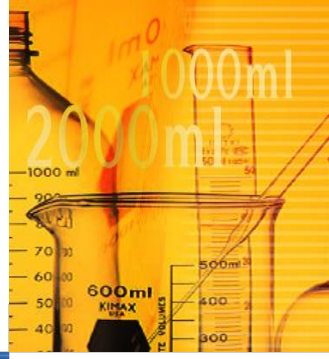
R 프로그래밍



Chapter 7

데이터 시각화 (1)

Sejong Oh

Bio Information technology Lab.

Content

- 패키지 설치
- 이변량 밀도
- 이변량 히스토그램
- 사각 타일
- 모자이크 플롯

분석에 필요한 패키지 설치및 사용하기

- 자료 분석에 필요한 함수들은 다양한 패키지들이 제공한다.
- 따라서 내 컴퓨터에 해당 패키지가 설치되어 있지 않으면 패키지를 먼저 설치해야 한다
- 한번 설치된 패키지는 지속적으로 사용가능 하며, library() 함수를 통해서 설치된 패키지를 호출한 후 포함된 함수를 이용하면 된다

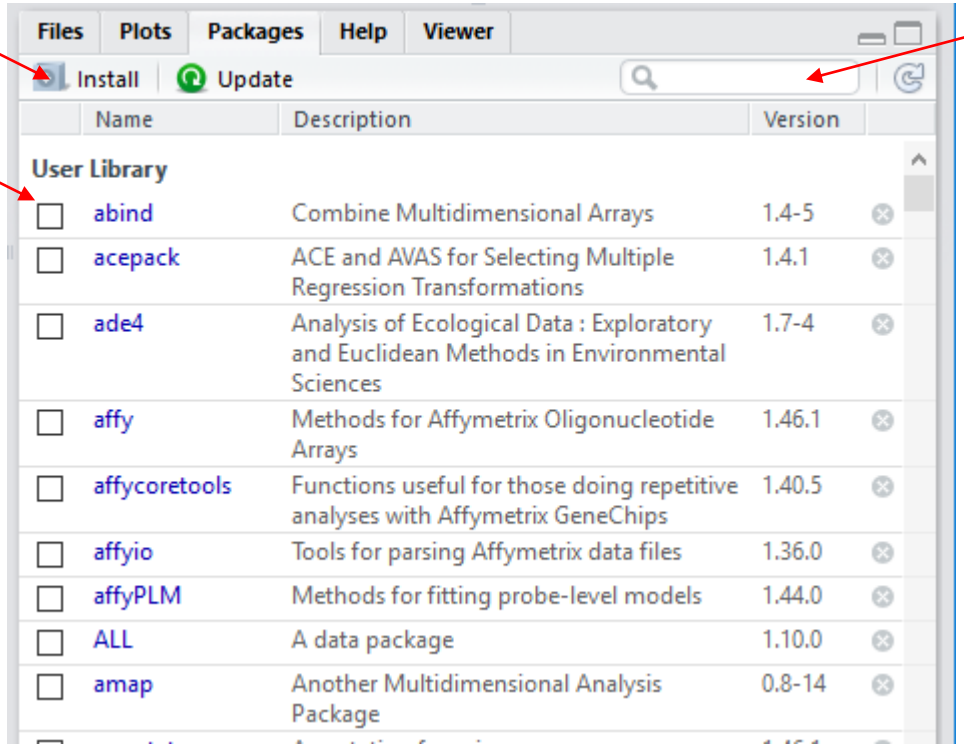
분석에 필요한 패키지 설치및 사용하기

- Rstudio 에서 패키지 설치하기 (1)

새 패키지 설치

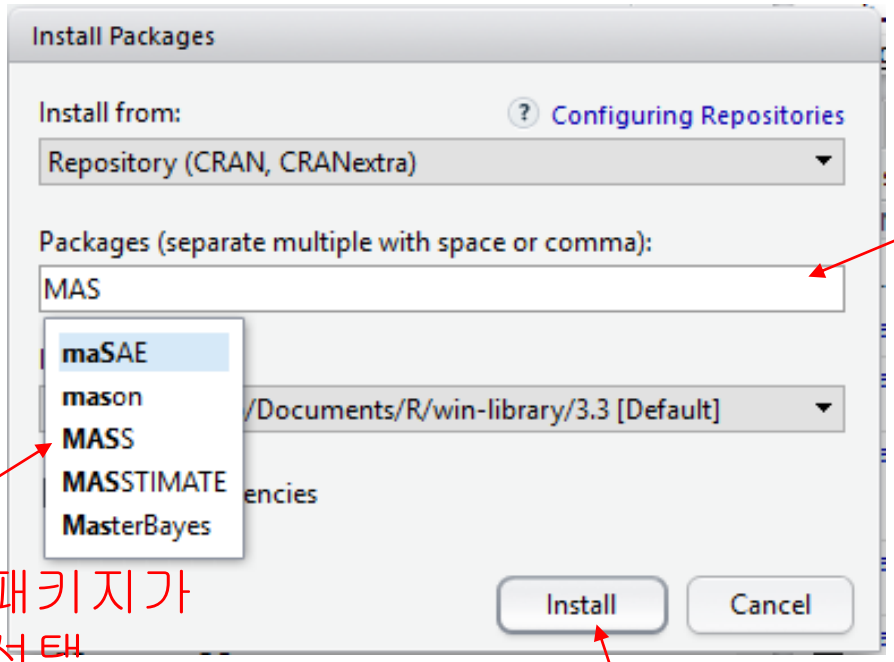
내컴퓨터에 설치된 패키지 목록

내컴퓨터에 설치된 패키지 검색



분석에 필요한 패키지 설치및 사용하기

- Rstudio 에서 패키지 설치하기 (2)



1. 여기에 패키지 이름 입력

2. 원하는 패키지가 표시되면 선택

3. 클릭

분석에 필요한 패키지 설치및 사용하기

- Rstudio 에서 패키지 설치하기 (3)

  - 완료시 다음과 같은 메시지 출력

```
package 'MASS' successfully unpacked and MD5 sums checked
Warning in install.packages :
  cannot remove prior installation of package 'MASS'

The downloaded binary packages are in
        C:\Users\mango\AppData\Local\Temp\RtmpWQlUl4\downloaded_packages
> |
```

  - [install] 버튼 클릭 후 오래된 package 들을 update 할지를 물어보는 경우가 있는데 [Yes] 를 클릭하면 된다

이변량 밀도(Bivariate Density)

- 등고선을 활용하여 산점도에서 잘 드러나지 않는 부분을 보여준다
- 실습에 필요한 패키지
  - MASS
  - KernSmooth

- 대상 데이터셋 :
  - MASS 패키지의 geyser
  - 옐로우스톤 국립공원의 간헐천에서 관측된 대기시간(waiting)과 지속시간(duration)으로 구성

이변량 밀도(Bivariate Density)

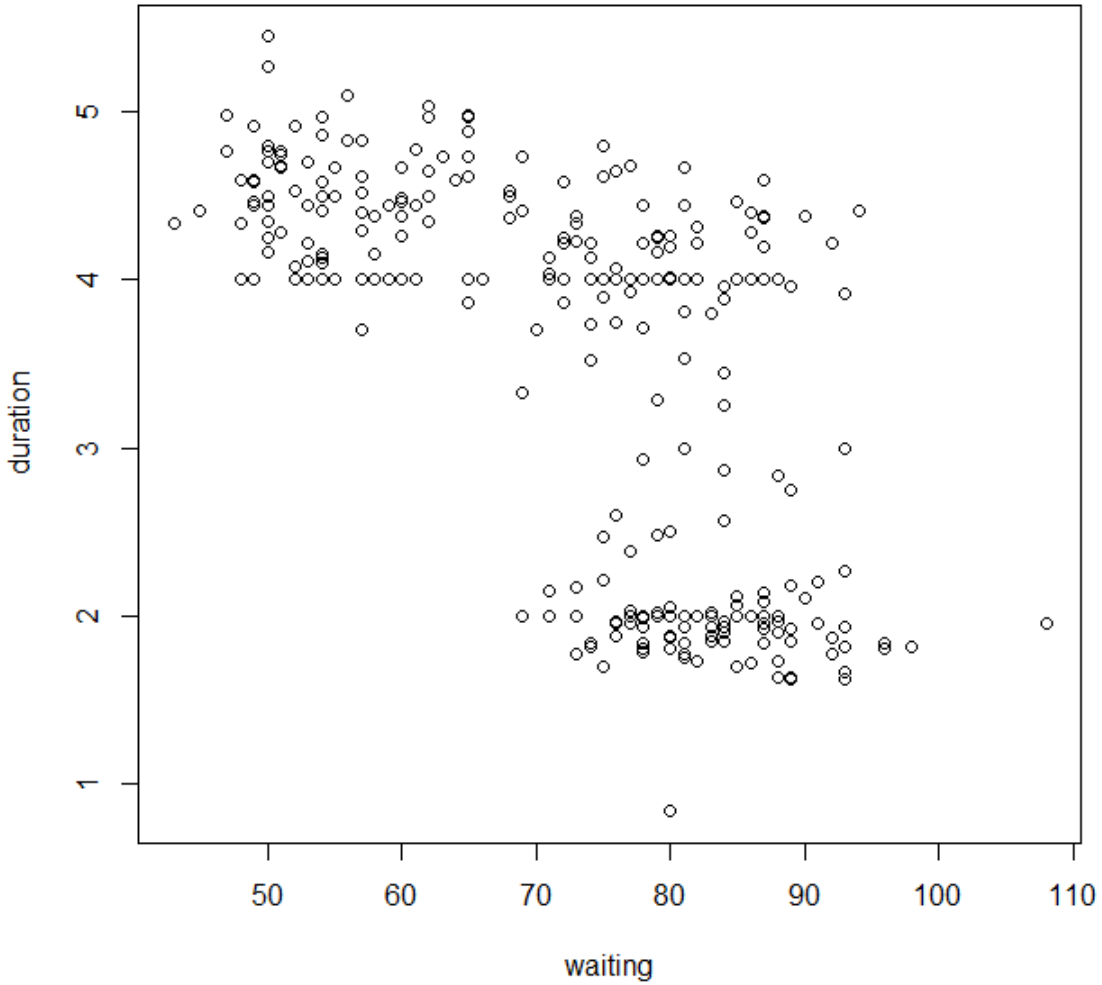
```
rm(list=ls())    # 앞의 작업 결과 clear
library(MASS)
head(geyser)
```

```
> library(MASS)
> head(geyser)
  waiting duration
1      80 4.016667
2      71 2.150000
3      57 4.000000
4      80 4.000000
5      75 4.000000
6      77 2.000000
```

