

막대그래프를 그려주는 함수

- 📖 막대그래프 그리기
- 📖 마이크로 데이터와 색깔
- 📖 누적 막대그래프와 비교 막대그래프
- 📖 범례 추가, 빗금 만들기(density, angle)
- 📖 barplot 함수의 출력

Loren ipsum dolor sit amet, ius an molestie facilisi erroribus, mutat nalerum delectus ei vis. Has ornatus conclusionemque id, an videri molestatis sit. In etqui praesent sit. An vel agan porro comprehensan, ad ludus constituto nea, et ius utroque scaevola assuaverit.

Vis cu nodus nulla feugait, oratio facilisi in usu, eili vitae sea te. Ea fabulas accusamus dissentias sea, facete tacinates definitiones et per. Nihil dicant mediocrem pro eu, no mei nostro sensibus platonem. Qui id sunno perpetua neglegantur. Vel ipsum novum copiosae ut. Quo et liber detracto probatus. Nam augue scribentur an. Sea oporteat percipitur incidere at. Qui viris nemore an.



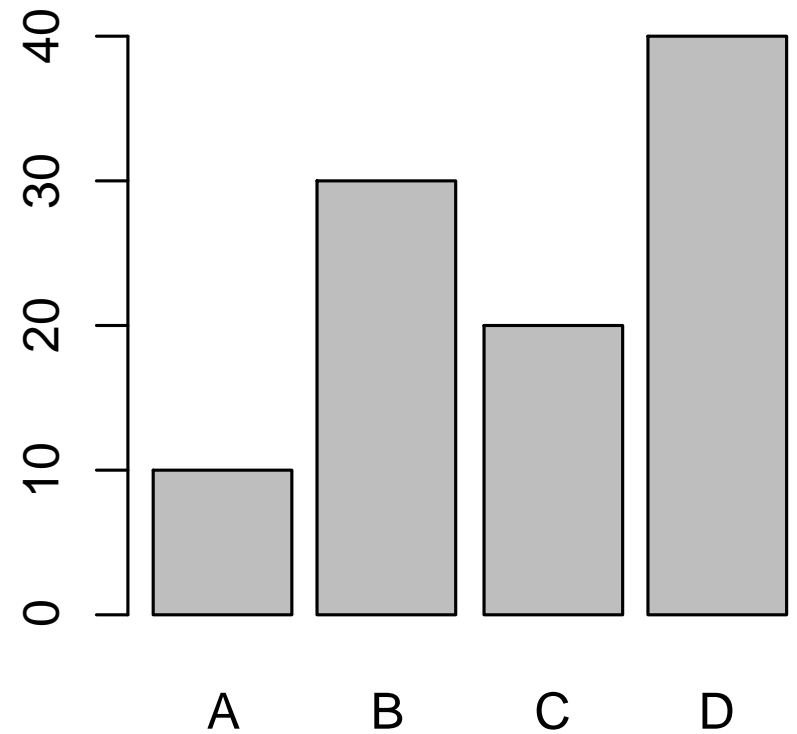
막대그래프 그리기



막대그래프의 사용

+ 주로 범주형 자료를 그림으로 표현하기 위해 사용

- 막대의 높이가 해당 범주의 빈도수 또는 비율이 됨
- 각 막대는 일반적으로 분리되도록 그림





막대그래프 그리기



사용 함수 : barplot

```
barplot(height, names.arg = NULL, legend.text = NULL, beside = FALSE,  
        horiz = FALSE, density = NULL, angle = 45,  
        col = NULL, border = par("fg"),  
        main = NULL, sub = NULL, ...)
```

+ 매개변수

- height : 기둥의 높이(빈도) 벡터 설정
- horiz : 기둥을 가로로 그릴지 세로로 그릴지 설정, 기본값은 FALSE로 세로 기둥
- beside : 비교 막대 그래프
- main, sub : 주제목과 보조제목 설정
- names.arg : 각 기둥에 이름 주기
- legend.text : 범례의 이름 주기
- col : 각 기둥에 사용할 색깔 설정 : 색깔의 이름이나 숫자
- density, angle : 기둥 내부를 빗금으로 처리할 때 빗금의 수 및 각도





막대그래프 그리기



예제



자료

A, B, C, D 네 개의 범주에 대한 빈도수가 각각 10, 30, 20, 40인 경우를 생각해보자.

빈도표

범주	A	B	C	D
빈도	10	30	20	40

이 자료를 사용하여 막대그래프를 그려보자.



