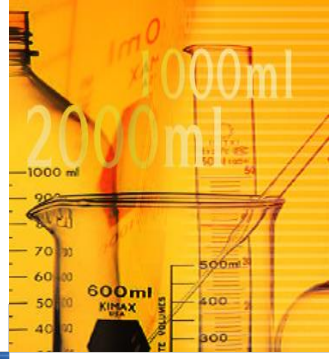
R 프로그래밍



Chapter 8

데이터 시각화 (2)

Sejong Oh

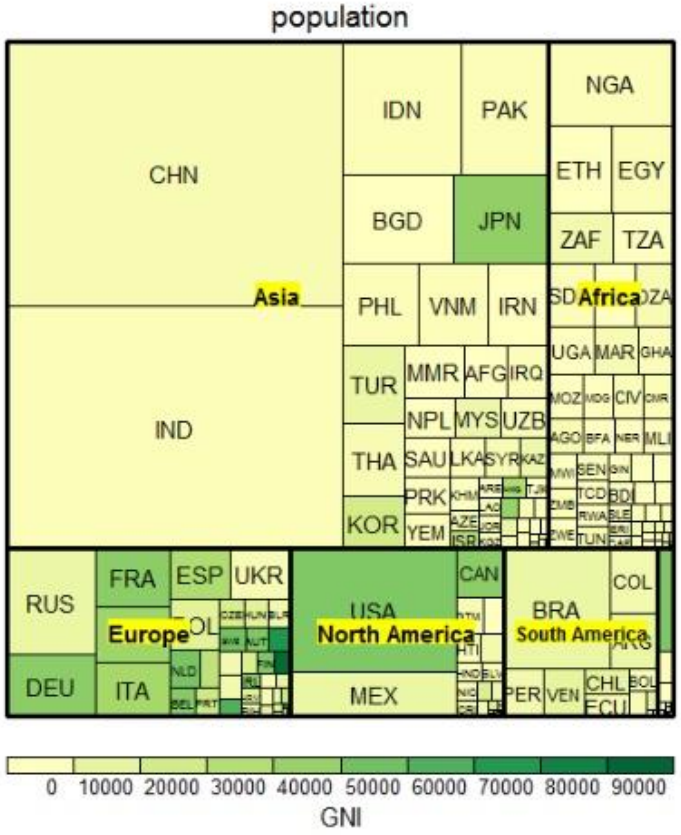
Bio Information technology Lab.

Content

- 나무지도
- 버블차트
- 다중상자그림
- 기타 그래프(self-study)

나무지도(tree map)

- 나무지도는 데이터가 갖는 계층구조를 타일 모양으로 표현한 것
- 타일은 계층적 속성을 가지며, 계층은 컬러로 표현된다



참조 : http://blog.daum.net/huh420/19

나무지도(tree map)

- 설치가 필요한 패키지
  - treemap

- 실습에 사용할 데이터셋
  - GNI2014 (treemap)
  - 208개 국가의 1인당 총소득(gross national income) 데이터
  - 국가는 대륙(continent)으로 그룹핑되고 국가명은 국제표준(iso3)으로 지칭된다.
  - 국가정보는 population(인구)과 GNI(1인당 국민소득)이다

```
> head(GNI2014)
  iso3           country     continent population    GNI
3  BMU           Bermuda North America      67837 106140
4  NOR            Norway        Europe    4676305 103630
5  QAT             Qatar          Asia     833285  92200
6  CHE       Switzerland        Europe    7604467  88120
7  MAC Macao SAR, China          Asia     559846  76270
8  LUX        Luxembourg        Europe     491775  75990
```

나무지도(tree map)

```
library(treemap)
data(GNI2014)                          # 데이터 불러오기
str(GNI2014)                           # 데이터 내용보기
treemap(GNI2014,
        index=c("continent","iso3"),
        vSize="population",   # 타일의 크기
        vColor="GNI",         # 타일의 컬러
        type="value",         # 타일 컬러링 방법
        bg.labels="yellow")   # 레이블의 배경색
```

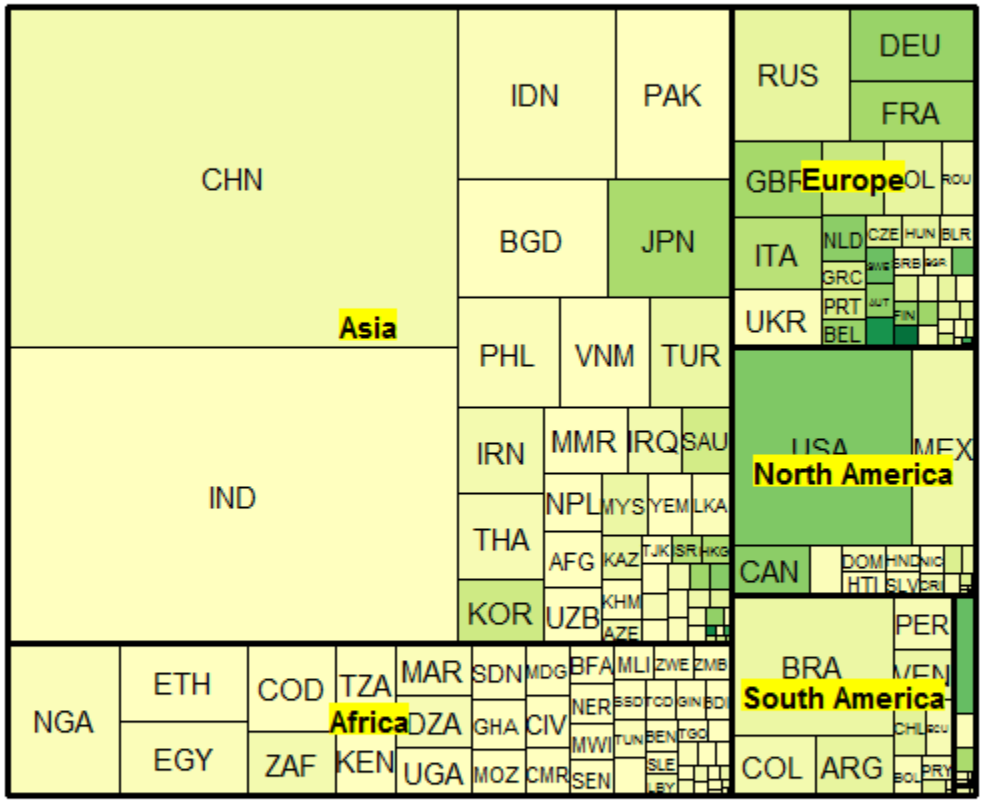
index=c("continent","iso3")