참조 : http://blog.daum.net/huh420/19

이변량 밀도(Bivariate Density)

```
plot(geyser)                                    # 산점도
```



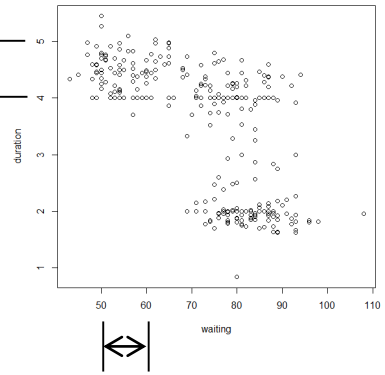
이변량 밀도(Bivariate Density)

```
# 산점도 그리기
plot(geyser,
     xlim=c(30,120),        # x축에 표시될 값의 범위
     ylim=c(0,6.5),         # y축에 표시될 값의 범위
     col="forestgreen",
     pch=20,
     main="geyser")

# 산점도 위에 등고선 그리기
library(KernSmooth)
density <- bkde2D(geyser, bandwidth=c(5,0.5))
par(new=T)
contour(density$x1, density$x2, density$fhat,
        xlim=c(30,120), ylim=c(0,6.5),
        col=heat.colors(7)[7:1],
        nlevels=7, lwd=2)
```

이변량 밀도(Bivariate Density)

- **bkde2D()** : 데이터의 분포 밀도를 추정하는 함수
  - **bandwidth=c(5,0.5)** : 등고선의 모양 조절 c(x축방향,y축방향). 데이터에 따라 특성이 잘 드러나도록 값을 조절하여 사용한다

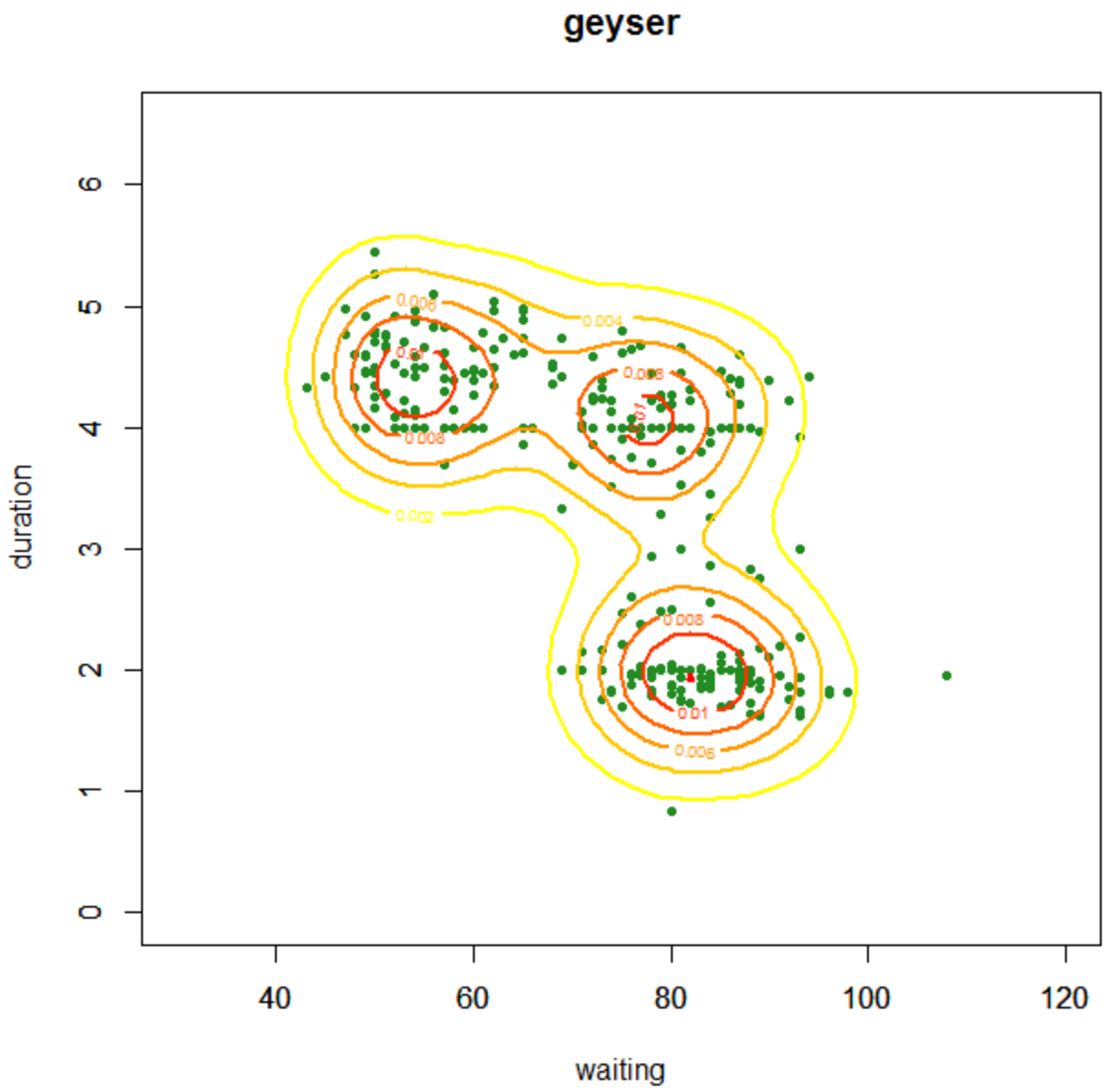
Y축방향은 이 값보다 작게



X축방향은 이 값보다 작게

- **par(new=T)** : 뒤의 그래프(contour)를 앞의 그래프(plot) 위에 겹쳐 그리도록 한다

- **Contour()** : 등고선을 그리는 함수
  - **col=heat.colors(7)[7:1]** : 등고선의 색을 데이터 값에 따라 7단계로 표시
  - **nlevels=7** : 등고선을 7 개 레벨로 표시


geyser
duration
waiting
0
1
2
3
4
5
6
40
60
80
100
120
0.002
0.004
0.006
0.008
0.01

이변량 밀도(Bivariate Density)

- <해석>
  - 대기 시간이 40~60 인 경우는 간헐천 분출 지속 시간이 4~5 이다
  - 대기 시간이 70~90 인 경우는 지속 시간이 두 그룹으로 나뉘는데, 하나는 1.5~2.5, 다른 하나는 3.5~4.5 이다

[연습 1]

1. state.x77 데이터셋에서 Income, Murder 에 대해 이변량 밀도 그래프를 작성해 보시오 (**`bandwidth`** 값을 조절). 그래프에서 관찰할 수 있는 정보는 무엇인가

2. iris 데이터셋에서 Petal.Length Petal.Width 에 대해 이변량 밀도 그래프를 작성해 보시오 (**`bandwidth`** 값을 조절). 그래프에서 관찰할 수 있는 정보는 무엇인가

