1. ggplot2 시작하기

ggplot을 그리는 2+3 단계

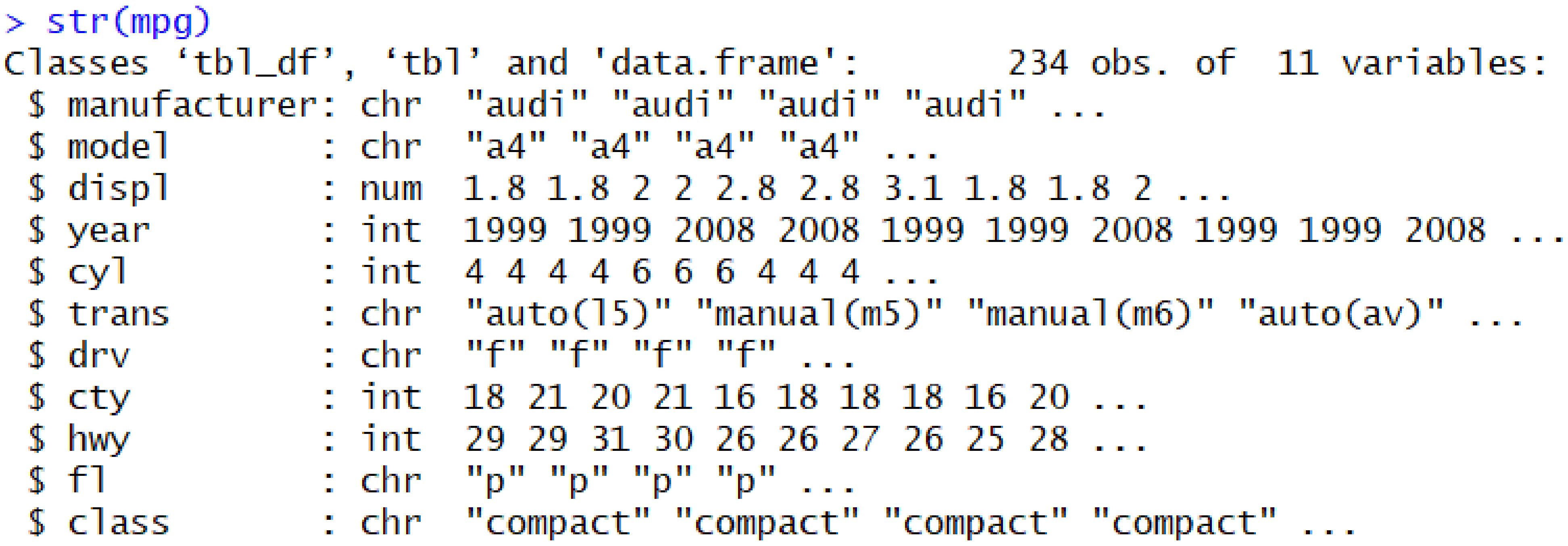
1. 평면 세팅	ggplot(diamonds, aes(x = , y =))
2. 도형선택	+ geom_point( )
3. 라벨	+ labs(title=“ ", x=“ ", y=“ ")
4. 테마	+ theme( )
5. 패싯	+ facet_wrap(~ cut, ncol = 3)

Source : https://blog.naver.com/definitice/221127682474

mpg 데이터셋 보기

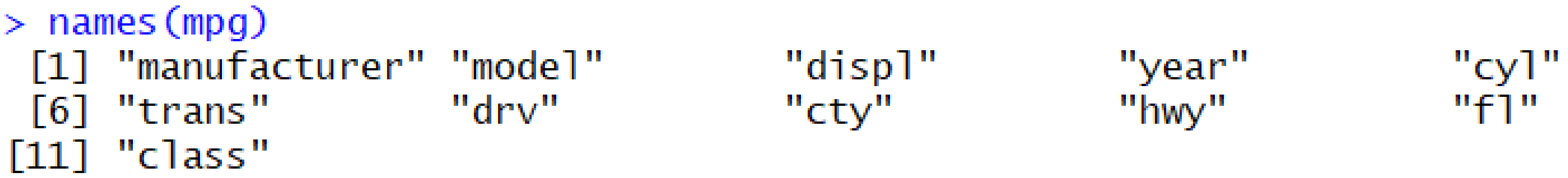
- ggplot2 패키지 설치 후 사용하는 부속패키지. 가장 많이 인용됨

- 미국 환경보호국 조사, 1999~2008 자동차 모델,제조사, 연료, 거리, 연비



```
> str(mpg)
Classes 'tbl_df', 'tbl' and 'data.frame':       234 obs. of  11 variables:
 $ manufacturer: chr  "audi" "audi" "audi" "audi" ...
 $ model       : chr  "a4" "a4" "a4" "a4" ...
 $ displ       : num  1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
 $ year        : int  1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
 $ cyl         : int  4 4 4 4 6 6 6 4 4 4 ...
 $ trans       : chr  "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
 $ drv         : chr  "f" "f" "f" "f" ...
 $ cty         : int  18 21 20 21 16 18 18 18 16 20 ...
 $ hwy         : int  29 29 31 30 26 26 27 26 25 28 ...
 $ fl          : chr  "p" "p" "p" "p" ...
 $ class       : chr  "compact" "compact" "compact" "compact" ...
```

mpg 데이터셋 보기



```
> names(mpg)
 [1] "manufacturer" "model"        "displ"        "year"         "cyl"
 [6] "trans"        "drv"          "cty"          "hwy"          "fl"
[11] "class"
```

- cty and hwy :   miles per gallon (mpg) for city and highway driving
- displ : engine displacement in litres. (배기량)
- drv: the drive train - front wheel (f), rear wheel (r) or four wheel (4).
- class : the “type” of car, two seater, SUV, compact, etc
- trans : type of transmission

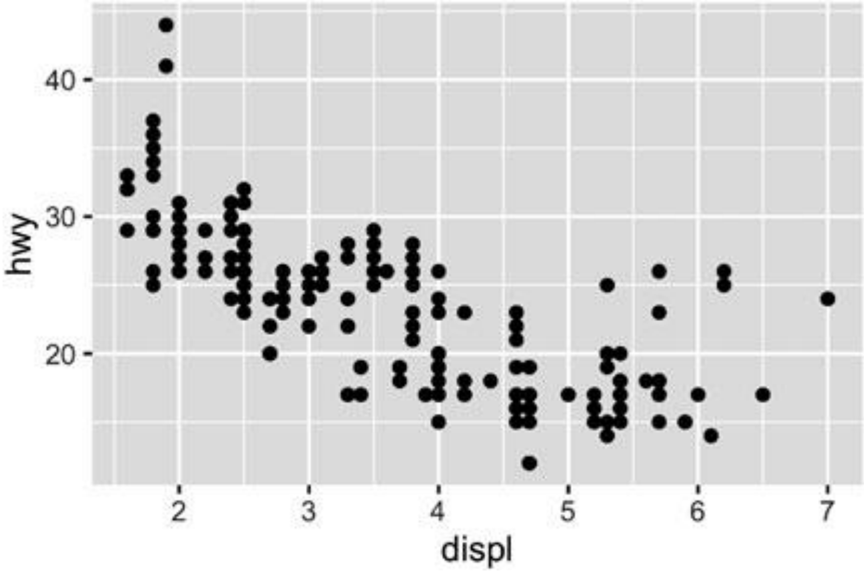
ggplot2 도움말

- https://ggplot2.tidyverse.org/reference/
- https://www.rstudio.com/resources/cheatsheets/
- https://www.rdocumentation.org/
- 그리고 google

