

R을 이용한 데이터 시각화

김성수

한국방송통신대학교 정보통계학과

sskim@knou.ac.kr

목차

1. R, R Studio, R Commander
2. 이산형 그래프
3. 연속형 그래프
4. 다변량 데이터 탐색
5. lattice 활용
6. ggplot2 활용

1. R, R Studio, R Commander

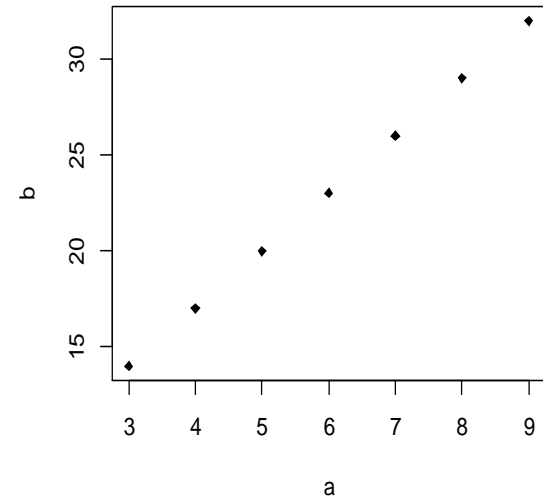
R의 소개

- R은 자료처리, 통계분석, 통계그래프 등에 뛰어난 기능을 가지고 있는 **무료 통계시스템**.
- R은 **대화형 프로그래밍 언어**(interpreted programming language)
- R은 **객체지향**(object-oriented) 시스템
 - 데이터, 변수, 행렬 등은 모두 객체(object)
 - 객체는 연산자 “<-”, 또는 “=”에 의해 생성됨.

예)

```
> x = 2:10
> y = 3*x + 5
> x
[1] 2 3 4 5 6 7 8 9 10
> y
[1] 11 14 17 20 23 26 29 32
35
```

```
> a <- 3:9
> b <- 3*a + 5
> plot(a,b, pch=18)
```



R의 태동

- **S의 탄생** : Becker and Chambers (AT&T Bell Lab) 가 1980년대에 새로 개발한 통계프로그램 언어를 S 라 명함 – S-PLUS 시스템으로 발전.
- **R의 탄생** : Ross Ihaka and Robert Gentleman(Univ. of Auckland, New Zealand) 가 교육 목적으로 S 의 축소버전 (reduced version) "R & R" 을 만듦
- **R의 발표** : 1995년 Martin Maechler가 Ross Ihaka and Robert Gentleman를 설득하여 Linux system 과 같이 Open Source Software 규약인 GPL(General Public Licence) 규약하에 R의 source code를 발표
- **R Core Team 의 결성** : 1997년 8월 R 시스템의 발전을 위한 국제적인 R core team의 결성됨. 이후 확장 발전하여 현재(2015년 7월) 21명의 멤버로 구성됨. 2000년 2월 29일 R version 1.0.0 발표됨. 2015년 7월 현재 R version 3.2.1.

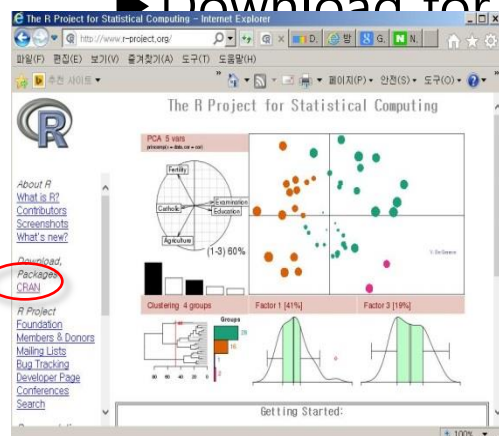
※ 참고 : www.r-project.org

Peter Dalgaard (2005), Introductory Statistics with R, Springer.

R 다운 받기

- www.r-project.org  CRAN   ▶ Mirrors     

▶ Download for Windows ▶ base ▶ Download R



R for Windows



CRAN Mirrors

The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. Some statistics on the status of the mirrors can be found here: [main page](#), [windows release](#), [windows old release](#).

0-Cloud	Rstudio, automatic redirection to servers worldwide
http://cran.rstudio.com/	
Argentina	
http://mirror.fcaglp.unlp.edu.ar/CRAN/	Universidad Nacional de La Plata
Australia	
http://cran.csiro.au/	CSIRO
http://cran.ms.unimelb.edu.au/	University of Melbourne
Austria	
http://cran.at.r-project.org/	Wirtschaftsuniversitaet Wien
Belgium	
http://www.freeststatistics.org/cran/	K.U.Leuven Association
Brazil	

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Subdirectories:

[base](#)

Binaries for base distribution (managed by Duncan Murdoch).

This is what you want to [install R for the first time](#).

[contrib](#)

Binaries of contributed packages (managed by Uwe Ligges).

There is also information on [third party software](#) available for CRAN Windows services and corresponding environment and make variables.

[Rtools](#)

Tools to build R and R packages (managed by Duncan Murdoch). This is what you want to build your own packages on Windows, or to build R itself.

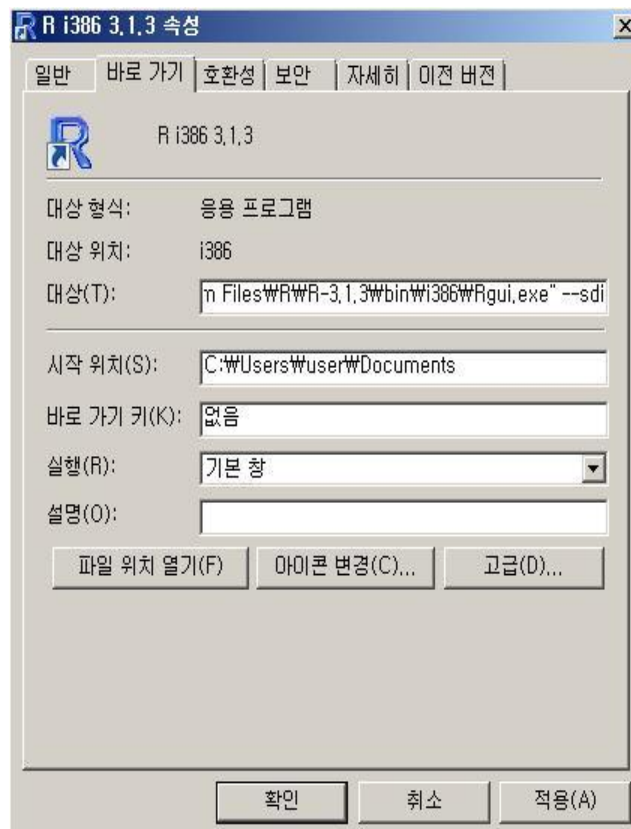
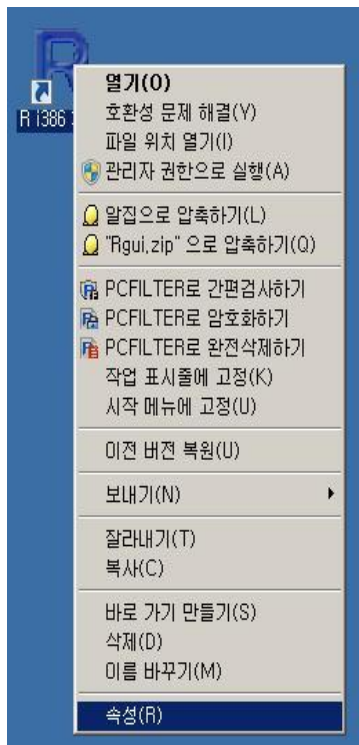
R-3.1.2 for Windows (32/64 bit)

[Download R 3.1.2 for Windows](#) (34 megabytes, 32/64 bit)

[Installation and other instructions](#)

[New features in this version](#)

R 환경 설정



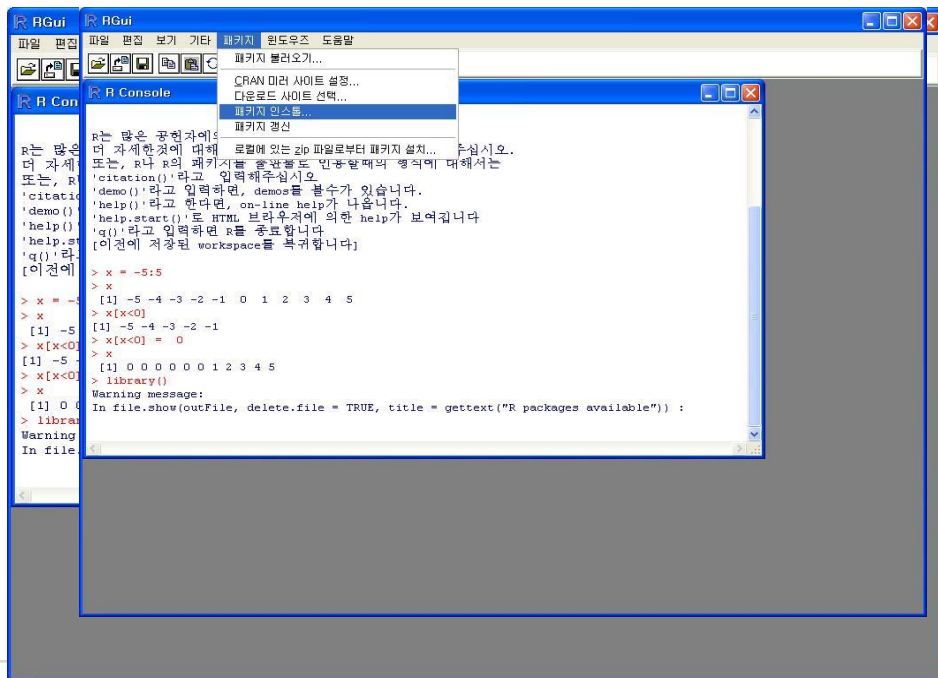
Only one space and
--sdi
(Single Document
Interface)

R 패키지

◆ 다양한 분석 방법들이 패키지(package)로 제공됨

◆ 예) 군집분석 패키지 “cluster”를 설치하는 방법

- ① [패키지] - [package 인스톨]을 선택 또는
> install.packages(“cluster”)

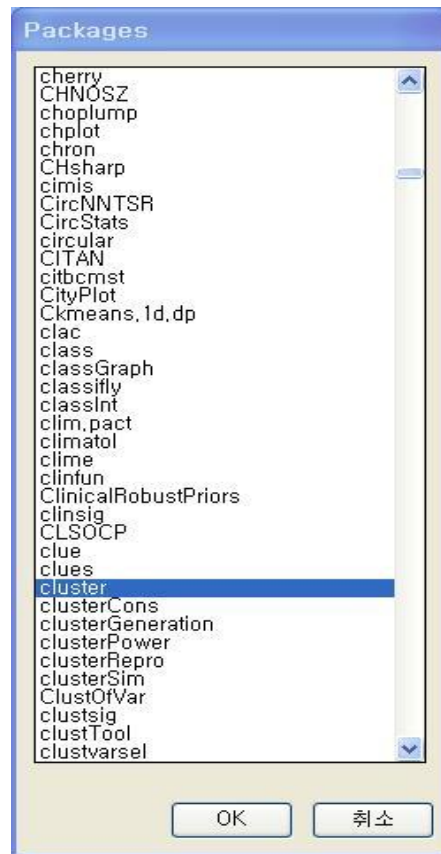


R 패키지 설치

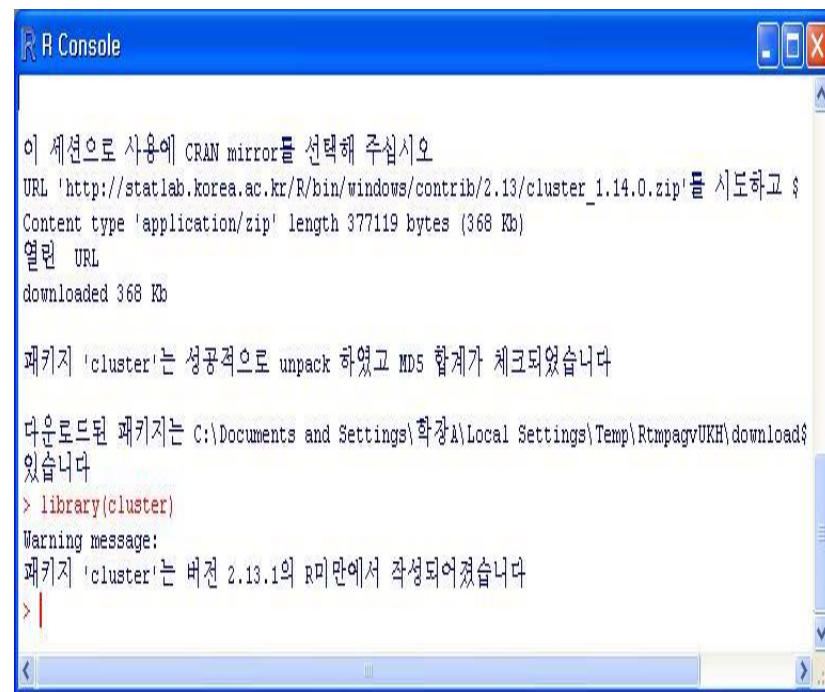
② Cran Mirror 선택



③ 패키지 cluster 선택



④ 가동 : > library(cluster)



작업영역 지정

- **작업 영역(Working directory)** : R에서 데이터를 가져오고 저장하는 디폴트 폴더를 지정해두면 편리하게 작업할 수 있음. 이를 작업 영역(Working directory) 이라 함.

```
> getwd() # shows the working directory
```

```
[1] "C:/Users/user/Documents"
```

```
> setwd("c:/Rfolder/data") # change the working directory
```

```
> getwd()
```

```
[1] "c:/Rfolder/data"
```

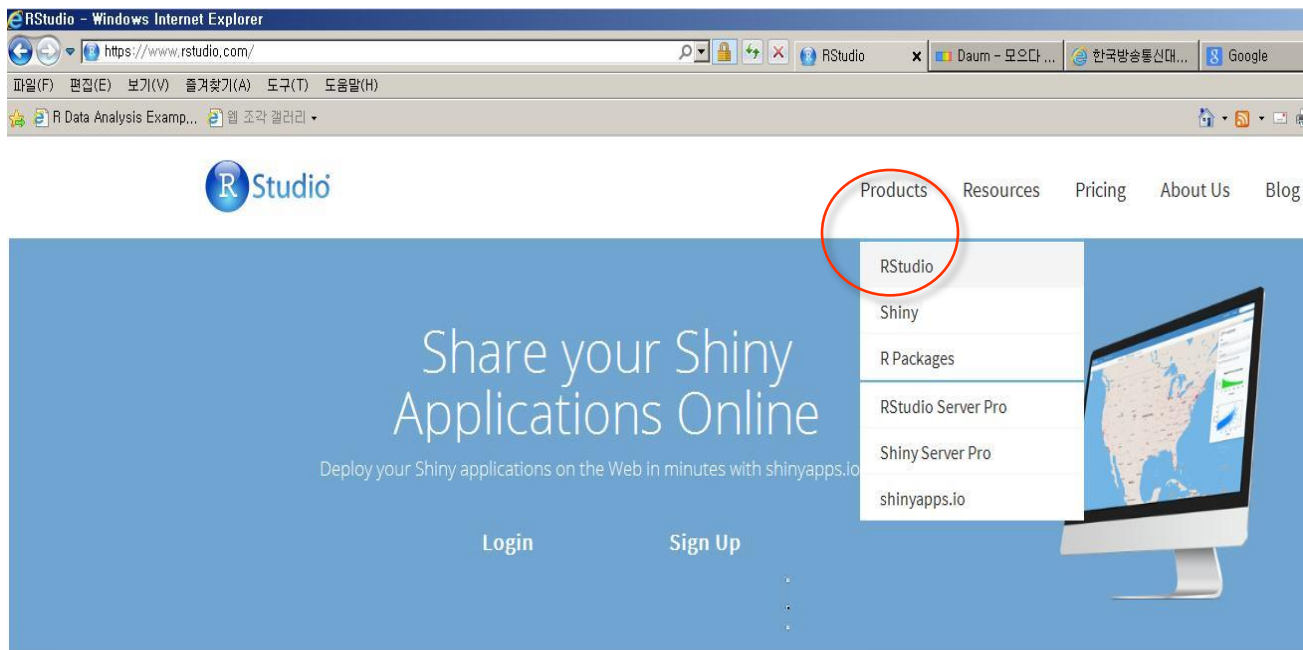
```
> setwd(choose.dir()) # select the working directory  
interactively
```

```
> getwd()
```

```
[1] "C:/Rfolder/data"
```