이변량 히스토그램(Bivariate Histogram)

- 이변량 밀도와 유사하나 3차원 구조로 분포를 보여준다

- 실습에 필요한 패키지
  - MASS
  - lattice
  - latticeExtra

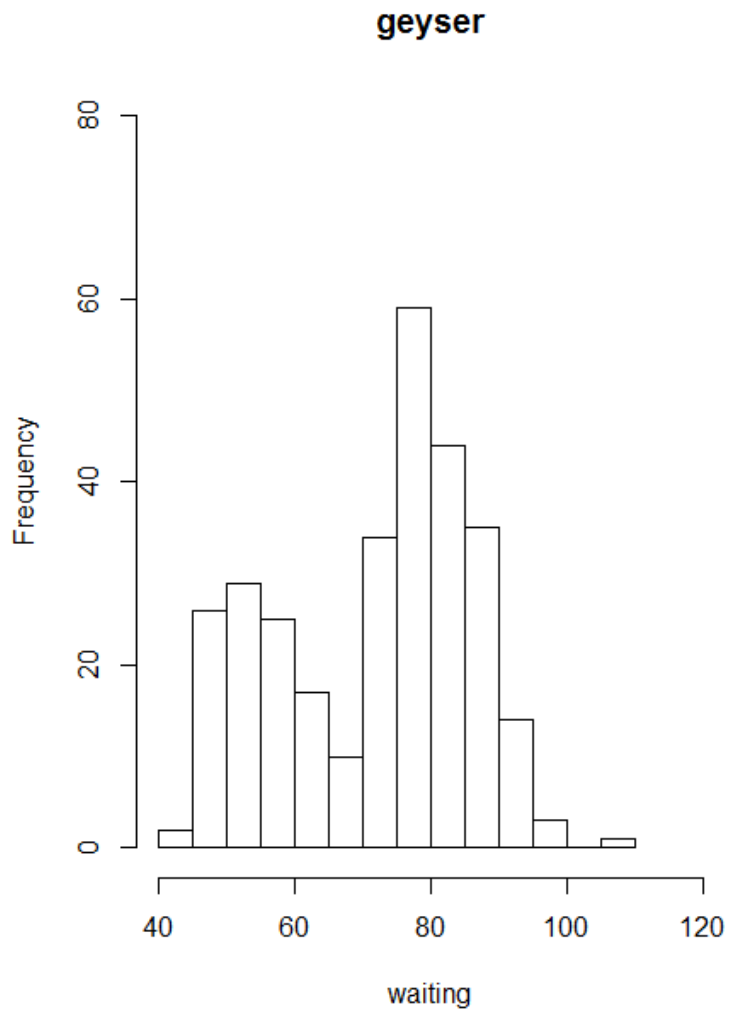
- 대상 데이터셋 :
  - MASS 패키지의 geyser
  - 옐로우스톤 국립공원의 간헐천에서 관측된 대기시간(waiting)과 지속시간(duration)으로 구성

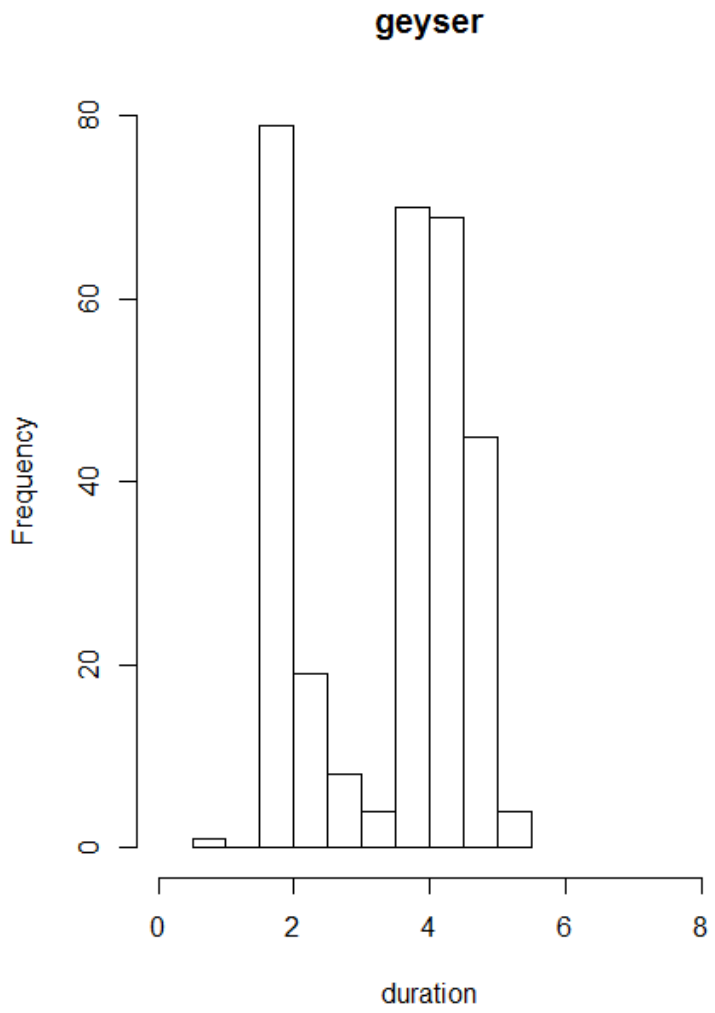
참조: http://blog.daum.net/huh420/19

이변량 히스토그램(Bivariate Histogram)

```
rm(list=ls())    # 앞의 작업 결과 clear

# 필요한 패키지 불러오기
library(lattice)
library(latticeExtra)
library(MASS)

# 일변량 히스토그램
par(mfrow=c(1,2))
hist(geyser$waiting, nclass=20, main="geyser",
xlab="waiting", xlim=c(40,120), ylim=c(0,80))
hist(geyser$duration, nclass=16, main="geyser",
xlab="duration", xlim=c(0,8), ylim=c(0,80))
```

이변량 히스토그램(Bivariate Histogram)





이변량 히스토그램(Bivariate Histogram)

```
rm(list=ls())    # 앞의 작업 결과 clear
# 이변량 히스토그램 데이터 준비
x.waiting <- cut(geyser$waiting,10)
levels(x.waiting) <- paste("w",1:10)
y.duration <- cut(geyser$duration,8)
levels(y.duration) <- paste("d",1:8)
geyser.freq <- table(x.waiting, y.duration)
```

cut(geyser$waiting,10)

: geyser$waiting 의 데이터를 10개의 구간으로 나눈다

levels(x.waiting) <- paste("w",1:10)

: 10개의 구간에 이름을 w1,w2,…,w10 과 같이 붙인다

table(x.waiting, y.duration)

: x.waiting, y.duration 의 구간 데이터를 가지고 교차테이블을 만든다

이변량 히스토그램(Bivariate Histogram)

```
> x.waiting <- cut(geyser$waiting,10)
> levels(x.waiting) <- paste("w",1:10)
> y.duration <- cut(geyser$duration,8)
> levels(y.duration) <- paste("d",1:8)
> geyser.freq <- table(x.waiting, y.duration)
>  geyser.freq
           y.duration
x.waiting d 1 d 2 d 3 d 4 d 5 d 6 d 7 d 8
      w 1     0    0    0    0    0    4   10    2
      w 2     0    0    0    0    0   17   21    5
      w 3     0    0    0    0    1   12   16    2
      w 4     0    0    1    0    1    4    9    3
      w 5     0    4    6    0    2   18    5    0
      w 6     1   18   19    3    3   29    6    0
      w 7     0   18   12    3    2   10    5    0
      w 8     0   12    4    2    0    3    2    0
      w 9     0    3    0    0    0    0    0    0
      w 10    0    1    0    0    0    0    0    0
```

여기에 있는 값을 가지고 탑을 쌓는다.
값의 크기가 탑의 높이로 표현됨

이변량 히스토그램(Bivariate Histogram)

```
# 이변량 히스토그램 그리기
cloud(geyser.freq,
  panel.3d.cloud = panel.3dbars,
  main="geyser",
  xlab="waiting", ylab="duration",zlab="freq",
  zlim = c(0, max(geyser.freq)*1.5),
  scales = list(arrows = FALSE, just = "right"),
  col.facet = level.colors(geyser.freq,
     at = do.breaks(range(geyser.freq), 24),
     col.regions = terrain.colors, colors = TRUE),
  screen = list(z = 30, x = -30))
```

cloud()
: 이변량 히스토그램을 작성하는 함수

이변량 히스토그램(Bivariate Histogram)

zlim = c(0, max(geyser.freq)*1.5)
: z축 천정의 높이를 geyser.freq 최대값의 1.5 배 정도로 설정. 이값을 지정안하면 제일 긴 막대가 천정에 닿는다