다음 질문에 생각해 볼 수 있다

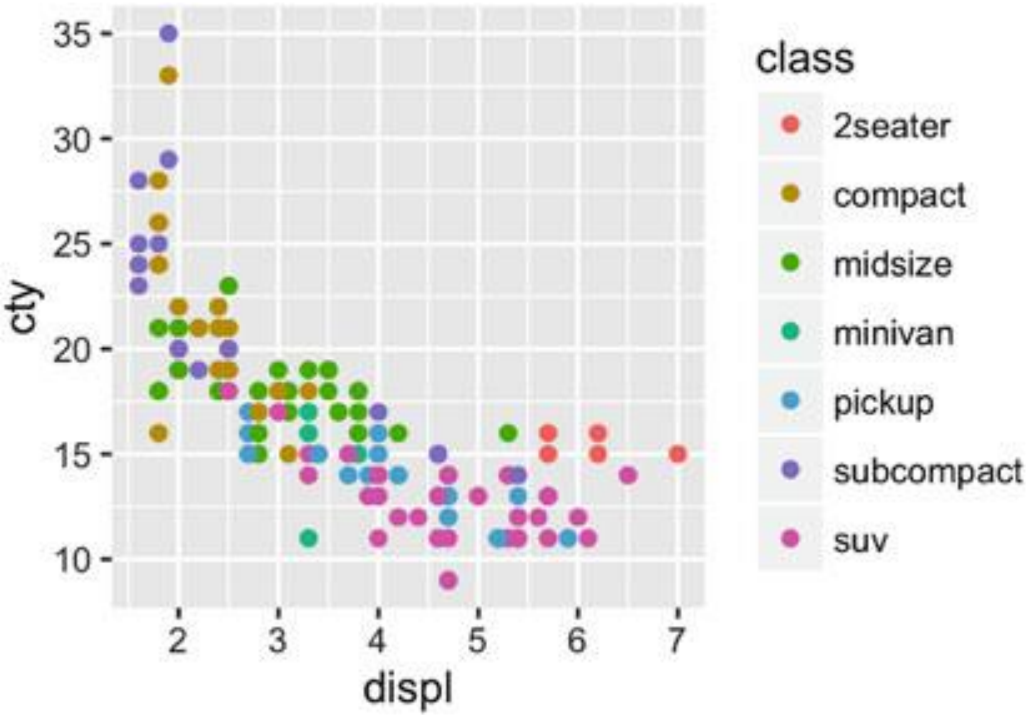
- 엔진 크기와 연비의 관계는 ?
- 어느 제조회사가 다른 회사보다 연비에 관심을 많이 기울이고 있을까?
- 지난 10년간 연비는 과연 향상되었을까?

```
ggplot(mpg, aes(x = displ, y = hwy)) +
  geom_point()
```



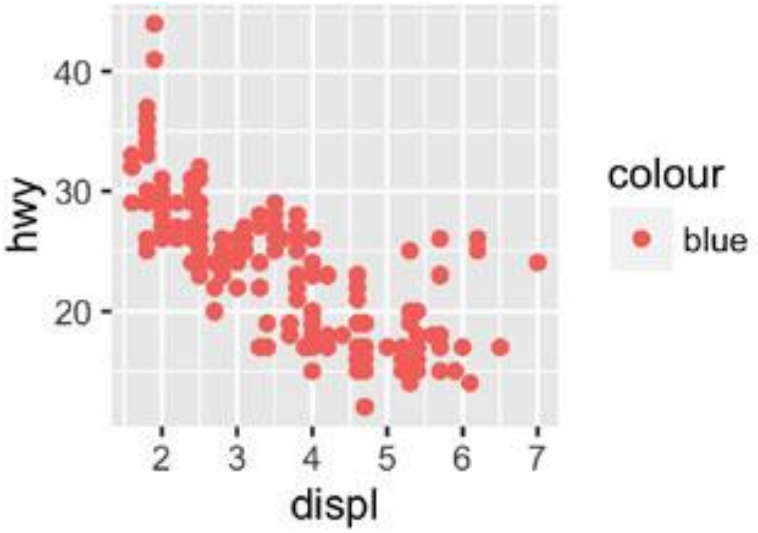
Colour

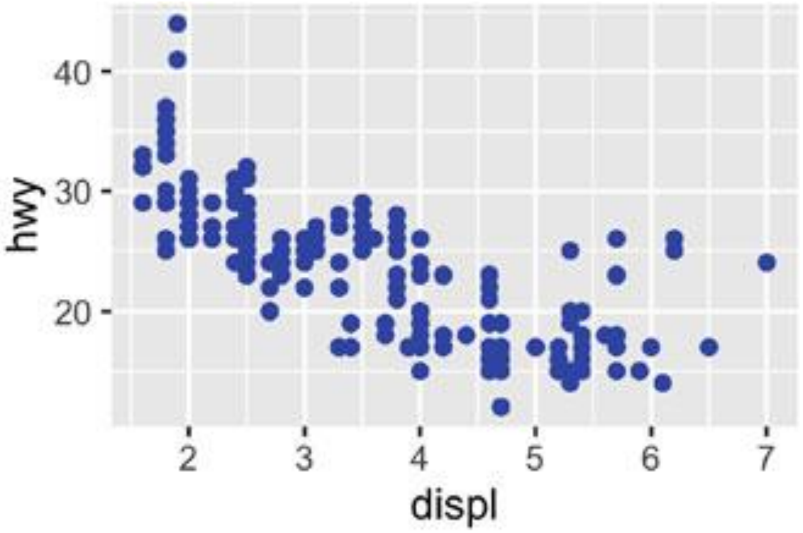
```
ggplot(mpg, aes(displ, cty, colour = class)) +
  geom_point()
```



Colour

```
ggplot(mpg, aes(displ, hwy)) + geom_point(aes(colour = "blue"))
ggplot(mpg, aes(displ, hwy)) + geom_point(colour = "blue")
```

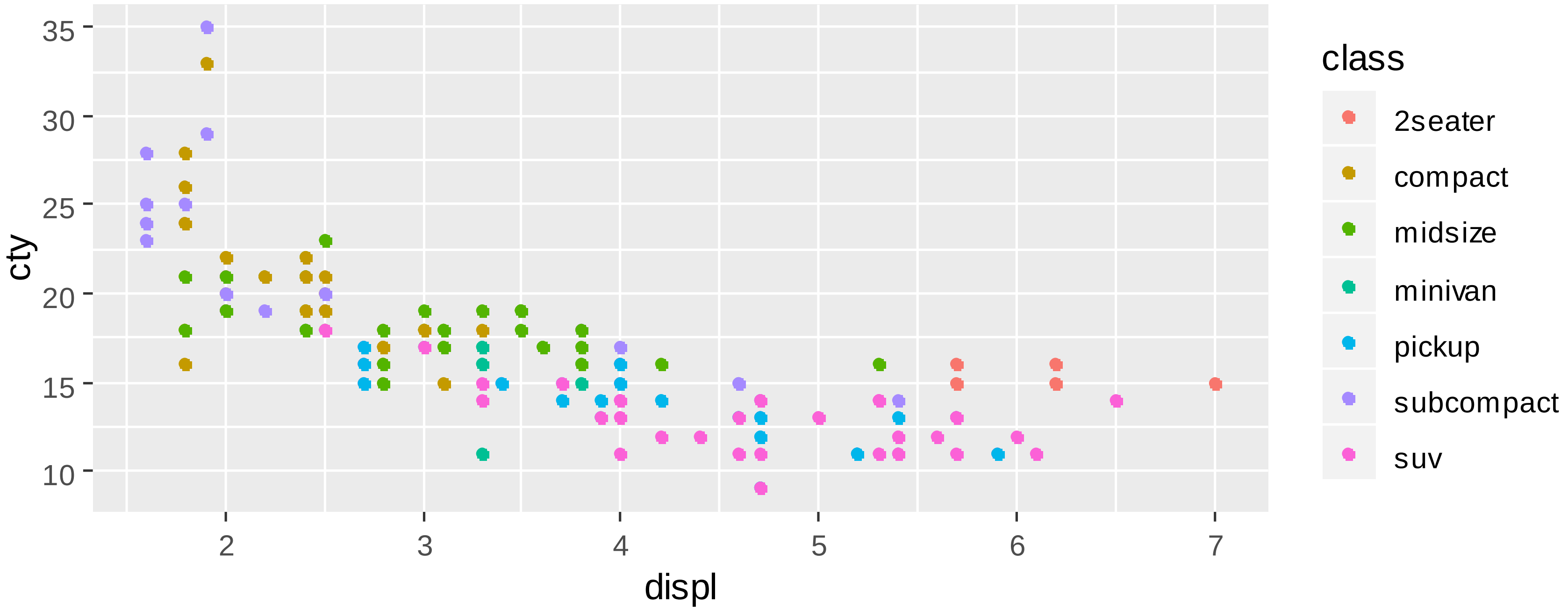




Colour와 연속형 변수

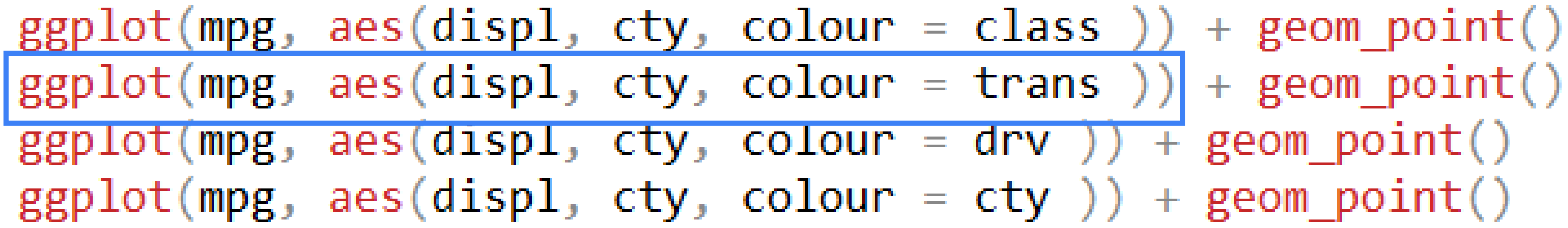
1. 위 그림에서 colour 요소에 다른 변수들을 넣어 봅시다

```
ggplot(mpg, aes(displ, cty, colour = class )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = trans )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = drv )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = cty )) + geom_point()
```