: 개체의 단위를 지정하는데 계층적 구조를 갖는 경우 상위 층을 먼저 넣는다. 대륙을 먼저 표현하고 그 안에 국가를 넣으라는 의미

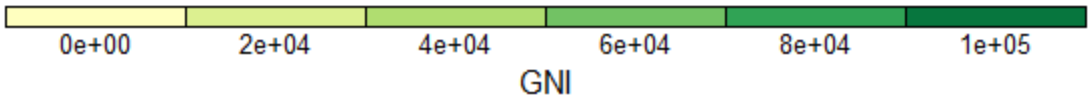
type="value"

: **vColor** 에서 지정한 값에 의해서 타일의 컬러가 결정됨

population



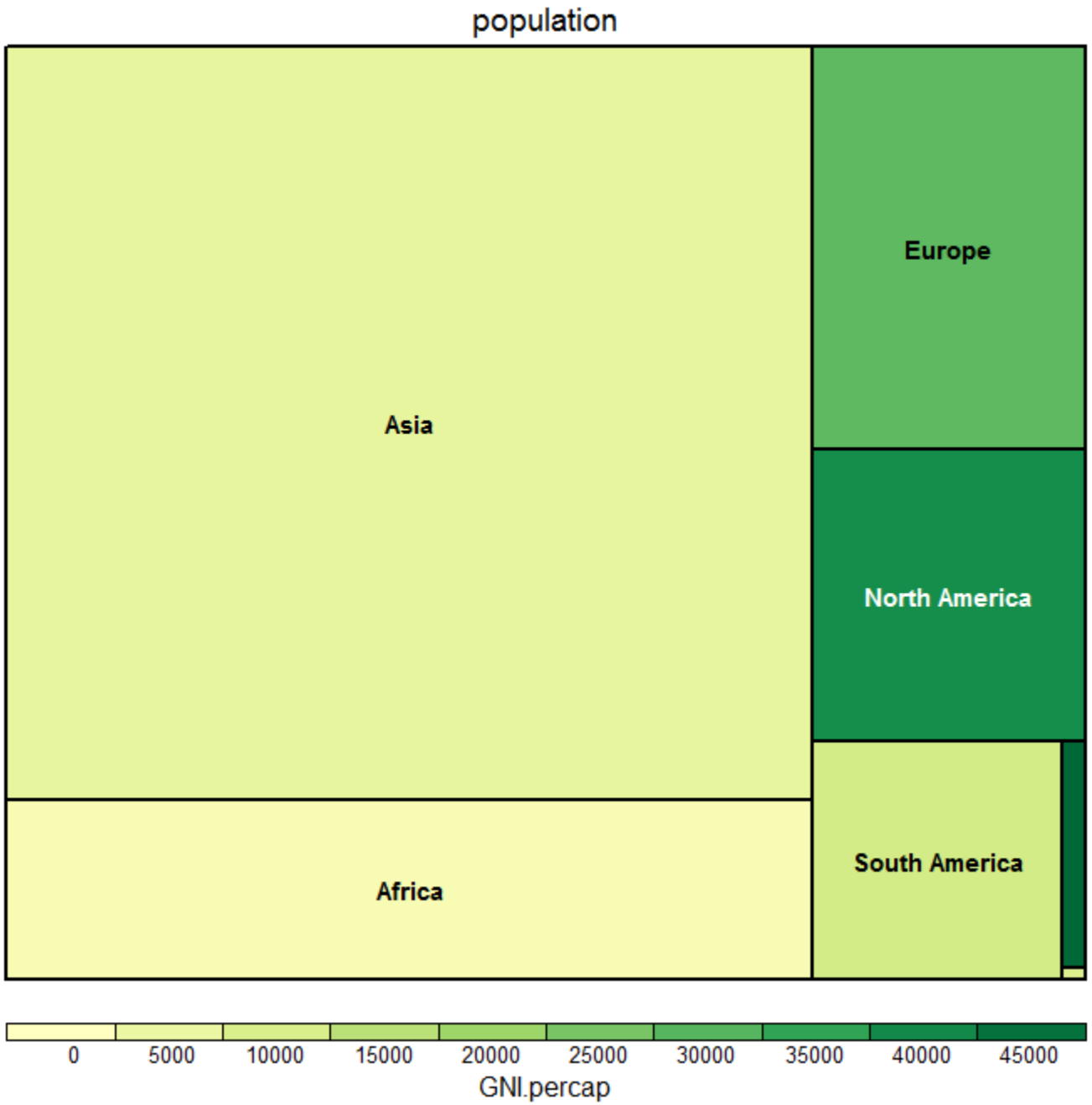
타일의 크기 : 국토면적
타일의 색 : 국민소득



나무지도(tree map)