빈도수, 각 범주의 이름을 각각 freq, category라고 하고 아래의 명령으로 두 개의 빈도표 그리기

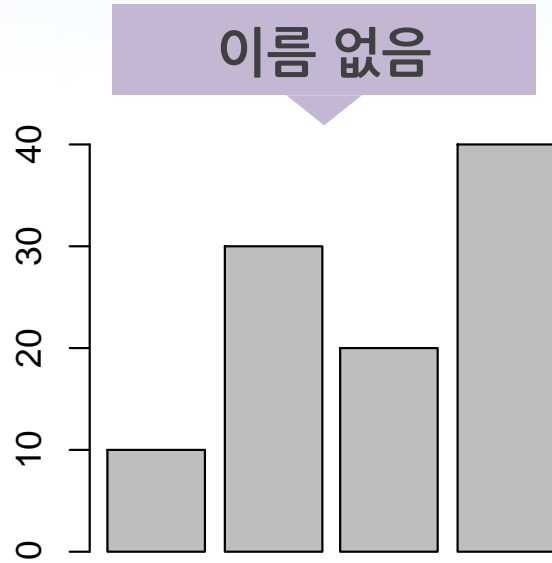
```
> freq <- c(10,30,20,40)
> barplot(freq, main="이름없음")
> category <- c("A", "B", "C", "D")
> barplot(freq, main="이름 있음", names.arg=category)
```



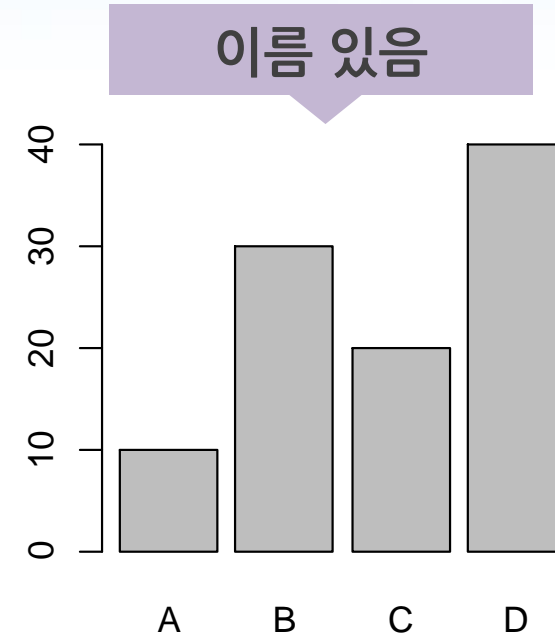
막대그래프 그리기



예제



- names.arg가 설정되지 않은 barplot 함수로 가로축에 각 범주의 이름이 설정되지 않았음



- names.arg가 설정되어 가로축에 이름이 나타남
- main 옵션은 그림의 제목을 설정함



막대그래프 그리기



예제

+ 참고 : 변수값에 대한 이름이 설정된 경우

- names.arg가 따로 설정되지 않아도 자동으로 가로축에 이름이 나타남
- 예제의 명령 대신 다음과 같이 이름을 설정하고 barplot 함수를 호출하면 위의 오른쪽 막대그래프와 동일한 결과를 얻게 됨

```
> freq <- c(10,30,20,40)  
> names(freq) <- c("A", "B", "C", "D")  
> barplot(freq, main="이름 있음")
```



마이크로 데이터



마이크로 데이터

사용할 자료

```
> BMI <- read.table(url("http://jupiter.hallym.ac.kr/ftpdata/data/bmi.txt"),  
  col.names=c("height", "weight", "year", "religion", "gender", "marriage"))
```



자료출처

- url 함수를 사용하여 인터넷 사이트에서 바로 읽어 올 수 있으며, 위의 인터넷주소를 브라우저 창에 붙여 넣으면 브라우저에서 볼 수 있음



내용 : 2000년, 177명에 대한 조사 결과

- 키, 몸무게, 출생년도
- 종교(Bu=불교, C1=개신교, C2=가톨릭, No=없음)
- 성별(F=여자, M=남자)
- 결혼여부(N=미혼, Y=기혼)



마이크로 데이터



마이크로 데이터

```
> BMI <- read.table(url("http://jupiter.hallym.ac.kr/ftpdata/data/bmi.txt"),  
  col.names=c("height", "weight", "year", "religion", "gender", "marriage"))  
> barplot(BMI$religion) # 오류  
> barplot( table(BMI$religion) ) # OK  
> barplot( table(BMI$religion), horiz= T ) # OK
```