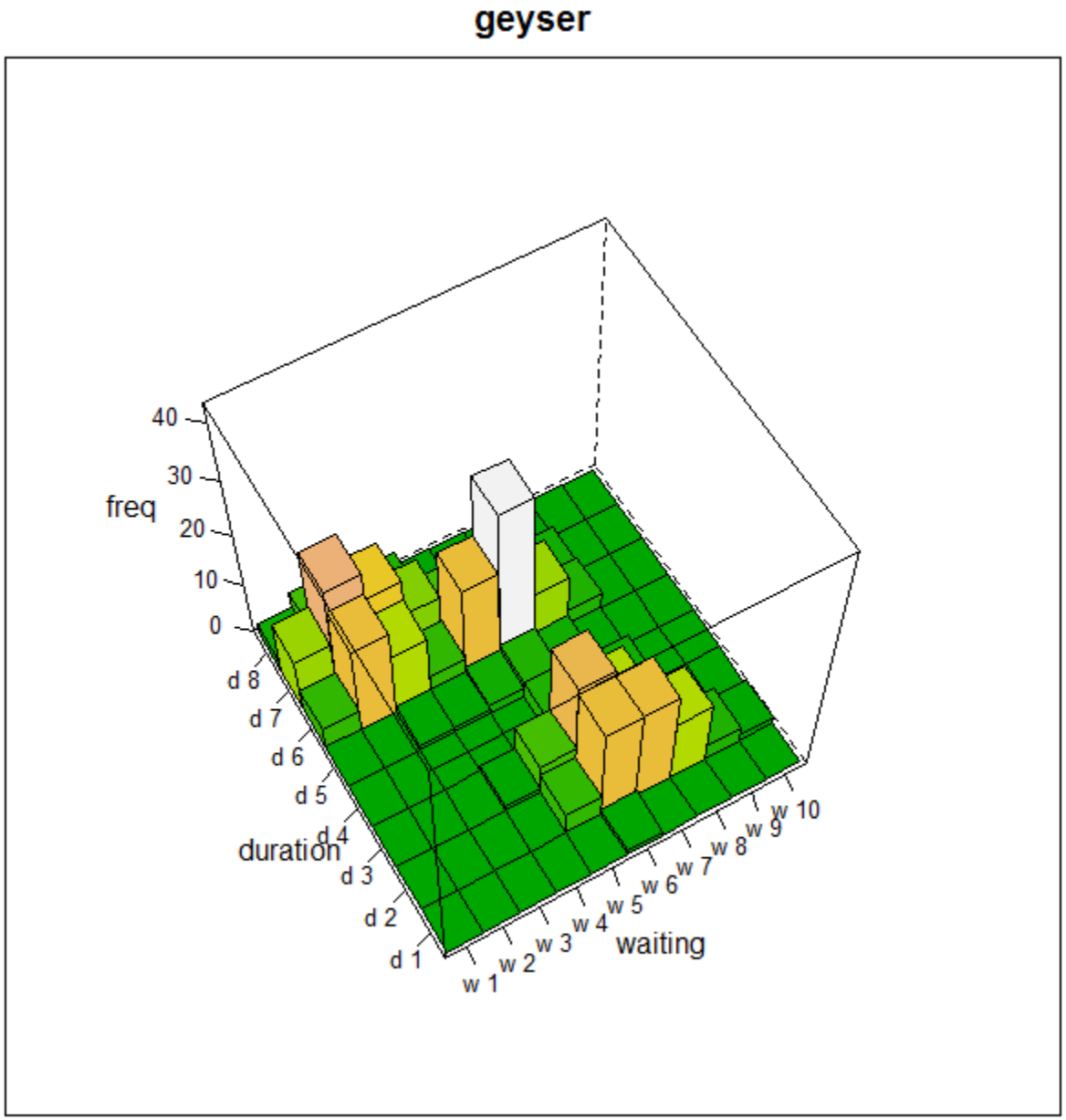
scales = list(arrows = FALSE, just = "right")
: z축에 화살표가 표시되는데 이것을 안보이게 함

```
col.facet = level.colors(geyser.freq,
     at = do.breaks(range(geyser.freq), 24),
     col.regions = terrain.colors, colors = TRUE)
```

: 막대의 컬러를 지정함. 막대의 높이에 적용된 컬러는 terrain.colors이다 (낮은 지역은 초록색으로 높은 지역은 흰색으로 칠해진다)

screen = list(z = 30, x = -30)
: z는 사각형 틀의 좌우 회전각을 지정하고 x는 사각형 틀을 보는 카메라의 높이를 지정 한다 (가장 높은 곳이 0이고 가장 낮은 곳이 –90이다). z,x 의 값을 바꾸어서 히스토그램을 그려보자.

이변량 히스토그램(Bivariate Histogram)



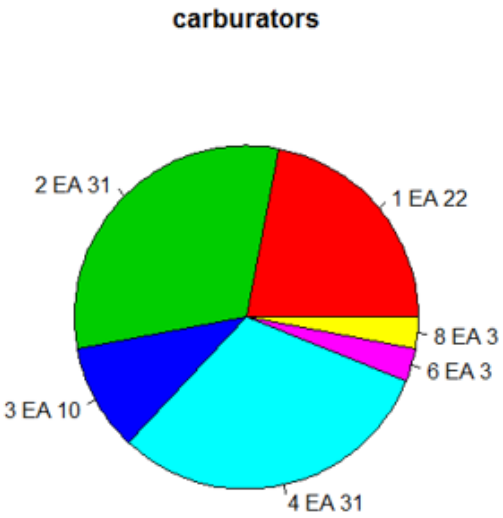
[연습 2]

1. state.x77 데이터셋에서 Income, Murder 에 대해 이변량 히스토그램을 작성해 보시오

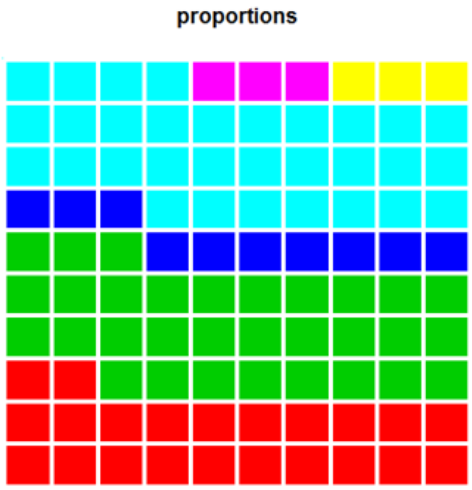
2. iris 데이터셋에서 Petal.Length Petal.Width 에 대해 이변량 밀도 히스토그램을 작성해 보시오

사각타일(square tile)

- 파이 차트(pie chart)는 특정 속성의 구성 비율을 시각화하기 위해 쓰이는 표준적 도구
- 파이 차트는 각(angle)의 크기로 비율을 나타내는데, 각(角)에 대한 인간의 인지력은 좋지 않기 때문에 대안으로서 사각타일에 제안됨
- 사각 타일은 사각형 틀에 배열한 100개의 타일을 구성 비율 대로 색깔을 달리하여 전체적인 구성비를 보다 쉽게 파악할 수 있도록 한다



<파이차트>



<사각타일>

사각타일(square tile)