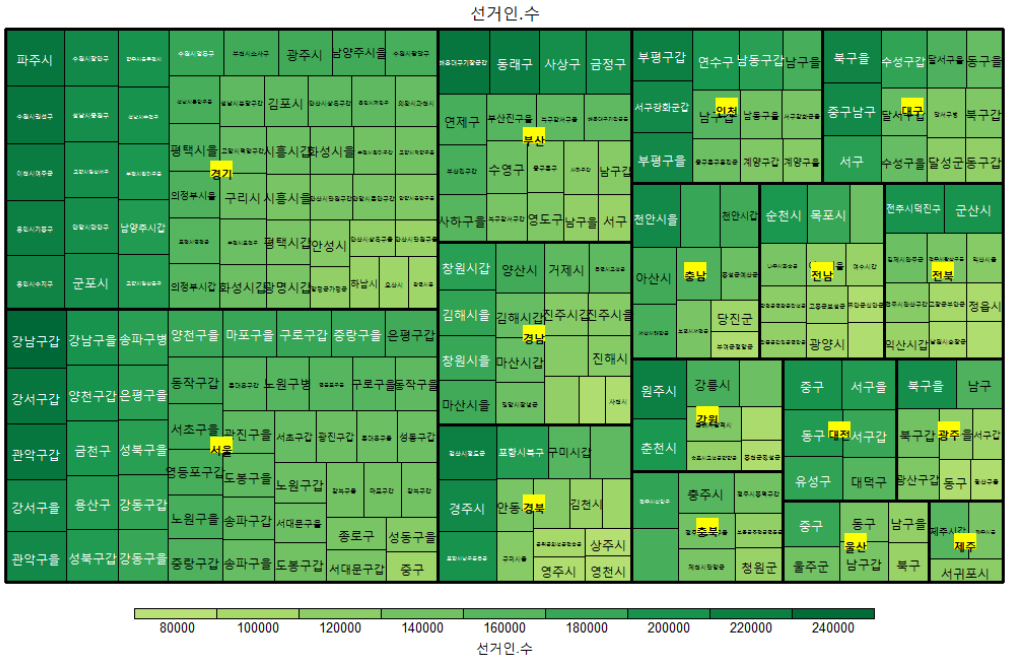
```
# 국가별 국민 총소득을 계산해서 GNI.total 컬럼에 저장
GNI2014$GNI.total <-
    GNI2014$population*GNI2014$GNI
head(GNI2014)
# 국가별 국민 총소득을 대륙별로 합계내서 GNI2014.a 에 저장
GNI2014.a <- aggregate(GNI2014[,4:6],
                by=list(GNI2014$continent),sum)
# 대륙별 합계를 대륙 인구수로 나누어 GNI.percap 컬럼에 저장
GNI2014.a$GNI.percap <-
    GNI2014.a$GNI.total/GNI2014.a$population

treemap(GNI2014.a,
        index=c("Group.1"),
        vSize="population",
        vColor="GNI.percap",
        type="value",
        bg.labels="yellow")
```


aggregate(): 배열에 저장된 값을 기준 컬럼에 의해 집계값를 계산한다


population
Asia
Europe
North America
South America
Africa
0
5000
10000
15000
20000
25000
30000
35000
40000
45000
GNI.percap

[연습 1]

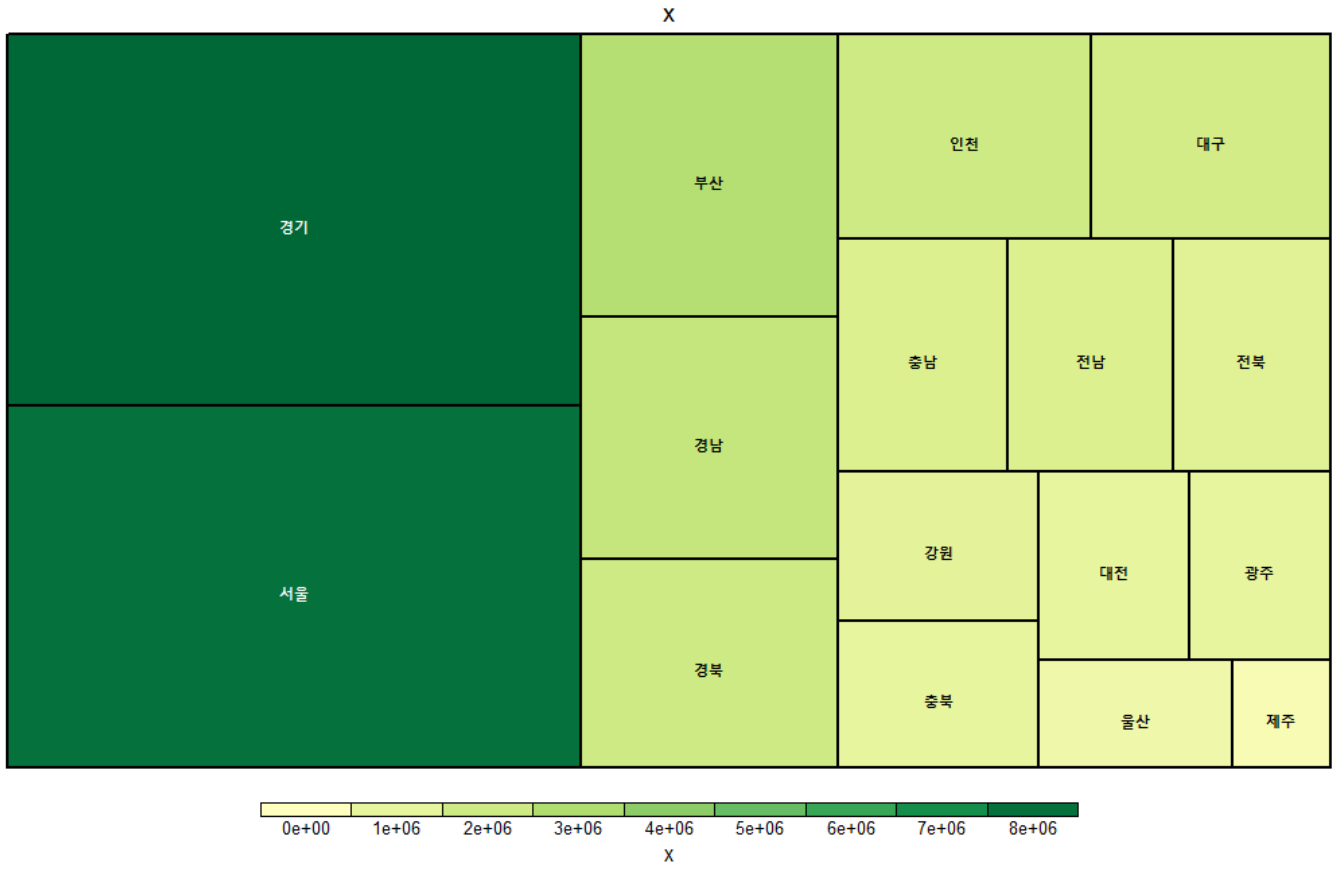
- 1. “국회의원_선거구_유권자수.csv” 파일의 내용을 가지고 다음과 같은 treemap 을 작성하시오



