel.cor <- function(x, y, digits=2, prefix="", cex.cor)
{
  usr <- par("usr"); on.exit(par(usr))

```

```

    par(usr = c(0, 1, 0, 1))
    r = (cor(x, y))
    txt <- format(c(r, 0.123456789), digits=digits)[1]
    txt <- paste(prefix, txt, sep="")
    if(missing(cex.cor)) cex <- 1.5
    text(0.5, 0.5, txt, cex = 1.5)
  }

pairs(d.demog, lower.panel=panel.smooth, upper.panel=panel.cor)

#####
##          하위수준 그림 함수          ##
#####

# points : 점추가
# lines : 선 추가
# abline : 기준선 추가
# mtext : 텍스트 추가
# legend : 설명(legend) 추가
# polygon : polygon 추가

##### 점, 선, 설명 추가 하기 #####

plot(pct.95$TIME, pct.95$PCT50, main="PK of Drug X"
      , type="l", xlab="Time (h)", ylab="Concentration (ng/ml)"
      , ylim=range(0,80), lty=1, col="red", lwd=2)

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], main="PK of Drug X"
      , type="n", xlab="Time (h)", ylab="Concentration (ng/ml)"
      , ylim=range(0,80))
points(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], pch = 16, cex=0.8)
lines(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], col="black", lwd=1)

```

```

abline(40, 0, col="red", lty=2)                                #abline(a,b): y=a+b*x
legend("topright", legend=c("Individual concentrations")
      , lty=1, col="black")

##### polygon 함수 #####

plot(c(1, 10), c(1, 6), type = "n")
polygon(c(2,8,8,2), c(5,4,3,2), col="lightgreen")

plot(c(1, 9), 1:2, type = "n")
polygon(1:9, c(2,1,2,1,1,2,1,2,1),
       col = c("red", "blue"),
       border = c("green", "yellow"),
       lwd = 3, lty = c("dashed", "solid"))

##### 그림 출력하기 #####

#--pdf graphics devices
pdf("PK_of_Drug_X.pdf")

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], main="PK of Drug X"
     , type="n", xlab="Time (h)", ylab="Concentration (ng/ml)"
     , ylim=range(0,80))
points(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], pch = 16, cex=0.8)
lines(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], col="black", lwd=1)
abline(40, 0, col="red", lty=2)                                #abline(a,b): y=a+b*x
legend("topright", legend=c("Individual concentrations")
      , lty=1, col="black")

dev.off()

#--PNG graphics devices

```

```
png("PK_of_Drug_X.png")

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], main="PK of Drug X"
      , type="n", xlab="Time (h)", ylab="Concentration (ng/ml)"
      , ylim=range(0,80))
points(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], pch = 16, cex=0.8)
lines(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], col="black", lwd=1)
abline(40, 0, col="red", lty=2) #abline(a,b): y=a+b*x
legend("topright", legend=c("Individual concentrations")
      , lty=1, col="black")

dev.off()

#--Windows graphics devices
win.metafile("PK_of_Drug_X.wmf")

plot(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], main="PK of Drug X"
      , type="n", xlab="Time (h)", ylab="Concentration (ng/ml)"
      , ylim=range(0,80))
points(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], pch = 16, cex=0.8)
lines(dta$TIME[dta$MDV==0], dta$DV[dta$MDV==0], col="black", lwd=1)
abline(40, 0, col="red", lty=2) #abline(a,b): y=a+b*x
legend("topright", legend=c("Individual concentrations")
      , lty=1, col="black")

dev.off()
```

A.2 Lecture 4

```
# 2017-03-29

setwd("D:/Rt")
dir()

mydata = read.csv("MyData2017.csv", as.is=TRUE)

Theoph
library(lattice) # trellis

xyplot(conc ~ Time | Subject, data=Theoph)

xyplot(conc ~ Time | Subject, data=Theoph, type="b")