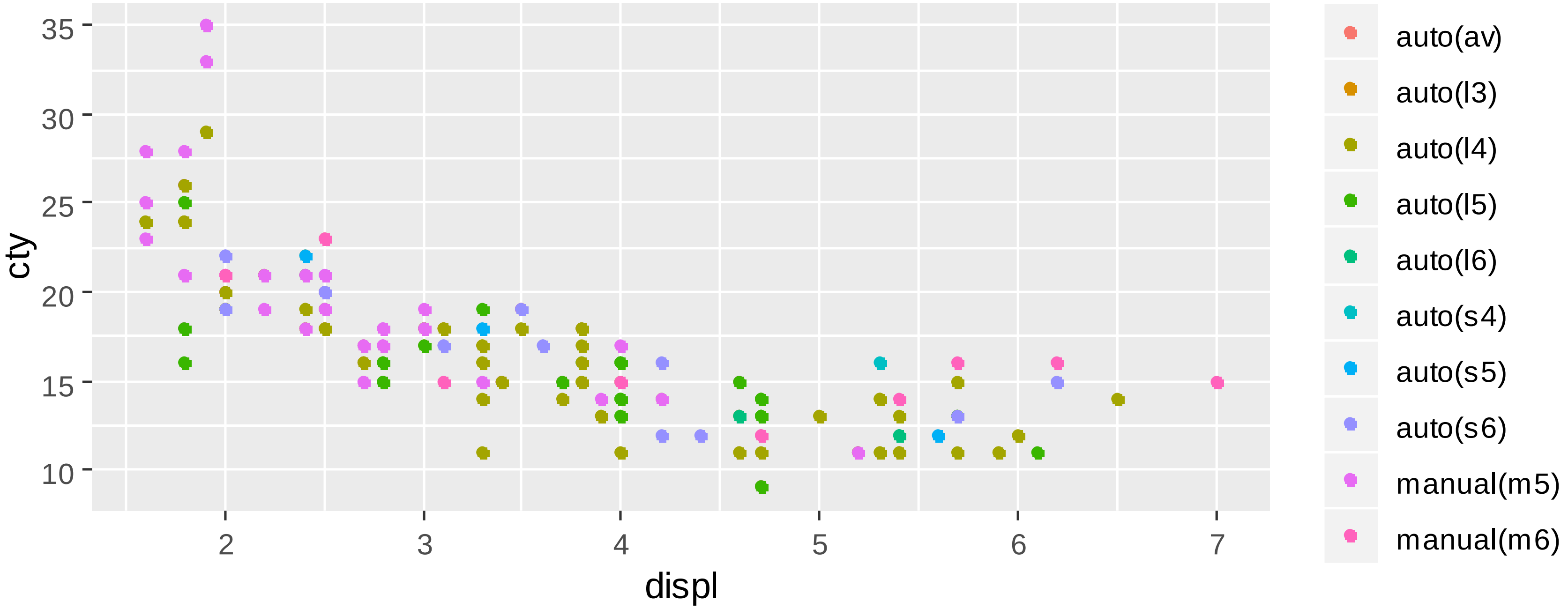


Colour와 연속형 변수

1. 위 그림에서 colour 요소에 다른 변수들을 넣어 봅시다

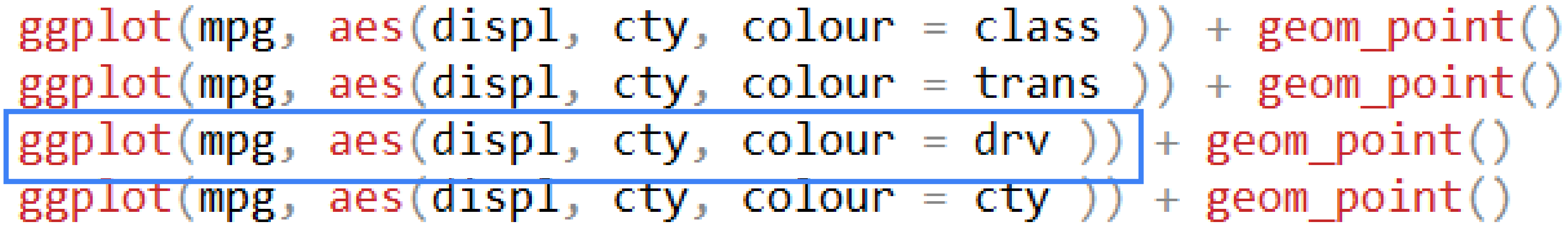


```
ggplot(mpg, aes(displ, cty, colour = class )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = trans )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = drv )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = cty )) + geom_point()
```