- 타일 하나는 각 선거구를 의미
- 굵은 검은띠 블록은 선거구가 속한 시도를 의미
- 타일의 면적, 색깔은 선거인수를 의미

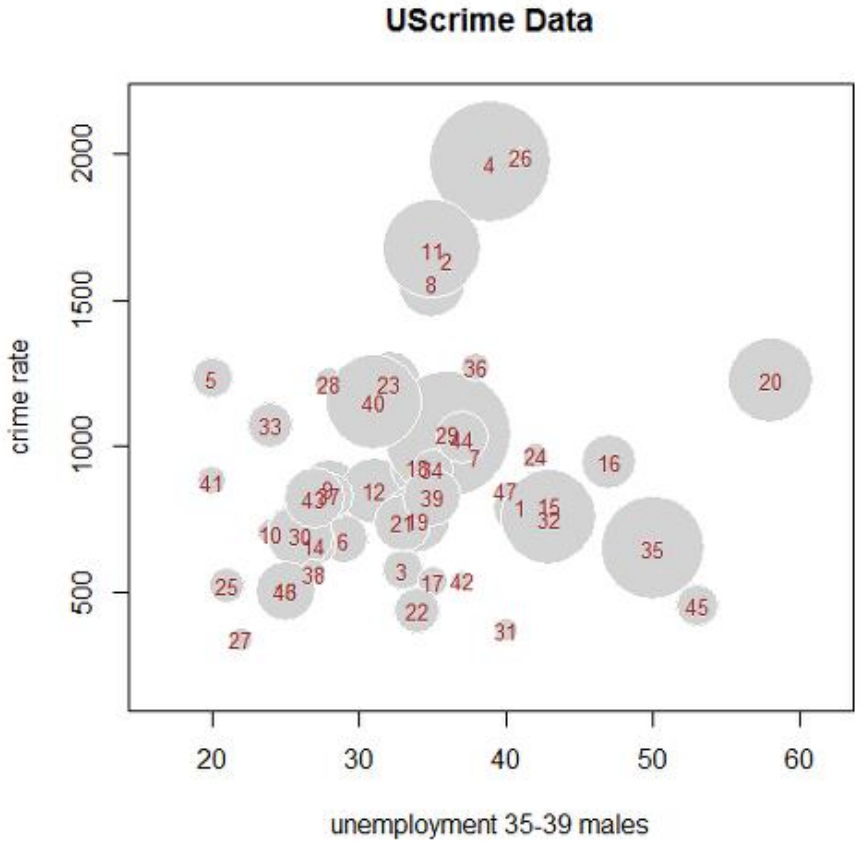
[연습 1]

- 2. “국회의원_선거구_유권자수.csv” 파일의 내용으로 부터 시도별 선거인수를 집계(합계계산)하여 다음과 같은 treemap 을 작성하시오



버블 차트 (bubble chart)

- 산점도는 두개의 변수간 상관 관계를 표시한다.
- 버블 차트는 산점도에 제3의 변수를 크기에 비례하는 버블(원)으로 표현한 그림이다.



실업률(남자 35-39세) x와 범죄율 y 간 관계를 보여주는 버블차트
(원의 넓이는 인구수)

버블 차트 (bubble chart)

- 설치가 필요한 패키지
  - MASS

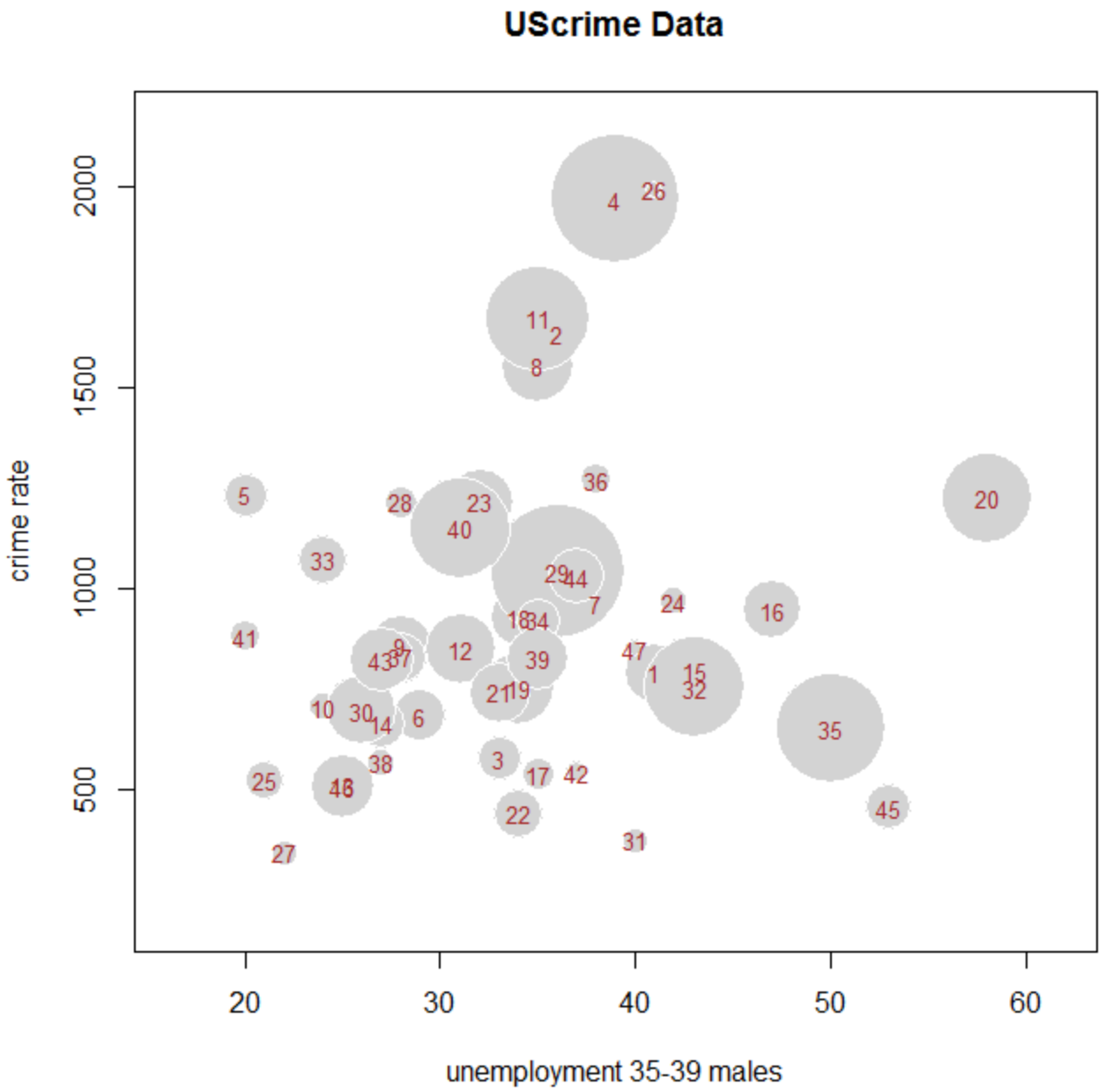
- 실습에 사용할 데이터셋
  - UScrime (MASS)

```
> head(UScrime)
    M So  Ed Po1 Po2  LF  M.F Pop  NW  U1 U2 GDP Ineq     Prob     Time    y
1 151  1  91  58  56 510  950  33 301 108 41 394  261 0.084602 26.2011  791
2 143  0 113 103  95 583 1012  13 102  96 36 557  194 0.029599 25.2999 1635
3 142  1  89  45  44 533  969  18 219  94 33 318  250 0.083401 24.3006  578
4 136  0 121 149 141 577  994 157  80 102 39 673  167 0.015801 29.9012 1969
5 141  0 121 109 101 591  985  18  30  91 20 578  174 0.041399 21.2998 1234
6 121  0 110 118 115 547  964  25  44  84 29 689  126 0.034201 20.9995  682
```