⊕ 첫 번째 호출은 자료가 마이크로 데이터인 형태이므로 오류가 발생함

➔ 매크로 데이터 형태(각 범주의 빈도수 또는 비율이 포함된 자료)로 교체

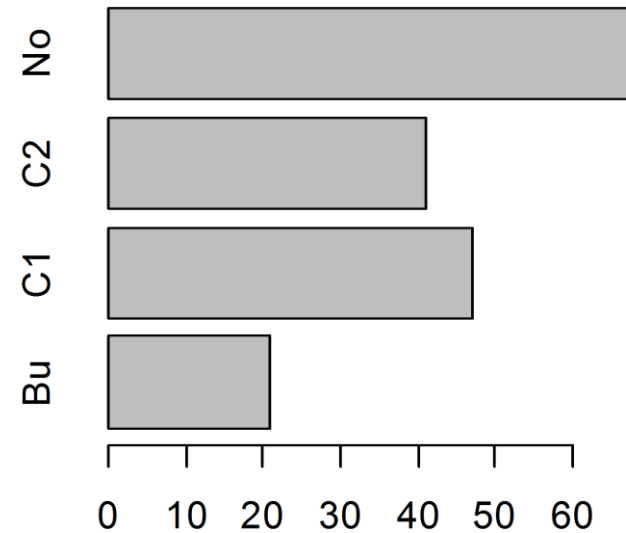
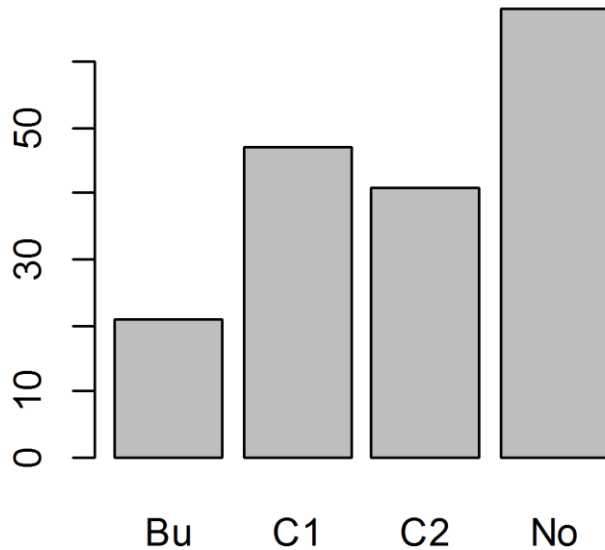


마이크로 데이터



마이크로 데이터

- + 두 번째 table 함수에 의해 얻은 도수분포표는 barplot 함수를 호출한 결과임(왼쪽 그래프)
 - names.arg가 설정되지 않아도 가로축에 이름이 나타난 것은 table 함수의 출력이 각 범주값을 해당 빈도의 이름으로 갖기 때문
- + 세 번째 barplot 함수에 의해 얻은 도수분포표는 horiz 옵션에 T로 설정하여 막대를 가로로 그리도록 설정한 결과임(오른쪽 그래프)





색깔 설정



색깔

+ barplot 함수(뿐 아니라 다른 R-그래픽에서도)에서 색의 설정

➤ 숫자로 0-8까지의 번호나 색깔의 영문이름 사용

0	1	2	3	4	5	6	7	8
white	black	red	green	blue	cyan	magenta	yellow	grey/gray

➤ 색깔 이름으로 사용할 수 있는 이름은 훨씬 많음

[예] orange, NavyBlue 등

➤ 색깔 이름에 따옴표를 사용함

[예] `barplot(table(BMI$religion), col="orange")`



색깔 설정



색깔

- + R-그래픽에서 사용할 수 있는 색의 이름 확인 파일

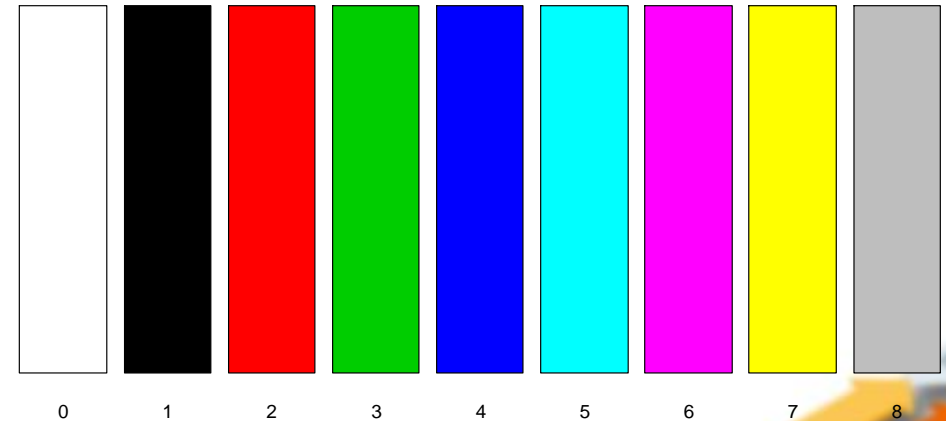
C:\Program Files\R\R-*.*.etc\rgb.txt

- *.*. : 설치된 R-언어의 버전이며 R-3.3.1의 경우 "AliceBlue"부터 "YellowGreen"까지 657개의 이름을 사용할 수 있음