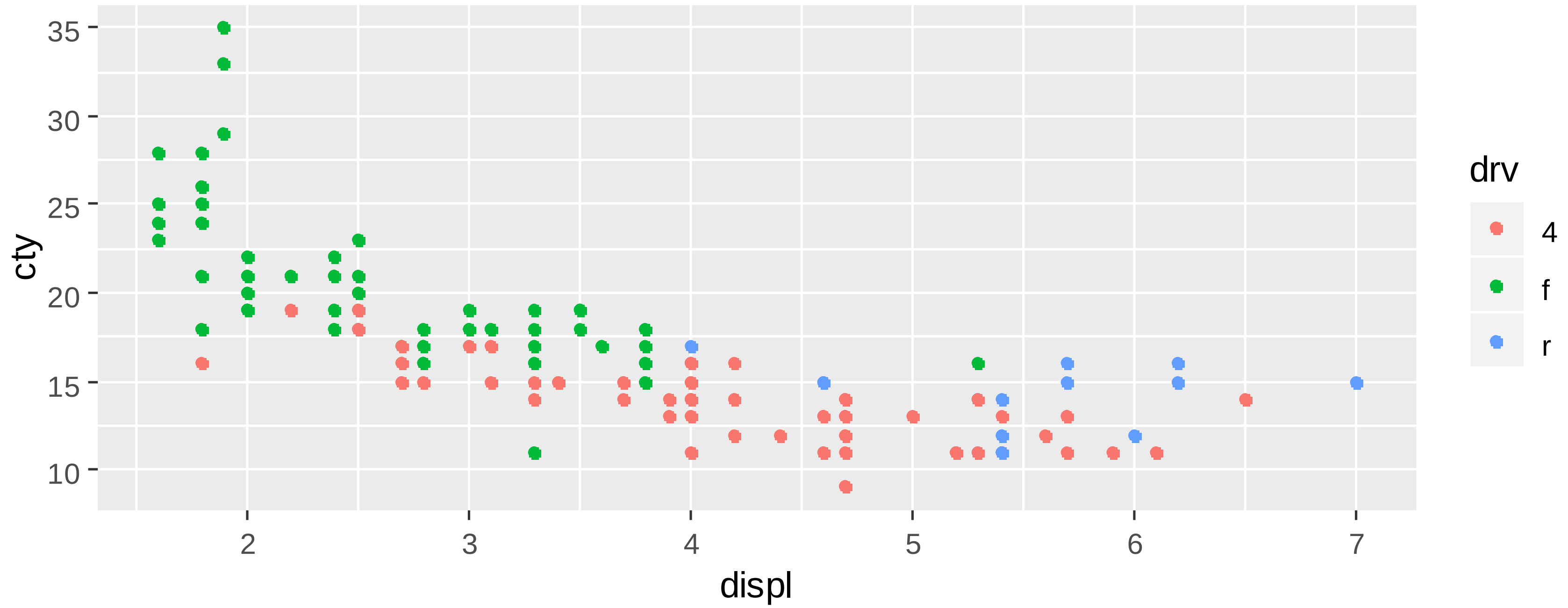


Colour와 연속형 변수

1. 위 그림에서 colour 요소에 다른 변수들을 넣어 봅시다



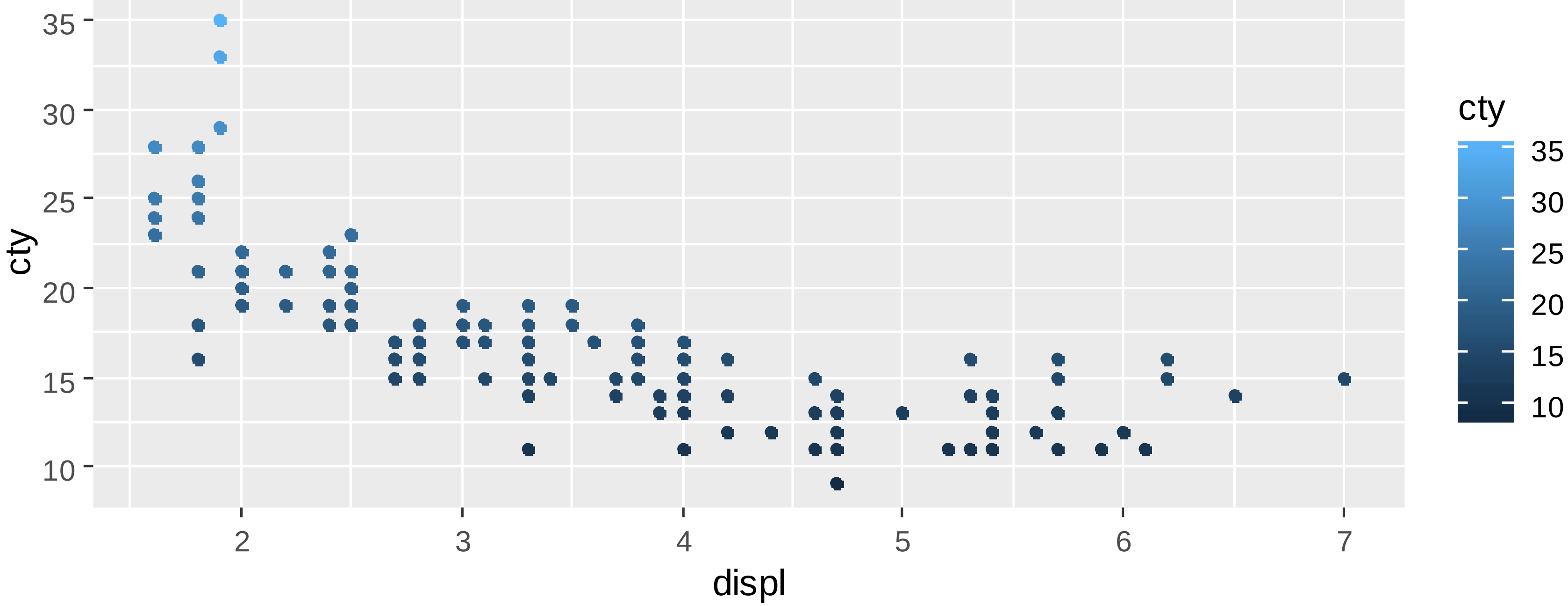
```
ggplot(mpg, aes(displ, cty, colour = class )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = trans )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = drv )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = cty )) + geom_point()
```



Colour와 연속형 변수

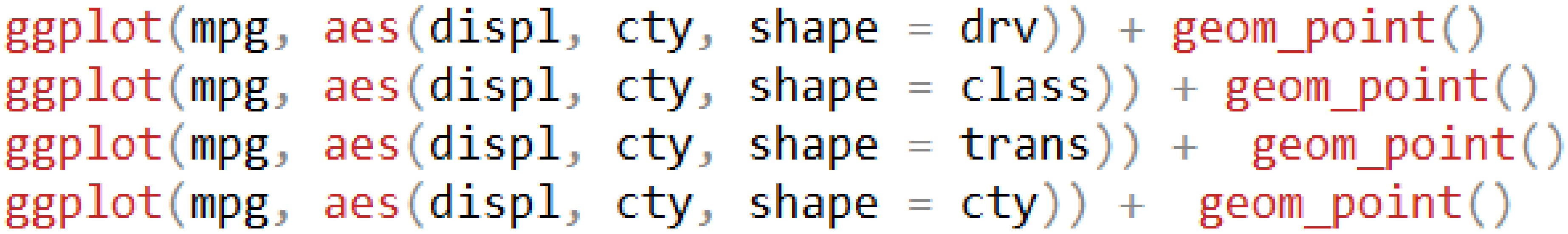
1. 위 그림에서 colour 요소에 다른 변수들을 넣어 봅시다

```r
ggplot(mpg, aes(displ, cty, colour = class )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = trans )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = drv )) + geom_point()
ggplot(mpg, aes(displ, cty, colour = cty )) + geom_point()
```



Colour, Shape

2. colour = 대신에 shape = 으로 바꾸면

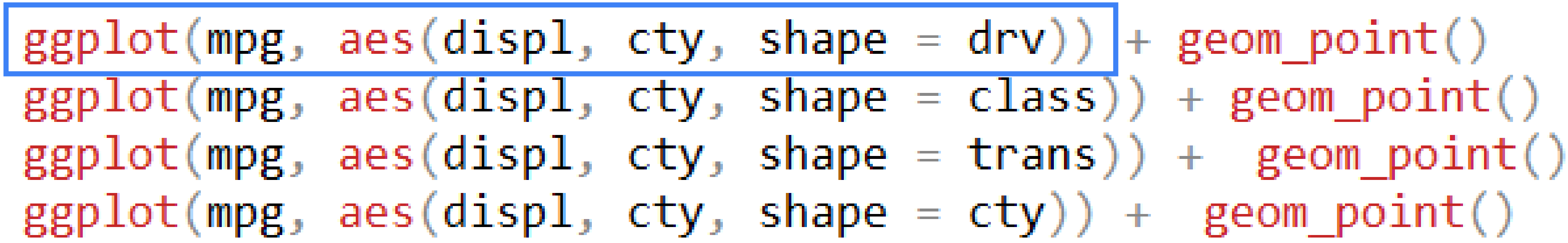


```
ggplot(mpg, aes(displ, cty, shape = drv)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = class)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = trans)) +  geom_point()
ggplot(mpg, aes(displ, cty, shape = cty)) +  geom_point()
```

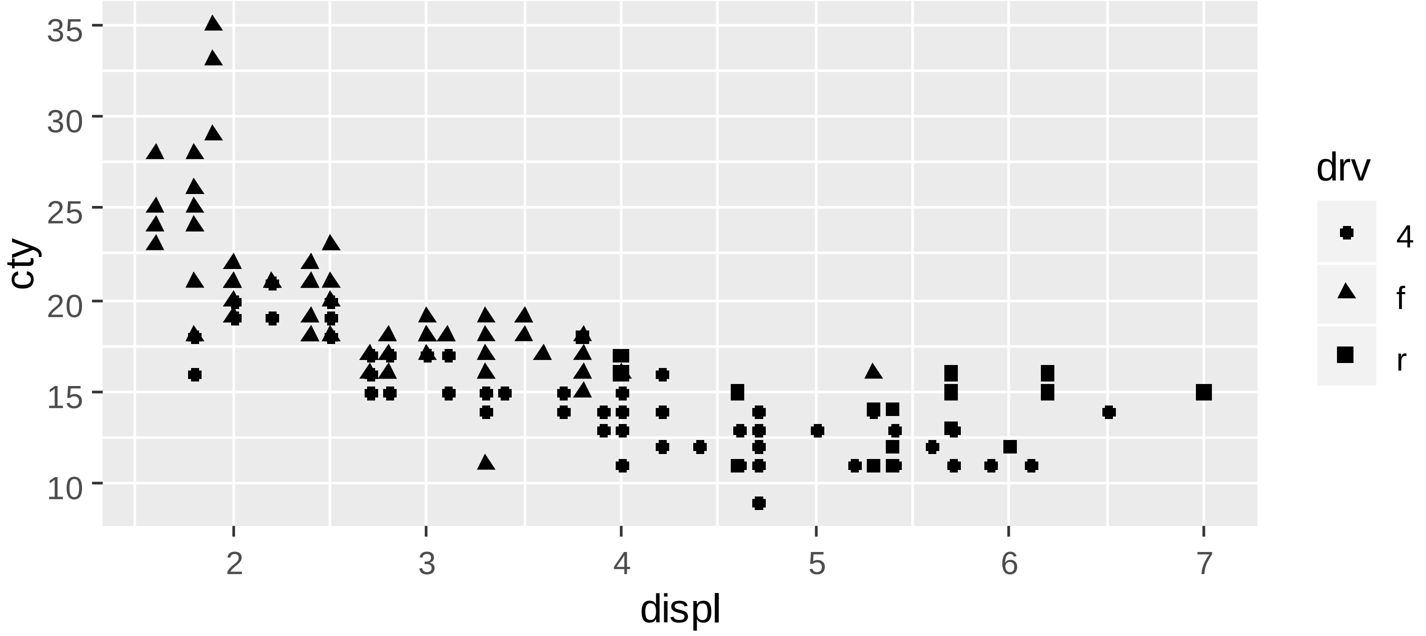
많은 warnings 과 error 가 뜹니다… 왜 그럴까요?

