고급 시각화 - (ggplot2)를 중심으로

Sangkon Han(sangkon@pusan.ac.kr)

2023-03-21

```
install.packages("lattice", repos = "https://cran.us.r-project.org")
install.packages("mlmRev", repos = "https://cran.us.r-project.org")
install.packages("ggplot2", repos = "https://cran.us.r-project.org")
```

```
## 패키지 'ggplot2'를 성공적으로 압축해제하였고 MD5 sums 이 확인되었습니다
##
## 다운로드된 바이너리 패키지들은 다음의 위치에 있습니다
##  C:\Users\sigma\AppData\Local\Temp\RtmpIpaeMF\downloaded_packages
```

패키지 설치 및 데이터 준비

lattice과 mlmRev를 활성화하세요.

```
library(lattice)
library(mlmRev)
```

해당 데이터를 불러옵니다.

```
data("Chem97")
```

먼저 데이터의 구성을 확인합니다.

```
str(Chem97)
```

```
## 'data.frame':    31022 obs. of  8 variables:
##  $ lea      : Factor w/ 131 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ school   : Factor w/ 2410 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ student  : Factor w/ 31022 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ score    : num  4 10 10 10 8 10 6 8 4 10 ...
##  $ gender   : Factor w/ 2 levels "M","F": 2 2 2 2 2 2 2 2 2 2 ...
##  $ age      : num  3 -3 -4 -2 -1 4 1 4 3 0 ...
##  $ gcsescore: num  6.62 7.62 7.25 7.5 6.44 ...
##  $ gcsecnt  : num  0.339 1.339 0.964 1.214 0.158 ...
```

데이터의 앞부분 30개를 살펴보겠습니다.

```r
head(Chem97, 30)
```

```
##    lea school student score gender age gcsescore     gcsecnt
## 1    1      1       1     4      F   3     6.625  0.33931571
## 2    1      1       2    10      F  -3     7.625  1.33931571
## 3    1      1       3    10      F  -4     7.250  0.96431571
## 4    1      1       4    10      F  -2     7.500  1.21431571
## 5    1      1       5     8      F  -1     6.444  0.15831571
## 6    1      1       6    10      F   4     7.750  1.46431571
## 7    1      1       7     6      F   1     6.750  0.46431571
## 8    1      1       8     8      F   4     6.909  0.62331571
## 9    1      1       9     4      F   3     6.375  0.08931571
## 10   1      1      10    10      F   0     7.750  1.46431571
## 11   1      1      11    10      F  -1     7.857  1.57131571
## 12   1      1      12     8      F   1     7.333  1.04731571
## 13   1      1      13    10      F   1     7.750  1.46431571
## 14   1      2      14    10      M   0     7.700  1.41431571
## 15   1      2      15    10      M  -4     6.300  0.01431571
## 16   1      2      16    10      M   5     7.300  1.01431571
## 17   1      2      17     8      M  -3     6.636  0.35031571
## 18   1      2      18    10      M   4     7.272  0.98631571
## 19   1      2      19    10      M   0     7.200  0.91431571
## 20   1      2      20     4      M  -3     6.454  0.16831571
## 21   1      2      21     6      M   4     6.818  0.53231571
## 22   1      2      22    10      M  -5     7.300  1.01431571
## 23   1      2      23     2      M  -1     6.200 -0.08568429
## 24   1      2      24    10      M  -2     7.111  0.82531571
## 25   1      2      25    10      M   2     6.800  0.51431571
## 26   1      2      26     8      M  -4     6.500  0.21431571
## 27   1      2      27    10      M  -5     6.727  0.44131571
## 28   1      2      28     6      M  -6     7.000  0.71431571
## 29   1      2      29    10      M  -2     7.700  1.41431571
## 30   1      2      30    10      M   3     7.300  1.01431571
```

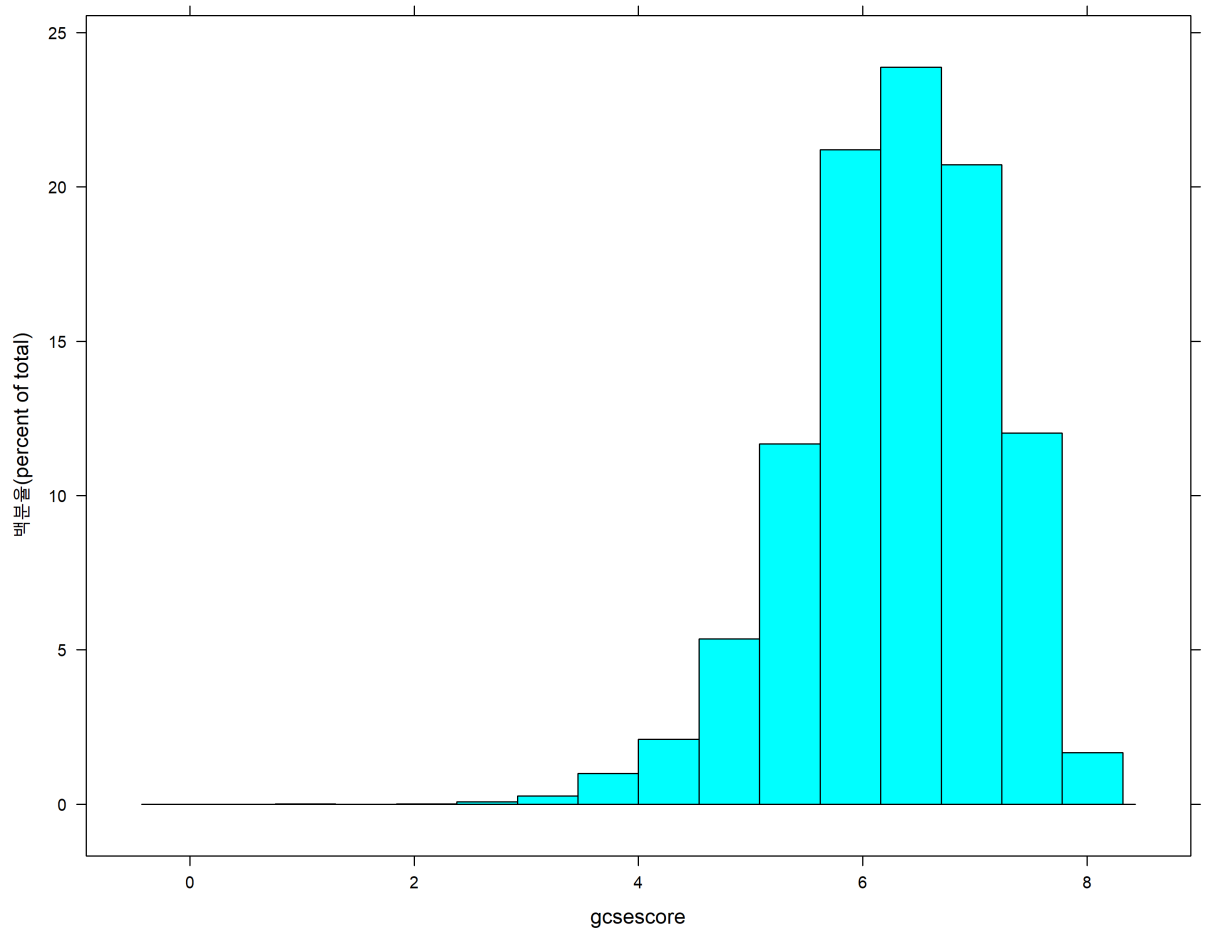
간단한 그래프 작성

lattice는 간단한 직교형태의 그래픽을 구성하는 방법을 포함하고 있습니다. R에서 제공하는 것과 별도로 작동하며, 데이터 셋의 특징을 전반적으로 보여주는 것이 주요한 특징입니다.

히스토그래프

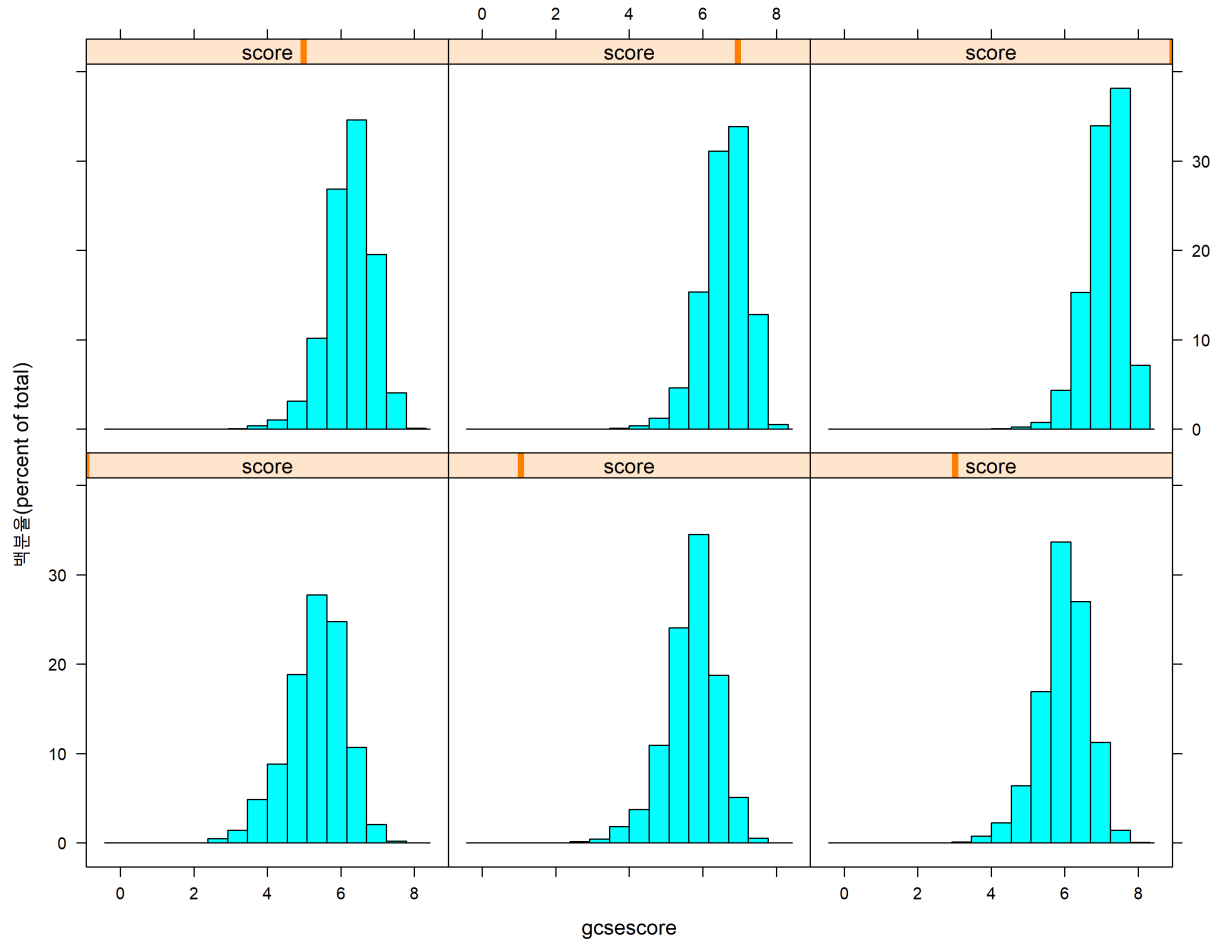
histogram() 함수를 이용하여 데이터 시각화를 시작해보겠습니다.

```r
histogram(~gcsescore, data = Chem97)
```