Theoph[, "ID"] = as.numeric(as.character(Theoph[, "Subject"]))

xyplot(conc ~ Time | ID, data=Theoph, type="b")

xyplot(conc ~ Time | as.factor(ID), data=Theoph, type="b")

write.csv(Theoph, "Theoph.csv", row.names=FALSE, quote=FALSE, na="")

IDs = sort(unique(Theoph[, "ID"])) ; IDs
nID = length(IDs) ; nID

demog = unique(Theoph[, c("ID", "Wt")])
colnames(demog) = c("ID", "BWT")
write.csv(demog, "1-demog.csv", row.names=FALSE, quote=FALSE, na="")

DV = Theoph[, c("ID", "Time", "conc")]
colnames(DV) = c("ID", "TIME", "DV")
```

```
write.csv(DV, "3-DV.csv", row.names=FALSE, quote=FALSE, na="")

adm = cbind(IDs, rep(0, nID), rep(320, nID))
colnames(adm) = c("ID", "TIME", "AMT")
write.csv(adm, "2-adm.csv", row.names=FALSE, quote=FALSE, na="")

demog = read.csv("1-demog.csv", as.is=TRUE)
adm = read.csv("2-adm.csv", as.is=TRUE)
dv = read.csv("3-dv.csv", as.is=TRUE)

AdmDv = merge(adm, dv, by=intersect(colnames(adm), colnames(dv)), all=TRUE)

DataAll = merge(demog, AdmDv, by=c("ID"), all=TRUE)
```


B

Using Coursera

PAGK에 보낸 이메일을 그대로 옮겼습니다.

배균섭 교수님의 추천을 받아 다음과 같은 강의와 책을 공유하고자 합니다. Coursera.com 에 유익한 R 강좌가 열렸습니다. “Mastering Software Development in R Specialization”이란 제목의 강좌인데 4개 Course를 무료로 들을 수 있게 되어있습니다. <https://www.coursera.org/specializations/r> 이것이 본래의 Link인데 여기서 각각의 Course를 찾거나 혹은 아래의 링크를 각각 클릭하여 하단에 나오는 “Audit”을 클릭하면 무료로 들을 수 있습니다.

Audit 버튼이 안보이신다고 하신 분들이 몇분 계셔서 첨언합니다. Coursera 회원가입하시고 로그인 한 뒤, Enroll Now를 누르시면 Audit 혹은 청강하기 라디오버튼을 보실 수 있습니다. 앱에서도 마찬가지입니다. 이외에도 코세라에는 많은 유익한 강의가 있는 것 같습니다. 다만 코스(Course)의 묶음인 “Specialization”에서는 유료등록(Enroll) 밖에 없으므로 Certificate가 필요하지 않다면, 각각의 코스를 구글검색 혹은 코세라 내에서 검색해서 “Audit(청강)” 하시면 무료로 강의를 들을 수 있습니다.

- <https://www.coursera.org/learn/r-programming-environment>
- <https://www.coursera.org/learn/advanced-r>
- <https://www.coursera.org/learn/r-packages>
- <https://www.coursera.org/learn/r-data-visualization>

<https://bookdown.org/rdpeng/RProgDA/> 이 링크는 무료로 공개된 강의 책자입니다. 강의를 듣지 않고 책으로 보고 싶으신 분은 참고하시면 됩니다.



C

R Tips

- Changing default R console size and etc : 배균섭 교수님께서 알려주신 tip을 참고하여 video clip을 만들었습니다. 매일같이 마주하게 되는 R console이 너무 작게 느껴지셨다면 다음의 동영상을 참고하셔서 초기 세팅 (Rconsole 파일)을 바꿔서 해결할 수 있습니다. <https://youtu.be/uSunEN8W5Mo>



D

Acknowledgement

이 웹북을 만드는데 도움을 주신 분들은 다음과 같습니다.

1. Dr. Jekyll
2. Hyde



Bibliography

Sarkar, D. (2017). *lattice: Trellis Graphics for R*. R package version 0.20-35.

Xie, Y. (2016). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.3.16.



Index

bookdown, [xi](#)

disqus, [xi](#)

lattice, [31](#)