- + barplot 함수 사용 결과

```
barplot(rep(10,9), col=0:8,
names.arg=as.character(seq(0,8)), yaxt="n")
```

* yaxt 옵션은 y 축 눈금표시를 설정하는데 "n"은 없으므로 y축의 축이 만들어지지 않았음



누적 막대그래프와 비교 막대그래프



누적 막대그래프

+ 두 개 이상의 열을 가진 행렬/데이터 프레임/표가 빈도로 입력될 경우 각 열의 값에 대해서 누적해서 막대 그래프를 그림

+ 그림의 순서

- 첫 번째 열은 첫 번째 기둥에, 두 번째 열은 두 번째 기둥 등등의 순서 열을 구분함
- 첫 번째 행은 가장 아래, 두 번째 행은 두 번째 층 등등

➔ 행렬에서 각 원소의 위치와 누적 막대그래프의 각 칸은 행의 순서만 역순인 위치에 있게 됨



누적 막대그래프와 비교 막대그래프

누적 막대그래프

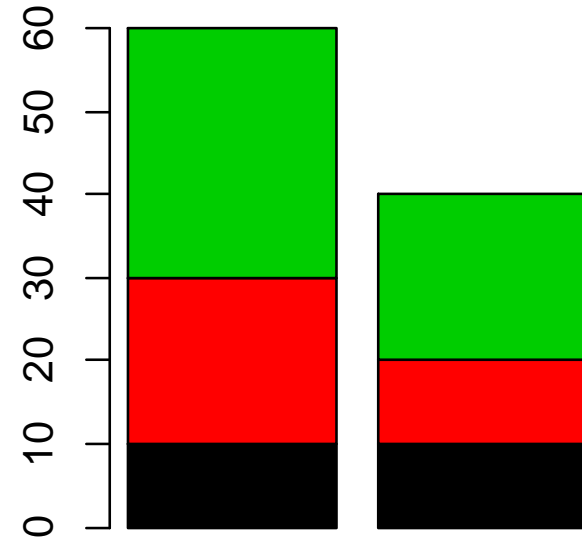


자료

두 개의 열을 가진 행렬을 사용하여 누적 막대그래프를 그려보자.

+ 첫 번째, freq에 자료의 값을 두 개의 열을 가진 행렬로 설정하고 이를 막대그래프로 그리기

```
freq <- cbind( c(10,20,30), c(10,10,20))  
par(mfrow=c(1,2))  
barplot(freq, col=1:3)
```



누적 막대그래프와 비교 막대그래프

누적 막대그래프



자료

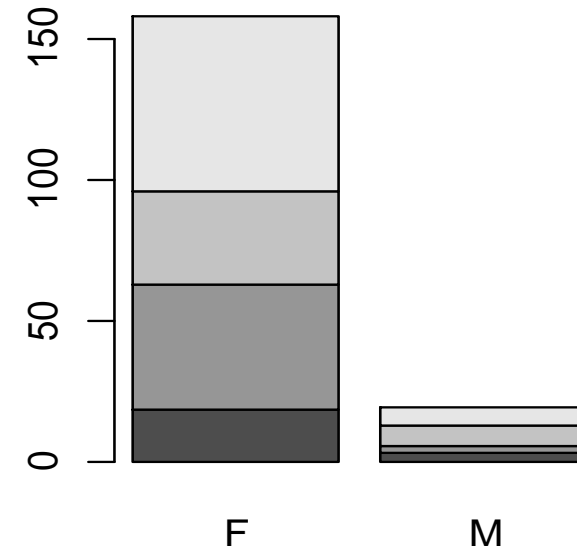
두 개의 열을 가진 행렬을 사용하여 누적 막대그래프를 그려보자.