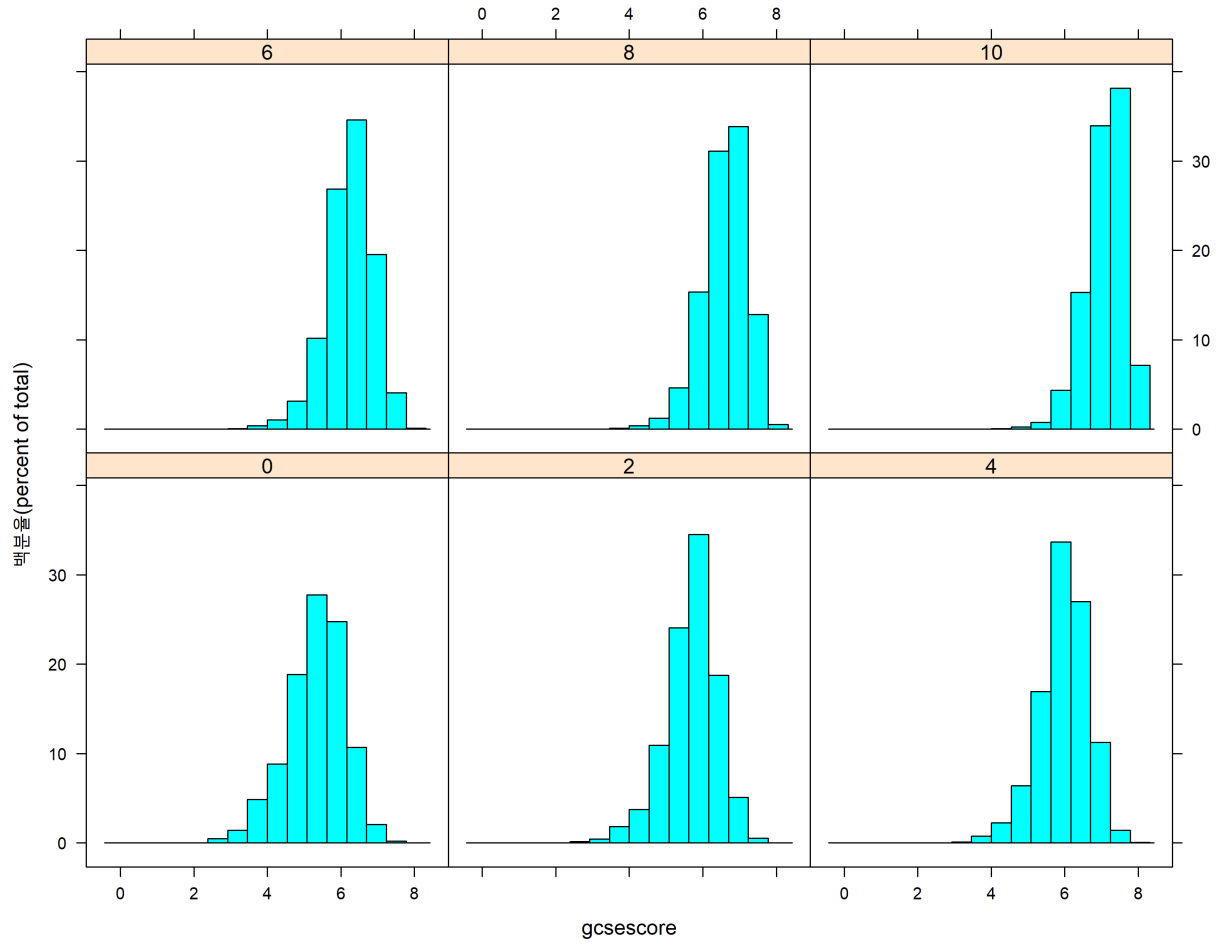
score 변수를 **조건**변수로 지정하여 데이터를 시각화하는 방법은 아래와 같습니다. score 변수를 사용하면 주황색 기준으로 격자형으로 분리해서 분포를 보여줍니다. 하지만 해당 주황색의 값을 정확하게 알기 쉽지 않습니다.

```
histogram(~gcsescore | score, data = Chem97)
```



`factor`를 사용하여 `score` 값을 손쉽게 확인할 수 있습니다. factor는 범주 값을 반환하는 것으로(0~10) score를 x축에 대입하면, 순서대로 적용됩니다.

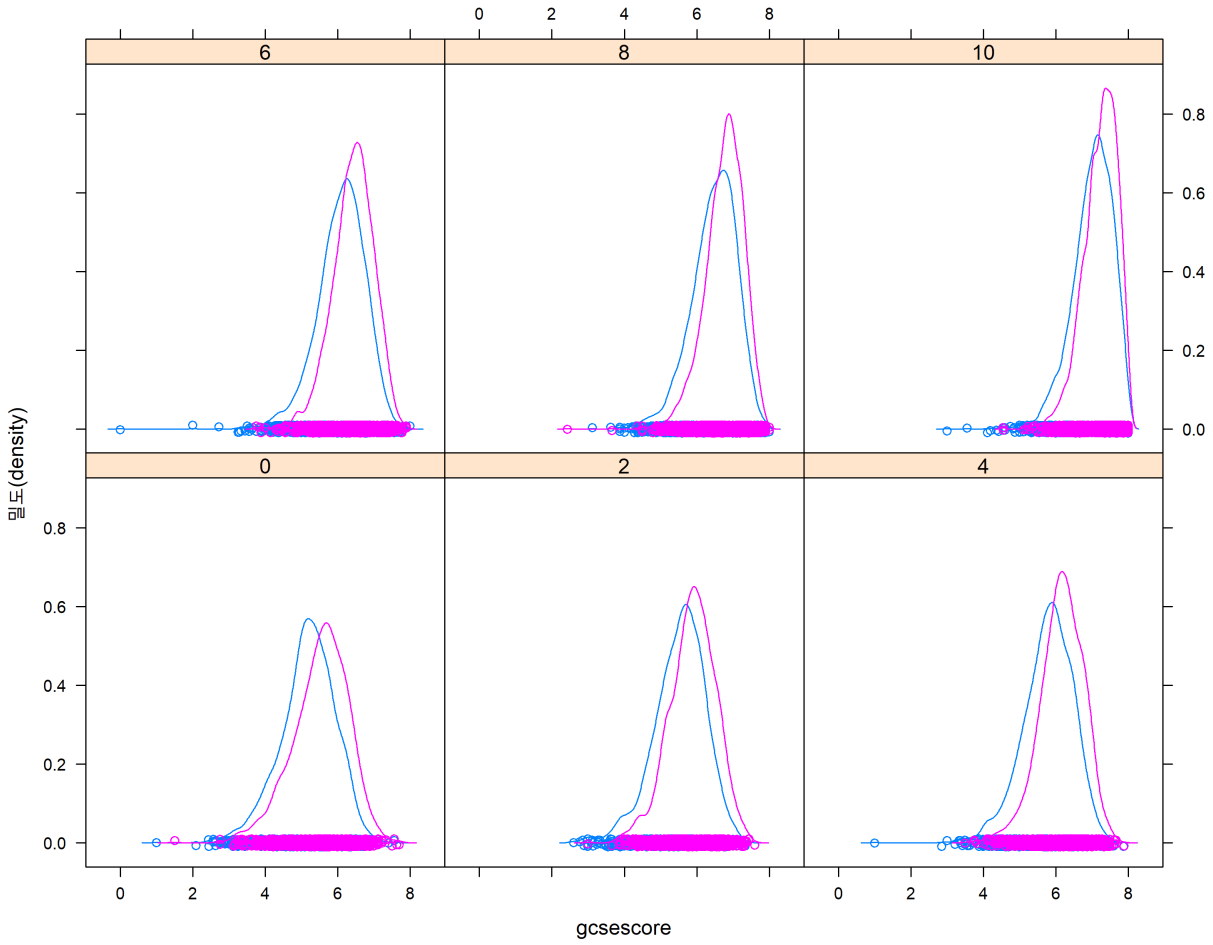
```
histogram(~gcsescore | factor(score), data = Chem97)
```



밀도 그래프

선을 그려서 값을 표현하는 형태로 R에서 제공하는 line과 유사한 기능입니다. 분포를 빠르게 이해하는데 표율적입니다. - plot.points는 밀도 점 표시 - auto.key는 범례 표시 여부

```
densityplot(~gcsescore | factor(score), data = Chem97, groups = gender, plot.Points = T, auto.ley = T)
```



막대 그래프

VADeaths 데이터를 사용하도록 하겠습니다.

```
data(VADeaths)
```

해당 데이터는 matrix로 구성되어 있습니다.

```
head(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
```

```
str(VADeaths)
```

```
##  num [1:5, 1:4] 11.7 18.1 26.9 41 66 8.7 11.7 20.3 30.9 54.3 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:5] "50-54" "55-59" "60-64" "65-69" ...
##   ..$ : chr [1:4] "Rural Male" "Rural Female" "Urban Male" "Urban Female"
```