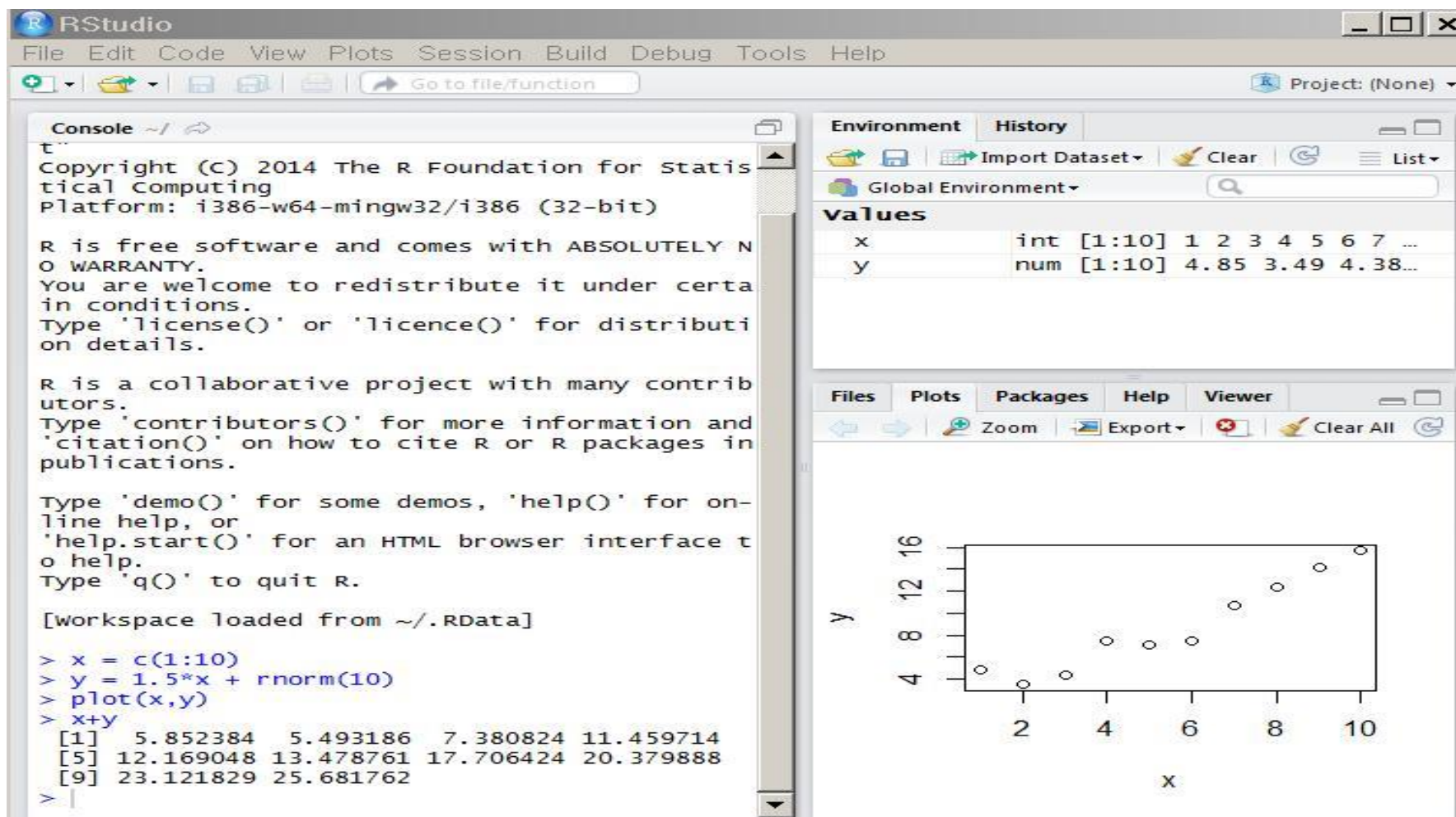
R Studio의 소개

- R Studio : 사용자가 친숙하게 R을 쉽게 사용할 수 있도록 개발된 R 통합환경 시스템
- 다운로드 : www.rstudio.com



참고 : “<http://dss.princeton.edu/training/RStudio101.pdf>”
“<http://www.rstudio.com>”

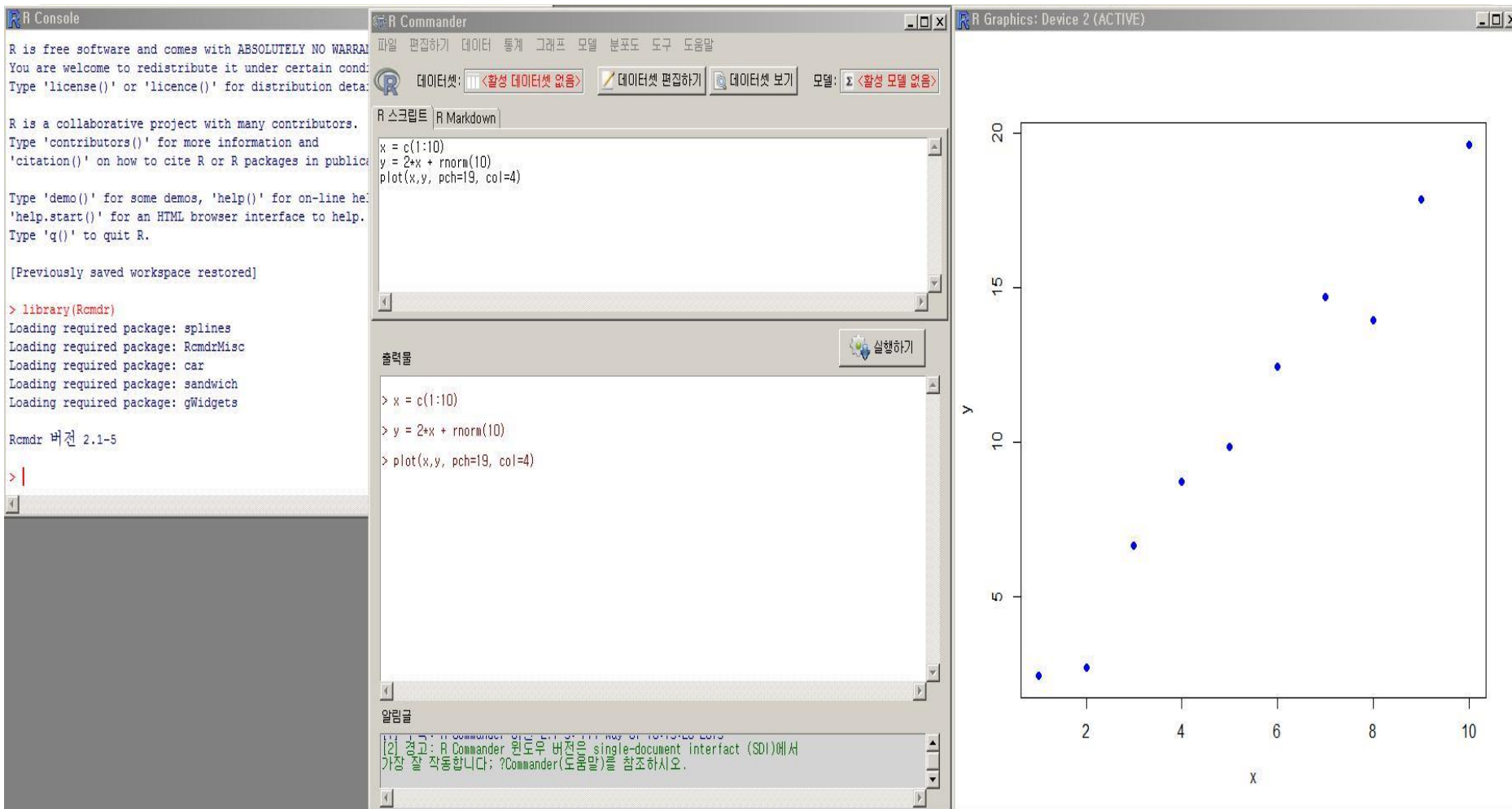
R Studio 화면



R Commander 소개

- R Commander : A GUI for R
 - menu 방식(menu-driven) 으로 처리할 수 있도록 개발된 R package
cf: R is command-driven
 - 개발자 : John Fox (McMaster University)
 - 통계학 입문 코스에 유용하게 이용
 - 복잡한 고급 기능에는 부적합
 - 현재 version 2.1-2 (11 Sep. 2014)

R Commander 화면



2. 이산형 그래프

예제 1

- 어느 집단에서 표본을 10명 추출하여 다음과 같은 4개 문항에 대하여 설문조사를 실시하였다.

문항 1. 귀하의 성별은?

- 1) 남자 2) 여자

문항 2. 귀하의 나이는? (단위 : 세)

문항 3. 교육정도는?

- 1) 중졸이하 2) 고졸 3) 대졸 및 그 이상

문항 4. 월수입(단위: 만원)



	sex	age	edu	salary
1	1	21	2	150
2	2	22	1	100
3	1	33	2	200
4	2	33	3	220
5	1	28	2	170
6	1	41	3	300
7	2	39	2	290
8	1	32	3	220
9	2	44	1	370
10	1	55	3	410

- 남자 여자의 수(이를 성별 도수분포표라 함)와 각 교육정도별 사람의 수(교육정도별 도수분포표)를 구하라.
- 교육정도별 도수분포를 나타내는 막대그림을 그려 어느 집단이 제일 많은지 관찰하라.

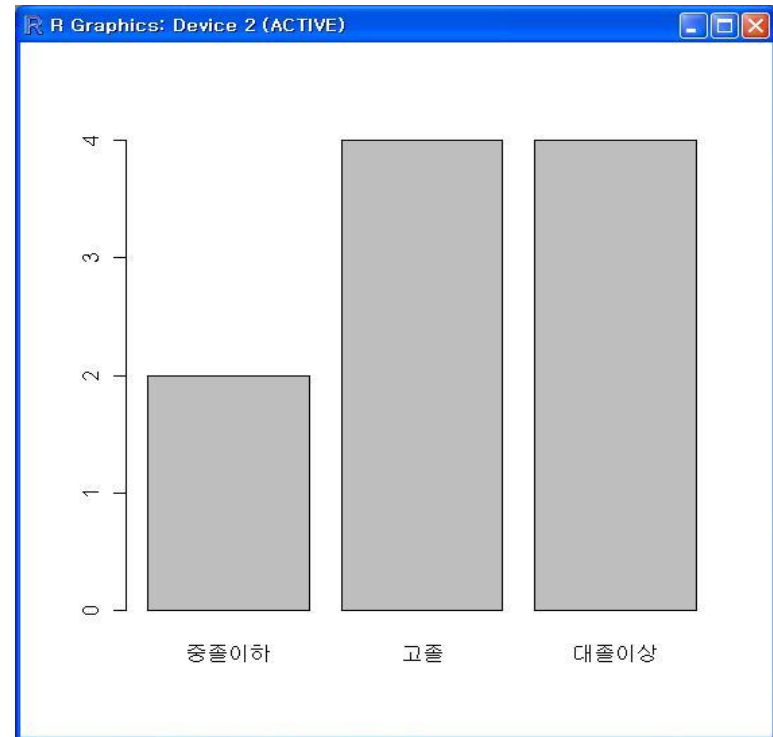
성별과 교육 정도의 도수분포표 및 막대그림

```
R R Console

> ex8 = read.table("c:/data/ex8.txt", header=T)
> attach(ex8)
> colnames(ex8)
[1] "sex"      "age"      "edu"      "salary"
> sex.tb = table(sex)
> sex.tb
sex
1 2
6 4
> edu.tb = table(edu)
> edu.tb
edu
1 2 3
2 4 4
> |
```

```
R R Console

> rownames(edu.tb) = c("중졸이하", "고졸", "대졸이상")
> edu.tb
edu
중졸이하    고졸    대졸이상
      2      4      4
> barplot(edu.tb)
> |
```



예제 2

- 세 그룹(C1,C2,C3)이 다섯 자선단체(T1 T5) 에 기부하는 가상자료를 예를 들어 막대그림, 원그림을 그리는 프로그램을 작성하여 보자.

< perc.txt >

	C1	C2	C3
T1	5.4	3.1	3.5
T2	5.7	8.6	25.0
T3	20.4	26.0	22.0
T4	36.3	34.1	28.0
T5	14.4	11.4	4.5

참고: 위와 같이 행렬 형태의 자료를 data.frame 이라고 한다.
read.table 함수를 이용하여 읽으면 된다.

데이터 읽기

```
> percData <- read.table("c:/data/perc.txt", header=T)
> percData <- as.matrix(percData)
> var.name <- colnames(percData)
> case.name <- rownames(percData)
```

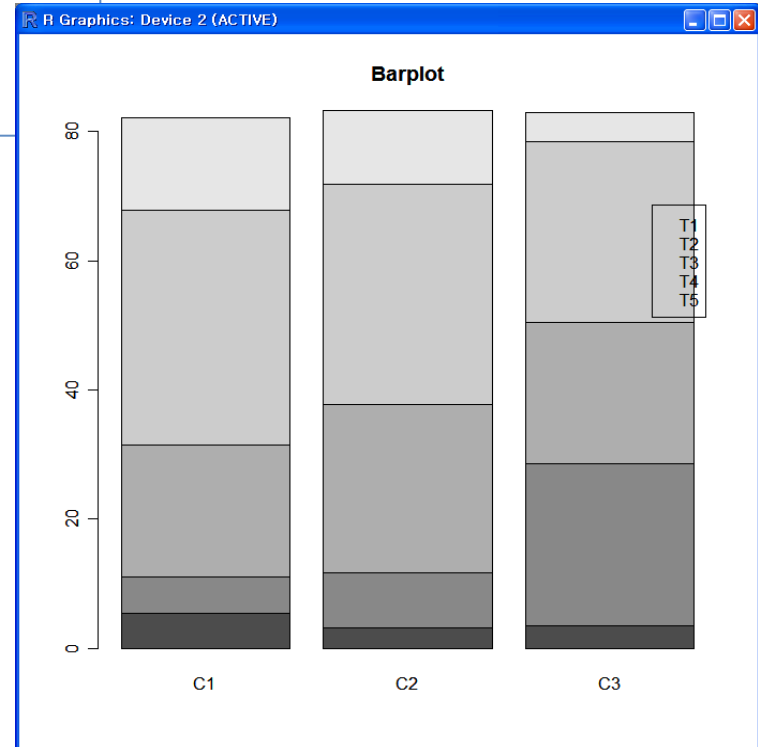
: 데이터의 변수이름과 행의 이름을 각각 `var.name`, `case.name` 에 저장

막대그림 그리기

```
> # barplot
> barplot(percData, names=var.name)
> legend(locator(1), case.name)
> title("Barplot")
```

: 막대그림함수(`barplot`),

: 마우스로 선택한 임의의 위치에(`locator(1)`) 범례(`legend`)를 나타냄



Generating tables

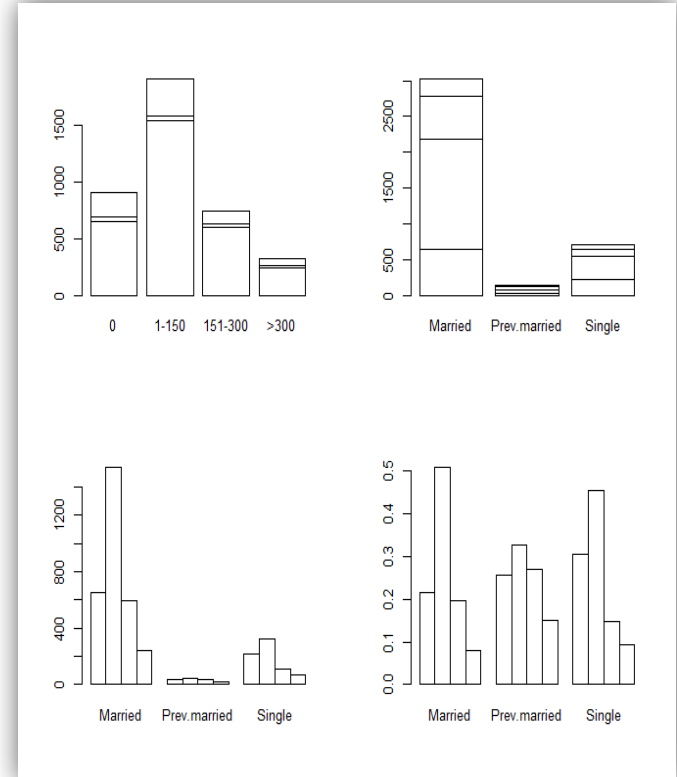
- Altman (1991, p. 242) contains an example on caffeine consumption by marital status among women giving birth. That table may be input as follows:

```
> caff.marital <-  
matrix(c(652,1537,598,242,36,46,38,21,218,327,106,67),  
       nrow=3,byrow=T)  
  
> caff.marital  
      [,1] [,2] [,3] [,4]  
[1,]  652 1537  598  242  
[2,]   36   46   38   21  
[3,]  218  327  106   67  
  
> colnames(caff.marital) <- c("0","1-150","151-300",>300")  
> rownames(caff.marital) <- c("Married","Prev.married","Single")  
> caff.marital  
      0 1-150 151-300 >300  
Married      652  1537    598  242  
Prev.married  36    46     38   21  
Single      218  327    106   67
```

Boxplot

- If the argument is a matrix, then `barplot` creates by default a “stacked barplot”, where the columns are partitioned according to the contributions from different rows of the table. If you want to place the row contributions beside each other instead, you can use the argument `beside=T`.

```
> par(mfrow=c(2,2))
> barplot(caff.marital, col="white")
> barplot(t(caff.marital), col="white")
> barplot(t(caff.marital), col="white", beside=T)
> barplot(prop.table(t(caff.marital),2), col="white",
+         beside=T)
> par(mfrow=c(1,1))
```

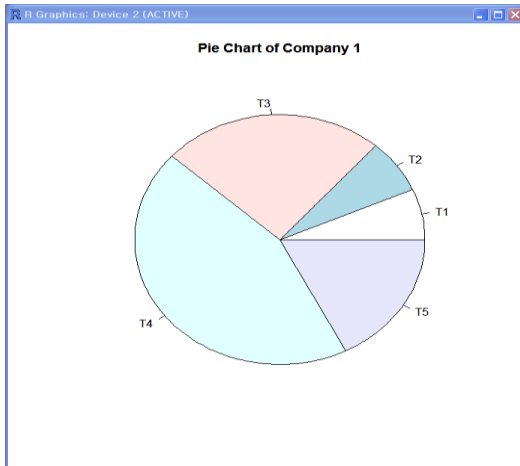




원그림 그리기

```
> #Piechart  
> pie(percData[,1], labels=case.name)  
> title("Pie Chart of Company 1")
```