- + 두 번째, barplot 함수는 키 몸무게 자료에서 성별에 따른 종교에 대한 교차표를 table 함수로 얻어 이를 막대그래프로 그리기

```
> barplot( table(BMI$religion, BMI$gender) )
```

```
> table(BMI$religion, BMI$gender)
```

	F	M
Bu	18	3
C1	45	2
C2	33	8
No	62	6

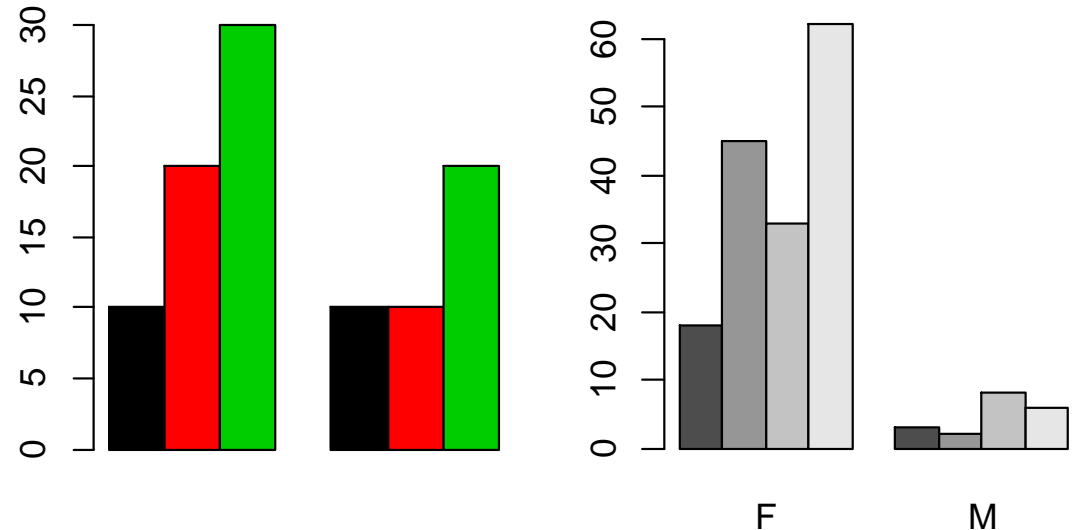


누적 막대그래프와 비교 막대그래프

비교 막대그래프

- + 각 그룹 별로 기둥을 좌우(막대가 가로인 경우 상하)로 배치하여 그룹간 비교를 직관적으로 할 수 있는 막대그래프로 빈도수를 포함한 행렬을 입력으로 받으며 beside 옵션이 설정되어야 함
- + 함수 사용 결과

```
freq <- cbind( c(10,20,30), c(10,10,20))  
par(mfrow=c(1,2))  
barplot(freq, col=1:3, beside=T)  
barplot( table(BMI$religion, BMI$gender),  
beside=T )
```





범례 추가, 빗금 만들기(density, angle)



범례(legend) 추가



- 누적 또는 비교 막대그래프는 가로축엔 이름이 자동 또는 names.arg 옵션으로 넣을 수 있으나 각 그룹 내의 기둥에 대한 구분이 되어 있지 않아 해석하기가 불편하므로, 이를 해결하기 위해 **범례(legend)를 추가**할 수 있음



빗금 만들기



- 색깔이 상자의 내부를 채우는 경우, 천연색이 아닌 흑백이면 구분이 안가는 경우도 있어 (또한, 잉크나 토너 절약) 빗금으로 기둥을 채우는 angle 옵션을 설정할 수 있음





범례 추가, 빗금 만들기(density, angle)



예제



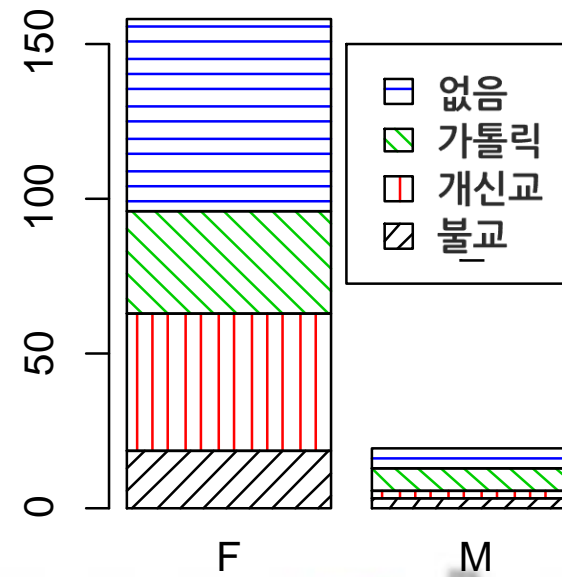
자료

다음은 범례와 빗금으로 키, 몸무게 자료에서 성별 종교를 막대그래프로 구현한 것이다. 아래에서 옵션에 사용한 legend.txt는 범례의 문자 설명이며 angle, density는 빗금의 각도 및 빗금의 밀도이다. 빗금의 밀도는 인치당 빗금의 수를, 빗금의 각도는 수평에서 반시계 방향의 각도를 설정한다.