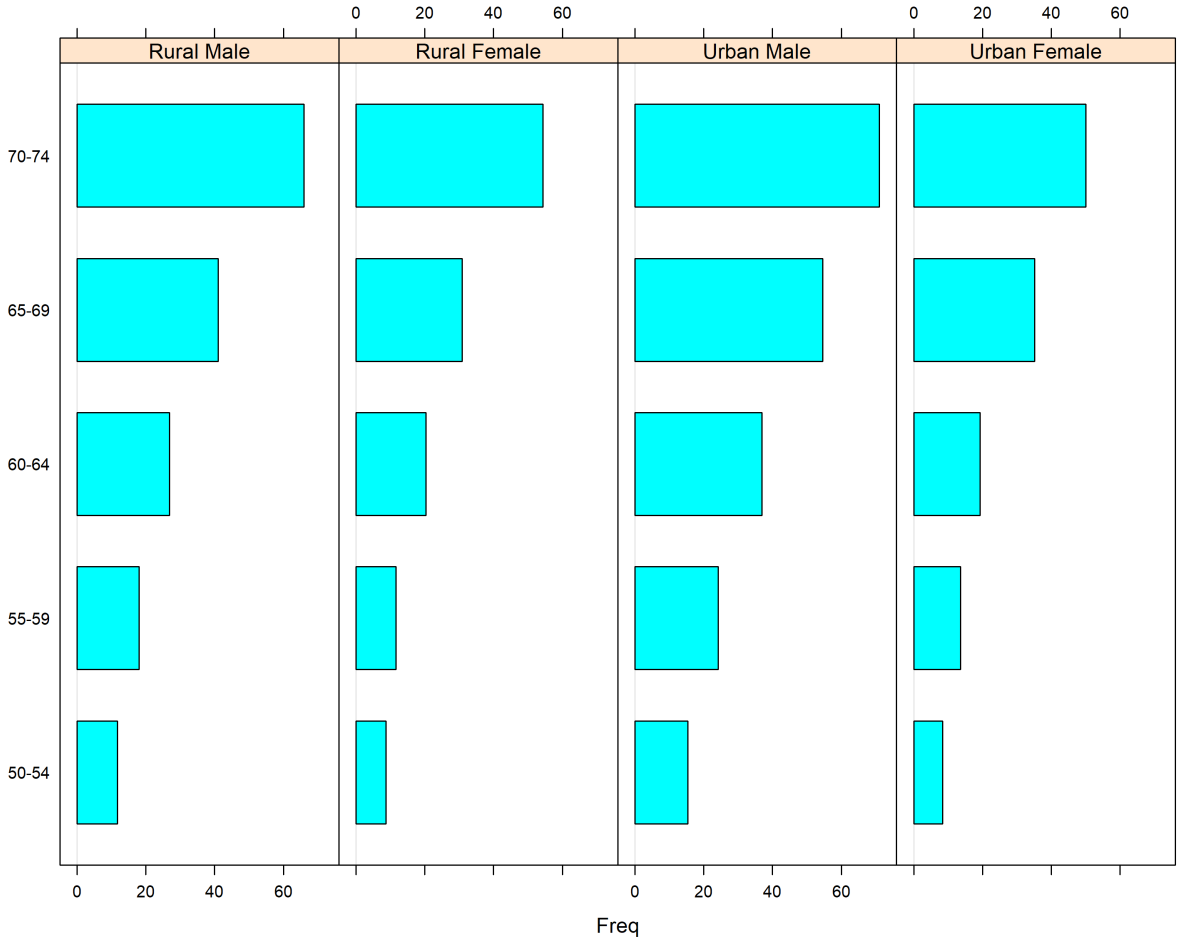
matrix 형식을 table 형식으로 변경합니다.

```
dft <- as.data.frame.table(VADeaths)
str(dft)
```

```
## 'data.frame':    20 obs. of  3 variables:
##  $ Var1: Factor w/ 5 levels "50-54","55-59",..: 1 2 3 4 5 1 2 3 4 5 ...
##  $ Var2: Factor w/ 4 levels "Rural Male","Rural Female",..: 1 1 1 1 1 2 2 2 2 2 ...
##  $ Freq: num  11.7 18.1 26.9 41 66 8.7 11.7 20.3 30.9 54.3 ...
```

막대 그래프를 그려줍니다. Var2를 기준으로 막대 그래프를 그려주며, c(4,1)는 4개를 1개의 행에 출력하라는 의미입니다. 4행 1열로 이해하지 않도록 주의를 해야 합니다.

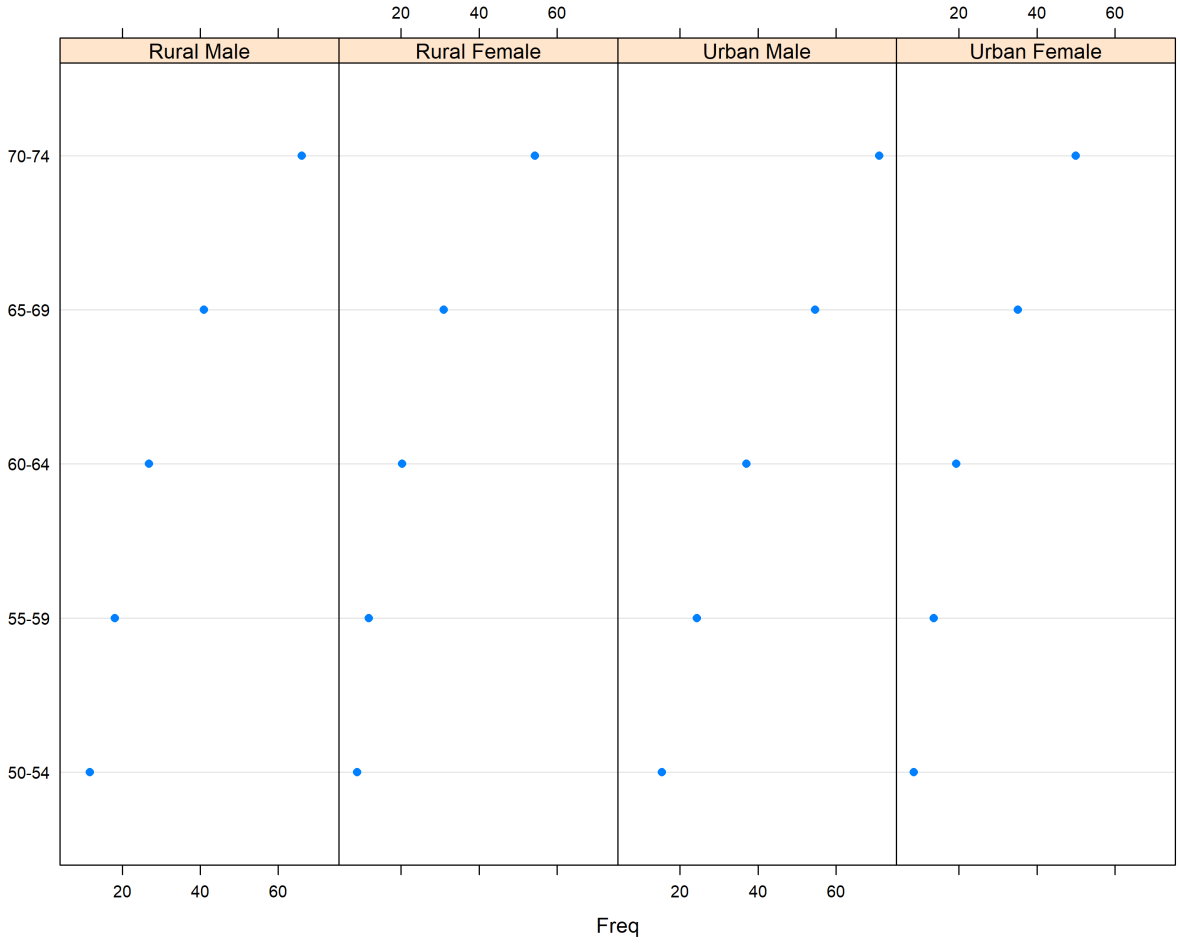
```
barchart(Var1 ~ Freq | Var2, data = dft, layout = c(4, 1), origin = 0 )
```



점 그래프

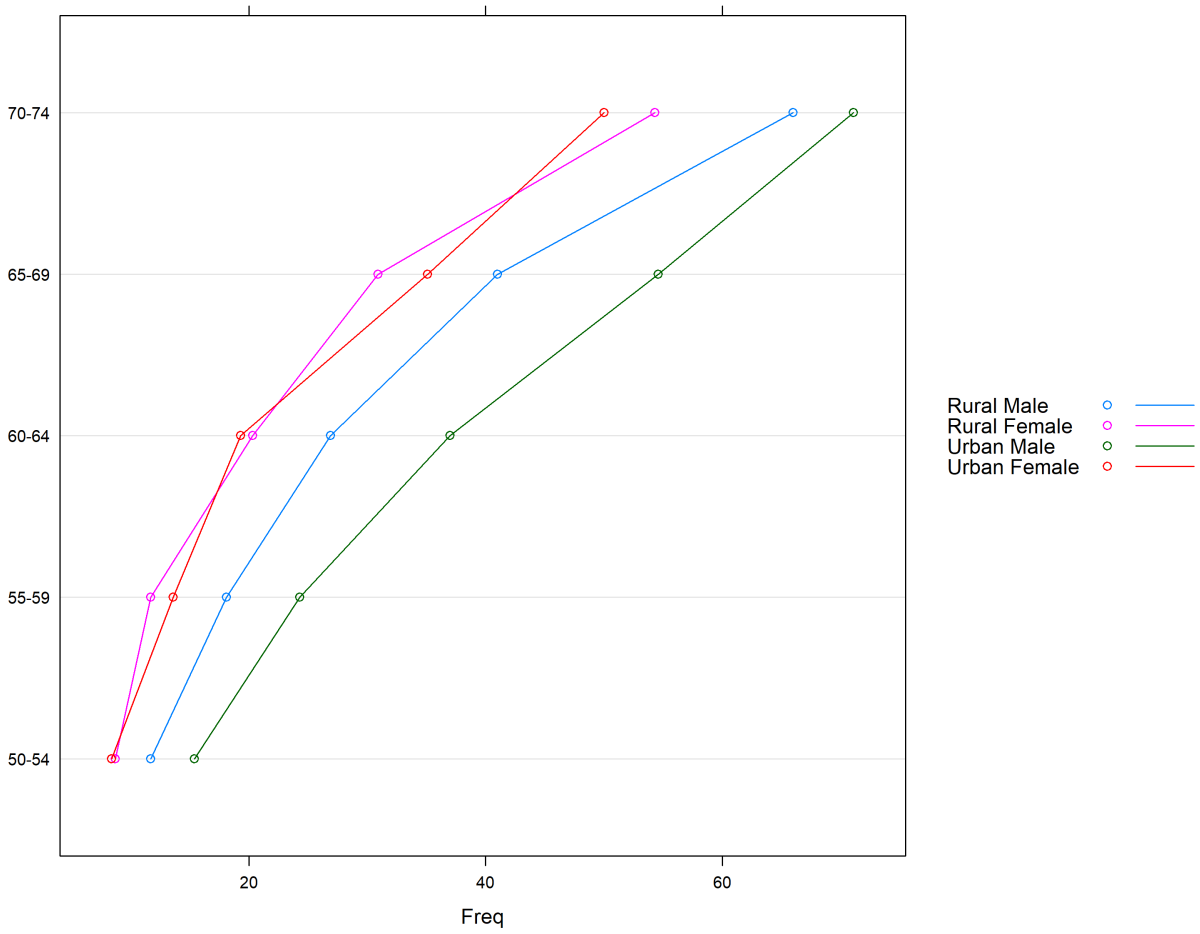
1점 그래는 아래와 같은 형태로 사용할 수 있습니다.

```
dotplot(Var1 ~ Freq | Var2, dft, layout = c(4,1))
```