: 처음 열변수의 원그림을 그리는 명령임



: 그룹1 (C1)에서 기부단체 비율을 원그림으로 나타낸 그림

🕒 **점그림 : Dot plots are a reasonable substitute for bar plots.**

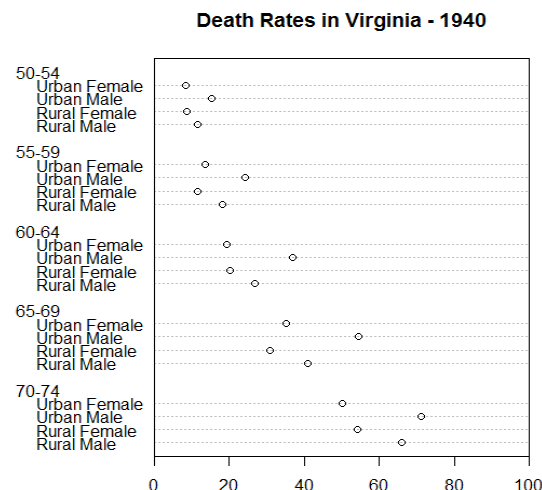
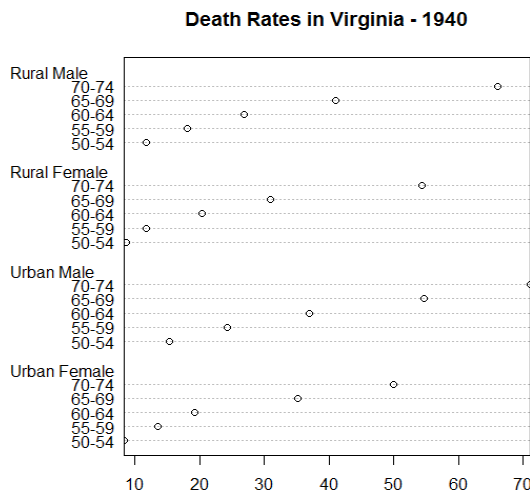
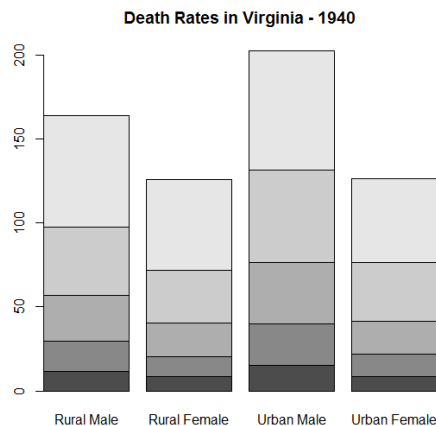
> **VADeaths**

	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, main = "Death Rates in Virginia - 1940") #1**

> **dotchart(VADeaths, main = "Death Rates in Virginia - 1940") #2**

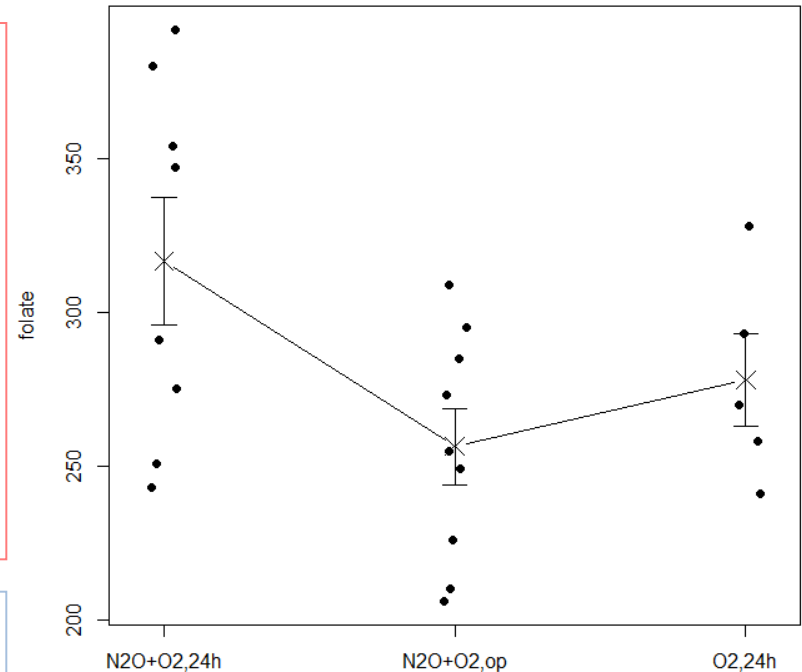
> **dotchart(t(VADeaths), xlim=c(0,100), main = "Death Rates in Virginia - 1940") #3**



- Stripchart where the raw data are plotted and overlaid with an indication of means and SEM(standard error of means).

```
> library(ISwR)
> attach(red.cell.folate)
> xbar <- tapply(folate, ventilation, mean)
> s <- tapply(folate, ventilation, sd)
> sem <- s/sqrt(n)
> stripchart(folate~ventilation, method="jitter",
+ jitter=0.05, pch=16, vert=T)
> arrows(1:3,xbar+sem,1:3,xbar-sem,
+       angle=90,code=3, length=.1)
> lines(1:3,xbar,pch=4,type="b",cex=2)
```

```
> red.cell.folate[c(1,9,18),]
   folate ventilation
1    243  N2O+O2,24h
9    206  N2O+O2,op
18   241    O2,24h
```



예제 3

한 설문조사에서 다음 6개 문항에 대하여 표본 추출된 40명을 대상으로 조사한 자료이다. R을 이용하여 교육 정도에 대한 수직형 막대그림을 그려라. 또 각각의 성별을 구분한 교육 정도의 수직형 막대그림을 그려라.

문항 1. 귀하의 성별은? 1) 남자 2) 여자

문항 2. 결혼하셨습니다까? 1) 미혼 2) 기혼 3) 이혼

문항 3. 귀하의 나이는? (단위: 세)

문항 4. 귀하의 직업은?

1) 회사원 2) 공무원 3) 노무자 4) 정치가

5) 학생 6) 기업가 7) 주부 8) 기타

문항 6. 가족의 월수입은? (단위: 만원)



	sex	marriage	age	job	edu	salary
1	1	1	21	1	4	60
2	1	1	22	5	5	100
3	1	1	33	1	4	200
4	2	2	33	7	4	120
5	1	2	28	1	4	70
6	1	1	21	5	5	80
7	2	2	39	7	4	190
8	1	1	32	1	4	100
9	1	2	44	3	1	120
10	1	2	55	4	4	110
11	2	2	46	7	5	150
12	1	1	20	1	4	50
13	1	2	31	6	4	210
14	1	1	27	1	4	60
15	2	1	21	5	5	80



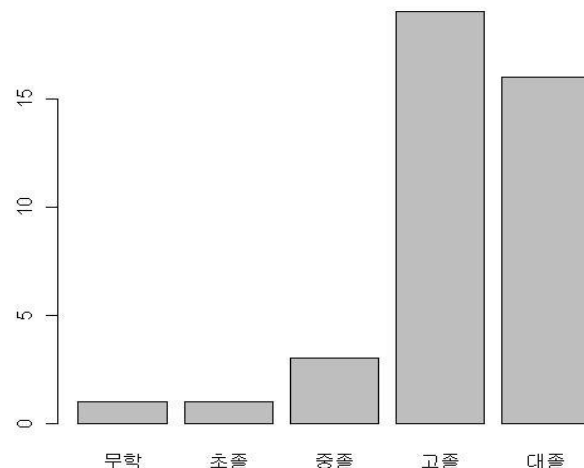
막대그림 그리기

R Console

R는 많은 공헌자에의한 공동 프로젝트입니다
더 자세한것에 대해서는 'contributors()'라고 입력해 주십시오.
또는, R나 R의 패키지를 출판물로 인용할때의 형식에 대해서는
'citation()'라고 입력해주십시오
'demo()'라고 입력하면, demos를 볼수가 있습니다.
'help()'라고 한다면, on-line help가 나옵니다.
'help.start()'로 HTML 브라우저에 의한 help가 보여집니다
'q()'라고 입력하면 R를 종료합니다
[이전에 저장된 workspace를 복귀합니다]

```
> ex91 = read.table("c:/data/ex9-1.txt", header=T)
> attach(ex91)
> colnames(ex91)
[1] "sex"      "marriage" "age"      "job"      "edu"      "salary"
> edu.tb = table(edu)
> edu.tb
edu
 1  2  3  4  5
 1  1  3 19 16
> rownames(edu.tb) = c("무학", "초졸", "중졸", "고졸", "대졸")
> barplot(edu.tb)
> |
```

R Graphics: Device 2 (ACTIVE)



⑥ 성별 구분 교육 정도 막대그림 그리기

```
R Console

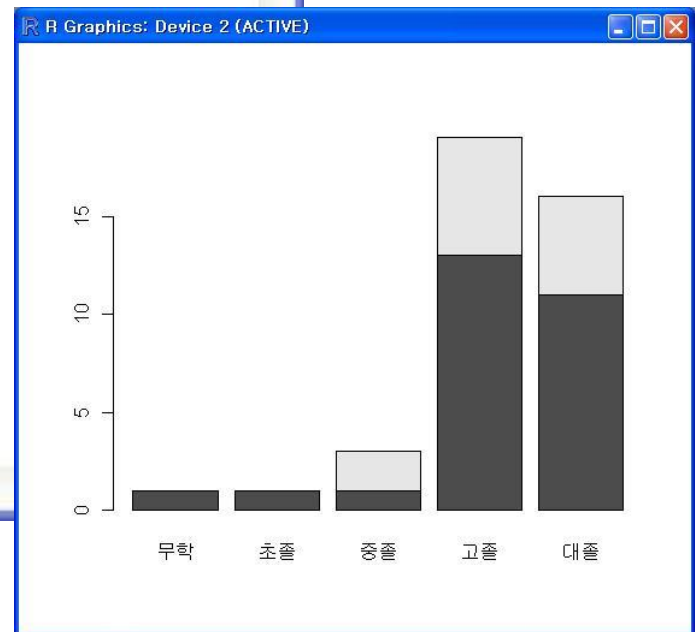
> sex.edu = list(sex,edu)
> sex.edu.tb = table(sex.edu)
> sex.edu.tb

      sex.edu.2
sex.edu.1 1  2  3  4  5
      1  1  1  1 13 11
      2  0  0  2  6  5

> colnames(sex.edu.tb) = c("무학","초졸","중졸","고졸","대졸")
> rownames(sex.edu.tb) = c("남성","여성")
> sex.edu.tb

      sex.edu.2
sex.edu.1 무학 초졸 중졸 고졸 대졸
      남성   1   1   1  13  11
      여성   0   0   2   6   5

> barplot(sex.edu.tb)
> |
```

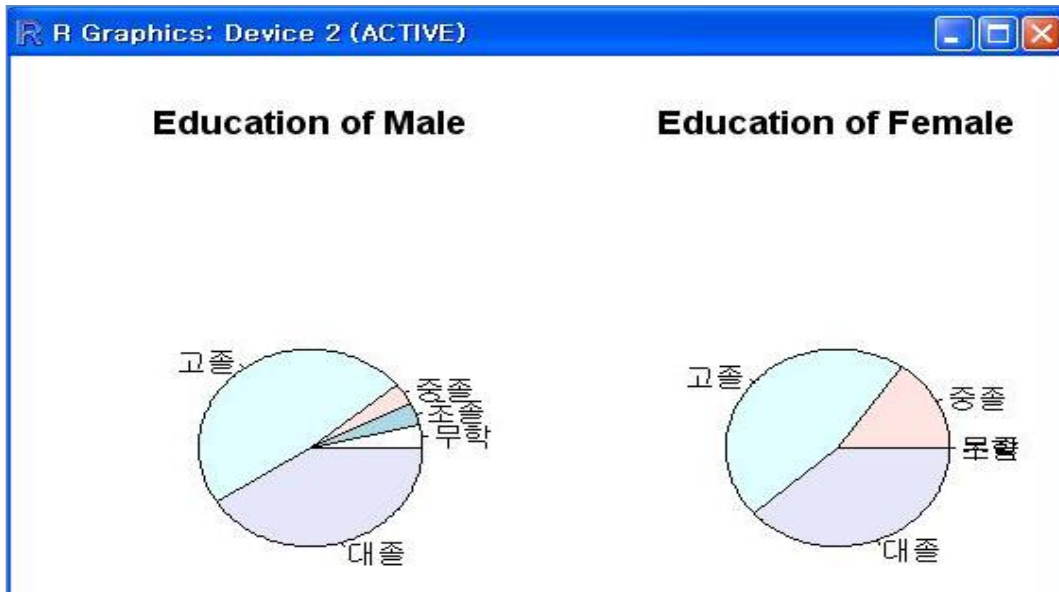




원그림 그리기

```
> #Piechart  
> pie(educ.tb)  
> par(mfrow=c(1,2))  
> pie(sex.educ.tb[1,])  
> title("Education of Male")  
> pie(sex.educ.tb[2,])  
> title("Education of Female")
```

: par문을 이용하여 한 화면에 여러 개의 그림을 그릴 수 있음



참고1 : 값 라벨 (Value labels)

- 숫자로 입력된 값을 라벨로 바꾸기

예) 변수 job 1=근로자, 2=사무직, 3=전문가
edu 1=무학, 2=국졸, 3=중졸, 4=고졸, 5=대졸

```
> insurance = read.table("c:/Rfolder/data/insurance.txt", header=T)
> insurance$job = factor(insurance$job, levels=c(1:3),
                        labels=c("근로자", "사무직", "전문가"))
> insurance$edu2 = ordered(insurance$edu, levels=c(1:5),
                          labels=c("무학", "국졸", "중졸", "고졸", "대졸"))
```

```
> head(insurance)
```

	id	sex	job	religion	edu	amount	salary	edu2
1	1	m	근로자	1	3	7.0	110	중졸
2	2	m	사무직	1	4	12.0	135	고졸
3	3	f	사무직	3	5	8.5	127	대졸
4	4	f	전문가	3	5	5.0	150	대졸
5	5	m	근로자	3	3	4.5	113	중졸
6	6	m	사무직	1	2	3.5	95	국졸



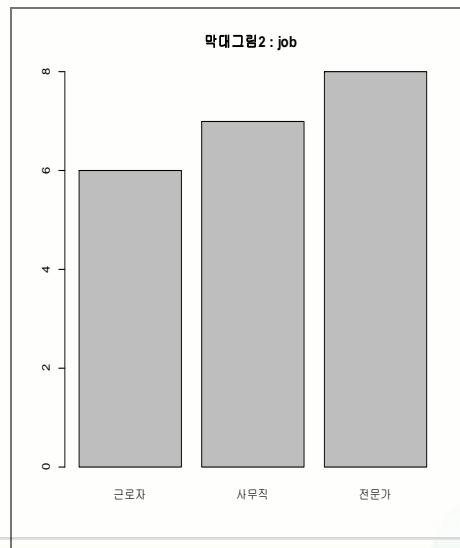
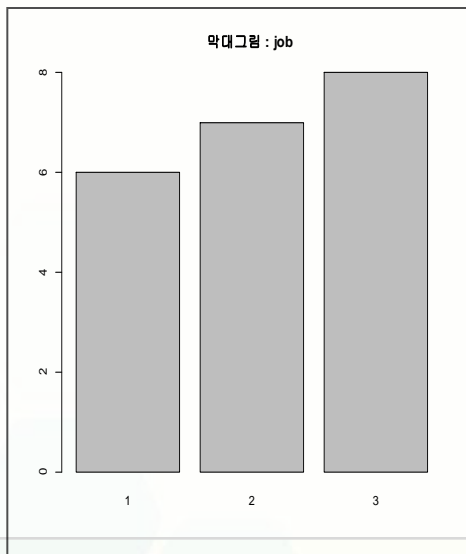
id	sex	job	religion	edu	amount	salary
1	m	1	3	7.0	110	
2	m	2	1	4	12.0	135
3	f	3	5	8.5	127	
4	f	3	5	5.0	150	
5	m	1	3	4.5	113	
6	m	2	1	2	3.5	95
7	m	3	2	4	4.0	102
8	f	3	2	4	4.0	122
9	f	2	3	4	4.5	110
10	m	1	3	5	17.0	200
11	f	1	1	3	22.0	NA
12	m	2	1	2	5.5	105
13	m	3	2	1	4.5	130
14	m	3	2	5	7.0	150
15	m	NA	3	4	6.0	110
16	f	1	3	NA	7.0	88
17	m	1	1	4	6.0	138
18	f	2	1	5	5.0	110
19	m	2	3	3	7.0	85
20	m	3	3	4	9.5	110
21	m	3	1	4	10.0	95
22	m	3	2	3	12.0	88

- 명목형(nominal data) :
factor() 함수
- 순서형(ordered data) :
ordered() 함수

값 라벨 (Value labels)

예) 막대그림 그리기

```
> job.freq = table(insurance$job)
> barplot(job.freq)
> title("막대그림 : job ")
> insurance$job = factor(insurance$job, levels=c(1:3),
                        labels=c("근로자", "사무직", "전문가"))
> job.freq2 = table(insurance$job)
> barplot(job.freq2)
> title("막대그림2 : job ")
```



참고2 : 변수 값 변환(recode)

예) 어느 제약제품의 약 구매 여부를 조사하였다. 나이별 구매내역은 다음과 같다. 반응변수 purchase 0=구매 안함, 1=구매함이다. 케이스 수는 100개이다. 이 자료에서 변수 나이(age)의 값을 "40 이하=1, 41~60=2, 60 보다 큰 값=3"으로 변환하여보자.