함수 사용 결과

```
barplot(table(BMI$religion, BMI$gender),  
legend.text=c("불교", "개신교", "가톨릭",  
"없음"), col=1:4, density=15, angle=c(45, 90,  
135, 180))
```





barplot 함수의 출력



각 기둥의 중간에 해당하는 x 좌표를 얻을 수 있음

```
> bpres <- barplot(table(BMI$religion, BMI$gender), legend.text=c("불교", "개신교",  
  "가톨릭", "없음"), col=1:4, density=15, angle=c(45, 90, 135, 180))  
> bpres  
[1] 0.7 1.9
```



그래픽 장치 설정과 그래픽 옵션 설정

📖 그래픽 장치 설정

📖 그래픽 옵션 설정

Loren ipsum dolor sit amet, ius an molestie facilisi erroribus, mutat nalerum delectus ei vis. Has ornatus conclusionemque id, an videri molestatis sit. In etqui praesent sit. An vel agan porro comprehensan, ad ludus constituto nea, et ius utroque scaevola assuaverit.

Vis cu nodus nulla feugait, oratio facilisi in usu, eili vitae sea te. Ea fabulas accusamus dissentias sea, facete tacinates definitiones et per. Nihil dicant mediocram pro eu, no mei nostro sensibus platonem. Qui id sunno perpetus neglegantur. Vel ipsum novum copiosae ut. Quo et liber detracto probatus. Nam augue scribentur an. Sea oporteat percipitur incidere at. Qui viris nemore an.





그래픽 장치 설정



사용 함수

- + `win.graph(width, height, ...)` : 그래픽 윈도우 열기
- + `png(filename = "Rplot%03d.png",...)`: png 파일로 바로 저장
- + `jpeg(filename = "Rplot%03d.jpg",...)`: jpg 파일로 바로 저장
- + `tiff(filename = "Rplot%03d.tif",...)`: tiff 파일로 바로 저장
- + `dev.off(which=dev.cur())`: 현재의 또는 지정된 그래픽 장치 닫기
- + `graphics.off()` : 모든 그래픽 장치 닫기



그래픽 장치 설정



함수 사용 결과

```
setwd("D:/hwp/STI")  
jpeg()  
barplot(table(BMI$religion, BMI$gender),  
legend.text=c("불교", "개신교", "가톨릭", "없음"),  
col=1:4, density=15, angle=c(45, 90, 135, 180))  
dev.off()  
null device  
1
```

* 결과로 D:/hwp/STI 폴더에 Rplot001.jpg 파일이 만들어짐

+ 주의 : 같은 이름의 파일이 있는 경우 묻지 않고 덮어쓰게 됨



그래픽 옵션 설정



par 함수

- + R-그래픽에서 사용하는 모든 옵션은 par 함수로 글로벌하게 설정할 수 있음
- + 개별 그래픽 함수에 별로 설정이 있는 경우 해당 그래픽에서는 그래픽 함수의 설정을 따름
- + par() 로 par 함수를 호출하면 현재의 그래픽 설정을 볼 수 있으며 그래픽 옵션은 아주 다양하므로 R-언어의 help 명령으로 도움말을 참고하기로 함



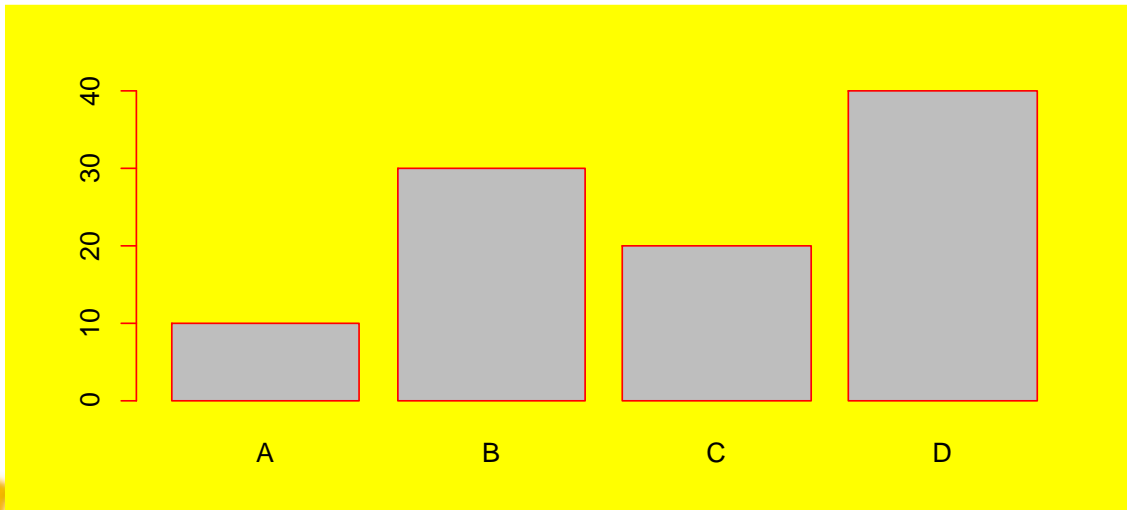
그래픽 옵션 설정



par 함수 사용 결과

```
freq <- c(10,30,20,40)  
category <- c("A", "B", "C", "D")  
par(bg="yellow", fg="red")  
barplot(freq, names.arg=category)
```