- 설치가 필요한 패키지
  - 없음

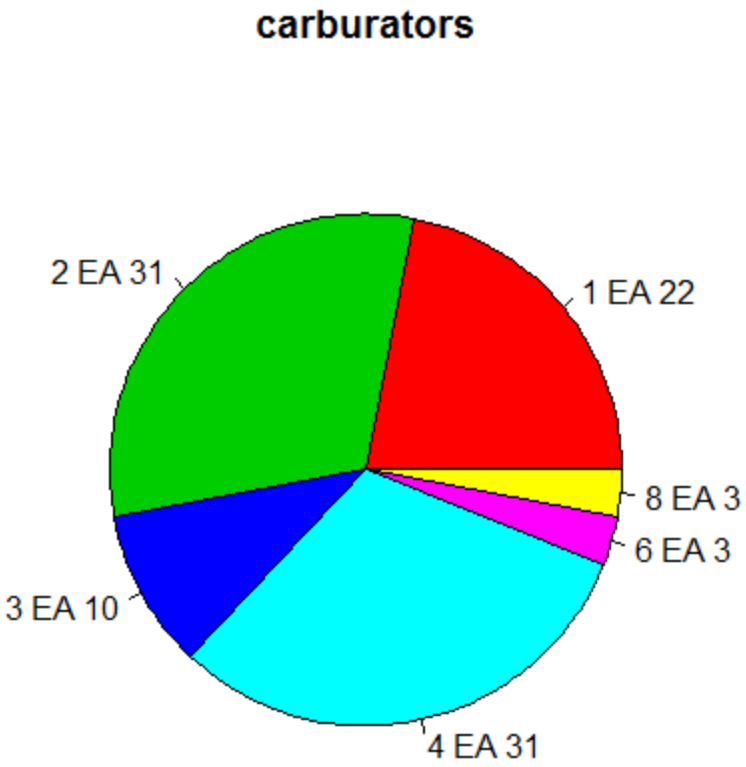
- 예제 데이터셋
  - mtcars 의 carb 변수

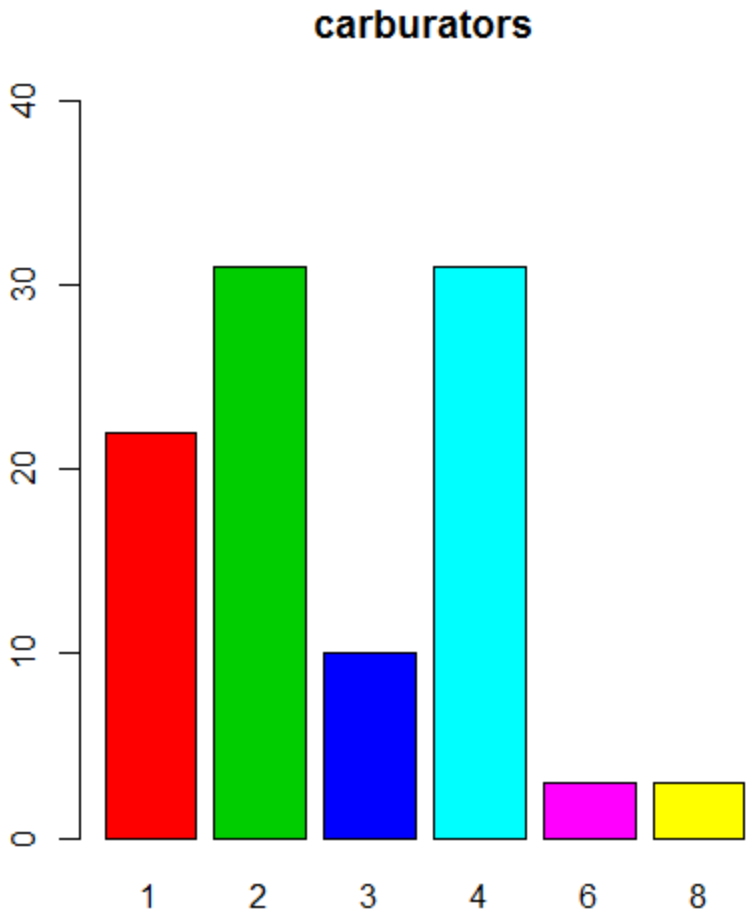
```
> mtcars$carb
 [1] 4 4 1 1 2 1 4 2 2 4 4 3 3 3 4 4 4 1 2 1 1 2 2 4 2 1 2 2 4 6 8 2
> table(mtcars$carb)

 1  2  3  4  6  8
 7 10  3 10  1  1
```

사각타일(square tile)

```
rm(list=ls())    # 앞의 작업 결과 clear
# 데이터 준비
carb <- mtcars$carb      # 변수의 데이터 추출
tbl <- table(carb)       # 도수 분포 계산
prop <- round((tbl/sum(tbl))*100, digits=0) #비율계산
sum(prop)                # 합이 100 되는지 확인하고
prop[3] <- prop[3]+1     # 아니면 100 되도록 맞춘다
m <- length(prop)

# pie chart
pie(prop, col=2:(m+1), main="carburators",
    labels=paste(names(tbl), "EA", prop))

# bar chart
barplot(prop, col=2:(m+1),main="carburators",
        ylim=c(0,40))
```

사각타일(square tile)





사각타일(square tile)

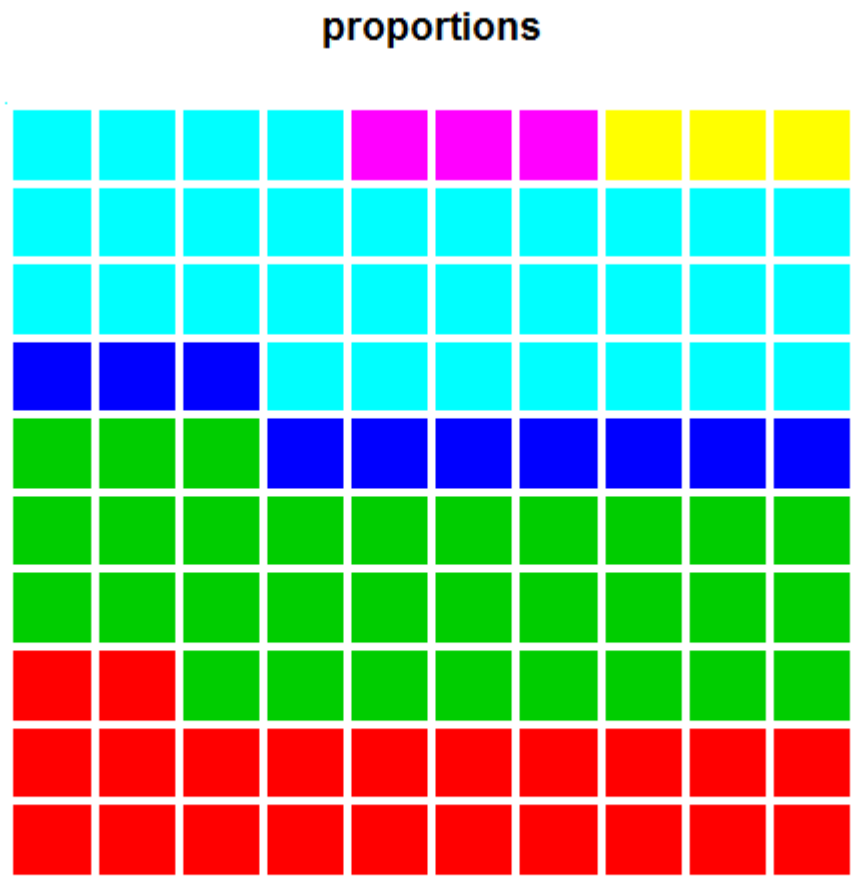
```
# square tiles
p.vec <- rep(1:m, prop)
P <- matrix(p.vec, 10, 10)
color <- 2:(m+1)
image(P, col=color, axes=F, main="proportions")

# 세로,가로 격자를 그린다
abline(h=seq(-0.05,1.05,1.1/10),col="white",lwd=4)
abline(v=seq(-0.05,1.05,1.1/10),col="white",lwd=4)
```

image() : 사각 타일을 그리는 함수

- **axes=F**: x,y축 및 눈금을 표시하지 않는다

```
> prop
carb
 1  2  3  4  6  8
22 31 10 31  3  3
> p.vec
  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [39] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4
 [77] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 5 5 5 6 6 6
```

사각타일(square tile)



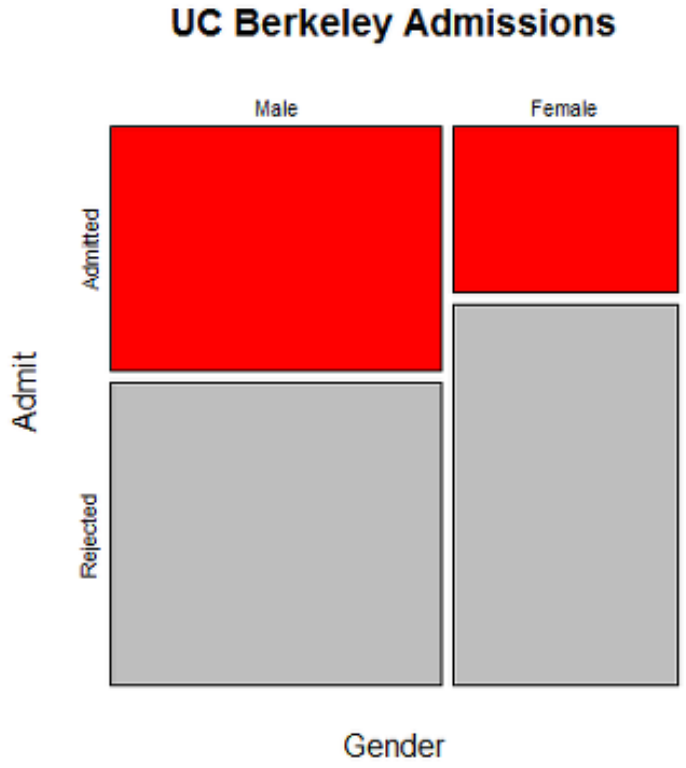
[연습 3]

1. infert 데이터셋의 parity 변수에 대해 파이그래프, 막대그래프, 사각 타일 그래프를 작성하시오

2. infert 데이터셋의 education 변수에 대해 파이그래프, 막대그래프, 사각 타일 그래프를 작성하시오

모자이크 플롯(mosaic plot)

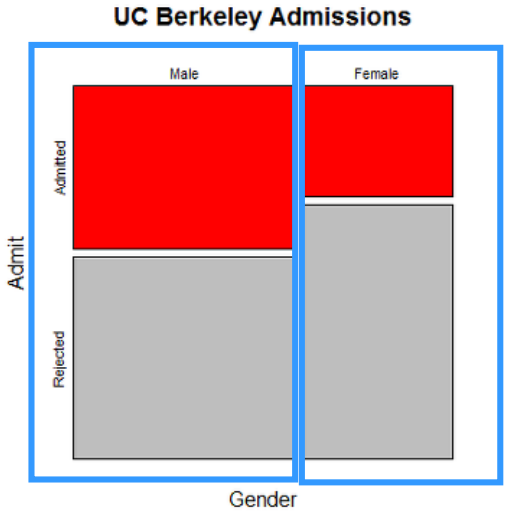
- 모자이크 플롯(mosaic plot)은 2원 3원 교차표의 시각화이다. 전체 정사각 도형을 교차표의 행 빈도에 비례하는 직사각 도형으로 나누고 다시 각 도형을 행 내 열의 빈도에 해당하는 직사각 도형으로 나눈다.