	A	B	C
1	id	age	purchase
2	1	20	0
3	2	23	0
4	3	24	0
5	4	25	1
6	5	26	0
7	6	27	0
8	7	27	0
9	8	28	0
10	9	29	0
11	10	29	0
12	11	30	0
13	12	30	0
14	13	30	0
15	14	30	1
16	15	32	0

```
> install.packages("xlsx")
> library(xlsx)
> drug = read.xlsx("c:/Rfolder/data/drug.xlsx", 1)
# Replace data in the field : Method 1
> drug$agr = drug$age
> drug$agr[drug$agr >= 20 & drug$agr <= 40 ] = 1
> drug$agr[drug$agr > 40 & drug$agr <= 60 ] = 2
> drug$agr[drug$agr > 60 ] = 3
> drug[c(1,20,40, 95),]
  id age purchase agr
1  1  20         0   1
20 20  34         0   1
40 40  41         0   2
95 95  61         1   3
```

변수 값 변환(recode)

예) car 패키지의 recode() 를 이용하는 예

```
> # Use recode function in car package : Method 2
> install.packages("car")
> library(car)
> drug$agr2 = drug$age
> drug$agr2 = recode(drug$age, "lo:40=1; 40:60=2; 60:hi=3")
> drug[c(1,20,40, 80),]
```

	id	age	purchase	agr	agr2
1	1	20	0	1	1
20	20	34	0	1	1
40	40	41	0	2	2
80	80	56	0	2	2

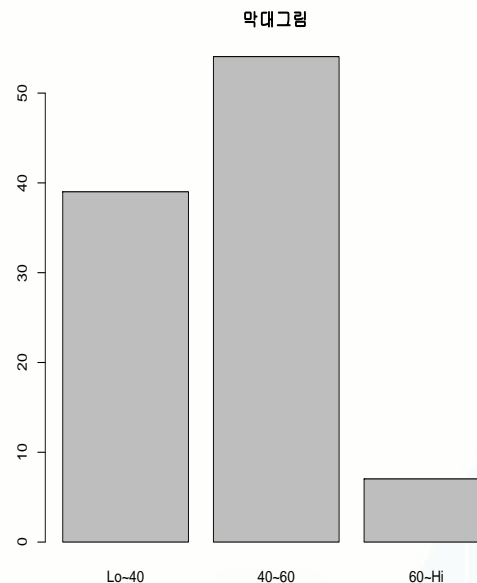
```
> drug$agr2 = ordered(drug$agr2, levels=c(1:3),
                      labels=c("Lo~40", "40~60", "60~Hi"))
```

```
> agr2.freq=table(drug$agr2)
```

```
> agr2.freq
```

Lo~40	40~60	60~Hi
39	54	7

```
> barplot(agr2.freq, main="막대그림")
```



참고3: 케이스 선택

예) insurance 자료에서 다음과 같은 조건을 만족하는 케이스를 추출해보자.

- ① 성별=m 인 경우,
- ② 성별=f 이고, 직업=2인 경우,

```
> insurance = read.table("c:/Rfolder/data/insurance.txt", header=T)
> select1 = insurance[insurance$sex=='m',]
> head(select1, n=3)
```

	id	sex	job	religion	edu	amount	salary	
1	1	m	1		1	3	7.0	110
2	2	m	2		1	4	12.0	135
5	5	m	1		3	3	4.5	113

```
> select2 = insurance[which(insurance$sex=='f' & insurance$job==2),]
> head(select2, n=3)
```

	id	sex	job	religion	edu	amount	salary
3	3	f	2	3	5	8.5	127
9	9	f	2	3	4	4.5	110
18	18	f	2	1	5	5.0	110

케이스 선택

예) insurance 자료에서 다음과 같은 조건을 만족하는 케이스를 추출해보자.

③ 직업=3 이고, 월수입=140 이상인 경우

```
> select3 = insurance[which(insurance$job==3 & insurance$salary >= 140),]
```

```
> head(select3, n=3)
```

	id	sex	job	religion	edu	amount	salary	
4	4	f	3		3	5	5	150
14	14	m	3		2	5	7	150

```
> select3 = insurance[insurance$job==3 & insurance$salary >= 140,]
```

```
> head(select3, n=3)
```

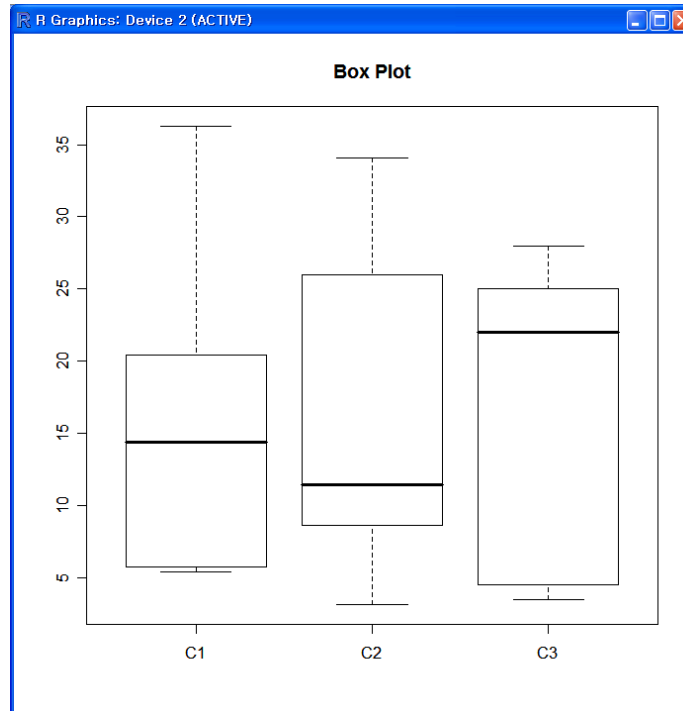
	id	sex	job	religion	edu	amount	salary	
4	4	f	3		3	5	5	150
14	14	m	3		2	5	7	150

3. 연속형 그래프

◆ 상자그림

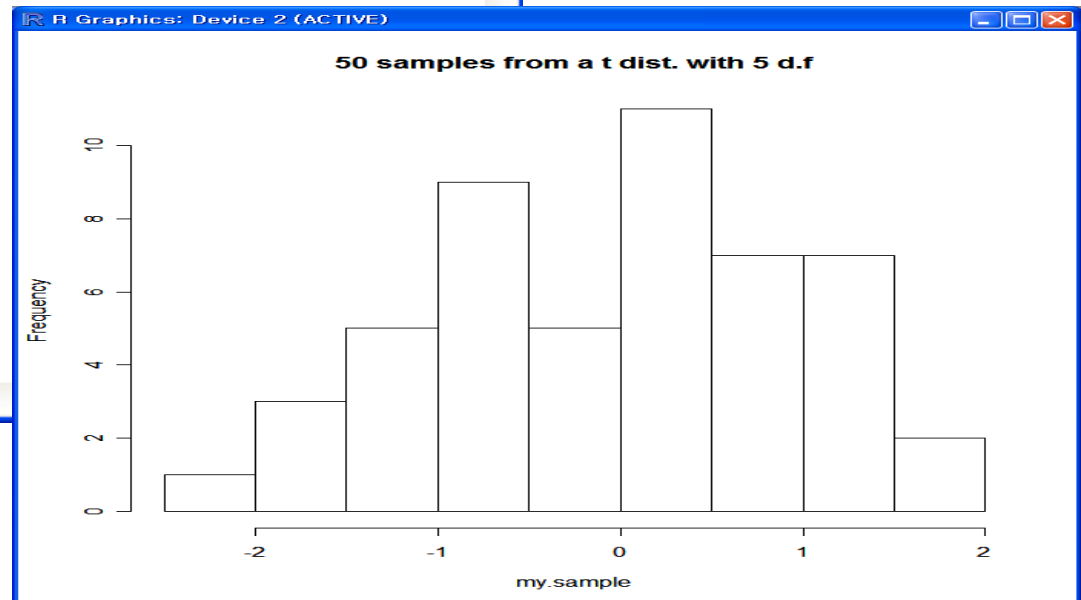
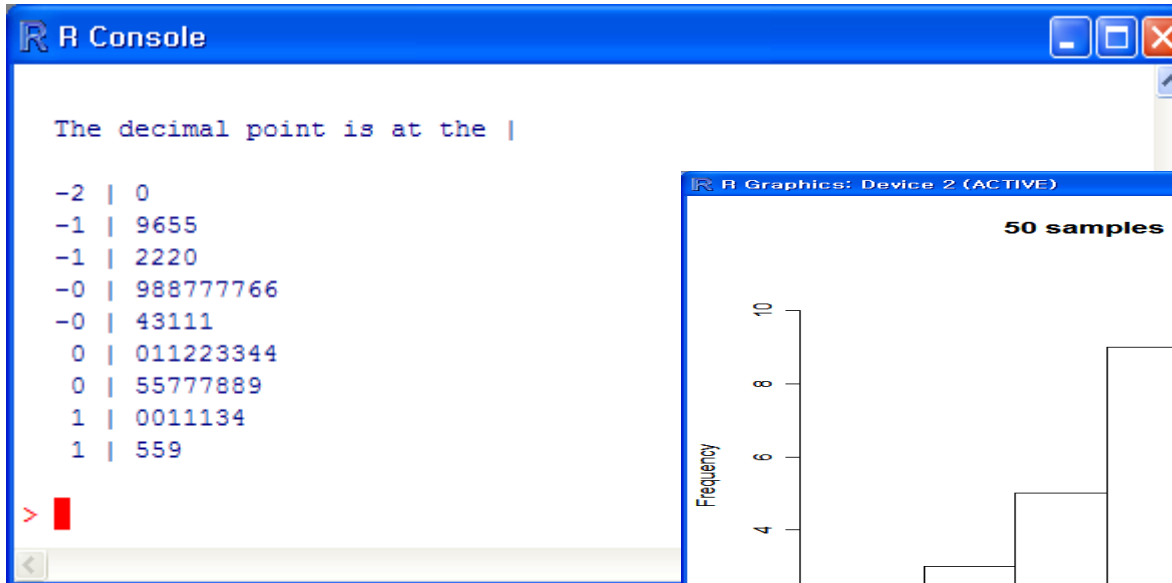
: percData를 이용해서 세 회사(C1, C2, C3)에 대한 상자그림 그리기

```
> boxplot(percData[,1],percData[,2],percData[,3],  
          names=var.name )  
> title("Box Plot")
```



- ◆ 자유도가 5인 t-분포를 따르는 난수 50개를 만들어 히스토그램 및 줄기-잎 그림 그리기

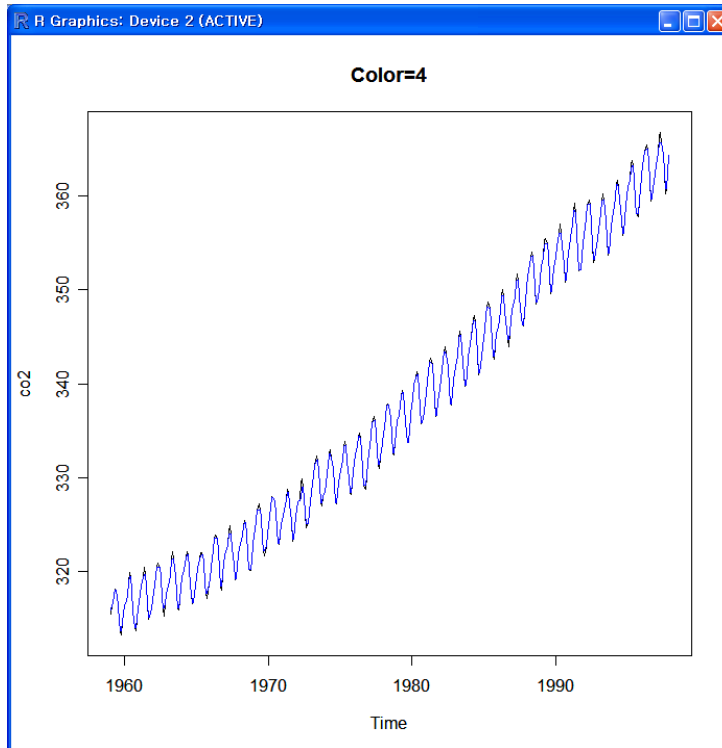
```
> my.sample <- rt(50,5) # 50 samples from t(d.f=5)  
> hist(my.sample, main="50 samples from a t dist. with 5 d.f")  
> stem(my.sample)
```



◆ R 시스템에 내장된 데이터 co2를 이용한 시계열 그림 그리기

```
> # plot using lines  
> plot(co2)  
> lines(smooth(co2),col="BLUE")
```

: 데이터 선으로 연결, col="BLUE" 는 파란색



◆ 함수 그리기

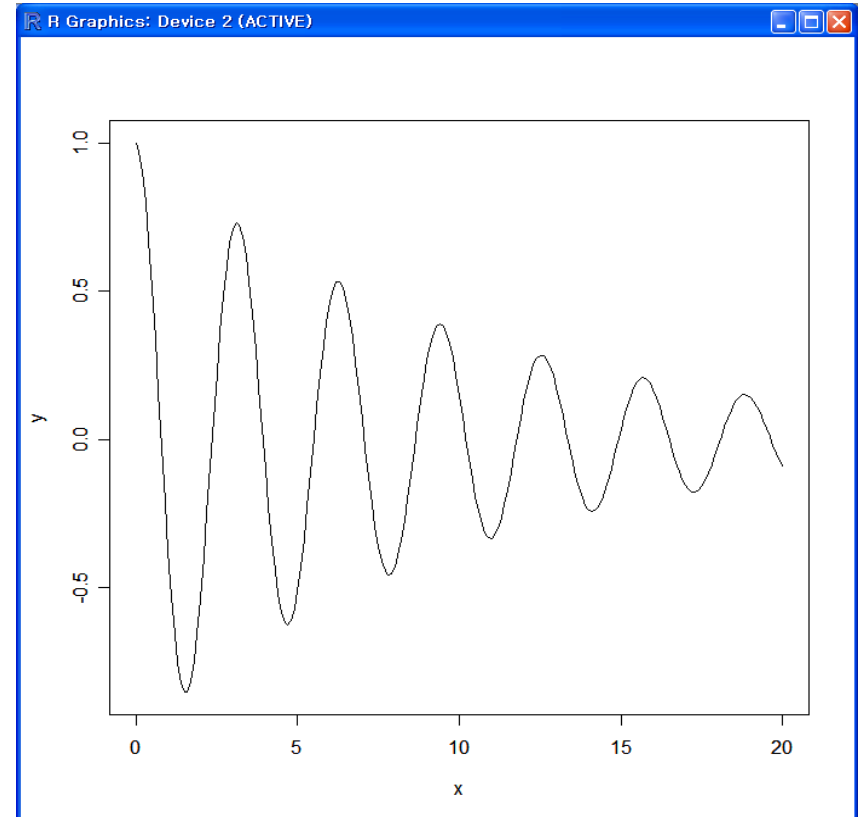
```
> # plot of mathematical functions  
> x <- seq(0, 20, 0.1)  
> y <- exp(-x/10)*cos(2*x)  
> plot(x,y,type="l")
```

```
x <- seq(0, 20, 0.1)
```

: 0~20 0.1단위로 데이터생성

```
plot(x, y, type="l")
```

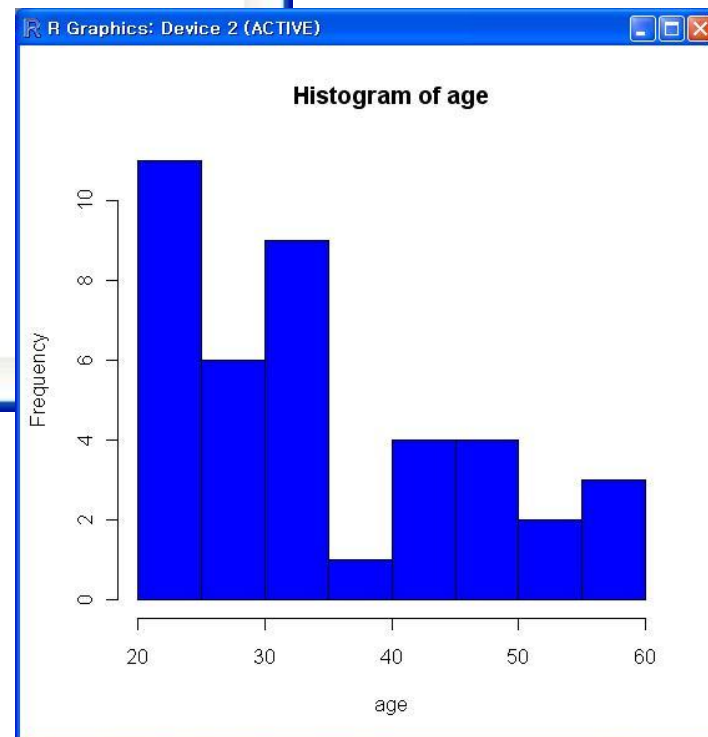
: type="l" 선으로 연결



◆ [예제3] 자료에서 나이에 대한 히스토그램 그리기

```
R R Console

> age.hist = hist(age, col="BLUE")
> bpoint = age.hist$counts
> bcount = age.hist$counts
> bpoint = age.hist$breaks
> bmid = age.hist$mid
> bpoint
[1] 20 25 30 35 40 45 50 55 60
> bcount
[1] 11  6  9  1  4  4  2  3
> bmid
[1] 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5
> |
```

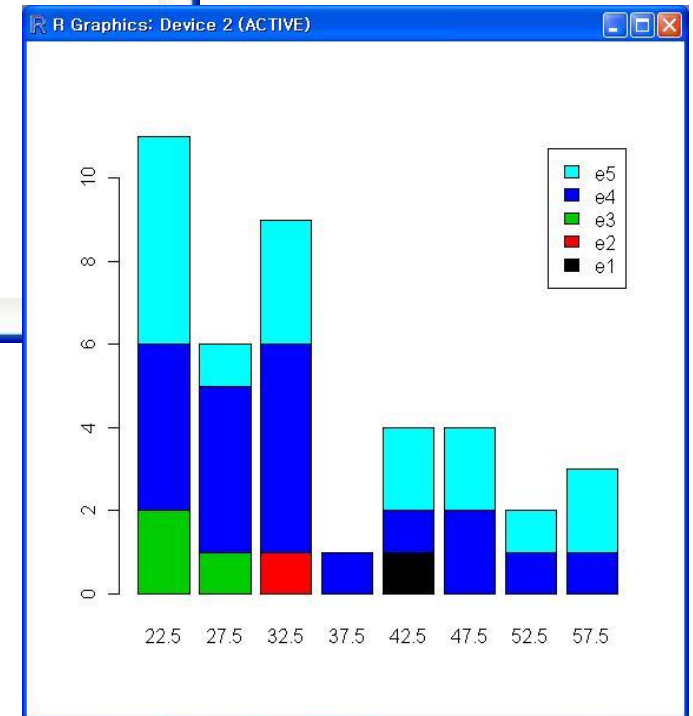


◆ [예제3] 자료에서 교육 정도 별 나이에 대한 히스토그램 그리기

```
R Console

> age.edu.count = matrix(0,ncol=8, nrow=5)
> colnames(age.edu.count) = bmid
> rownames(age.edu.count) = c("e1","e2","e3","e4","e5")
> for(i in 1:5) { age.e = age[edu==i]
+   age.hist = hist(age.e, breaks=bpoint)
+   age.edu.count[i,] = age.hist$counts }
> age.edu.count
      22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5
e1      0      0      0      0      1      0      0      0
e2      0      0      1      0      0      0      0      0
e3      2      1      0      0      0      0      0      0
e4      4      4      5      1      1      2      1      1
e5      5      1      3      0      2      2      1      2

> barplot(age.edu.count, col=c(1:5), legend=rownames(age.edu.count))
> |
```



커널(Kernel)밀도 그림

예제) 다음은 69개국에서 각 1000명씩을 조사해서 출생률과 사망률을 조사한 **birth.txt** 데이터의 일부이다. **birth.txt**로 밀도그림을 그려보고 히스토그램과 비교해 보아라.