Var2 변수 단위로 그룹화하여 점을 연결하는 것을 지원하며, 산점도 타입(type)과 범례(auto.key)를 추가하여 아래와 같은 형태로 작성할 수 있습니다.

```
dotplot(Var1 ~ Freq, data = dft, groups = Var2, type = "o", auto.key = list(space = "right", points = T
```



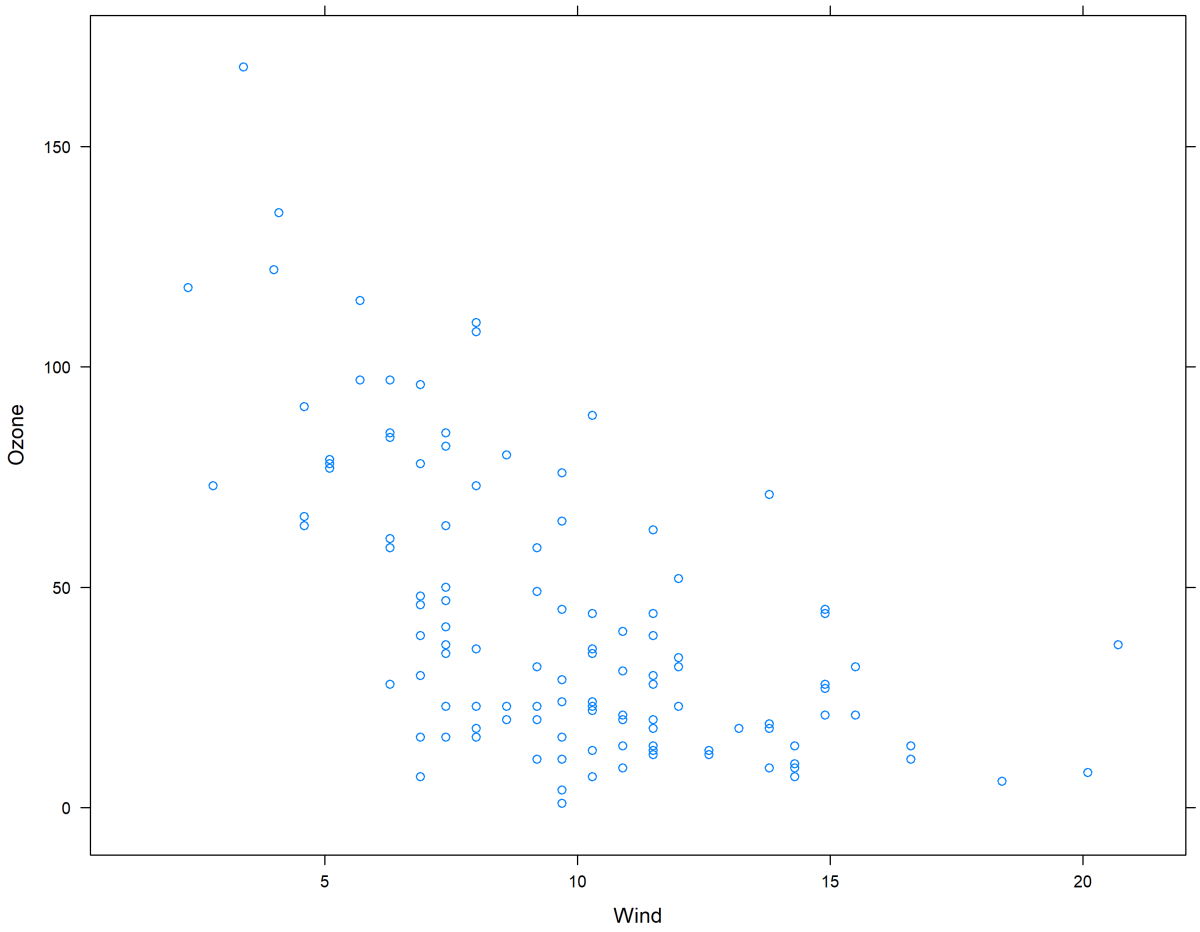
산점도 그래프

```
library(datasets)
str(airquality)
```

```
## 'data.frame':    153 obs. of  6 variables:
##  $ Ozone  : int  41 36 12 18 NA 28 23 19 8 NA ...
##  $ Solar.R: int  190 118 149 313 NA NA 299 99 19 194 ...
##  $ Wind   : num  7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
##  $ Temp   : int  67 72 74 62 56 66 65 59 61 69 ...
##  $ Month  : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ Day    : int  1 2 3 4 5 6 7 8 9 10 ...
```

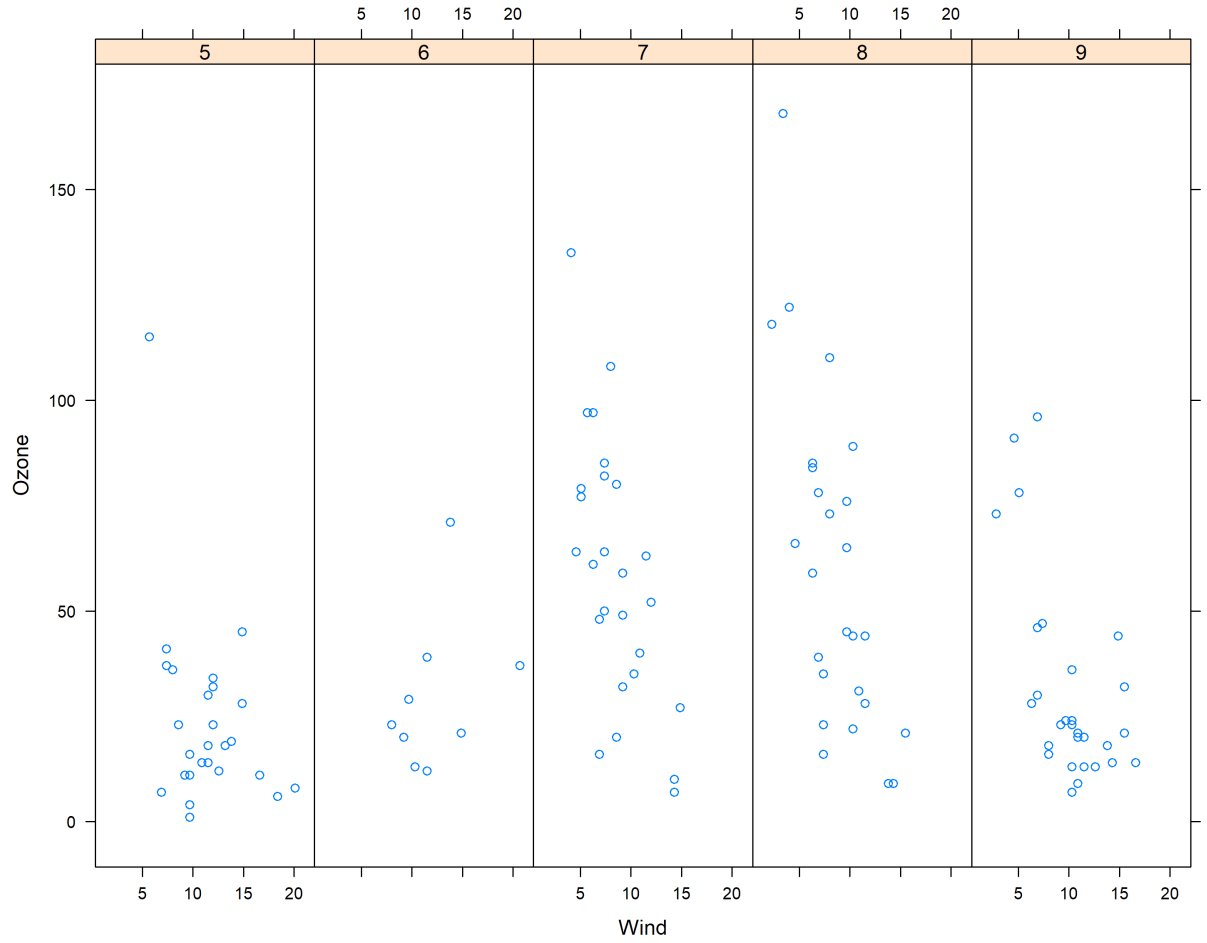
점 그래프와 마찬가지로, 간단한 산점도 그래프는 아래와 같은 형태로 작성할 수 있습니다.

```
xyplot(Ozone ~ Wind, data = airquality)
```



월별로 산점도 그래프를 그릴 수 있습니다.