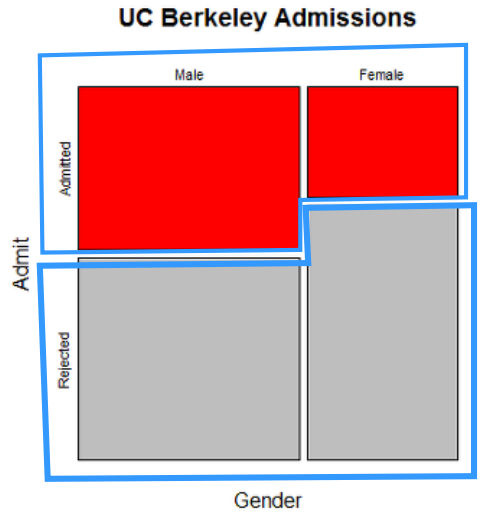
UC Berkeley 대학원 입시 통계

참조 : http://blog.daum.net/huh420/20

모자이크 플롯 (mosaic plot)

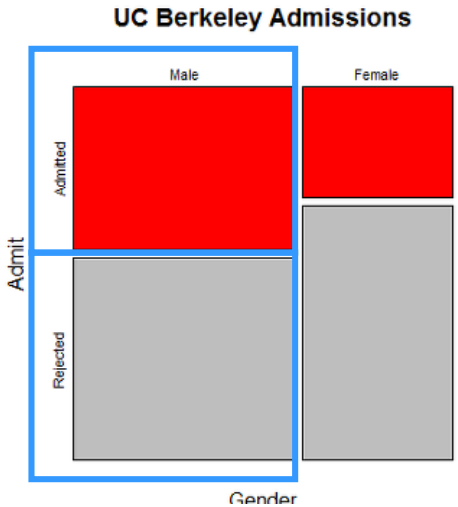


버클리 지원자 중 남성, 여성의 비율
(면적의 크기가 비율을 나타낸다)

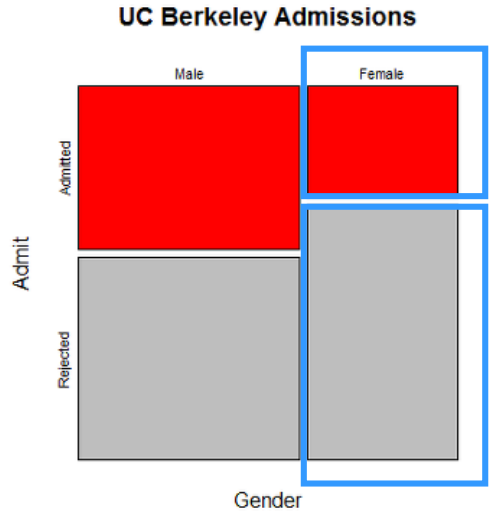


버클리 지원자 중 합격자와, 불합격자의 비율
(면적의 크기가 비율을 나타낸다)

모자이크 플롯 (mosaic plot)

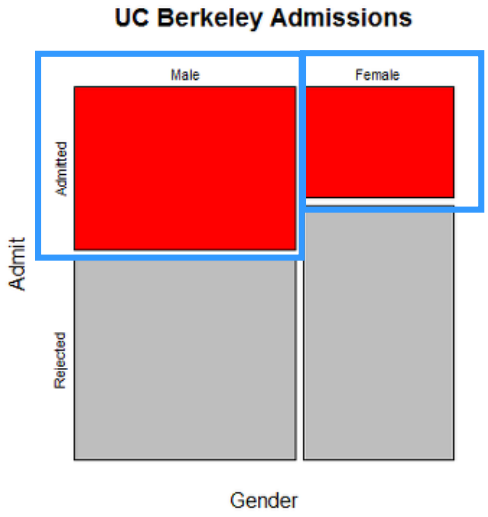


버클리 남성 지원자 중 합격자, 불합격자의 비율
(면적의 크기가 비율을 나타낸다)



버클리 여성 지원자 중 합격자, 불합격자의 비율
(면적의 크기가 비율을 나타낸다)

모자이크 플롯 (mosaic plot)



버클리 남성 합격자와, 여성 합격자의 비율
(면적의 크기가 비율을 나타낸다)

(전체적으로는 남성이 합격자 수, 합격률에 있어서 여성보다 앞서는 것을 알 수 있다. -> 남녀차별 문제 제기)

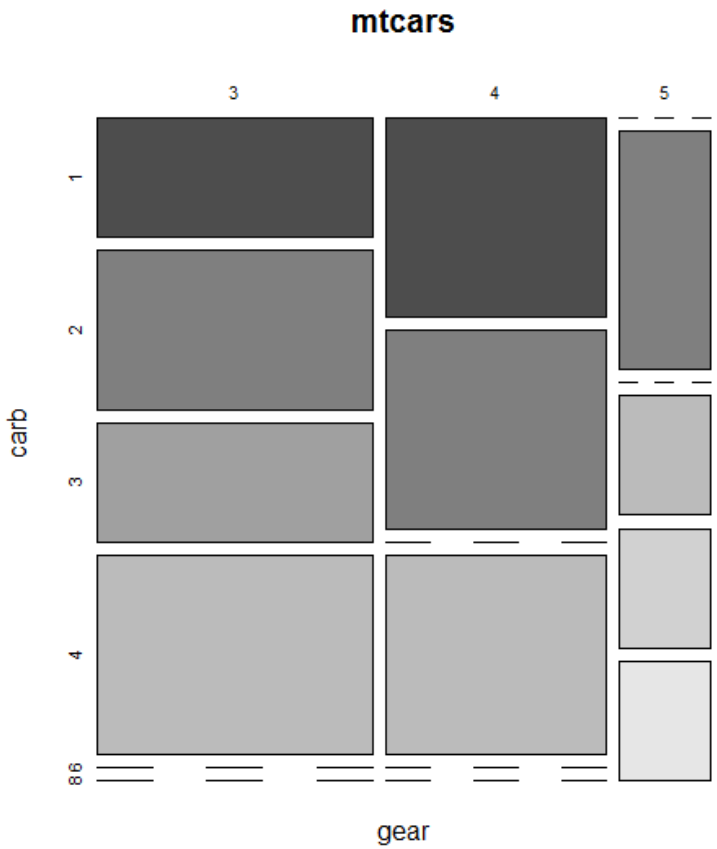
이와 같이 모자이크 플롯은 여러가지 정보를 한눈에 표현할 수 있다

모자이크 플롯 (mosaic plot)