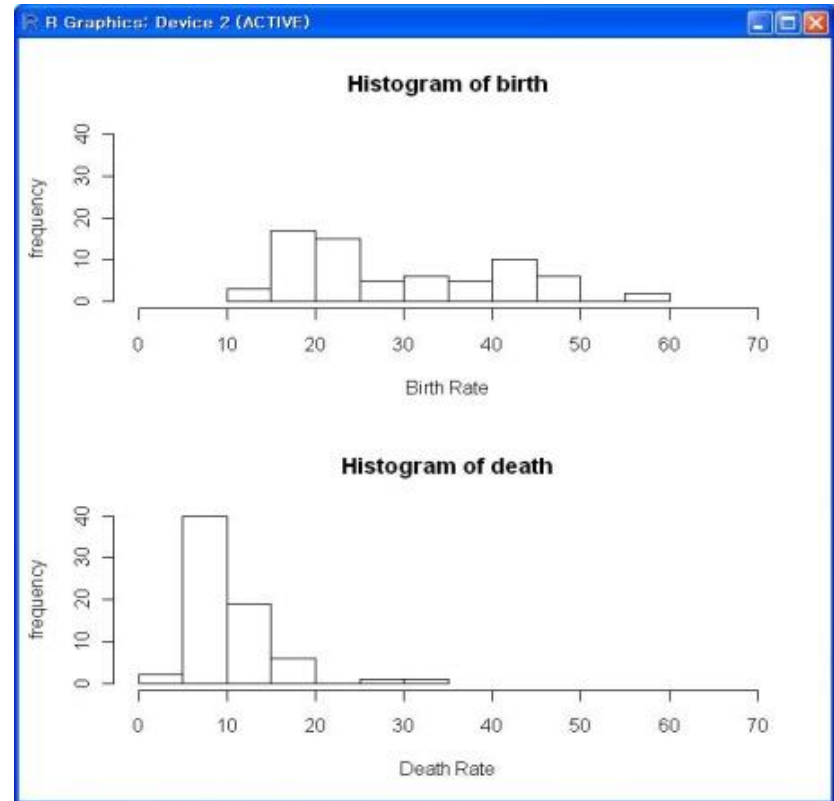


The image shows a screenshot of a Windows Notepad window titled "birth.txt - 메모장". The window contains a table with three columns: "country", "birth", and "death". The data is as follows:

country	birth	death
alg	36.4	14.6
con	37.3	8
egy	42.1	15.3
gha	55.8	25.6
ict	56.1	33.1
mag	41.8	15.8
mor	46.1	18.7
tun	41.7	10.1
cam	41.4	19.7
cey	35.8	8.5
chi	34	11

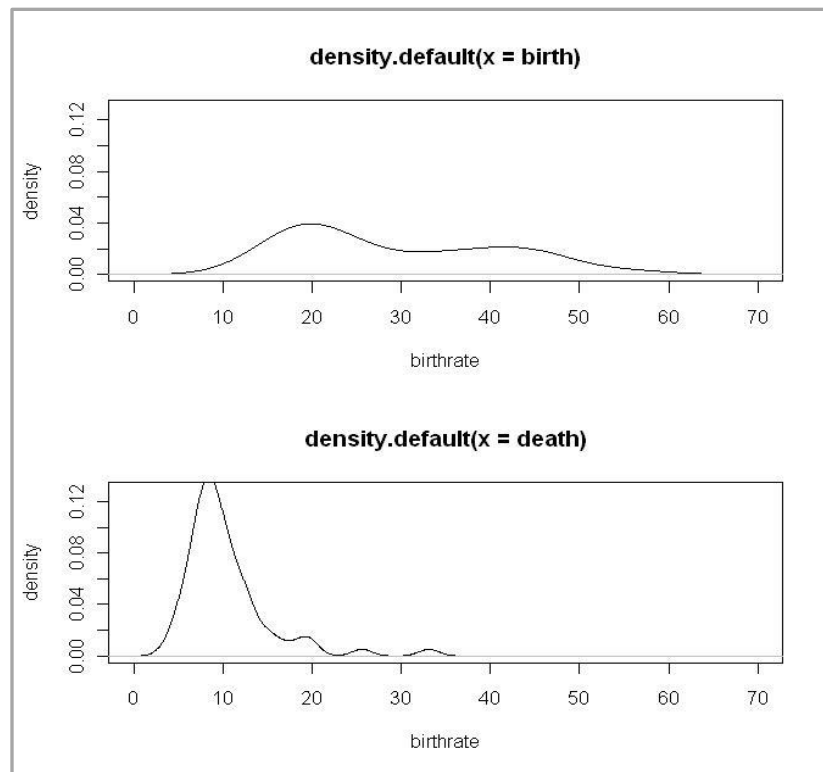
■ 히스토그램

```
> # 데이터 읽기
> rates = read.table("c:/data/birth.txt", header=T)
> attach(rates)
> # 그림이 그려질 공간확보
> par(mfrow = c(2,1))
> # 출생률에 대한 히스토그램 생성
> hist(birth, xlab="Birth Rate", ylab="frequency",
      xlim=c(0,70), ylim=c(0,40))
> # 사망률에 대한 히스토그램 생성
> hist(death, xlab="Death Rate", ylab="frequency",
      xlim=c(0,70), ylim=c(0,40))
```



커널(Kernel)밀도 추정법

```
> rates = read.table("c:/data/birth.txt", header=T)
> colnames (rates)
[1] "country" "birth" "death"
> attach(rates)
> par(mfrow = c(2,1))
> # 출생률에 대한 밀도그림 생성
> plot(density(birth), xlab="birthrate", ylab="density",
xlim=c(0,70), ylim=c(0, 0.13), type="l", axes=T)
> # 사망률에 대한 밀도그림 생성
> plot(density(death), xlab="birthrate", ylab="density",
xlim=c(0,70), ylim=c(0, 0.13), type="l", axes=T)
```



밀도 함수는 히스토그램과 전체적인 모습은 크게 다르지 않음. 하지만 히스토그램의 단점을 고려하지 않아도 되고, 분포의 부드러운 그림을 확인 할 수 있는 장점이 있음

◆ [예제3] 자료에서 남녀별 나이에 대한 줄기-잎그림 및 상자그림 그리기

```
R R Console

> stem(age[sex==1])

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

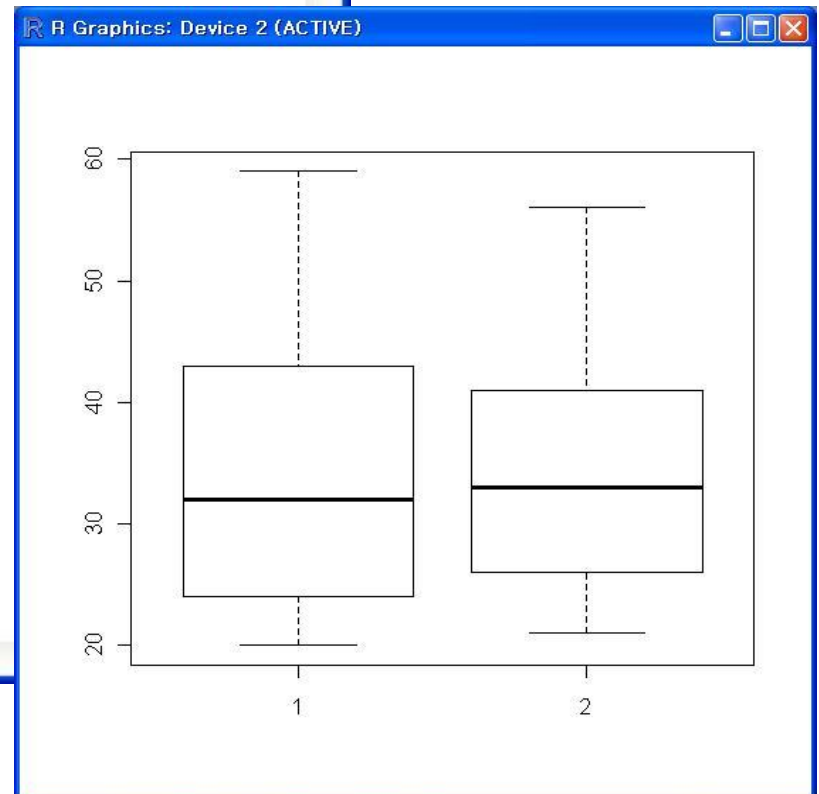
2 | 0111223
2 | 5678
3 | 1122234
3 | 5
4 | 24
4 | 67
5 | 2
5 | 569

> stem(age[sex==2])

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

2 | 124679
3 | 39
4 | 1169
5 | 6

> |
```



> boxplot(age ~ sex, data=ex91)

◆ 산점도 : 두 변수간의 관계를 나타낸 그림

예제) 다음은 Dr. Channi Kumar 가 엄마들의 심리상태에 따른 아이들의 IQ와 행동평점을 조사한 데이터 (iqdata.txt)의 일부이다. 변수 IQ와 BP의 산점도를 그려라.

또한 산점도가 그려진 상태에서, 경향을 알아볼 수 있는 직선을 추가하여라.

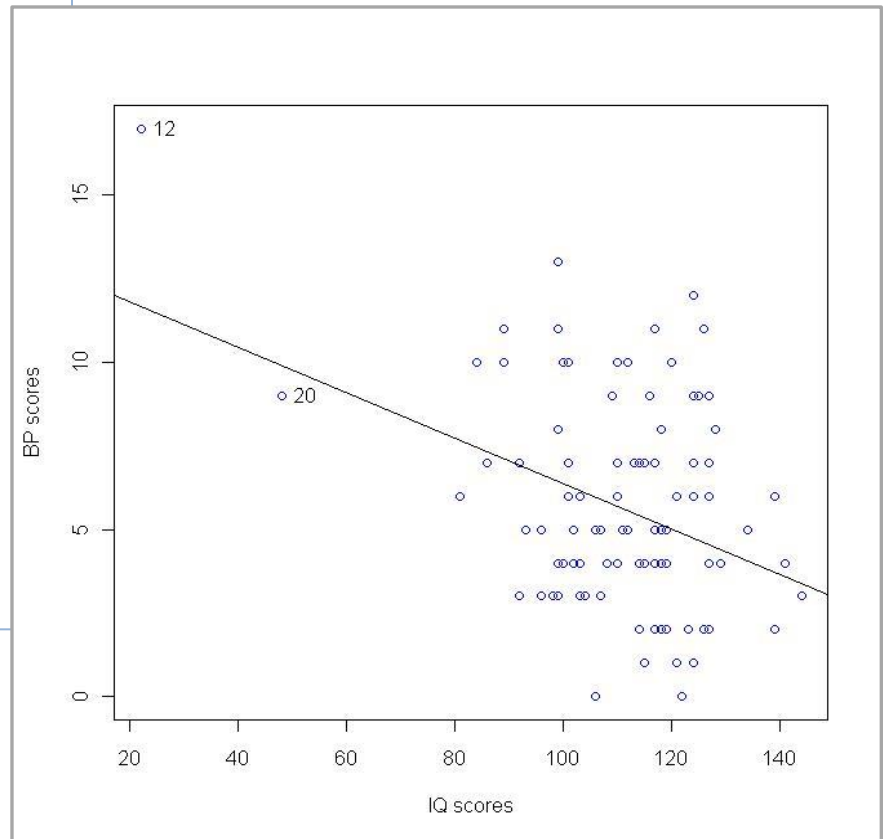
<iqdata.txt>



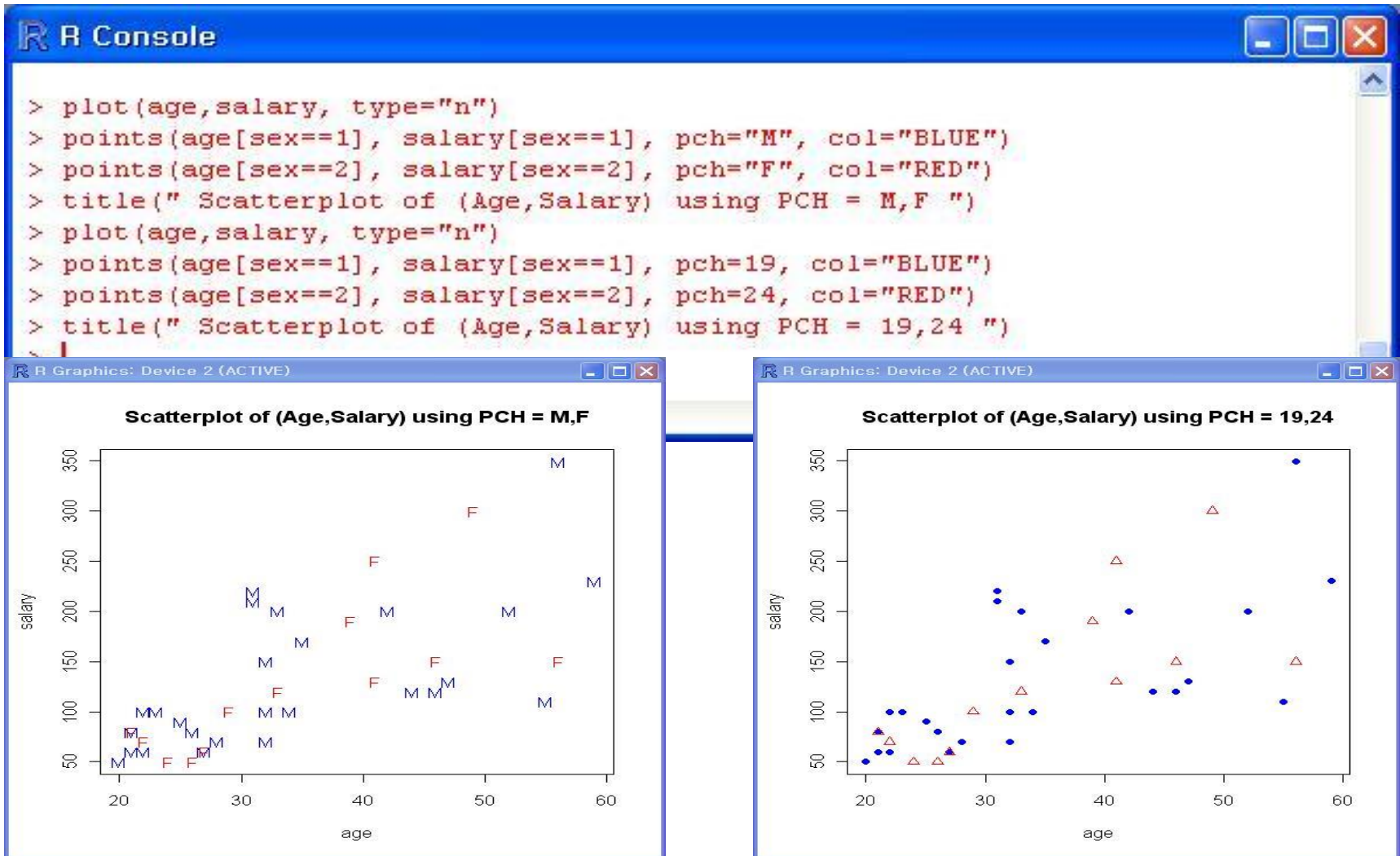
ND	103	4
ND	124	12
ND	124	9
ND	104	3
D	96	3
ND	92	3
ND	124	6
ND	99	4
ND	92	3
ND	116	9
ND	99	11
D	22	17