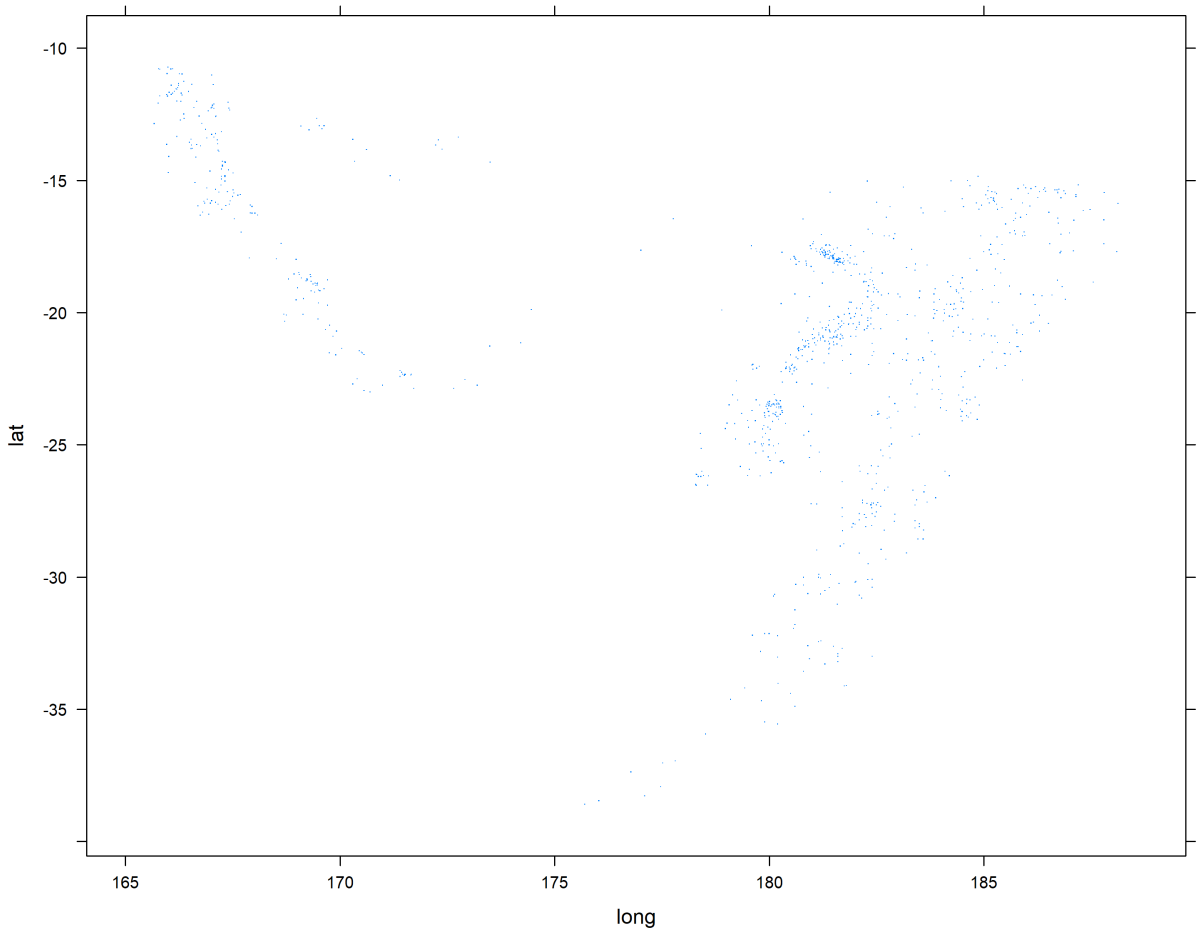
```
xyplot(Ozone ~ Wind | factor(Month), data = airquality, layout=c(5,1))
```



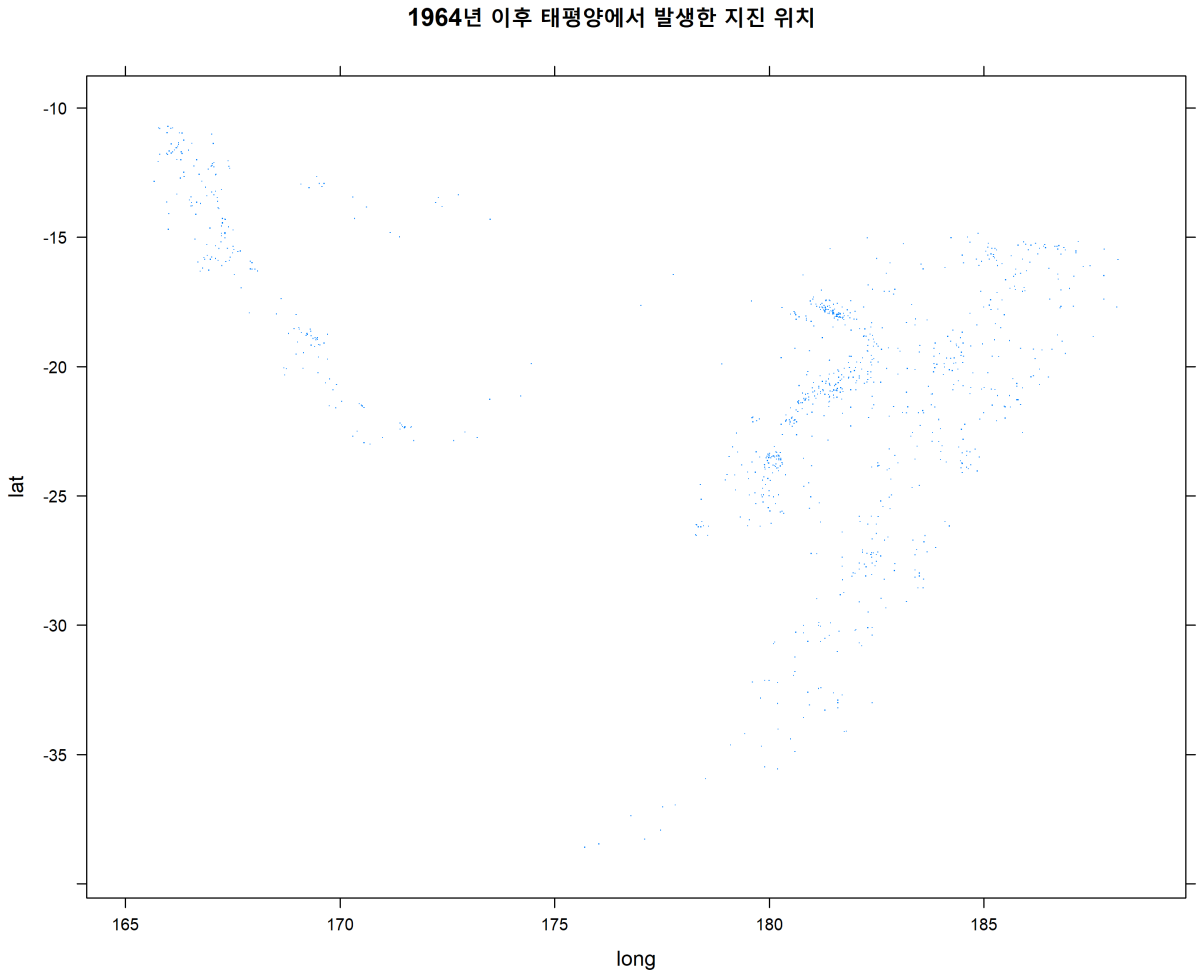

```
str(quakes)
```

```
## 'data.frame':    1000 obs. of  5 variables:
##  $ lat     : num  -20.4 -20.6 -26 -18 -20.4 ...
##  $ long    : num  182 181 184 182 182 ...
##  $ depth   : int  562 650 42 626 649 195 82 194 211 622 ...
##  $ mag     : num  4.8 4.2 5.4 4.1 4 4 4.8 4.4 4.7 4.3 ...
##  $ stations: int  41 15 43 19 11 12 43 15 35 19 ...
```

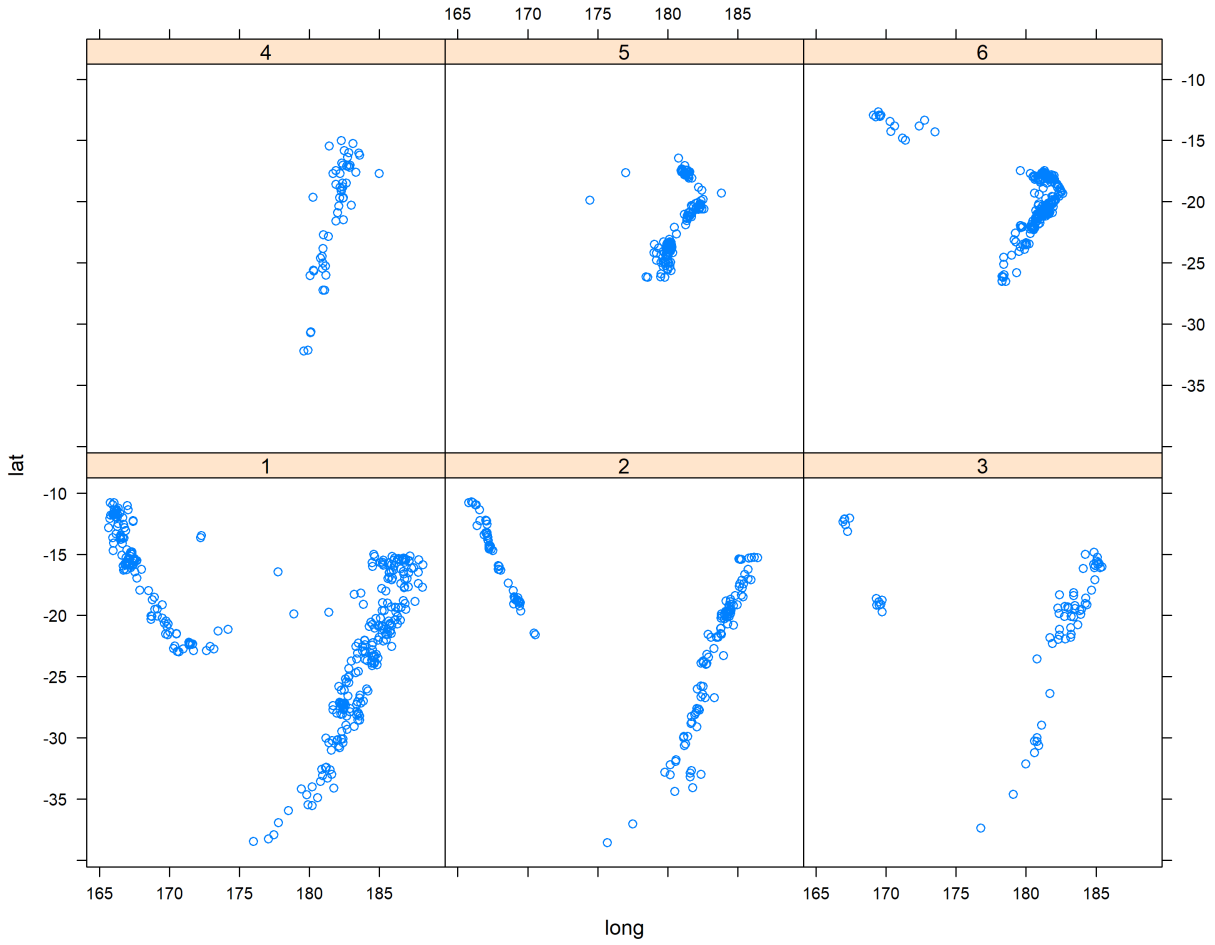
```
xyplot(lat ~ long, data = quakes, pch = ".")
```