- Pop : 인구수
- U2 : 실업률(35~39세)
- y : 범죄율

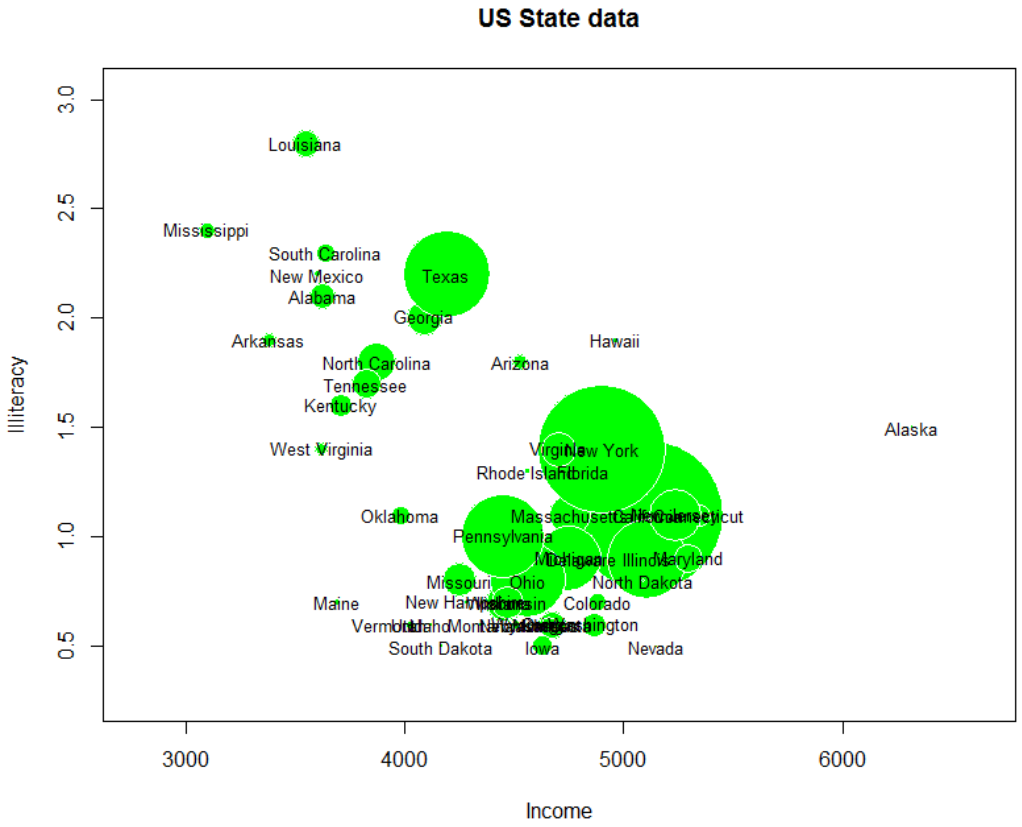
버블 차트 (bubble chart)

```
library(MASS)
head(UScrime)
radius <- sqrt(UScrime$Pop) # 원의 반지름(값이커서 줄임)
symbols(UScrime$U2, UScrime$y, # x,y 좌표값
    circles=radius,           # 원의 반지름값
    inches=0.4,               # 원의 크기 조절값
    fg="white",               # 원의 테두리 색
    bg="lightgray",           # 원의 바탕색
    lwd=1.5,                  # 원의 테두리선 두께
    xlab="unemployment 35-39 males",
    ylab="crime rate",
    main="UScrime Data")
text(UScrime$U2, UScrime$y,   # 텍스트가 출력될 x,y좌표
    1:nrow(UScrime),          # 출력할 텍스트
    cex=0.8,                  # 폰트 크기
    col="brown")              # 폰트 color
```


UScrime Data
crime rate
500
1000
1500
2000
20
30
40
50
60
unemployment 35-39 males

[연습 2]