산점도 그리기

```
> # 데이터 입력
> iq.data=read.table("c:/data/iqdata.txt", col.names=
c("dep", "iq", "bp"))
> iq=iq.data[,2]
> bp=iq.data[,3]
> par(mfrow = c(2,1))
> # 산점도 생성
> plot(iq, bp, xlab="IQ scores", ylab="BP scores")
> # 산점도가 그려진 후 적합한 직선추가
> abline(lsfit(iq, bp))
> # 관심있는 관찰값 표시
> identify(iq, bp)
```



◆ 그룹구분 산점도 : [예제3] 자료에서 나이(X축)와 월수입(Y축)에 대한 남녀별 산점도 그리기



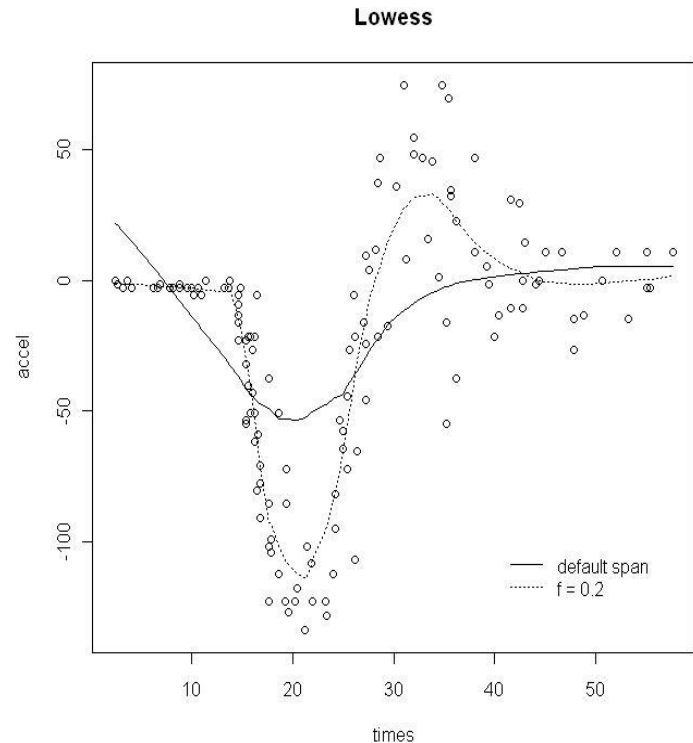
산점도 평활(Scatter smoothing)

- ◆ 두 변수 사이의 관계에 있어서 직선의 관계로 설명되지 못할 경우 산점도 평활(Scatter Smoothing)방법을 사용. 이 방법을 구성하는 두 요소로는 윈도우(Window)와 가중최소제곱법(Weighted Least Squares Method)이 있음
- ◆ 윈도우 : 산점도의 일부만을 볼 수 있게 열어 놓은 창틀의 개념
- ◆ 가중최소제곱법 : 자료가 X 값에 의하여 정렬되어 있다고 가정하고, 윈도우를 중심으로 중앙에 가까울수록 큰 가중치를 주어 회귀계수를 추정할 때 적용하는 방법이 가중최소제곱법 (WLS, Weighted Least Squares)임
- ◆ 각 개체의 X 값을 중심으로 윈도우를 만들고 가중최소제곱법을 모든 개체에 대하여 수행하면 n 개의 산점도 평활점이 만들어짐. 이 점들을 연결하면 자연스러운 곡선의 형태가 표출되는데, 이런 산점도 평활법을 LOWESS(Locally Weighted Regression Scatter- Plot Smoothing)방법이라고 함

산점도 평활(Scatter smoothing) 그리기

R 패키지 MASS에는 자동차 사고에 관한 데이터 `mcycle`이 있다. 변수는 시간(`time`)과 가속도(`accel`) 2개가 사용된다. 이 데이터에 대한 산점도를 그리고 LOWESS방법에 의해 두 변수간의 관계를 탐색하라.

```
> install.packages("MASS")
> library(MASS)
> attach(mcycle)
> # 산점도 생성
> plot(times, accel, main="Lowess")
> # 윈도우를 default 값으로 LOWESS적합
> lines(lowess(times, accel))
> # 윈도우를 작게 해서 LOWESS 적합
> lines(lowess(times, accel, f=0.2), lty=3)
> # 범례 표시
> legend(40, -100, c("default span", "f=0.2"), lty=c(1, 3),
, bty="n")
```

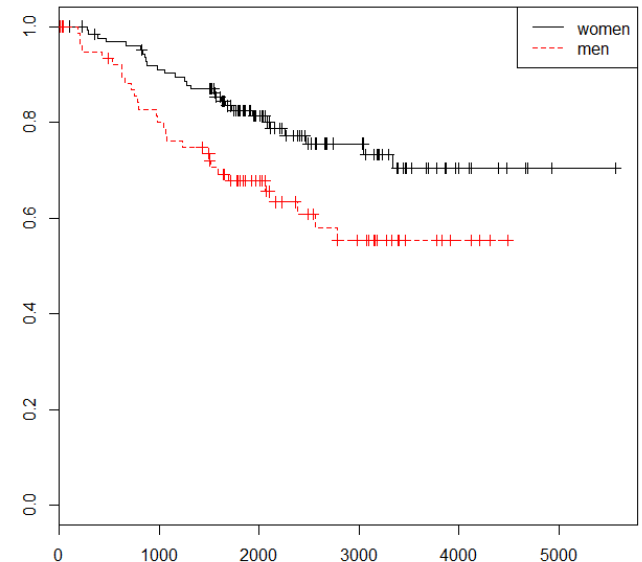


A plot of Kaplan-Meier estimates

It is often useful to plot two or more survival functions on the same plot so that they can be directly compared. To obtain survival functions split by gender, do the following:

```
> library(ISwR)
> attach(melanom)
> head(melanom)
  no status days ulc thick sex
1 789      3   10    1  676   2
2  13      3   30    2   65   2
3  97      2   35    2  134   2
```

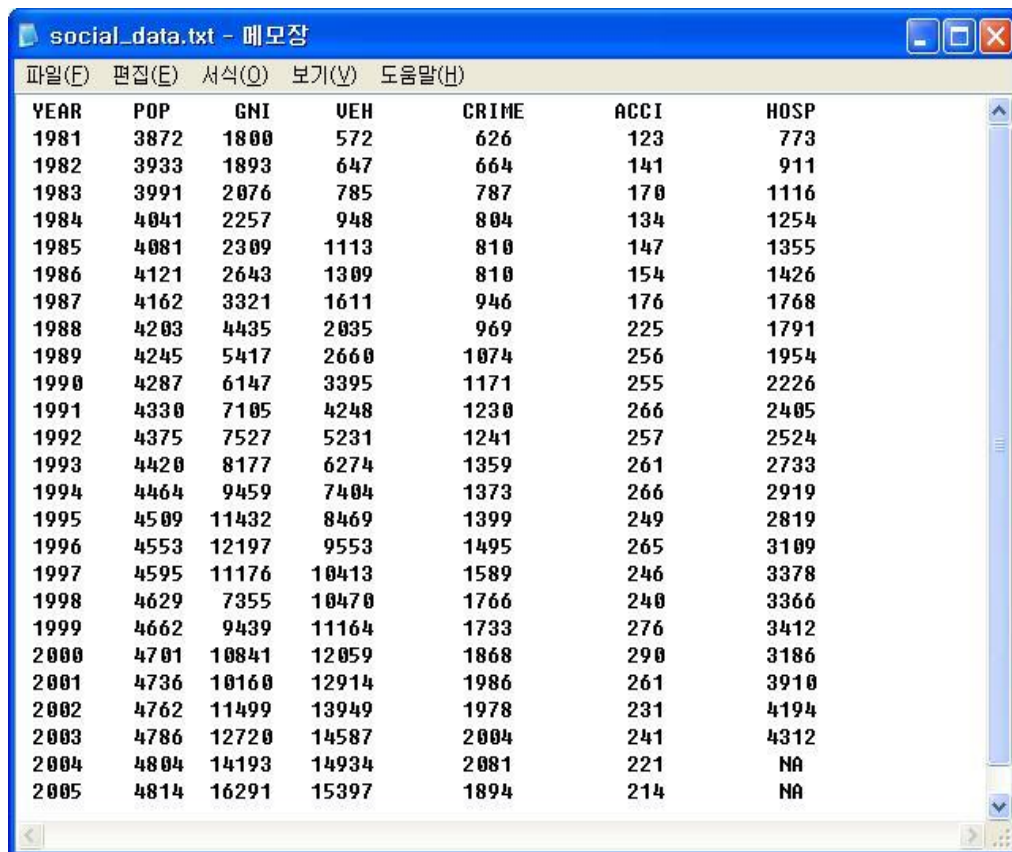
```
> sex = factor(sex, levels=c(1:2), labels=c("women", "men"))
> surv.bysex <- survfit(Surv(days,status==1)~sex)
> plot(surv.bysex, col=1:2, lty=1:2)
> legend("topright", legend=levels(sex), col=1:2, lty=1:2)
```



4. 다변량 데이터 탐색

- ◆ 예제) 한국의 각종 사회 통계(2006 한국의 사회지표)에서 산점도행렬을 그려서 변수들 간의 관계를 살펴보아라.

<social_data.txt>

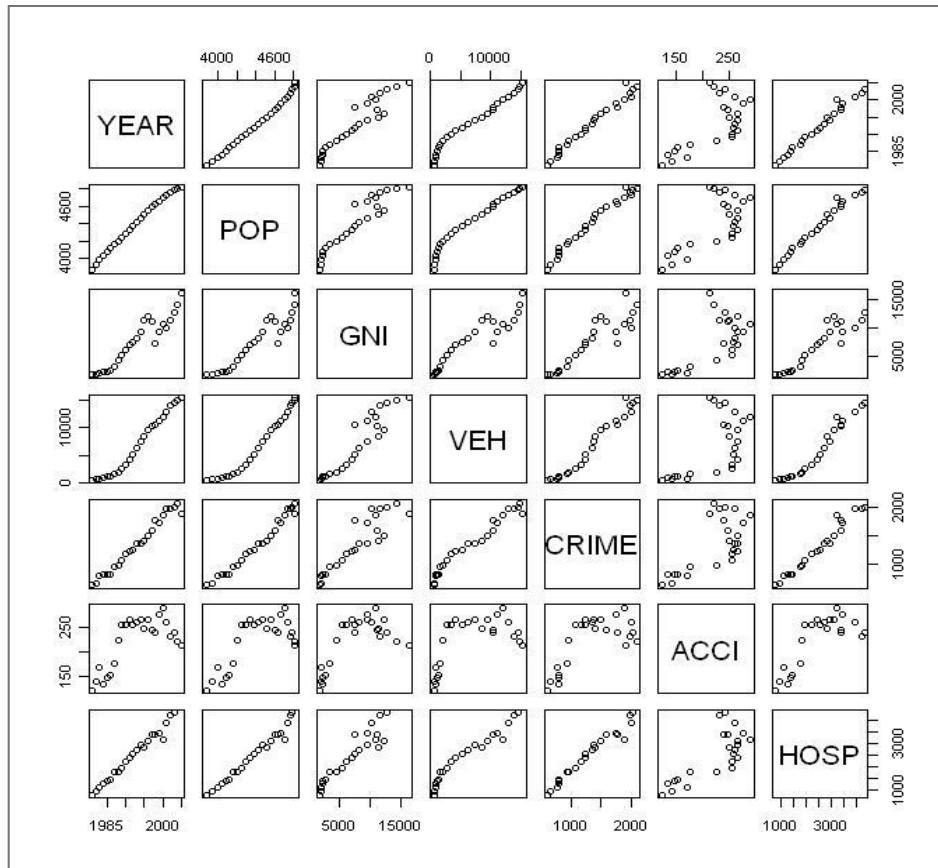


The screenshot shows a Notepad window with the title 'social_data.txt - 메모장'. The menu bar includes '파일(F)', '편집(E)', '서식(O)', '보기(V)', and '도움말(H)'. The text area contains a table with 7 columns: YEAR, POP, GNI, UEH, CRIME, ACCI, and HOSP. The data spans from 1981 to 2005, with the last two years (2004 and 2005) having 'NA' for the HOSP variable.

YEAR	POP	GNI	UEH	CRIME	ACCI	HOSP
1981	3872	1800	572	626	123	773
1982	3933	1893	647	664	141	911
1983	3991	2076	785	787	170	1116
1984	4041	2257	948	804	134	1254
1985	4081	2309	1113	810	147	1355
1986	4121	2643	1309	810	154	1426
1987	4162	3321	1611	946	176	1768
1988	4203	4435	2035	969	225	1791
1989	4245	5417	2660	1074	256	1954
1990	4287	6147	3395	1171	255	2226
1991	4330	7105	4248	1230	266	2405
1992	4375	7527	5231	1241	257	2524
1993	4420	8177	6274	1359	261	2733
1994	4464	9459	7404	1373	266	2919
1995	4509	11432	8469	1399	249	2819
1996	4553	12197	9553	1495	265	3109
1997	4595	11176	10413	1589	246	3378
1998	4629	7355	10470	1766	240	3366
1999	4662	9439	11164	1733	276	3412
2000	4701	10841	12059	1868	290	3186
2001	4736	10160	12914	1986	261	3910
2002	4762	11499	13949	1978	231	4194
2003	4786	12720	14587	2004	241	4312
2004	4804	14193	14934	2081	221	NA
2005	4814	16291	15397	1894	214	NA

산점도 행렬 그리기

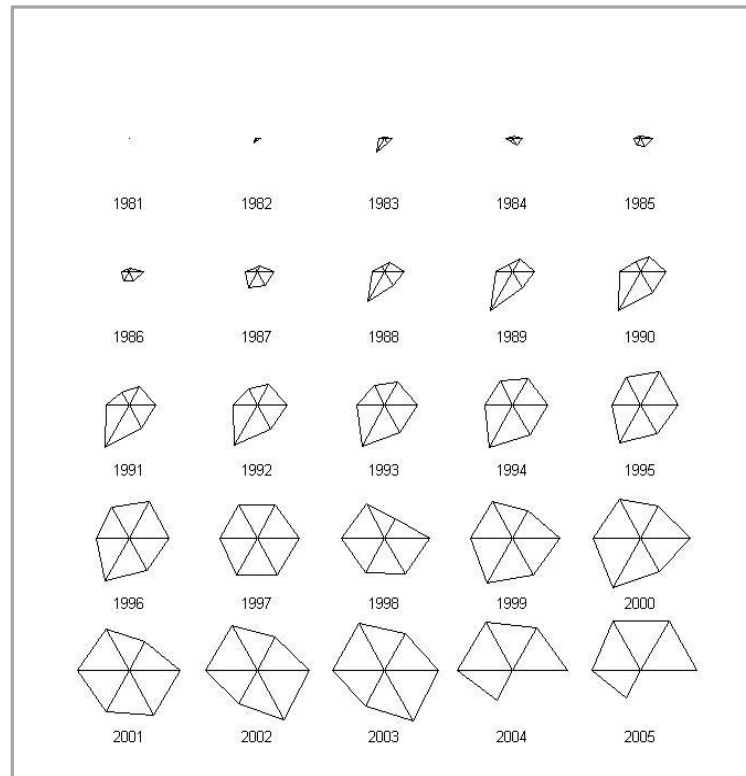
```
> social.data=read.table("c:/data/social_data.txt", header=T)  
> pairs(social.data)
```



별그림(Star Plot)

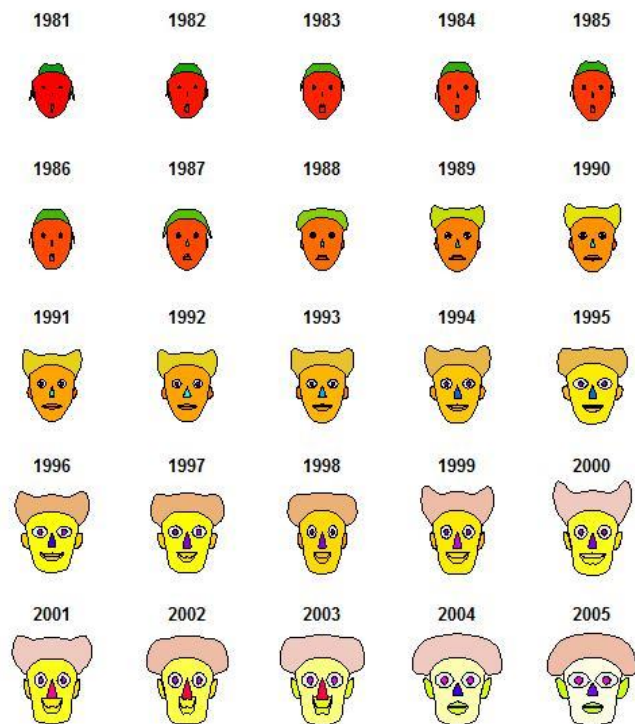
- ◆ 별그림 : 별모양의 점을 각각의 변수에 대응하도록 한 뒤, 각각의 변수 값에 비례하도록 반경(radius)을 나타내도록 하여 관찰 값을 표시한 것
- ◆ 별의 크기와 모양을 가지고 변수의 관계 및 유사한 관찰 값을 찾는데 이용됨

```
> social_data=read.table("c:/data/social_data.txt",  
  header=T)  
> social= social_data[,-1]  
> id_name= social_data[,1]  
> row_names(social)=id.name  
> stars(social)
```



얼굴그림(Face Plot)