- 설치 필요 패키지
  - 없음
- 실습용 데이터셋
  - mtcars
  - Titnic

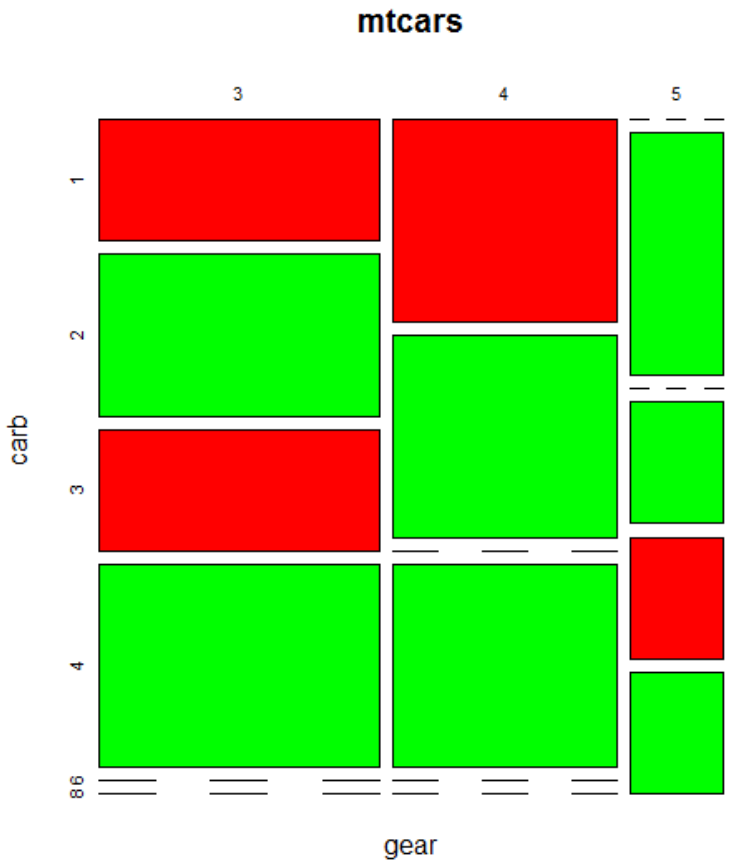
모자이크 플롯 (mosaic plot)

```
rm(list=ls())    # 앞의 작업 결과 clear
# matrix 형태로 데이터가 존재하는 경우
head(mtcars)
mosaicplot(~gear+carb, data = mtcars, color = TRUE)
```



모자이크 플롯 (mosaic plot)

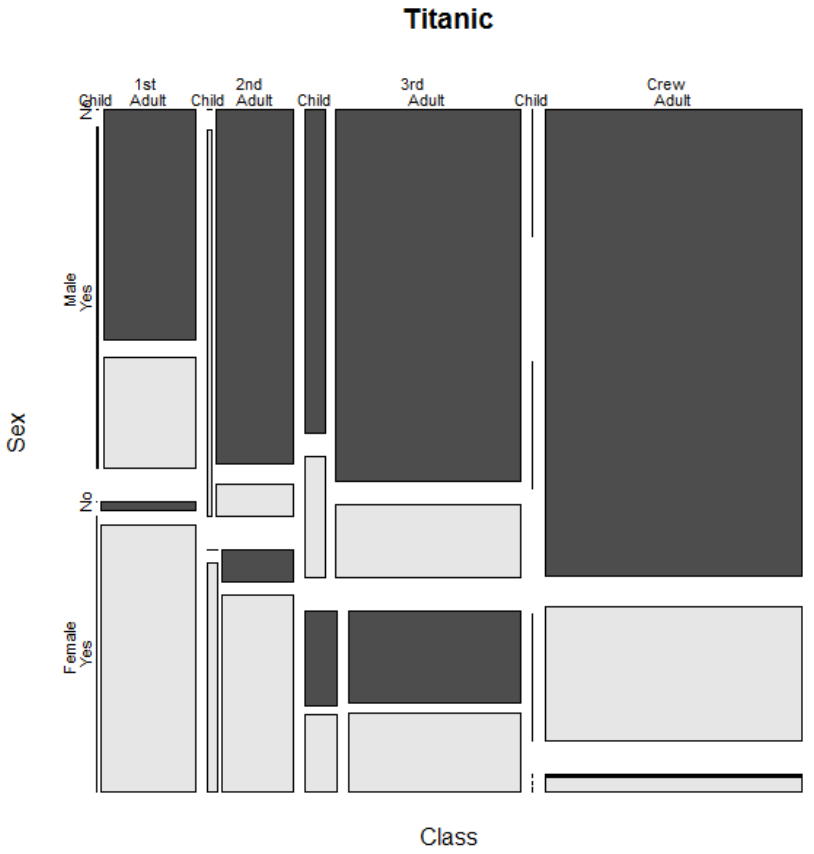
```
mosaicplot(~gear+carb, data = mtcars,
           color =  c("red","green"))
```



모자이크 플롯 (mosaic plot)

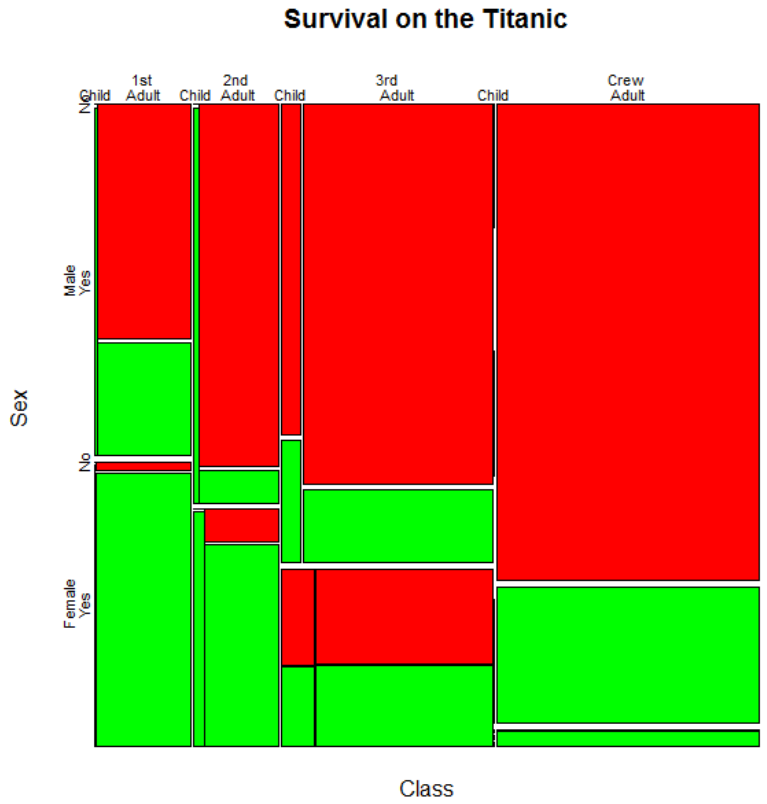
```
# 교차표 형태로 데이터가 존재하는 경우
Titanic
mosaicplot(Titanic, color = TRUE, off=5)
```

```
> Titanic
, , Age = Child, Survived = No

      Sex
Class  Male Female
  1st     0      0
  2nd     0      0
  3rd    35     17
  Crew    0      0

, , Age = Adult, Survived = No

      Sex
Class  Male Female
  1st   118      4
  2nd   154     13
```



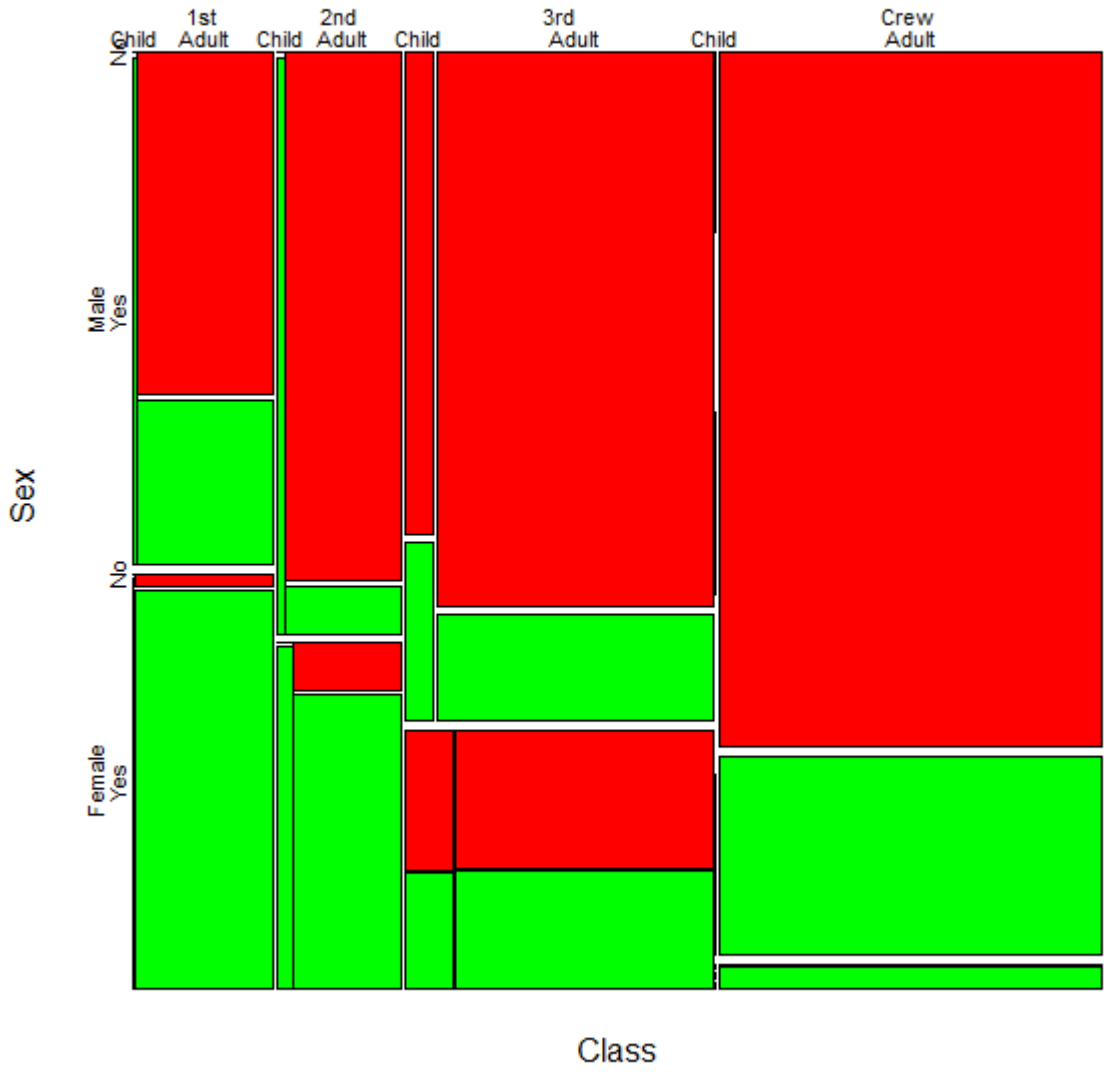
모자이크 플롯 (mosaic plot)

```
mosaicplot(Titanic,
           main = "Survival on the Titanic",
           color = c("red","green"),
           off=1) # 블럭들 사이의 간격 지정
```