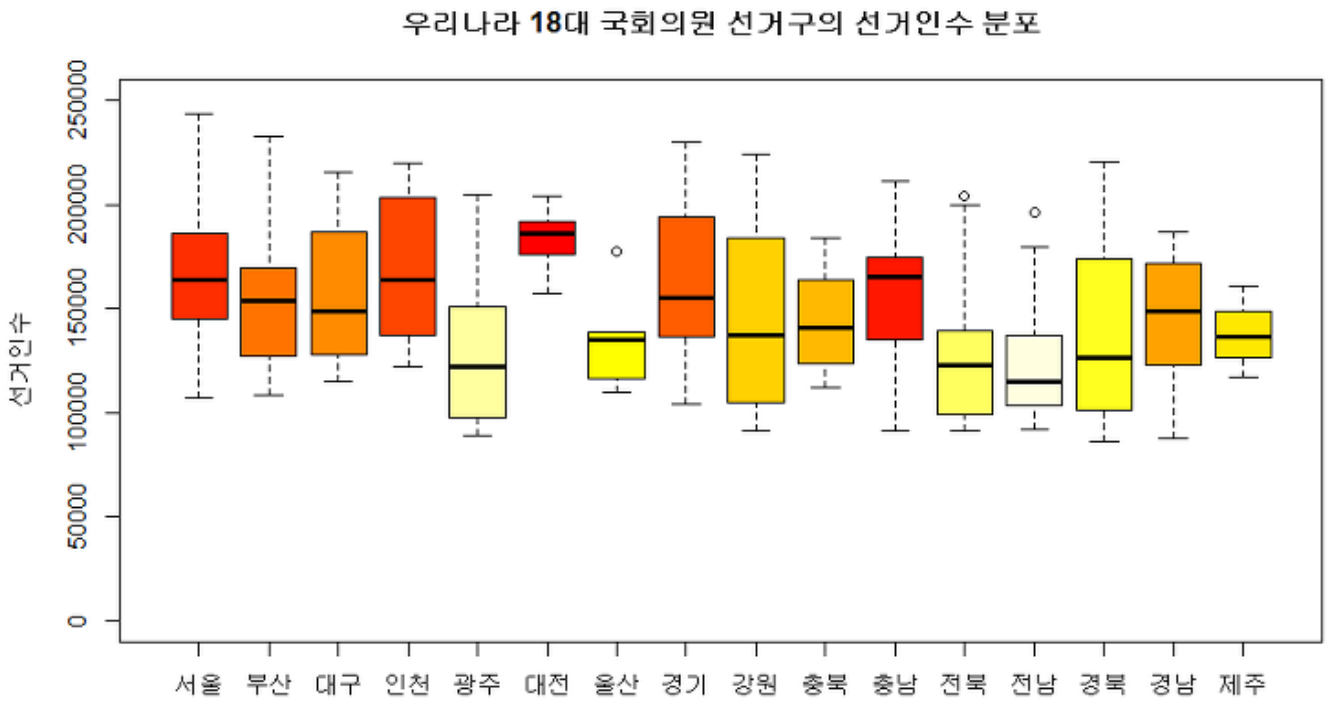
- 1. state.x77 데이터로 부터 다음과 같은 버블차트를 작성하시오
  - st <- data.frame(state.x77) 과 같이 matrix를 data frame 으로 변환하여 사용
  - 원의 크기는 인구(Population) 수를 의미



  - 이 그래프로부터 관찰할 수 있는 것은 무엇인가

다중 상자그림(Boxplot)

- 상자그림(box plot)은 일변량 연속형 자료를 상자와 선, 그리고 점으로 표현한 그림
- 다중 상자 그림은 총 자료가 여러 개의 자료 그룹(data batch)으로 구성되어 있는 경우 그룹 간 비교에 있어 시각적 효과가 탁월하다



다중 상자그림(Boxplot)

- 설치가 필요한 패키지
  - 없음

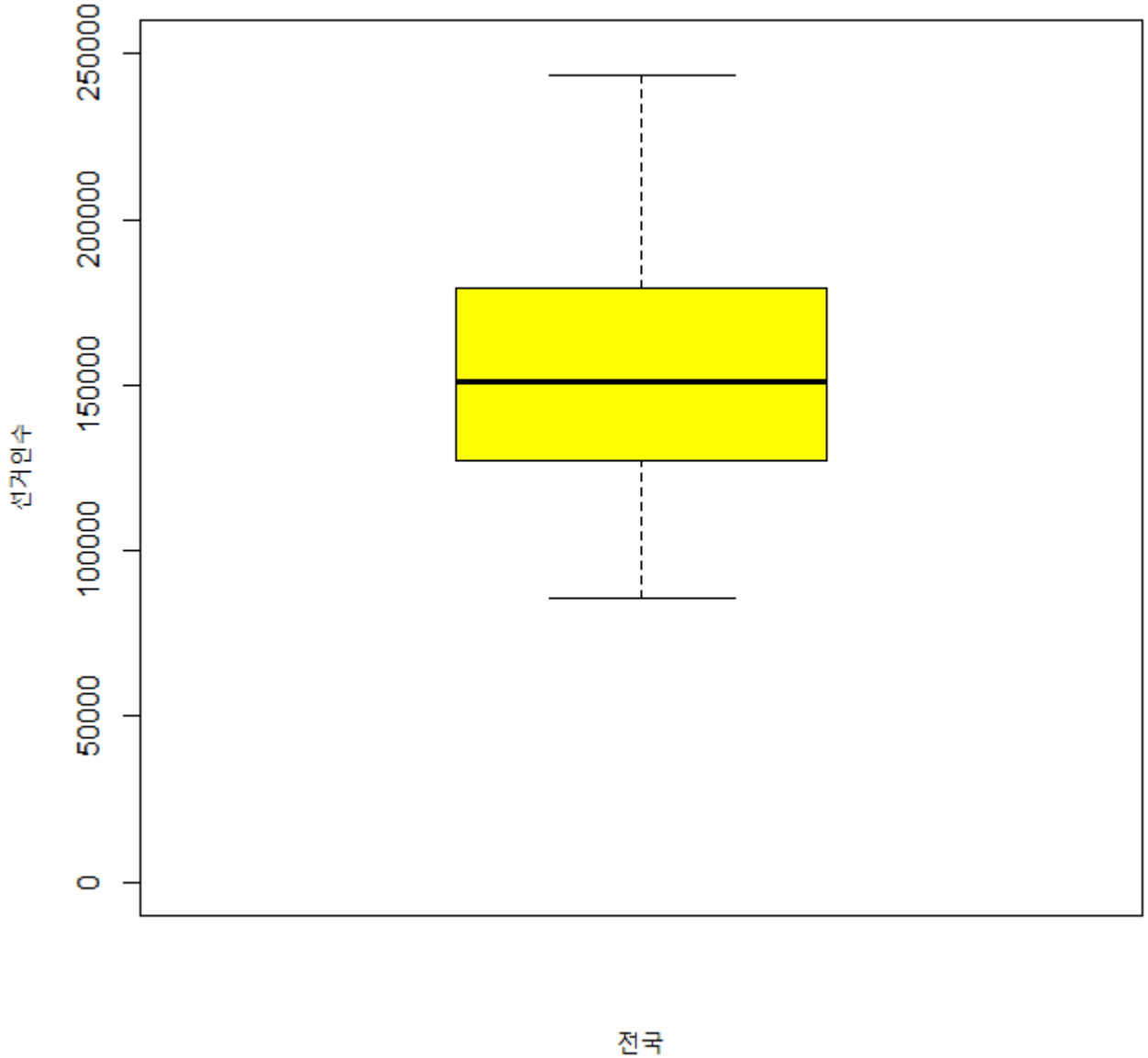
- 실습에 사용할 데이터셋
  - 국회의원_선거구_유권자수.csv

```
> head(ds)
  시도 시도번호 선거구명 선거구번호 선거인.수
1 서울        1   종로구          1    135727
2 서울        1     중구          2    106880
3 서울        1   용산구          3    192033
4 서울        1 성동구갑          4    143798
5 서울        1 성동구을          5    121869
6 서울        1 광진구갑          6    145896
```