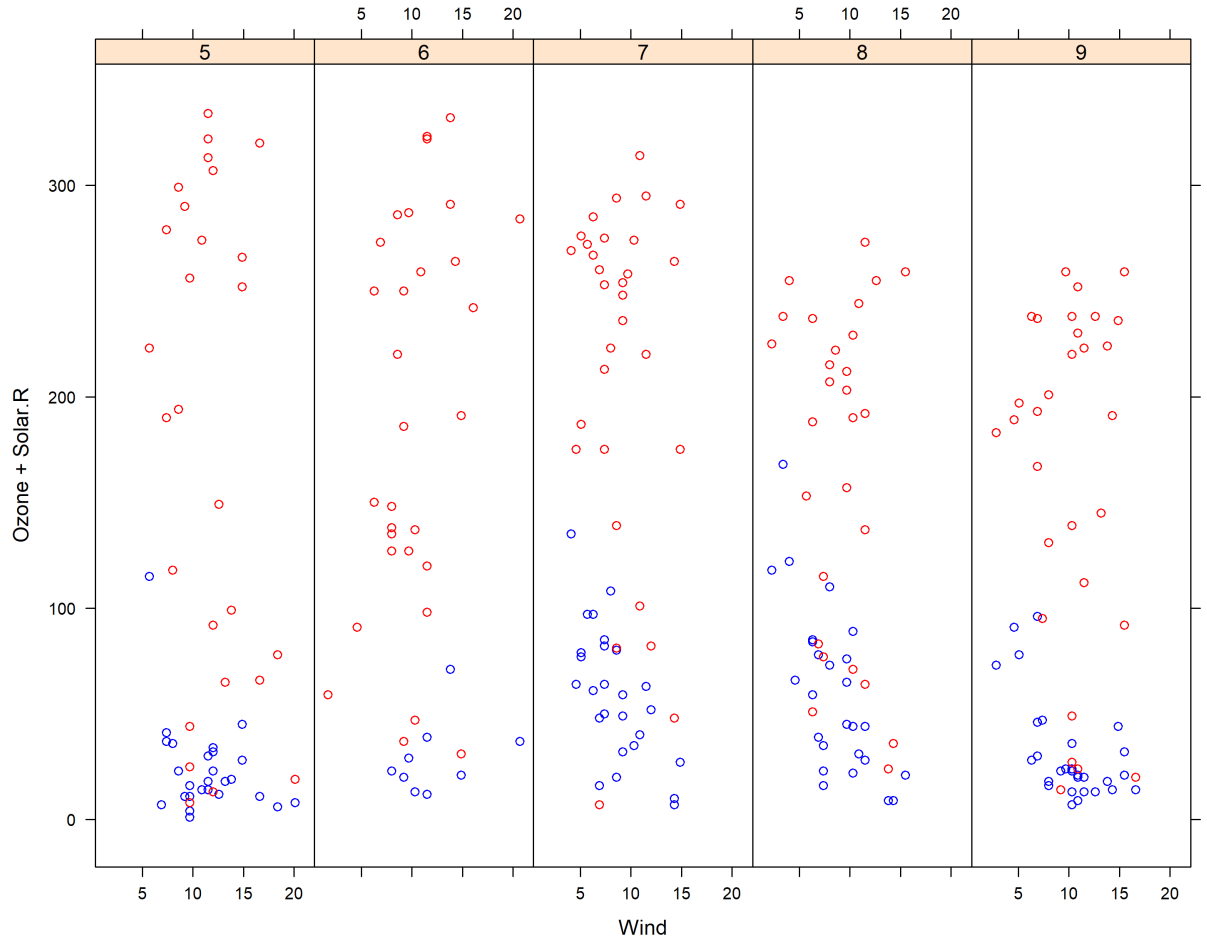

```
tplot <- xyplot(lat ~ long, data = quakes, pch = ".")
tplot <- update(tplot, main = "1964년 이후 태평양에서 발생한 지진 위치")
print(tplot)
```




```
quakes$depth2[quakes$depth >= 40 & quakes$depth <= 150] <- 1
quakes$depth2[quakes$depth >= 151 & quakes$depth <= 250] <- 2
quakes$depth2[quakes$depth >= 251 & quakes$depth <= 350] <- 3
quakes$depth2[quakes$depth >= 351 & quakes$depth <= 450] <- 4
quakes$depth2[quakes$depth >= 451 & quakes$depth <= 550] <- 5
quakes$depth2[quakes$depth >= 551 & quakes$depth <= 680] <- 6
convert <- transform(quakes, depth2 = factor(depth2))
xyplot(lat ~ long | depth2, data = convert)
```



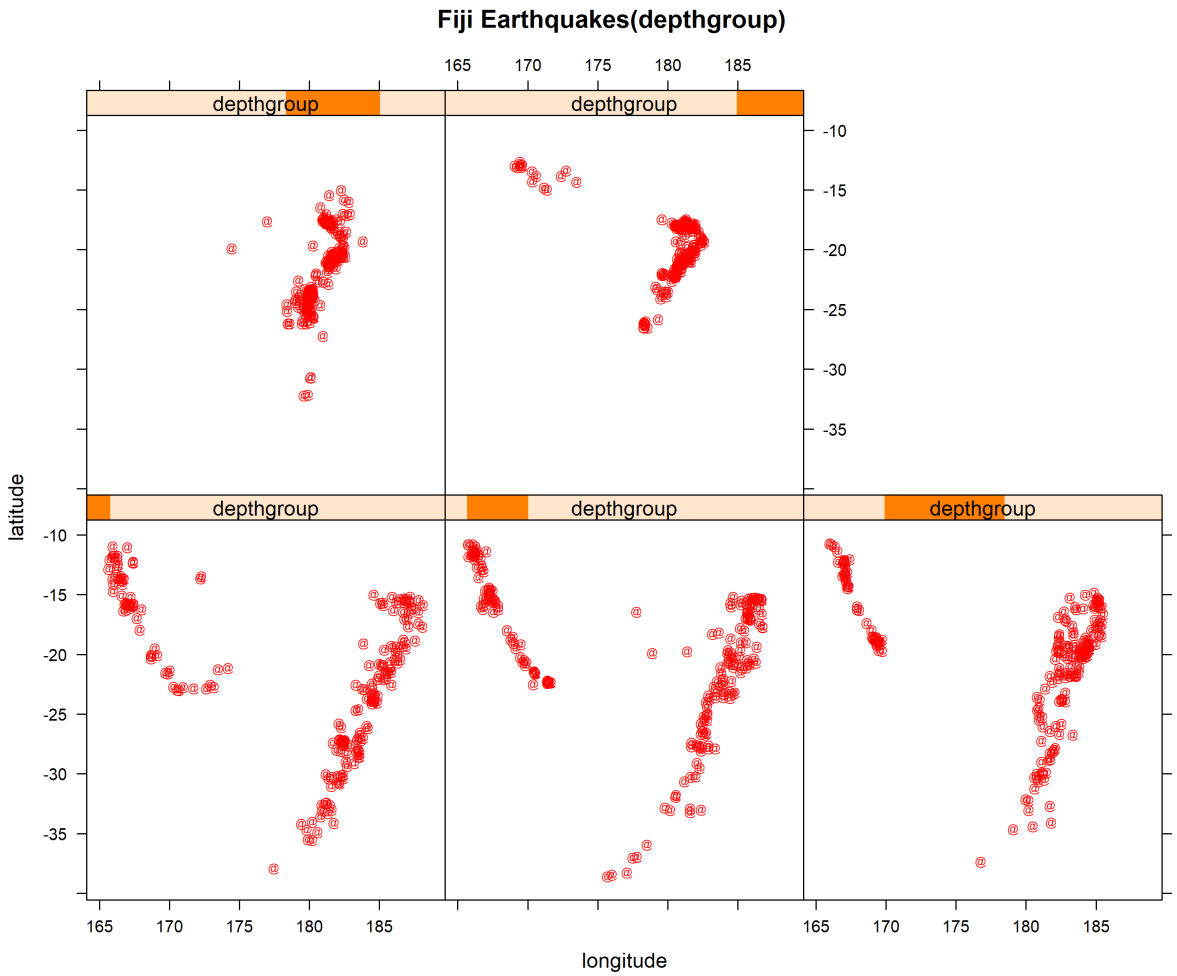

```
xyplot(Ozone + Solar.R ~ Wind | factor(Month), data = airquality, col = c("blue", "red"), layout = c(5,
```



데이터 범주화

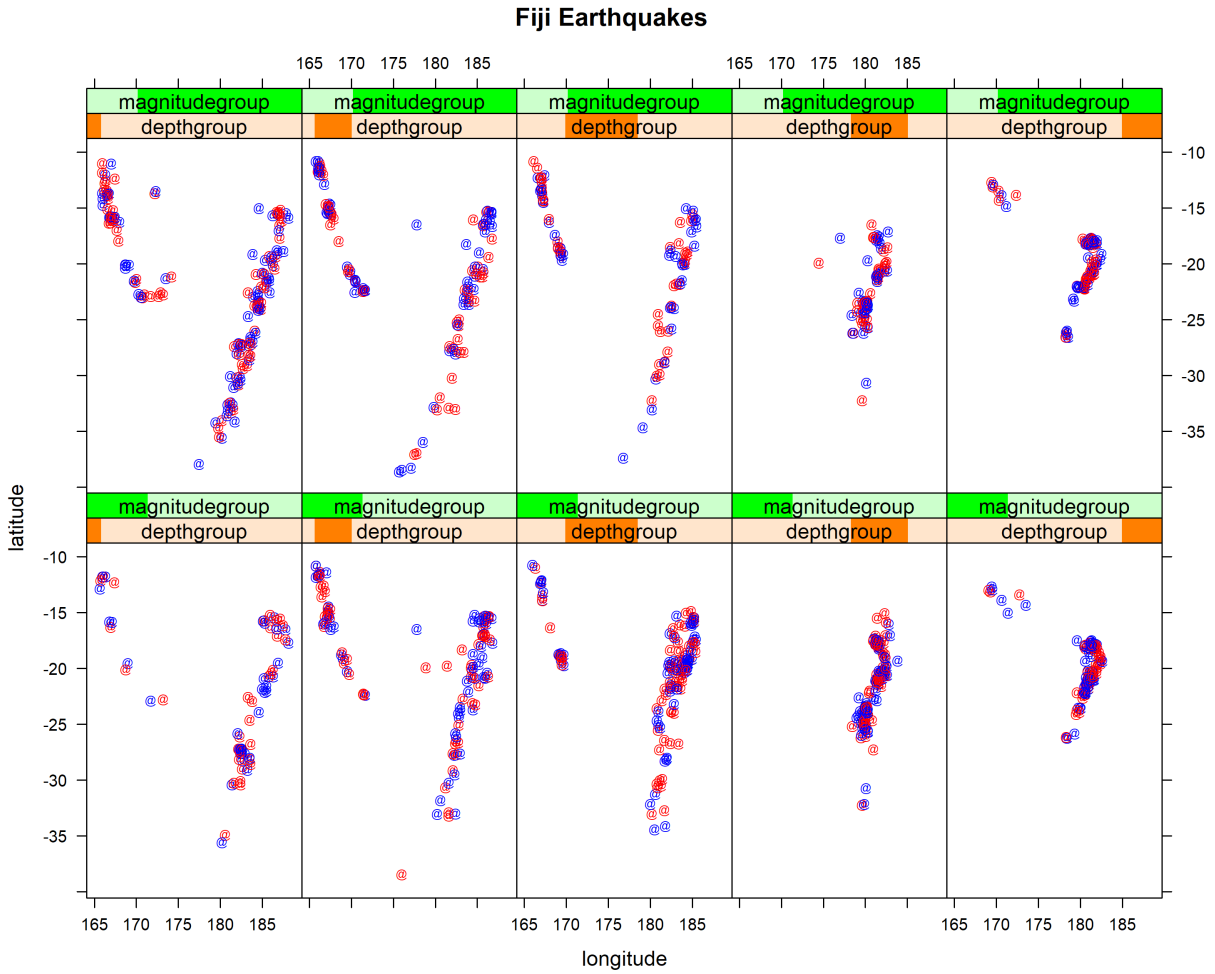
equal.count를 사용하면 데이터를 범주화 할 수 있습니다. numgroup은 1 ~ 150을 대상으로 겹치지 않게 4개 영역으로 범주화하는 방식을 보여주는 것으로 depthgroup은 지진 데이터를 5개의 영역으러 범주화 합니다.

```
numgroup <- equal.count(1:150, number = 4, overlap = 0)
depthgroup <- equal.count(quakes$depth, number = 5, overlap = 0)
xyplot(lat ~ long | depthgroup, data = quakes, main = "Fiji Earthquakes(depthgroup)", ylab = "latitude"
```