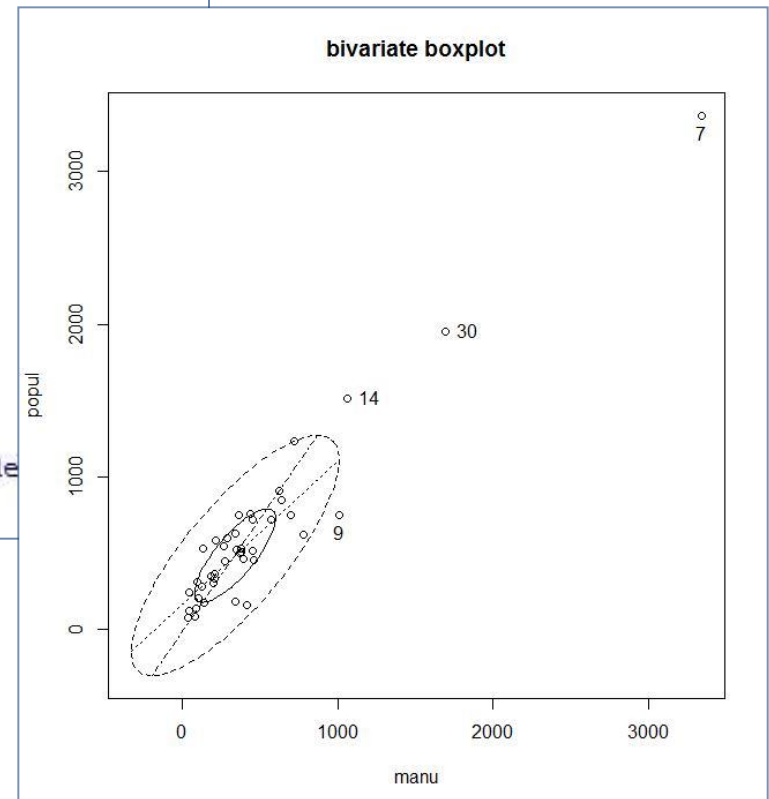
```
> library(aplpack)
> faces(social, face.type=1)
[1] "Warning: NA elements have been exchanged by mean values!!"
effect of variables:
modified item      Var
"height of face"   " "POP"
"width of face"    " "GNI"
"structure of face" " "VEH"
"height of mouth"  " "CRIME"
"width of mouth"   " "ACCI"
"smiling"          " "HOSP"
"height of eyes"   " "POP"
"width of eyes"    " "GNI"
"height of hair"   " "VEH"
"width of hair"    " "CRIME"
"style of hair"    " "ACCI"
"height of nose"   " "HOSP"
"width of nose"    " "POP"
"width of ear"     " "GNI"
"height of ear"    " "VEH"
> |
```



Bivariate boxplot 그리기

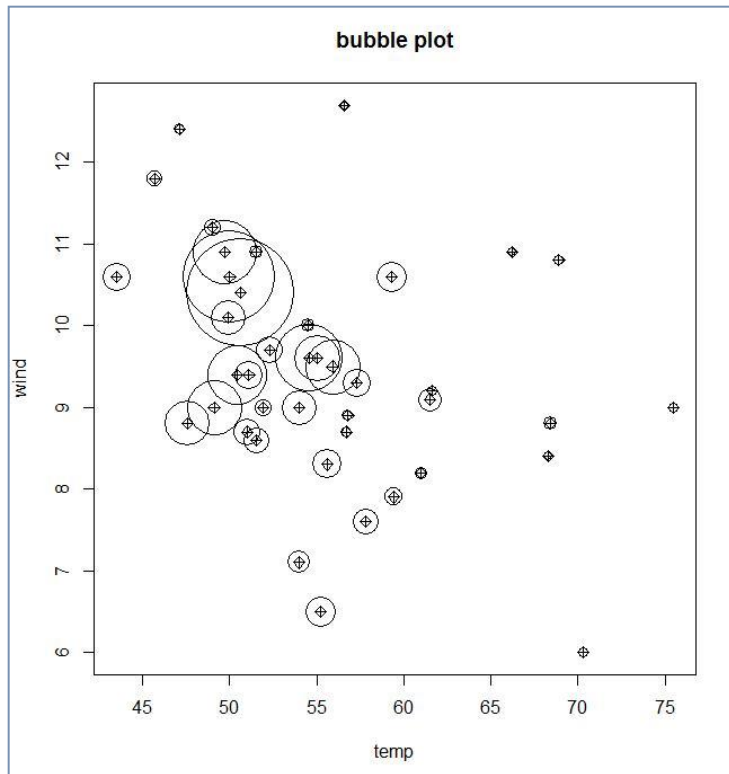
```
> library(HSAUR2)
> library(MVA)
> data(USairpollution)
> head(USairpollution)
      SO2 temp manu popul wind precip predays
Albany   46 47.6   44  116  8.8  33.36    135
Albuquerque 11 56.8   46  244  8.9   7.77     58
Atlanta   24 61.5  368  497  9.1  48.34    115
Baltimore 47 55.0  625  905  9.6  41.31    111
Buffalo   11 47.1  391  463 12.4  36.11    166
Charleston 31 55.2   35   71  6.5  40.75    148
> plot(manu, popul)
> x = USairpollution[,c(3,4)]
> bvbox(x, xlab="manu", ylab="popul")
> title("bivariate boxplot")
> identify(x)
[1] 7 9 14 30
> rownames(x)[c(7,9,14,30)]
[1] "Chicago"      "Cleveland"     "Detroit"       "Philadelphia"
> |
```

bvbox : bivariate boxplot 함수 (R package MVA)
identify(x) : 특이값을 표시



Bubble plot 그리기

```
> plot(wind~temp, data=USairpollution, pch=9)  
> with(USairpollution, symbols(temp, wind, circles=SO2, inches=0.5, add=T))  
> title("bubble plot")  
> |
```



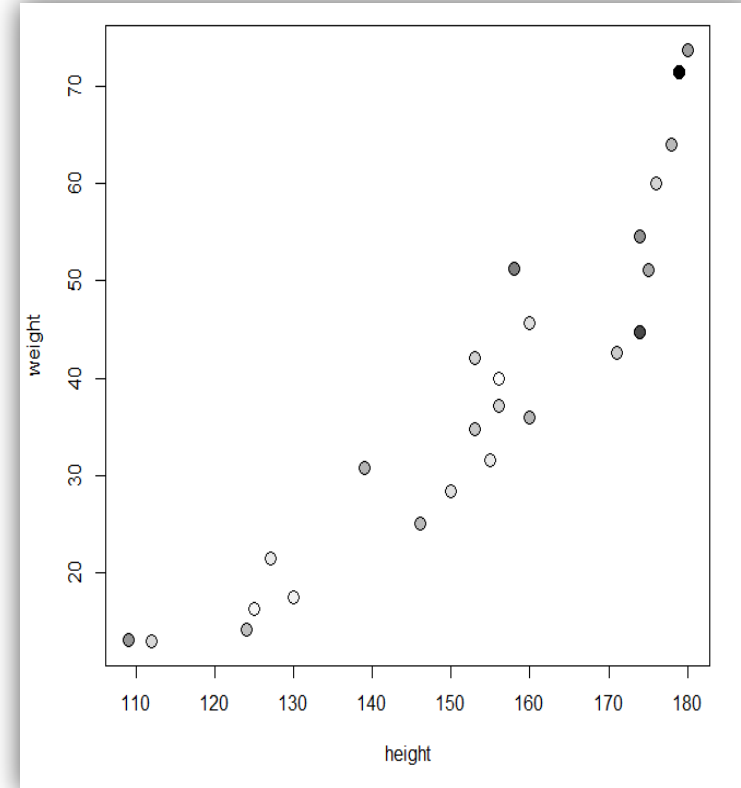
(temp, wind)의 산점도에 제3의 변수인 SO2의
정보의 크기에 따라 원으로 나타낸 그림

Color Coding

➤ How to present the quantities graphically ?

Here, we see how to display the value of Cook's distance (which is always positive) graphically for a model where pemax is described using height and weight using **color coding** .

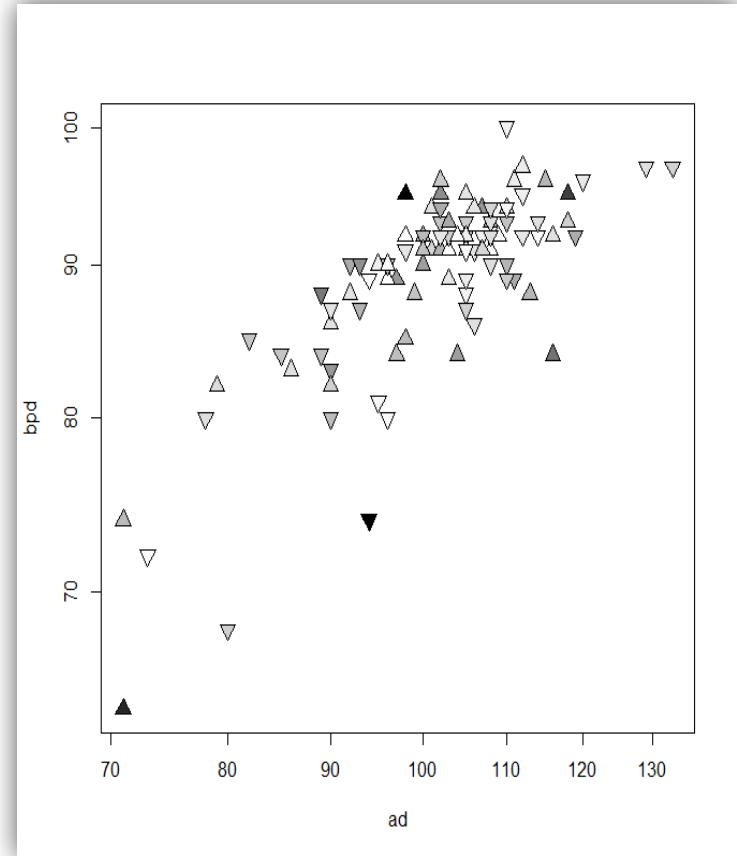
```
> library(lswR)
> data(cystfibr)
> cookd <- cooks.distance(lm(pemax~height+weight,
                             data=cystfibr))
> cookd <- cookd/max(cookd)
> cook.colors <- gray(1-sqrt(cookd))
> plot(height,weight,bg=cook.colors,pch=21,cex=1.5)
> points(height,weight,pch=1,cex=1.5)
```



Using different symbols

- You can use similar techniques to describe other influence measures. In the case of signed measures, you might use different symbols for positive and negative values.

```
> library(ISwR)
> data(secher)
> attach(secher)
> rst <- rstudent(lm(log10(bwt)~log10(ad)+log10(bpd)))
> range(rst)
[1] -3.707509 3.674050
> rst <- rst/3.71
> plot(ad,bpd,log="xy",bg=gray(1-abs(rst)),
       pch=ifelse(rst>0, 24, 25), cex=1.5)
>
```

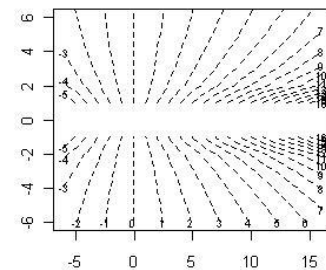
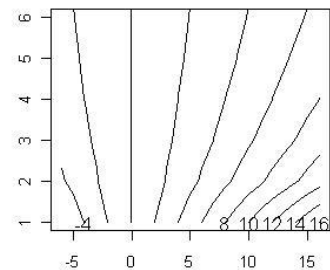
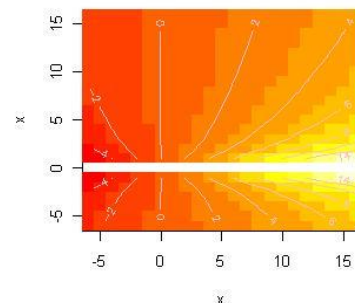
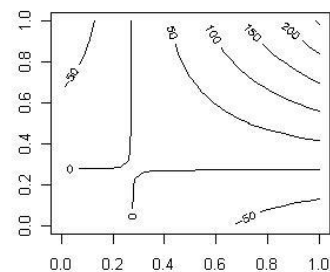


등고선 그림(Contour Plot)과 이미지 그림(Image Plot)

◆ 등고선그림 : 자료를 나타내기 위하여 두 좌표의 격자(grad)위에 등고선으로 화면을 그려줌

◆ 이미지 그림 : 색깔을 이용하여 z변수의 이미지를 표현한 그래프

```
> x<- -6:16
> par(mfrow=c(2, 2))
> contour(outer(x, x), method="edge", vfont=c('sans serif', "plain"))
> z <- outer(x, sqrt(abs(x)), FUN="/")
> image(x, x, z)
> contour(x, x, z, col="pink", add=TRUE,
  method="edge", vfont=c("sans serif", "plain"))
> contour(x, x, z, ylim=c(1,6), method="simple",
  laboex=1)
> contour(x, x, z, ylim=c(-6,6), nlev=20, lty=2 ,
  method="simple")
```



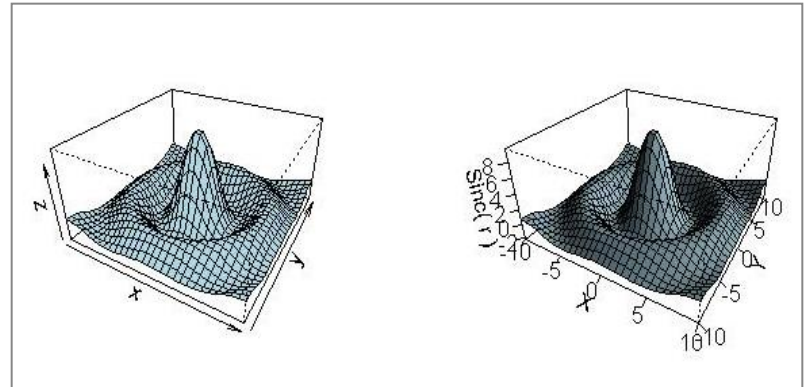
겨냥도 그림(Perspective Plot)

◆ 겨냥도 그림 :

등간격의 격자상에 높이 값을 갖는 행렬 자료에 대한 3차원 표현 방법 높이는 선으로 연결

◆ 두 변수에 의한 밀도 추정값을 입체적으로 시각적으로 표현한 그림

```
> par(mfrow=c(1, 2))  
> x <- seq(-10, 10, length=30)  
> y <- x  
> f <- function(x, y) {r <- sqrt(x^2+y^2); 10*sin(r)/r}  
> z <- outer(x, y, f)  
> z[is.na(z)] <- 1  
> persp(x, y, z, theta=30, phi=30, expand=0.5,  
  col="lightblue")  
> persp(x, y, z, theta=30, phi=30, expand=0.5,  
  col="lightblue", ltheta=120, shade=0.75,  
  ticktype="detailed", xlab="X", ylab="Y",  
  zlab="Sinc(r)")
```



5. lattice 활용

lattice 패키지

- ◆ lattice : R에서 Trellis 그림을 그릴 수 있는 패키지
- ◆ Trellis 그림 : 자료의 정보들을 통계그래픽스를 통하여 보다 정확하고 충실하게 전달할 수 있도록 하기 위하여 Bill Cleveland가 제안한 그래픽의 '디자인 원칙'을 구현해 놓은 그림. 가장 큰 특징은 R base에서 제공하고 있는 기본 그래픽 함수와는 달리 멀티패널 조건 (multipanel conditioning)을 제공하고 있어 쉽게 자료를 범주형 변수의 범주에 따라 나누어 그림을 그려 비교해 볼 수 있도록 함.