- par 함수에서 배경색(bg 옵션; background 의미)과 전경색(fg 옵션; foreground 의미)을 각각 bg="yellow"와 fg="red"로 설정하였음



* bg="yellow " 의 기본값 : white

* fg="red " 의 기본값 : black

3차원 막대그래프

- 📖 cloud 함수의 사용
- 📖 3차원 막대그래프 그리기

Loren ipsum dolor sit amet, ius an molestie facilisi erroribus, mutat natorum delectus ei vis. Has ornatus conclusionemque id, an videri molestatis sit. In etqui praesent sit. An vel agan porro comprehensan, ad ludus constituto nea, et ius utroque scaevola assumaverit.

Vis cu nodus nulla feugait, oratio facilisi in usu, eili vitae sea te. Ea fabulas accusamus dissentias sea, facete tacinates definitiones et per. Nihil dicant mediocrem pro eu, no mei nostro sensibus platonem. Qui id sunno perpetua neglegentur. Vel ipsum novum copiosae ut. Quo et liber detracto probatus. Nam augue scribentur an. Sea oporteat percipitur incidereat ab. Qui viris nemore an.

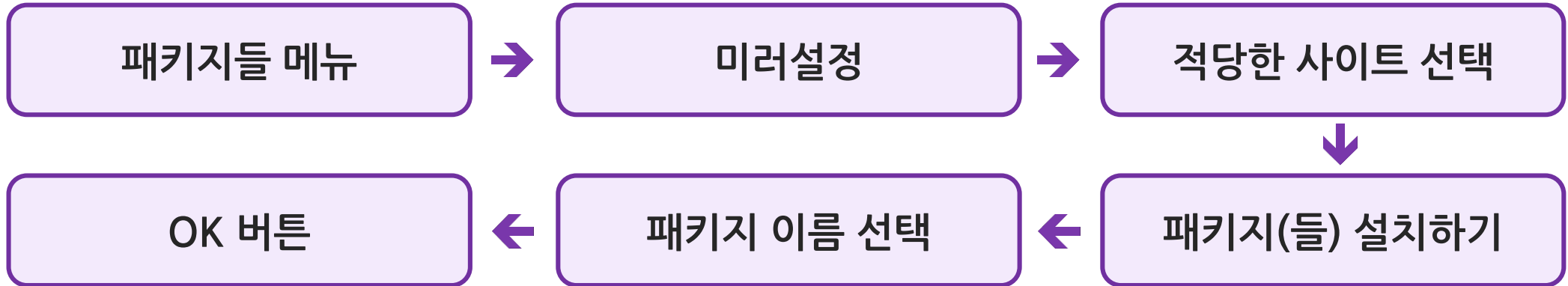


cloud 함수의 사용



3차원 막대그래프는 latticeExtra 패키지의 cloud 함수를 사용

+ 패키지의 설치



+ 설치된 패키지를 사용하려면 library 함수로 해당 패키지를 불러옴

+ **주의** : 3차원 막대그래프은 어떤 형태이든 왜곡이 발생됨



cloud 함수의 사용



latticeExtra 패키지에서 제공되는 cloud 함수의 사용



사용 함수

```
cloud(x, data, panel.3d.cloud=panel.3dbars, xbase, ybase, col.facet, col, scales ...)
```



매개변수

- x : 행렬이 설정되며 각 원소는 기둥의 높이를 나타낸다. x에 formula: freq ~ x+y 형식으로 빈도수, x축, y축에 해당하는 변수를 설정할 수도 있다.
- data : 데이터 프레임 이름
- panel.3d.cloud : 그릴 그림의 형식을 설정하며 3차원 막대그래프는 panel.3d.bars로 설정
- xbase, ybase : 기둥의 바닥 면적의 비율
- col.facet, col : 기둥의 면의 색깔 및 선의 색깔 설정
- scales : 그림의 스케일에 대한 속성 설정