수심과 리히터 규모를 동시에 표현하는 방법(depthgroup * magnitudegroup)은 아래와 같습니다.

```
magnitudegroup <- equal.count(quakes$mag, number = 2, overlap =0)
xyplot(lat ~ long | depthgroup * magnitudegroup, data = quakes, main = "Fiji Earthquakes", ylab = "lati
```



ggplot2 그래프

ggplot2는 그래프를 만들 때 사용하는 패키지로 'layer' 구조로 되어 있습니다. (layer 구조 - 기본 + 옵션1 + 옵션2) 방식으로 쌓아올리는 형식을 사용합니다. 일반적으로 간단하게 시각화 하고 싶을 때 사용합니다. - 기본(x,y축 설정) + 옵션1(그래프 유형선택 - 점, 선, 막대) + 옵션2 (색상, 표식 등등) - 기하학적 객체들(점,선,막대등)에 미적특성(색상, 모양,크기)을 설정하여 플로팅 - 그래픽 생성 기능과 통계 변환을 포함 - ggplot2의 기본함수 qplot()-aesthetics(크기,모양,색상)과 geoms(점,선등) 으로 구성

```
library(ggplot2)
str(mpg)
```

```
## tibble [234 x 11] (S3: tbl_df/tbl/data.frame)
##  $ manufacturer: chr [1:234] "audi" "audi" "audi" "audi" ...
##  $ model       : chr [1:234] "a4" "a4" "a4" "a4" ...
##  $ displ       : num [1:234] 1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
##  $ year        : int [1:234] 1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
##  $ cyl         : int [1:234] 4 4 4 4 6 6 6 4 4 4 ...
##  $ trans       : chr [1:234] "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
##  $ drv         : chr [1:234] "f" "f" "f" "f" ...
##  $ cty         : int [1:234] 18 21 20 21 16 18 18 18 16 20 ...
```

```
##  $ hwy          : int [1:234] 29 29 31 30 26 26 27 26 25 28 ...
##  $ fl           : chr [1:234] "p" "p" "p" "p" ...
##  $ class        : chr [1:234] "compact" "compact" "compact" "compact" ...
```

```
summary(mpg)
```

```
##  manufacturer          model               displ           year
##  Length:234         Length:234         Min.   :1.600   Min.   :1999
##  Class :character   Class :character   1st Qu.:2.400   1st Qu.:1999
##  Mode  :character   Mode  :character   Median :3.300   Median :2004
##                                        Mean   :3.472   Mean   :2004
##                                        3rd Qu.:4.600   3rd Qu.:2008
##                                        Max.   :7.000   Max.   :2008
##       cyl           trans               drv                 cty
##  Min.   :4.000   Length:234         Length:234         Min.   : 9.00
##  1st Qu.:4.000   Class :character   Class :character   1st Qu.:14.00
##  Median :6.000   Mode  :character   Mode  :character   Median :17.00
##  Mean   :5.889                                         Mean   :16.86
##  3rd Qu.:8.000                                         3rd Qu.:19.00
##  Max.   :8.000                                         Max.   :35.00
##       hwy             fl                class
##  Min.   :12.00   Length:234         Length:234
##  1st Qu.:18.00   Class :character   Class :character
##  Median :24.00   Mode  :character   Mode  :character
##  Mean   :23.44
##  3rd Qu.:27.00
##  Max.   :44.00
```

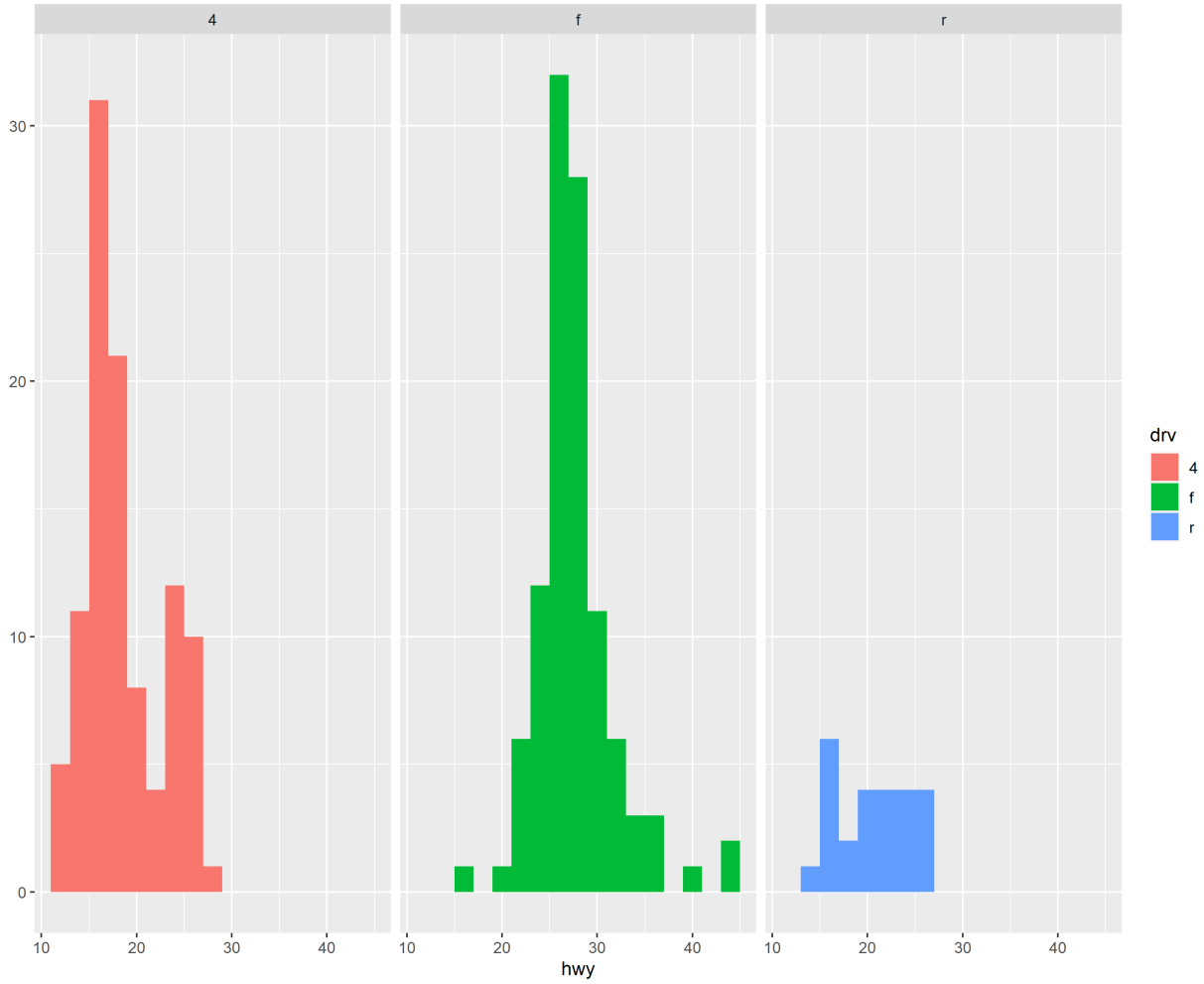
```
table(mpg$drv)
```

```
##
##   4   f   r
## 103 106  25
```

막대 그래프

fill은 막대 그래프 상에서 색상으로 구별하여 시각화하는 것으로 변수에 대한 특징이 가시적으로 확인 가능합니다. binwidth는 막대의 폭 크기를 지정합니다. facets는 열 단위로 패널을 생성할 수 있습니다. 행단위로 보고 싶으시면 facets = drv~.로 구성하시면 됩니다.

```
qplot(hwy, data = mpg, fill = drv, binwidth = 2, facets = .~drv)
```