참고문헌 :

Sarkar, D. (2008). Lattice: multivariate data visualization with R. Springer Science & Business Media.

http://www.isid.ac.in/~deepayan/R-tutorials/labs/04_lattice_slides.pdf

lattice 패키지 활용 예

◆ 데이터 : tipping

레스토랑 고객들의 팁에 대한 습성을 알아보기 위하여 미국 뉴욕 근교에서 수집된 자료로 전체 가격(totbill), 팁(tip), 계산한 사람의 성별(sex), 흡연석/금연석(smoker), 요일(day), 점심/저녁(time), 일행 수(size)에 관한 정보들이 있는 자료

```
> library(lattice)
> tipping = read.csv("c:/data/tips.csv", header=T)
> tipping$tiprate <- tipping$tip/tipping$totbill * 100
> dim(tipping)
[1] 244  9
> head(tipping,3)
  obs totbill  tip  sex  smoker day  time size  tiprate
1   1  16.99 1.01 Female Non-smoker Sun Dinner    2  5.944673
2   2  10.34 1.66  Male Non-smoker Sun Dinner    3 16.054159
3   3  21.01 3.50  Male Non-smoker Sun Dinner    3 16.658734
```

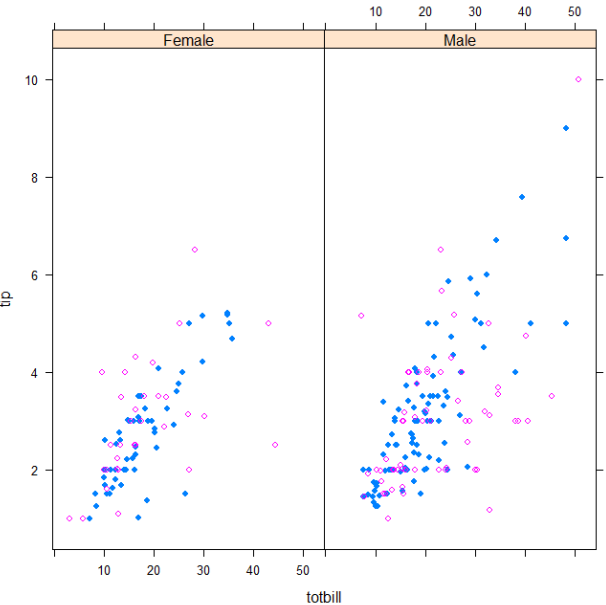
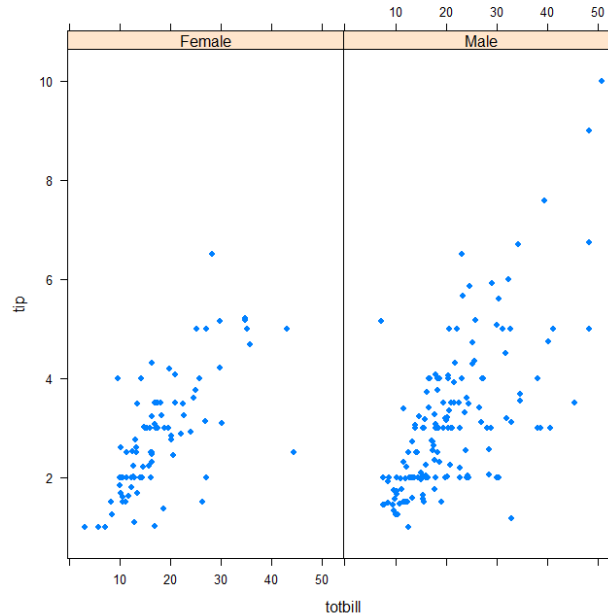
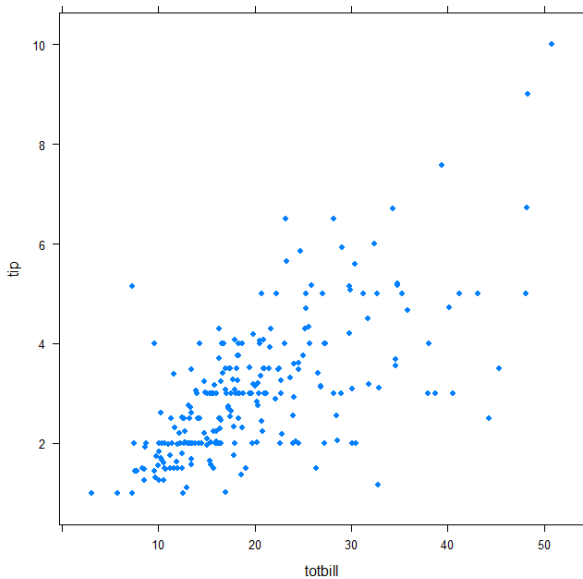
lattice 패키지 활용 예

1) 이변량 그래프 : 연속변수 vs 연속변수

```
> xyplot(tip ~ totbill, pch=16, data = tipping) #1
```

```
> xyplot(tip~totbill | sex, pch=16, data = tipping) #2
```

```
> xyplot(tip~totbill | sex, group = smoker, pch = c(16,1), data = tipping) #3
```



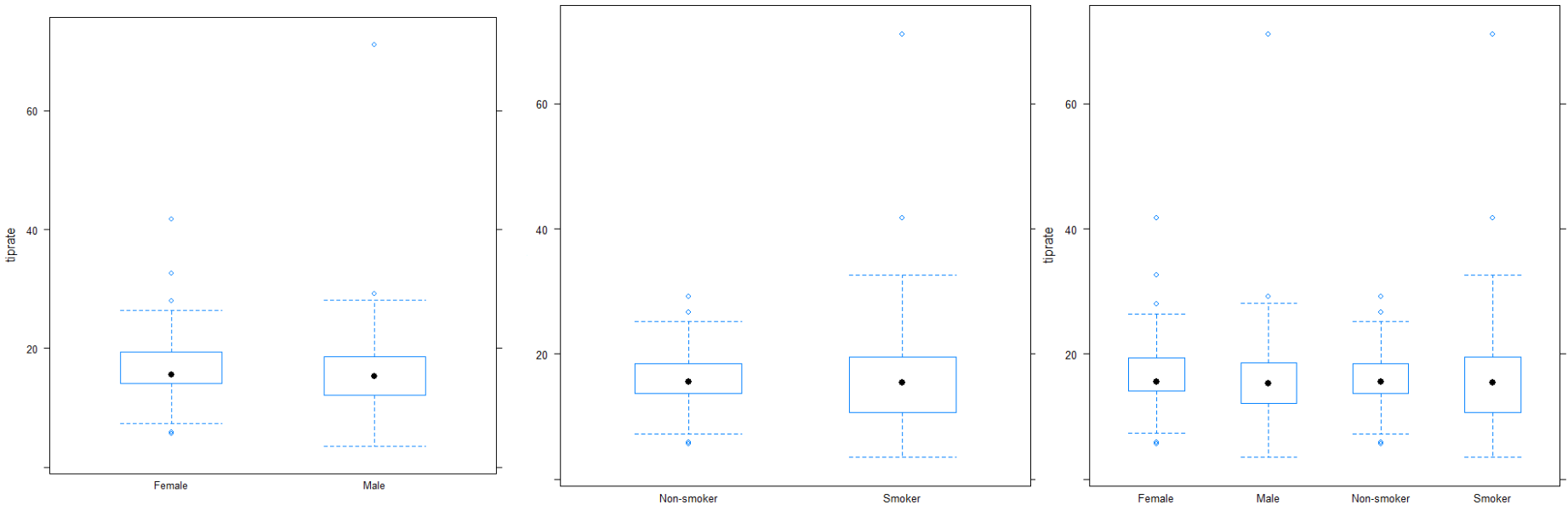
lattice 패키지 활용 예

2) 이변량 그래프 : 연속변수 vs 범주형변수

```
> bwplot( tiprate ~ sex, data=tipping) #1
```

```
> bwplot( tiprate ~ smoker, data=tipping) #2
```

```
> bwplot( tiprate ~ sex+smoker , data=tipping) #3
```



lattice 패키지 활용 예

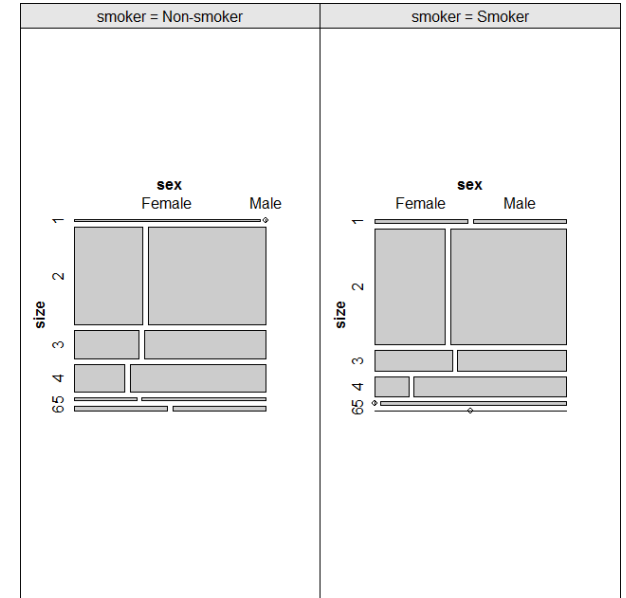
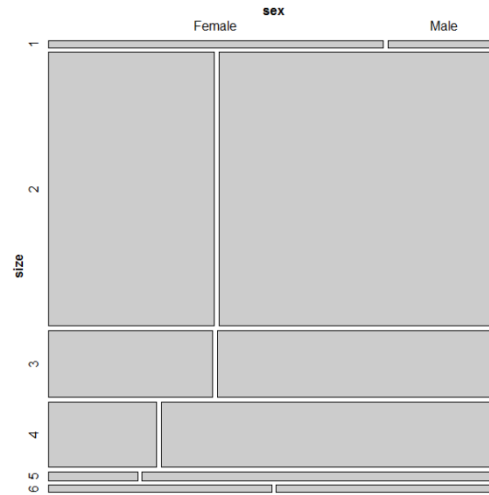
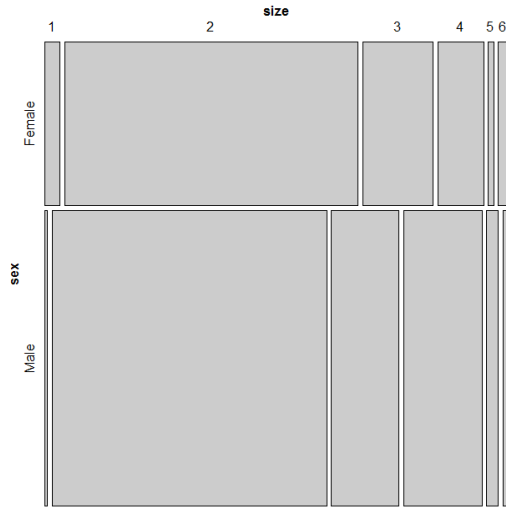
3) 이변량 그래프 : 범주형 변수 vs 범주형 변수 (mosaic 그림)

```
> library(vcd)
```

```
> mosaic(~ sex + size, data = tipping) # sex 각 범주내에서 size 변수의 범주 비율 표시 1
```

```
> mosaic(~ size + sex, data = tipping) # size 각 범주내에서 sex 변수의 범주 비율 표시 2
```

```
> cotabplot(~ size+sex | smoker, data = tipping, panel = cotab_mosaic) # 3
```



6. ggplot2 활용

ggplot2

- ◆ High-level graphics system
- ◆ Implements grammar of graphics from Leland Wilkinson(2005)

소개 : http://www.ling.upenn.edu/~joseff/rstudy/summer2010_ggplot2_intro.html

매뉴얼 및 책자 : <http://ggplot2.org/>

```
> install.packages("ggplot2")
```

함수

- **qplot** : ggplot2에서 제공하고 있는 함수로 R의 base에서 제공하고 있는 plot 함수보다 좀 더 편리하게 그림을 그릴 수 있다
- **ggplot** 함수 : 그래픽 문법의 요소들을 다양한 자료에 적용하여 하나의 그림으로 표현할 수 있도록 되어 있어 qplot보다 좀 더 복잡하고 정교한 그림을 그릴 수 있다.

ggplot2 활용 예

◆ 데이터

abalone 자료 – UCI machine learning repository

(<http://archive.ics.uci.edu/ml/datasets/Auto+MPG>)에 저장

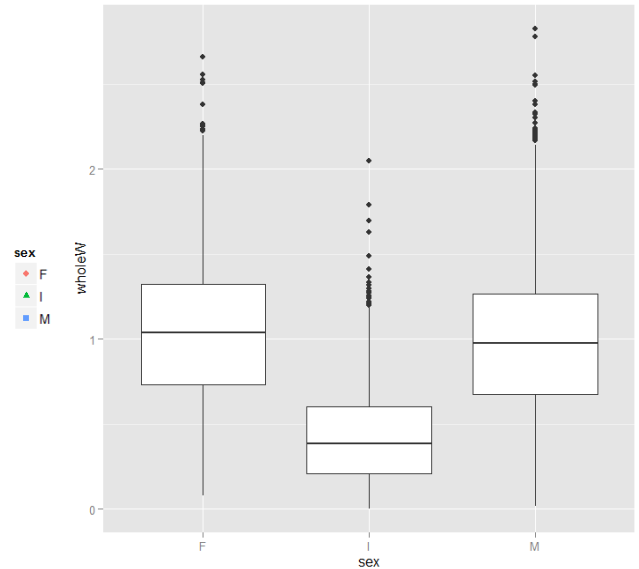
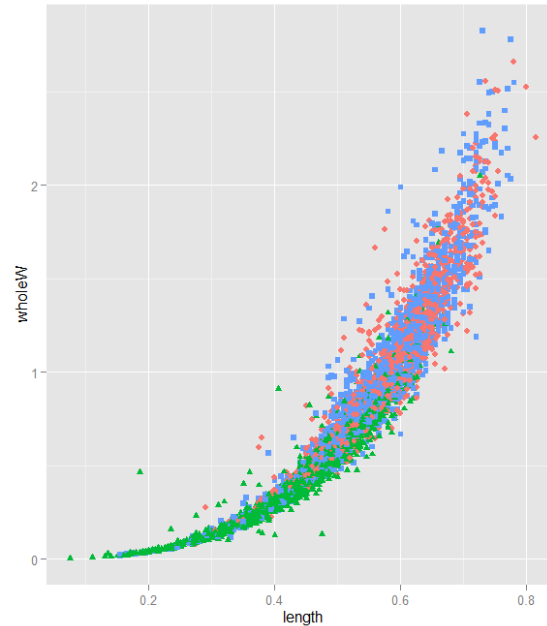
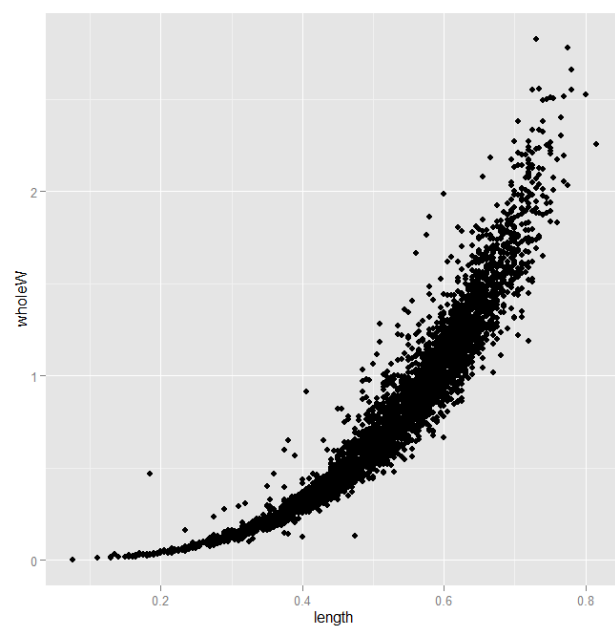
전복의 나이를 알 수 위하여 다른 신체적 측정치들을 이용하여 나이를 예측하는 방법을 개발하기 위하여 수집된 자료

```
> abalone<-read.csv("e:/data/abalone.csv", header=TRUE)
> dim(abalone)
[1] 4177    9
> head(abalone,3)
  sex length diameter height wholeW shuckedW visceraW shellW rings
1  M  0.455    0.365  0.095 0.5140   0.2245   0.1010   0.15    15
2  M  0.350    0.265  0.090 0.2255   0.0995   0.0485   0.07     7
3  F  0.530    0.420  0.135 0.6770   0.2565   0.1415   0.21     9
```

qplot 예

■ 산점도 , 상자그림

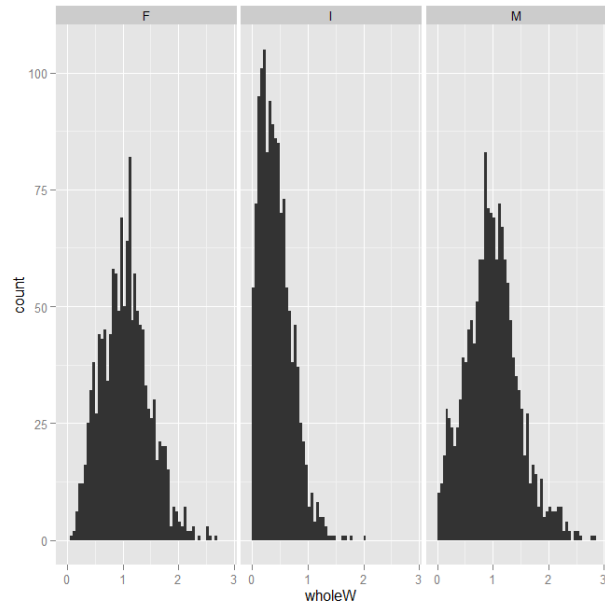
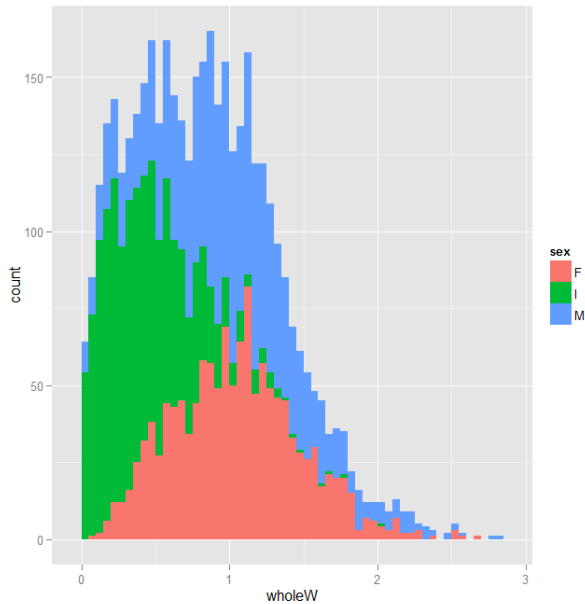
```
> library(ggplot2)
> qplot(length, wholeW, data = abalone) #1
> qplot(length, wholeW, data = abalone, colour=sex, shape=sex) #2
> qplot(sex, wholeW, data = abalone, geom="boxplot") #3
```



qplot 예

■ 히스토그램

```
> qplot(wholeW, data = abalone, geom = "histogram" , binwidth = 0.05,  
      fill = sex)  
> qplot(wholeW,data = abalone, geom="histogram", binwidth=0.05,  
      facets=~sex)
```



ggplot 함수를 이용한 층화 그래픽(layered graphics) 예 : 산점도

```
> tipping <- read.csv("tips.csv", header=T)
> tipping$tiprate <- tipping$tip/tipping$totbill*100
> plot.basic <- ggplot(tipping,
  aes(x=totbill, y=tip, color=sex, shape=sex, size=tiprate))
> plot.basic + layer(geom="point") #1
> plot.basic + geom_point() + geom_smooth(aes(group=sex)) #2
> plot.scale1 <- ggplot(tipping, aes(x=totbill, y = tip,
  color = sex, shape = sex, size = size)) + geom_point()
> plot.scale1 + scale_color_hue("Gender", labels = c("여자", "남자")) #3
```