Colour, Shape

2. colour = 대신에 shape = 으로 바꾸면



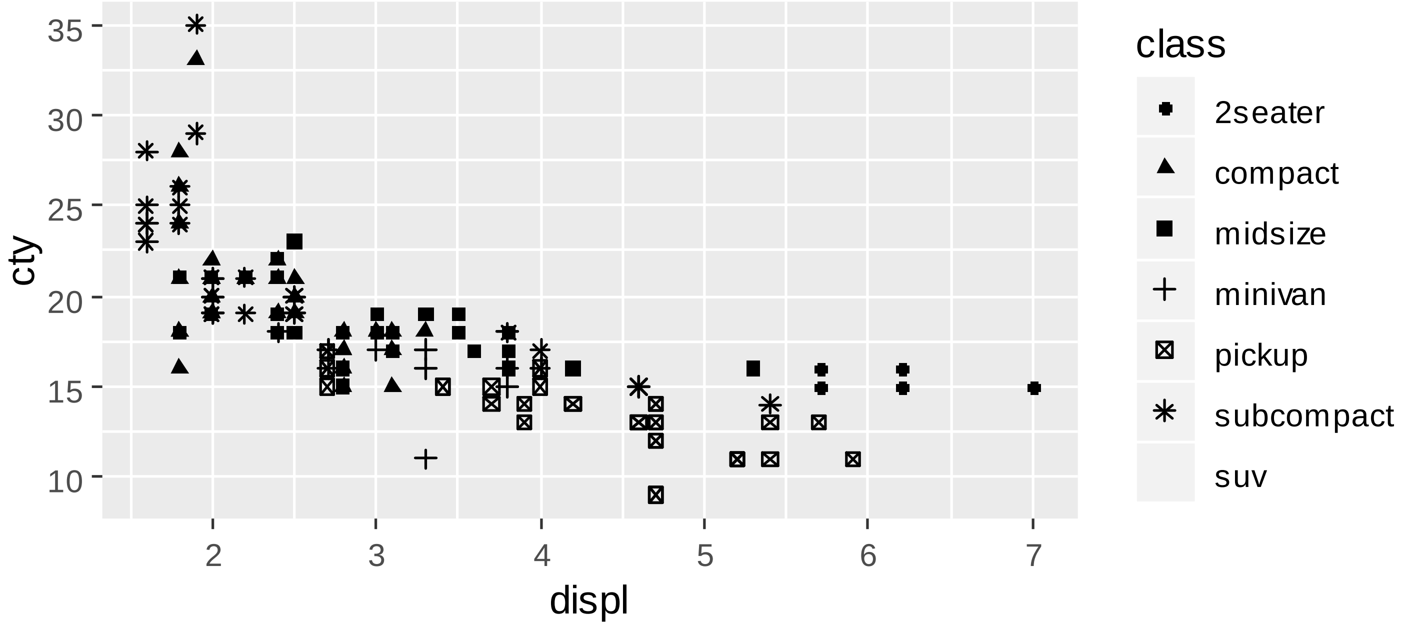
```
ggplot(mpg, aes(displ, cty, shape = drv)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = class)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = trans)) +  geom_point()
ggplot(mpg, aes(displ, cty, shape = cty)) +  geom_point()
```



Colour, Shape

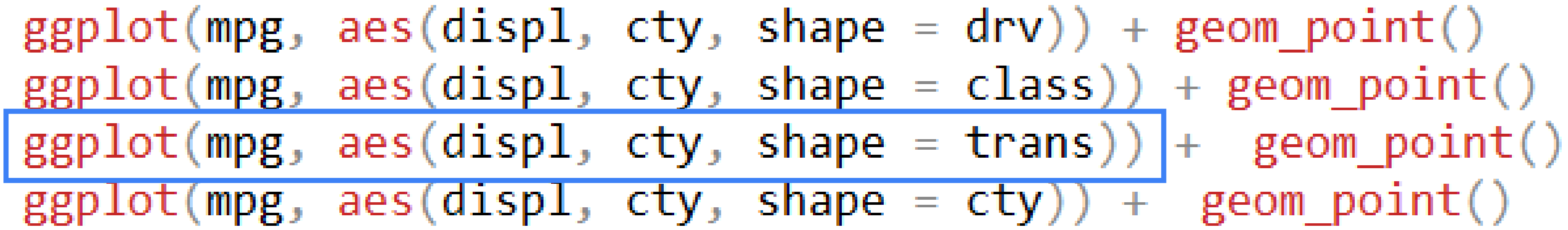
2. colour = 대신에 shape = 으로 바꾸면

```
ggplot(mpg, aes(displ, cty, shape = drv)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = class)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = trans)) +  geom_point()
ggplot(mpg, aes(displ, cty, shape = cty)) +  geom_point()
```

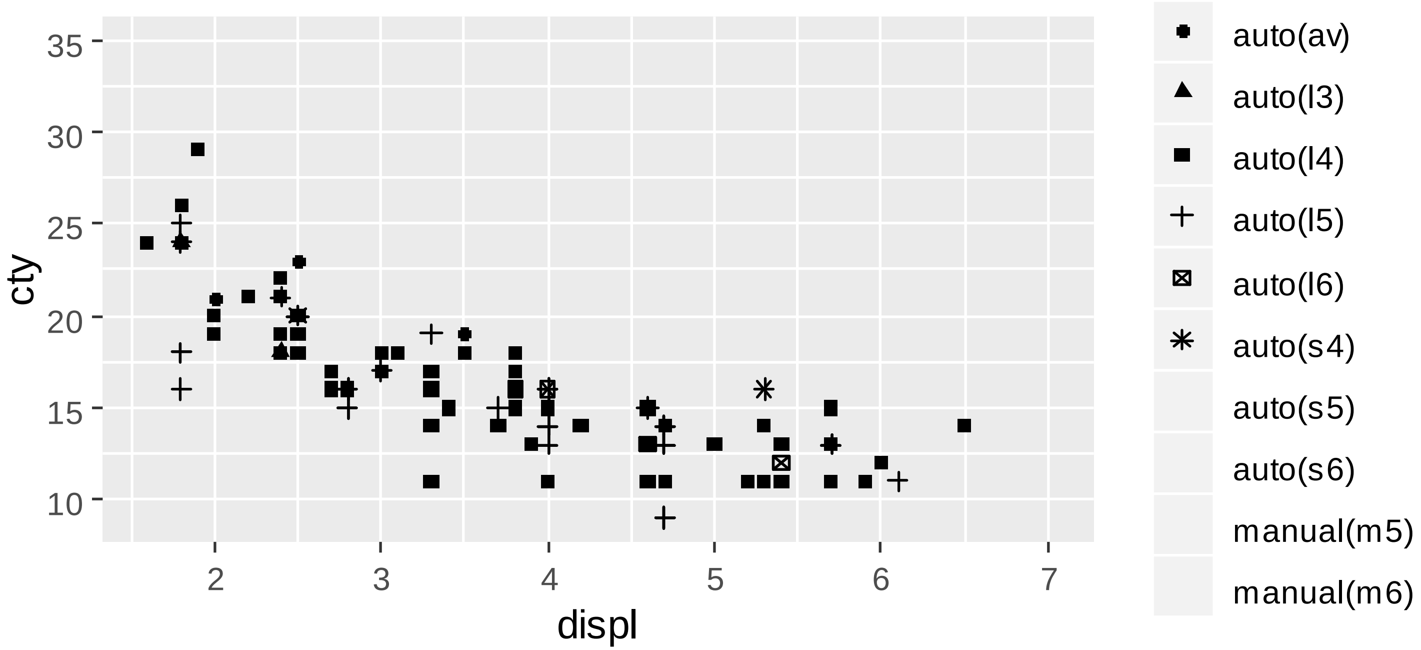


Colour, Shape

2. colour = 대신에 shape = 으로 바꾸면

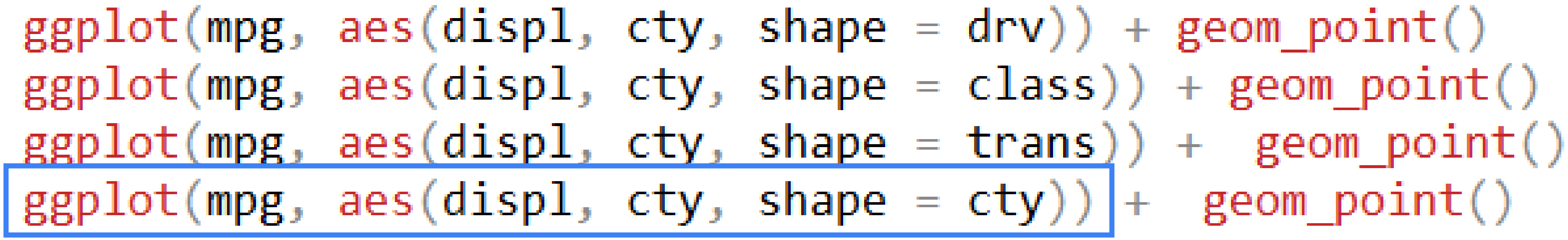


```
ggplot(mpg, aes(displ, cty, shape = drv)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = class)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = trans)) +  geom_point()
ggplot(mpg, aes(displ, cty, shape = cty)) +  geom_point()
```