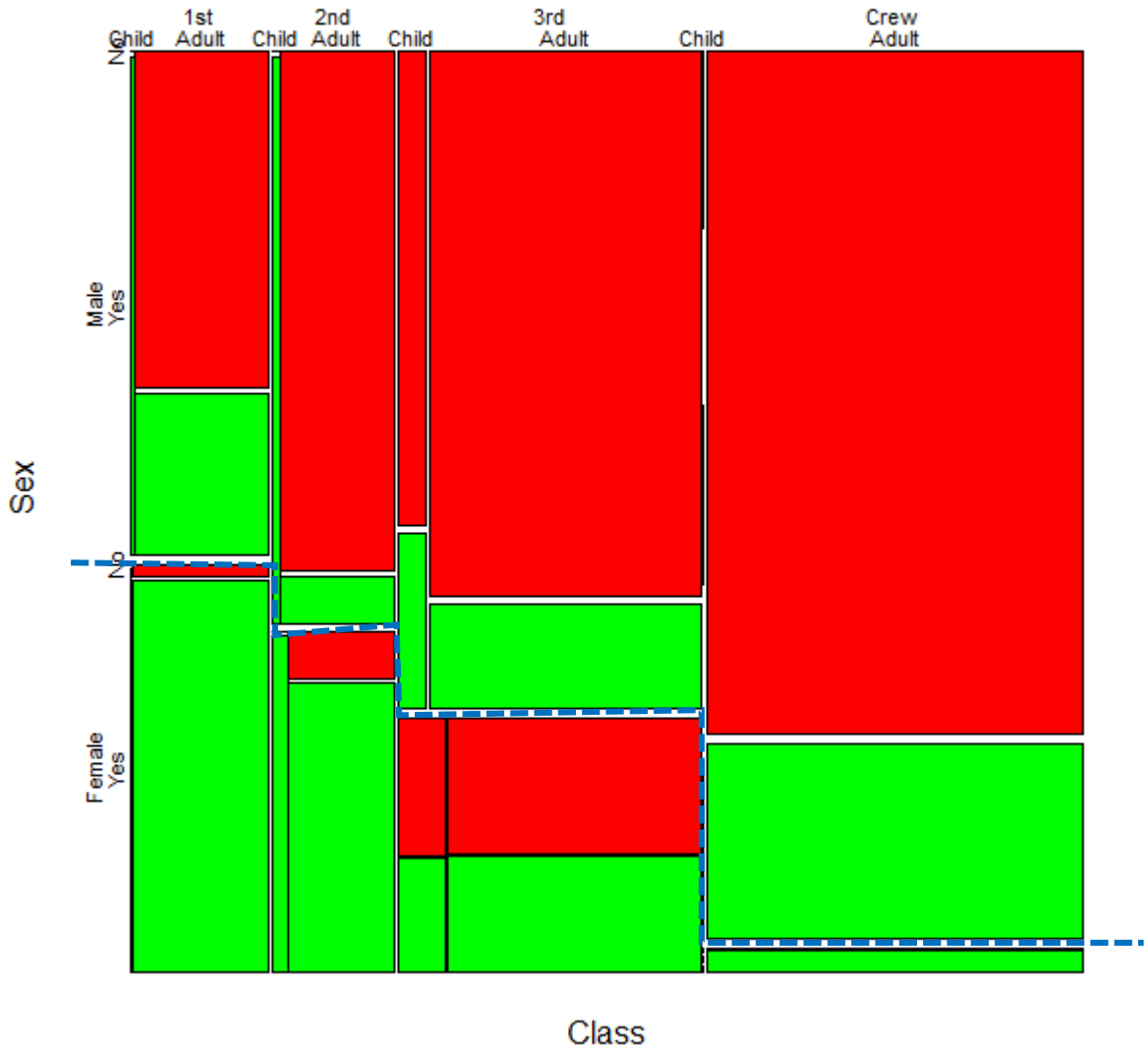
붉은색 : 사망
연두색 생존

Survival on the Titanic



붉은색 : 사망
연두색 생존

Survival on the Titanic



붉은색 : 사망
연두색 생존

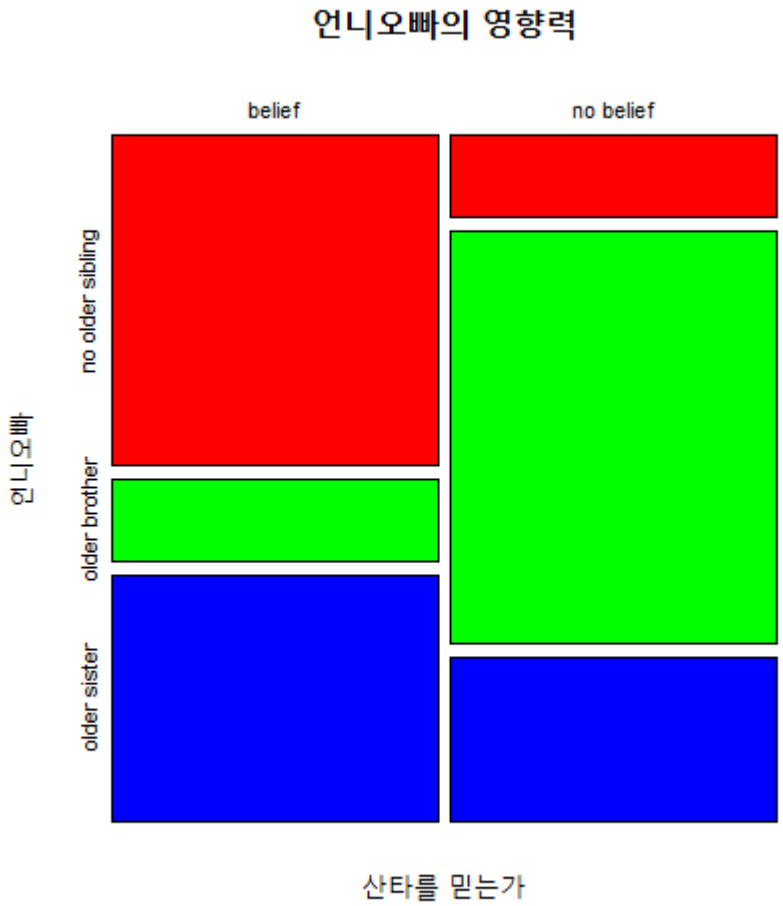
[연습 4]

1. HairEyeColor 데이터셋에 대해 모자이크 플롯을 작성하시오. 여기서 관찰할 수 있는 정보는 무엇인가

2. 다음의 santa data 에 대해 모자이크 플롯을 작성하시오 (다음 slide 참조). 여기서 관찰할 수 있는 정보는 무엇인가

```
santa <- data.frame(belief=c('no belief','no belief','no belief','no belief',
                             'belief','belief','belief','belief',
                             'belief','belief','no belief','no belief',
                             'belief','belief','no belief','no belief'),
                    sibling=c('older brother','older brother','older brother','older sister',
                              'no older sibling','no older sibling','no older sibling','older sister',
                              'older brother','older sister','older brother','older sister',
                              'no older sibling','older sister','older brother','no older sibling')
                    )
```

* belief : 산타를 믿는지여부, sibling: 손위 형제가 있는지 여부



http://www.cyclismo.org/tutorial/R/intermediatePlotting.html