다중 상자그림(Boxplot)

```
setwd("c:/works") # 읽어올 데이터 파일이 있는 폴더지정
ds <- read.csv("국회의원_선거구_유권자수.csv", header=T)
head(ds)
summary(ds$선거인.수)

# 선거구별 선거인수 (전국)
boxplot(ds$선거인.수,
    col="yellow",
    ylim=c(0,250000),
    xlab="전국",
    ylab="선거인수")
```

위의 예처럼 컬럼이름이나 변수이름에 한글을 쓸수 있지만
권장하지 않음

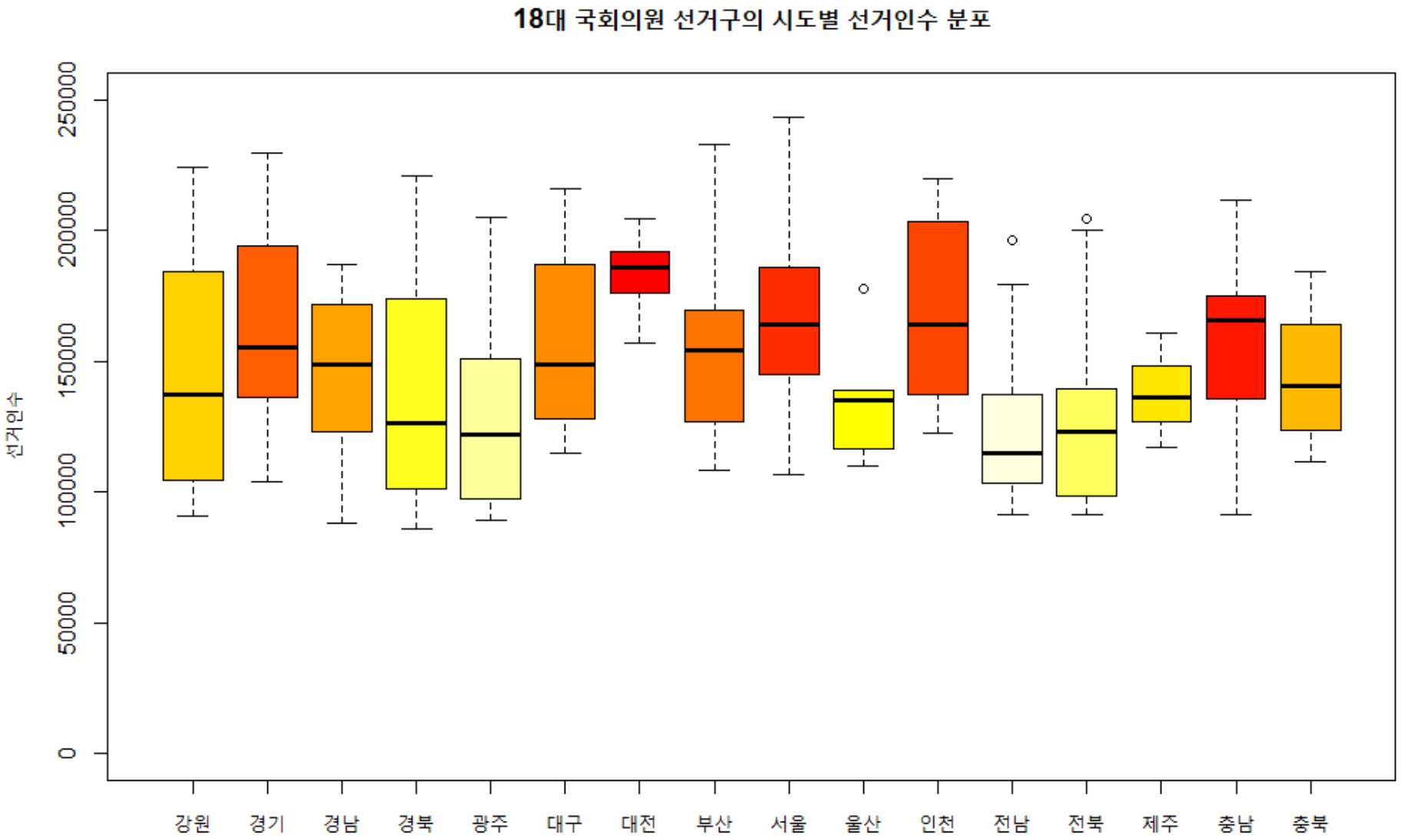

선거인수
0
50000
100000
150000
200000
250000
전국

다중 상자그림(Boxplot)

```
# 시도번호로 그룹핑한후 그룹열로 선거인.수의 중간값을 계산
cnt <- aggregate(ds[,5], by=list(ds$시도),median)

# 선거인.수의 중간값이 큰순서에서 작은 순서로 순위 계산
odr <- rank(-cnt$x)

# 시도를 기준으로 그룹핑하여 다중 상자그림을 그린다
boxplot(선거인.수~시도, data=ds,
      col=heat.colors(16)[odr], # 상자의 색을 지정
      ylim=c(0,250000),
      ylab="선거인수",
      main="18대 국회의원 선거구의 시도별 선거인수 분포")
```

col=heat.colors(16)[odr]

: 각 box 의 색을 heat.colors 에서 16개의 색을 취하여 그린다.
어느값을 취할지는 odr 에 따른다.