3차원 막대그래프 그리기



빈도수가 freq인 3차원 막대그래프 그리기



자료

다음의 자료는 x값, 빈도수, y값을 사용하여 (x,y)에서 빈도수가 freq인 3차원 막대그래프를 그리는 자료이다. 자료를 설정하고 3차원 막대그래프를 그려보자.

```
df<- read.table(  
text=' x   freq   y  
level1    5  high  
level1   21  med  
level1    4  low  
level2    8  high  
level2   19  med  
level2    3  low  
level3    5  high  
level3   12  low  
level3   50  med', header=TRUE)
```



3차원 막대그래프 그리기



빈도수가 freq인 3차원 막대그래프 그리기



- 자료 df를 사용하여 3차원 막대그래프를 그리기 위해, library 함수를 사용하여 패키지를 load하고 cloud 함수 호출하기

```
library(latticeExtra)
```

```
cloud(freq~x+y, data=df, panel.3d.cloud=panel.3dbars, xbase=0.4, ybase=0.4,  
col.facet=3, col=2)
```

```
cloud(freq~x+y, data=df, panel.3d.cloud=panel.3dbars, xbase=0.4, ybase=0.4,  
      xlab="", ylab="",  
      scales=list(arrows=F, just='right'),  
      col.facet = level.colors(df$freq, at = do.breaks(range(df$freq), 20),  
                               col.regions = topo.colors,  
                               colors = TRUE),  
      colorkey = list(col = topo.colors, at = do.breaks(range(df$freq), 20))  
)
```



3차원 막대그래프 그리기

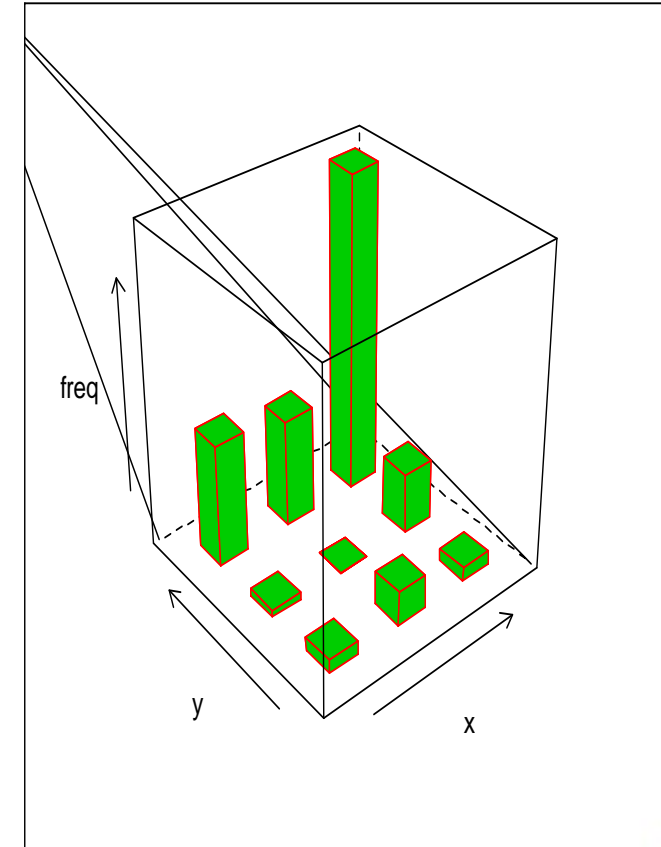


빈도수가 freq인 3차원 막대그래프 그리기



첫 번째 cloud 함수

- 기본적인 설정만 사용하여 (x,y) 에서의 빈도가 freq인 $\text{freq} \sim x+y$ 로 관계식을 설정함
- 기둥의 바닥면에 대한 비율을 0.4로, 기둥면과 기둥모서리의 색은 각각 3번색(green), 2번색(red)로 설정함





3차원 막대그래프 그리기



빈도수가 freq인 3차원 막대그래프 그리기



두 번째 cloud 함수

- 기본 설정은 첫 번째와 같지만 xlab, ylab에 모두 ""로 설정하여 축의 이름을 삭제하고 **scales에 arrow=F, just='right'로 설정하여** 화살표 출력대신 x, y의 값을 출력하며 이 문자열은 오른쪽 정렬을 하도록 하였음
- col.facet에는 freq의 값을 20개의 구간으로 (따라서 21개의 값)나눈 구간의 끝점에 따라 색을 다르게 설정하며, 이 때 사용할 색은 topo.colors임
- topo.colors는 **지도에서 색을 사용할 때 사용하는 색임**
- colorkey는 그림의 오른쪽에 색에 따른 빈도수 설명을 만들며 이 설명은 topo.colors를 col.facet에 설정된 구간에 적용하였음