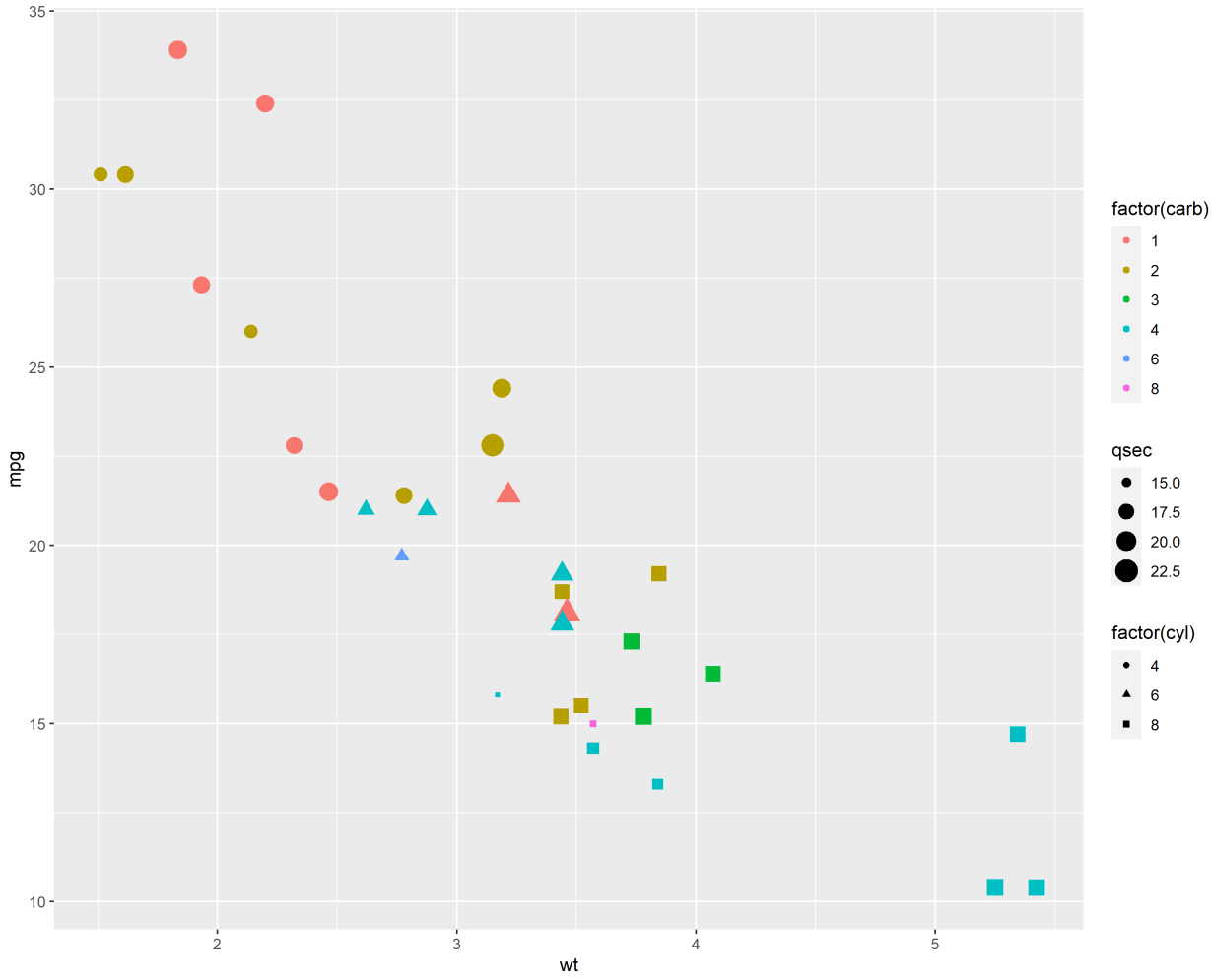


산점도 그래프

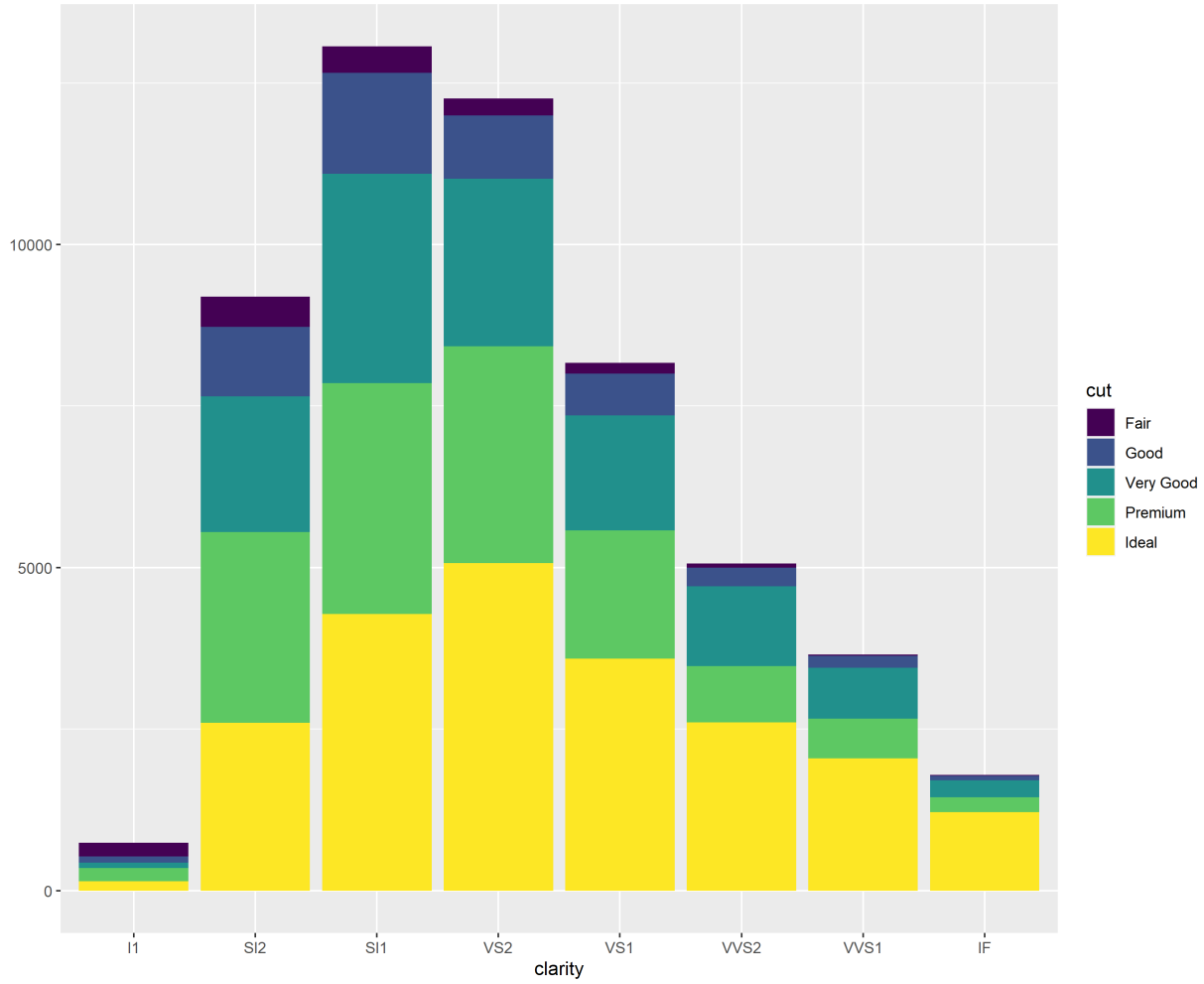
mpg 데이터 셋의 displ 과 hwy 변수 이용하여 산점도를 그려줍니다.

```
qplot(displ, hwy, data=mpg, color = drv, facets = .~drv)
```

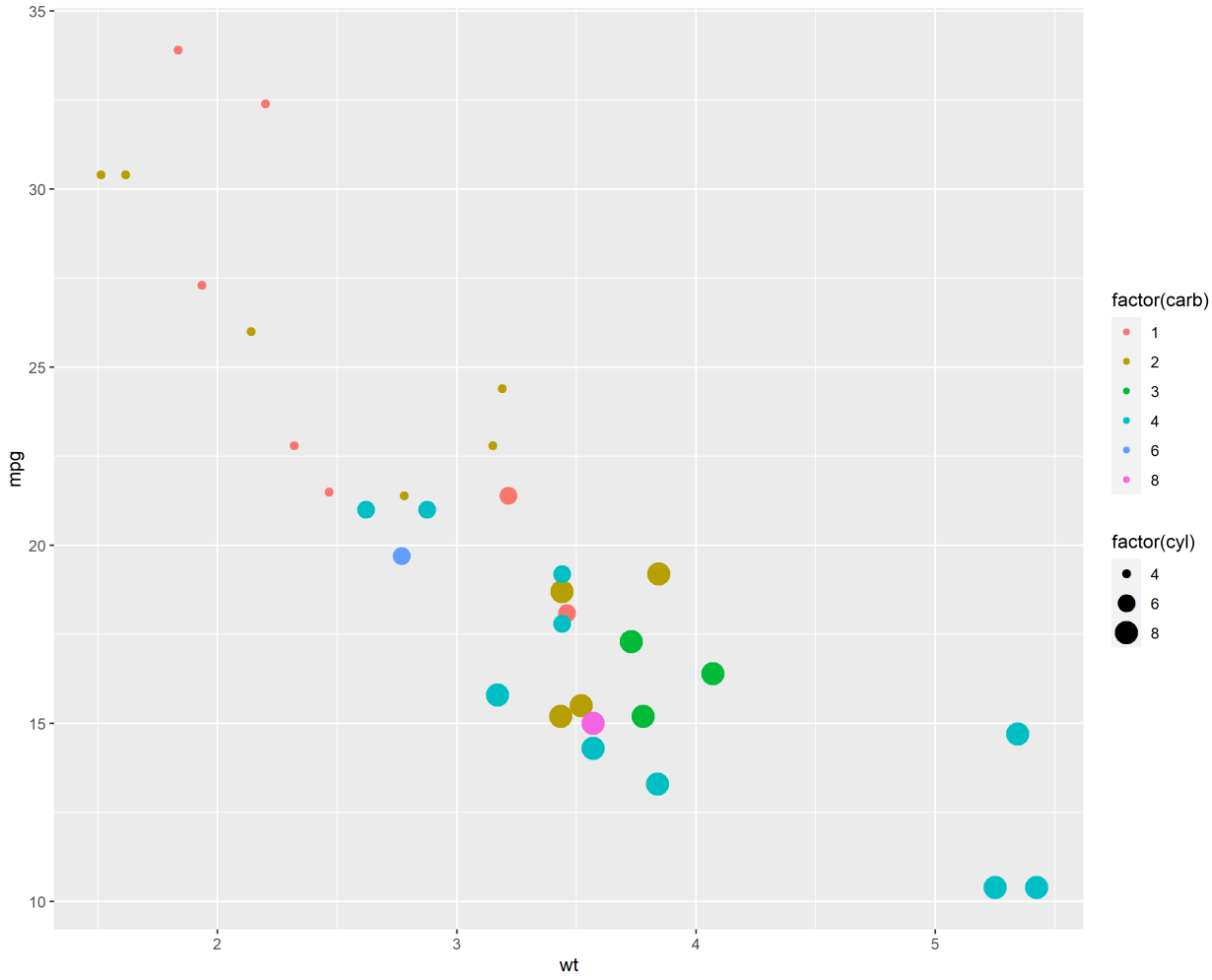

```
qplot(wt,mpg, data=mtcars, color=factor(carb), size=qsec, shape=factor(cyl))
```




```
qplot(clarity, data=diamonds, fill=cut, geom = "bar") # 레이아웃에 색 채우기
```




```
qplot(wt,mpg, data=mtcars, color=factor(carb), size=factor(cyl), geom = "point")
```


35
30
25
20
15
10
mpg
2
3
4
5
wt
factor(carb)
1
2
3
4
6
8
factor(cyl)
4
6
8