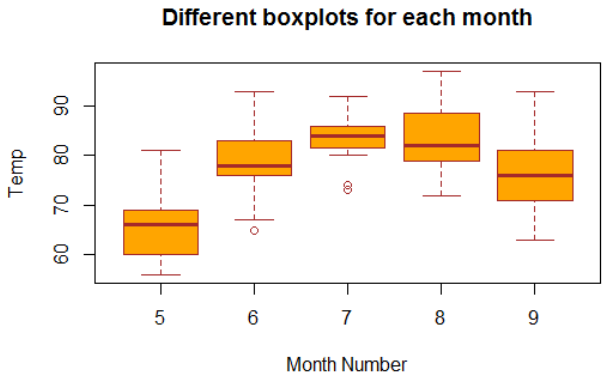


평균선거인수가 많을수록 붉은색, 적을수록 연한 노랑색

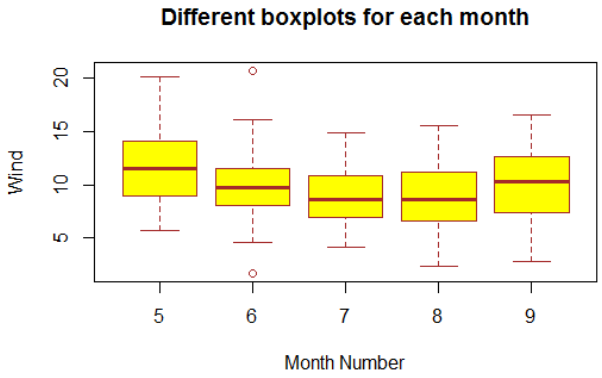
[연습 3]

- R 에서 제공하는 airquality 데이터셋을 이용하여 다음 문제를 해결하시오

1. 월별(Month) 기온(Temp)을 boxplot 으로 작성하시오

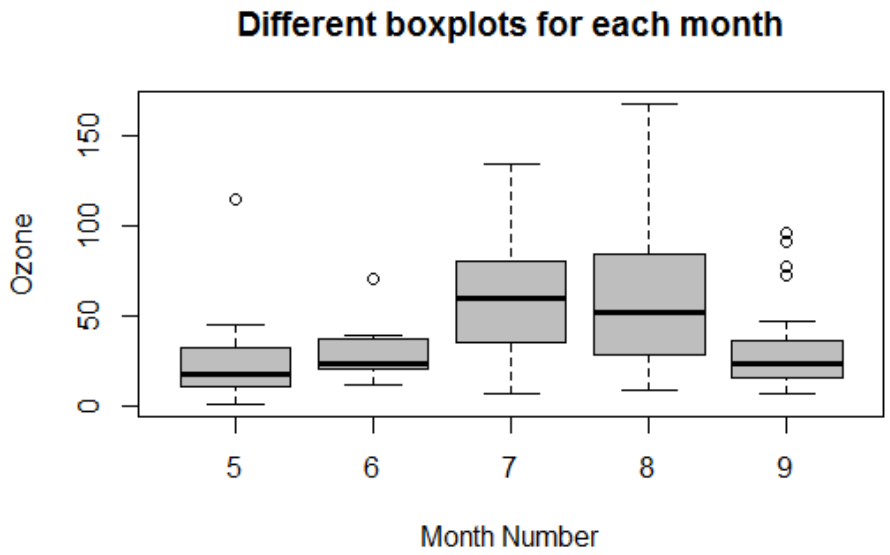


2. 월별(Month) 풍속(Wind)을 boxplot 으로 작성하시오



[연습 3]

3. 월별(Month) 오존농도(Ozone)을 boxplot 으로 작성하시오



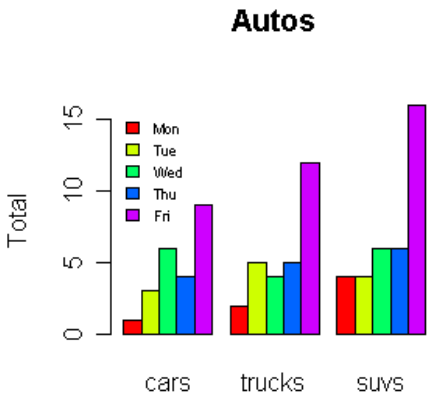
4. 각각의 boxplot 으로 부터 관찰할 수 있는 정보는 무엇인가

기타 그래프

- Self-study
  - http://www.harding.edu/fmccown/r

Now let's graph the total number of autos per day using some color and show a legend:

```
# Read values from tab-delimited autos.dat
autos_data <- read.table("C:/R/autos.dat", header=T, sep="\t")

# Graph autos with adjacent bars using rainbow colors
barplot(as.matrix(autos_data), main="Autos", ylab= "Total",
   beside=TRUE, col=rainbow(5))

# Place the legend at the top-left corner with no frame
# using rainbow colors
legend("topleft", c("Mon","Tue","Wed","Thu","Fri"), cex=0.6,
   bty="n", fill=rainbow(5));
```



Let's graph the total number of autos per day using a stacked bar chart and place the legend outside of the plot area:

```
# Read values from tab-delimited autos.dat
autos_data <- read.table("C:/R/autos.dat", header=T, sep="\t")

# Expand right side of clipping rect to make room for the legend
par(xpd=T, mar=par()$mar+c(0,0,0,4))

# Graph autos (transposing the matrix) using heat colors,
# put 10% of the space between each bar, and make labels
# smaller with horizontal y-axis labels
barplot(t(autos_data), main="Autos", ylab="Total",
   col=heat.colors(3), space=0.1, cex.axis=0.8, las=1,
   names.arg=c("Mon","Tue","Wed","Thu","Fri"), cex=0.8)

# Place the legend at (6,30) using heat colors
legend(6, 30, names(autos_data), cex=0.8, fill=heat.colors(3));

# Restore default clipping rect
par(mar=c(5, 4, 4, 2) + 0.1)
```