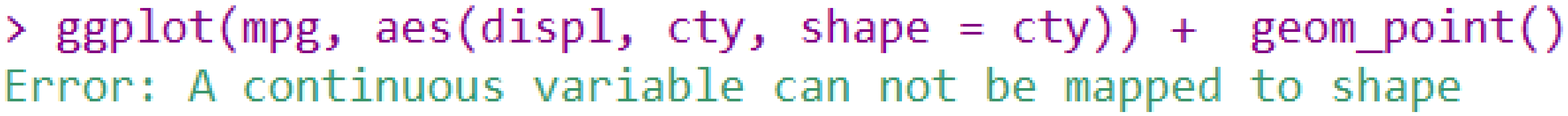


Colour, Shape

2. colour = 대신에 shape = 으로 바꾸면



```
ggplot(mpg, aes(displ, cty, shape = drv)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = class)) + geom_point()
ggplot(mpg, aes(displ, cty, shape = trans)) +  geom_point()
ggplot(mpg, aes(displ, cty, shape = cty)) +  geom_point()
```

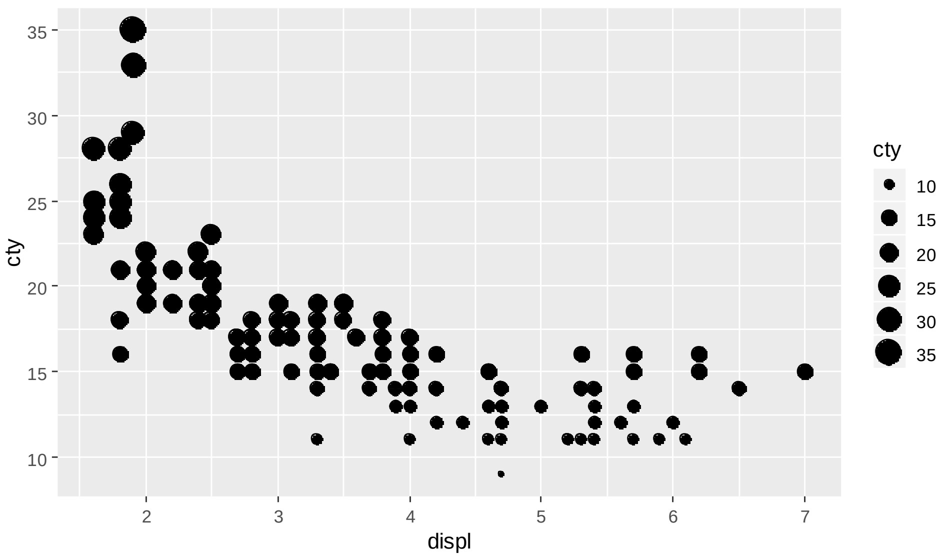


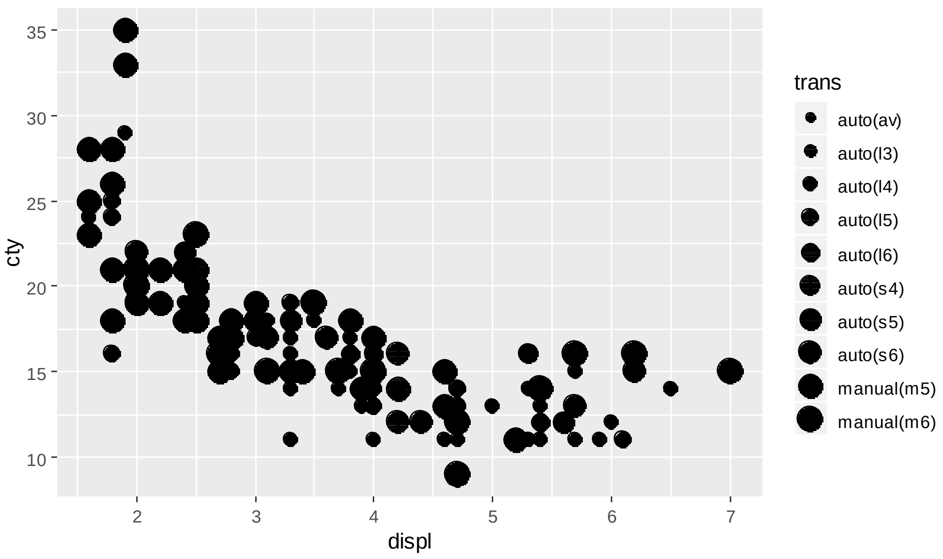
```
> ggplot(mpg, aes(displ, cty, shape = cty)) +  geom_point()
Error: A continuous variable can not be mapped to shape
```

Colour, Shape, Size

4. 연속형 변수에 size = 를 하면,

```
ggplot(mpg, aes(displ, cty, size = cty )) + geom_point()

ggplot(mpg, aes(displ, cty, size = trans )) + geom_point()
```

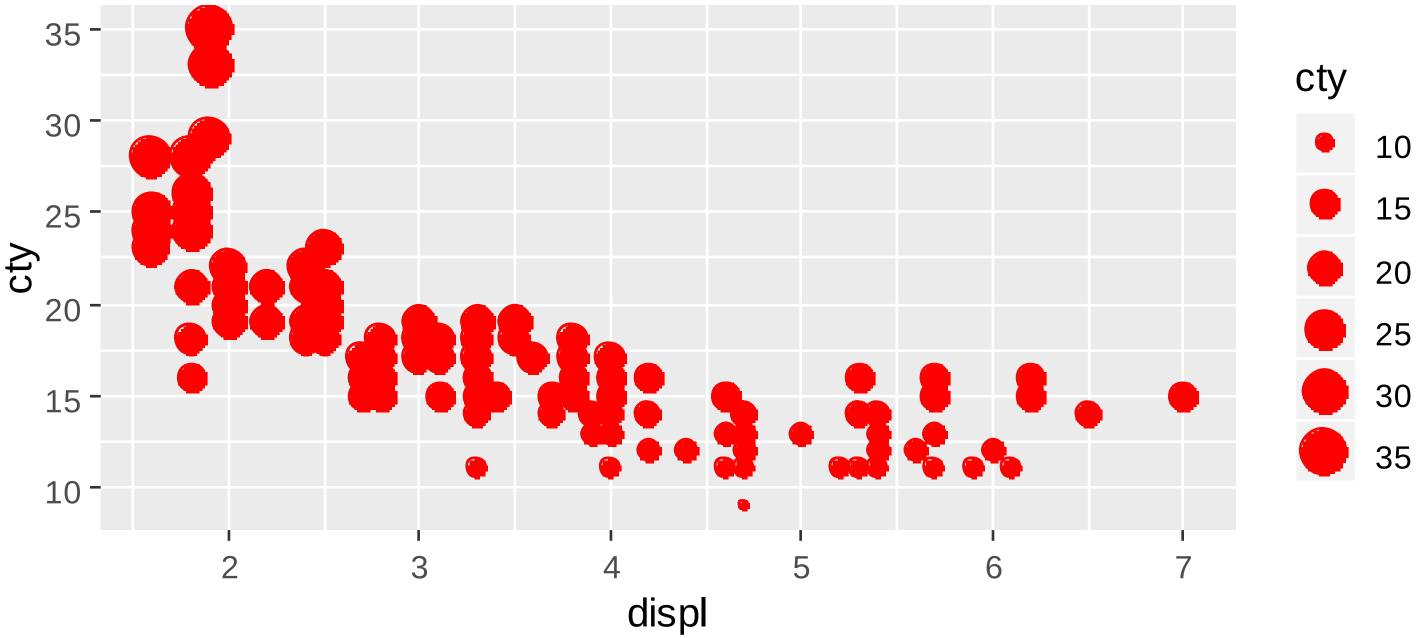




Colour, Size, Shape

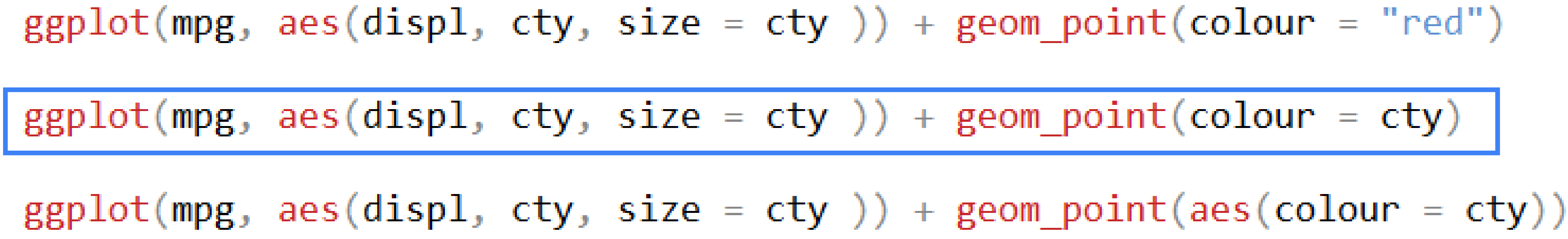
5. geom_point( ) 에서 색을 직접 지정할 수 있어요

```
ggplot(mpg, aes(displ, cty, size = cty )) + geom_point(colour = "red")

ggplot(mpg, aes(displ, cty, size = cty )) + geom_point(colour = cty)

ggplot(mpg, aes(displ, cty, size = cty )) + geom_point(aes(colour = cty))
```

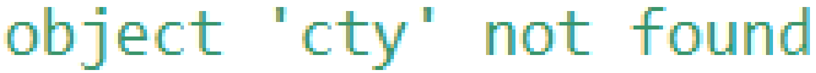


Colour, Size, Shape

5. geom_point( ) 에 직접 색을 지정할 수 있어요



```
ggplot(mpg, aes(displ, cty, size = cty )) + geom_point(colour = "red")

ggplot(mpg, aes(displ, cty, size = cty )) + geom_point(colour = cty)

ggplot(mpg, aes(displ, cty, size = cty )) + geom_point(aes(colour = cty))
```

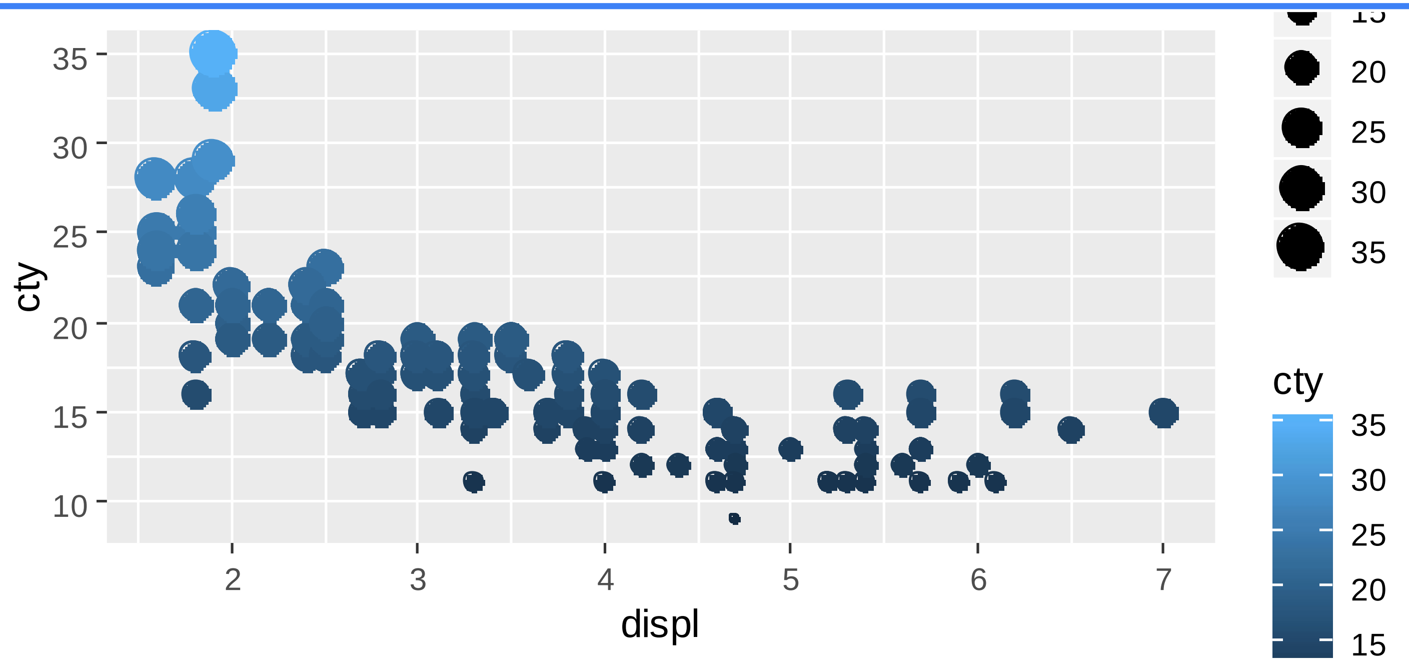


```
object 'cty' not found
```

Colour, Size, Shape

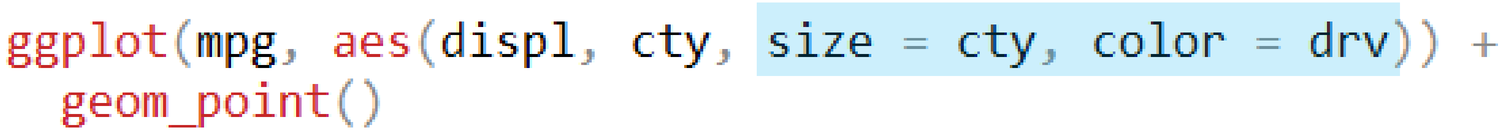
5. geom_point( ) 에 직접 색을 지정할 수 있어요

```
ggplot(mpg, aes(displ, cty, size = cty )) + geom_point(colour = "red")

ggplot(mpg, aes(displ, cty, size = cty )) + geom_point(colour = cty)

ggplot(mpg, aes(displ, cty, size = cty )) + geom_point(aes(colour = cty))
```

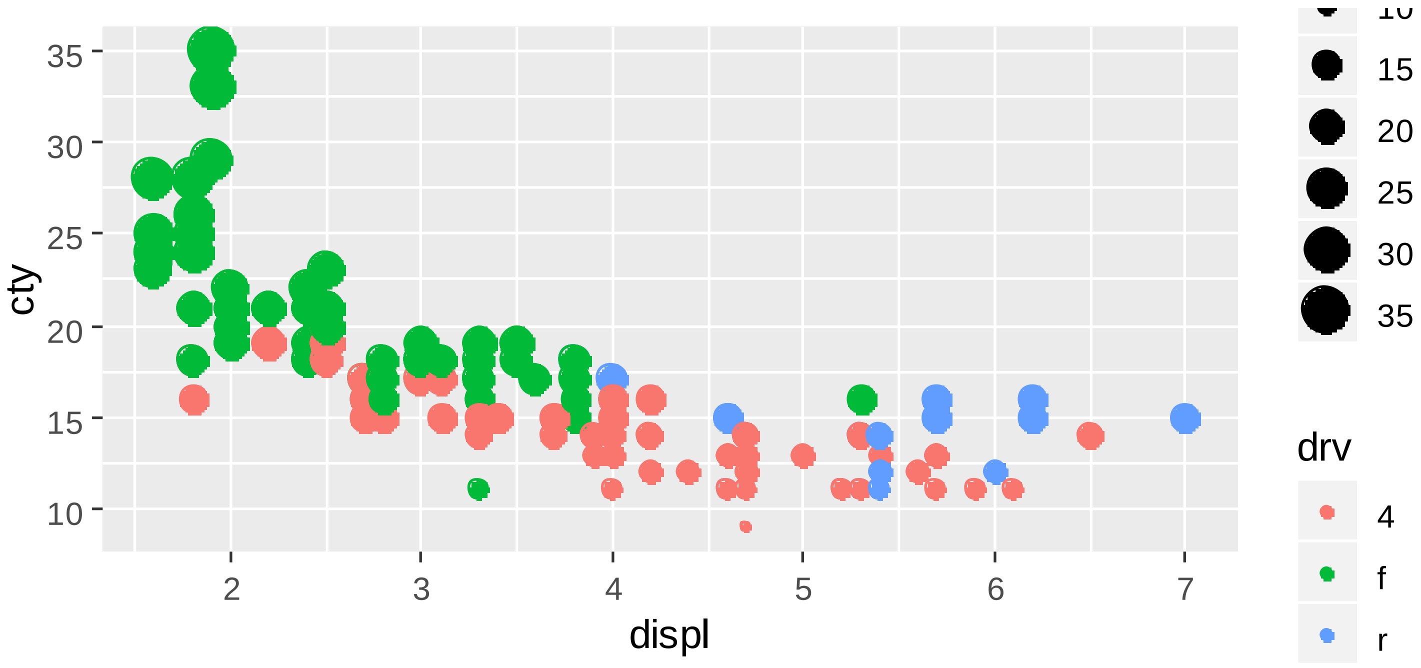


Colour, Size, Shape

6. 만약 size와 color를 다르게 주면 어떤 그림을 그려 낼까요



```
ggplot(mpg, aes(displ, cty, size = cty, color = drv)) +
  geom_point()
```



다음 그림을 미리 예상해 보고, 실제 연습해 봅시다

1. `ggplot(mpg, aes(cty, hwy)) + geom_point(  )`

2. `ggplot(diamonds, aes(carat, price)) + geom_point( )`

3. ggplot(economics, aes(date, unemploy)) + geom line( )

4. ggplot(mpg, aes(cty)) + geom histogram( )

5. ggplot(mpg, aes(cty)) + geom histogram(bins= 20)

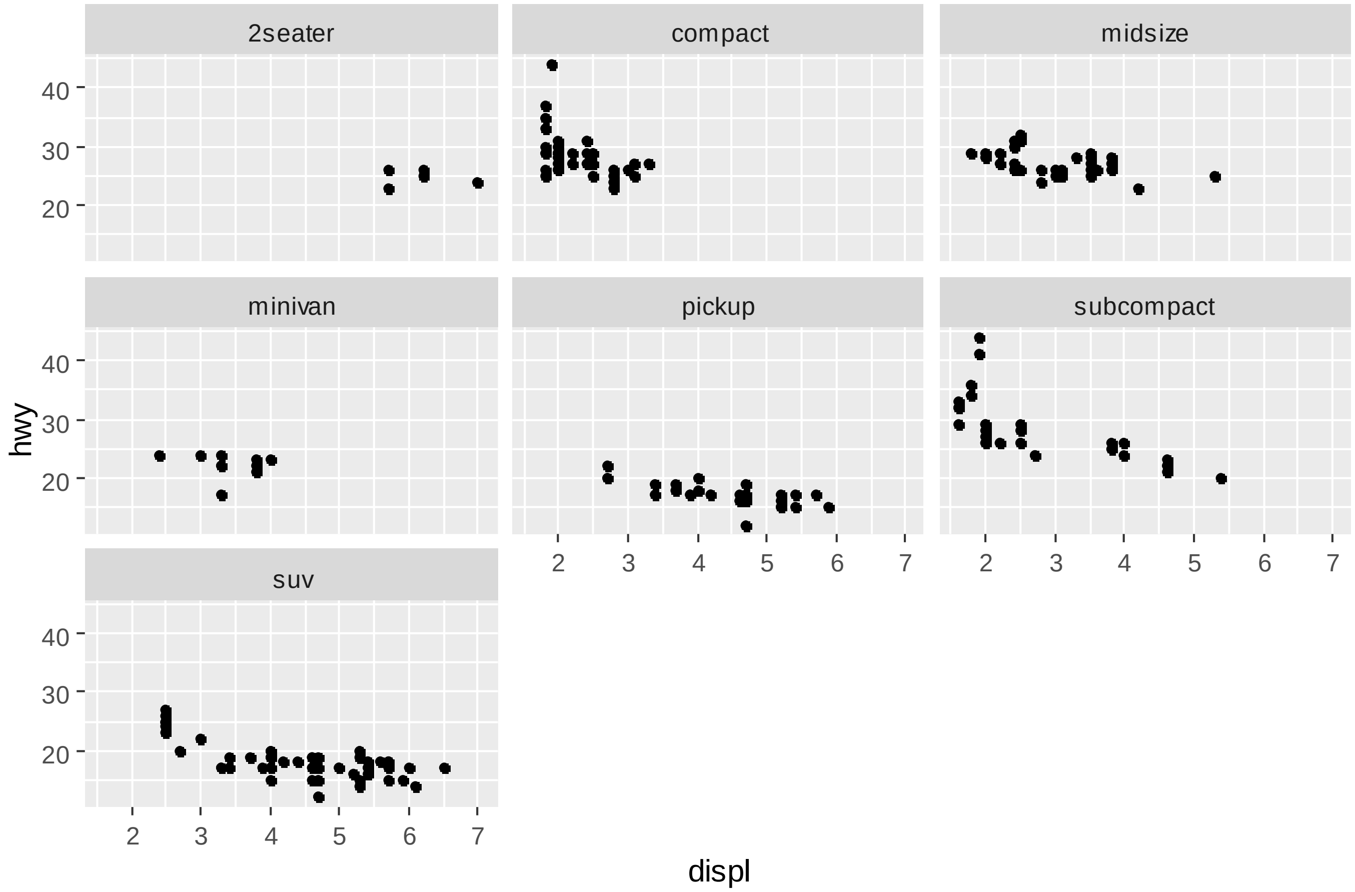
Facetting

Another technique for displaying additional categorical variables on a plot is facetting.
Facetting creates tables of graphics by splitting the data into
subsets and displaying the same graph for each subset.

There are two types of facetting: grid and wrapped.
To facet a plot you simply add a facetting specification with
facet_wrap( ) , facet_grid( )
which takes the name of a variable preceded by ˜ .

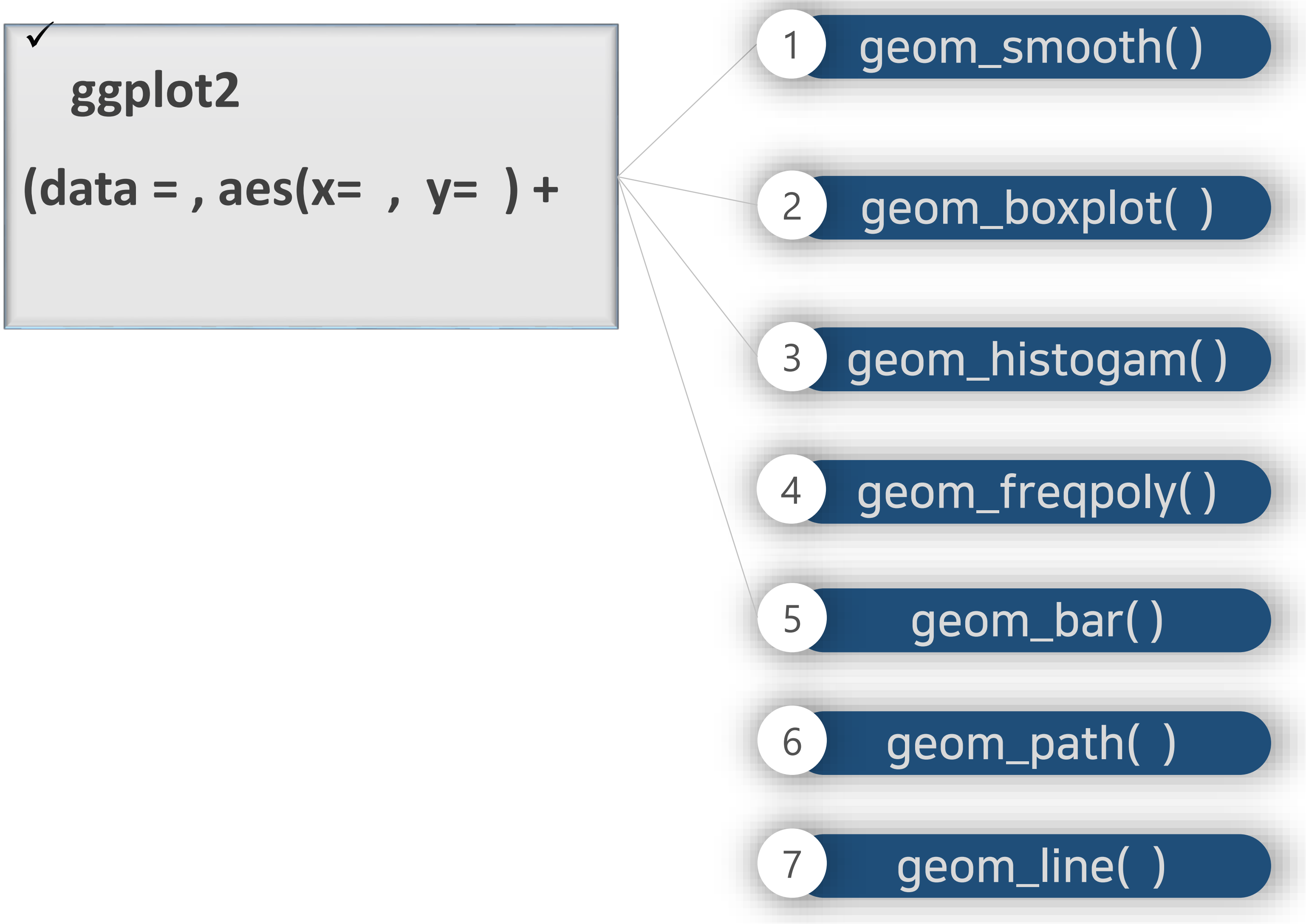
Facetting

```
ggplot(mpg, aes(displ, hwy)) +
  geom_point() +
  facet_wrap(~class)
```



WeDataLab
데이터컨설팅 데이터아카데미

2. geom_ * 요소 살펴보기



✓

ggplot2

(data = , aes(x= , y=) +

1 geom_smooth( )

2 geom_boxplot( )

3 geom_histogam( )

4 geom_freqpoly( )

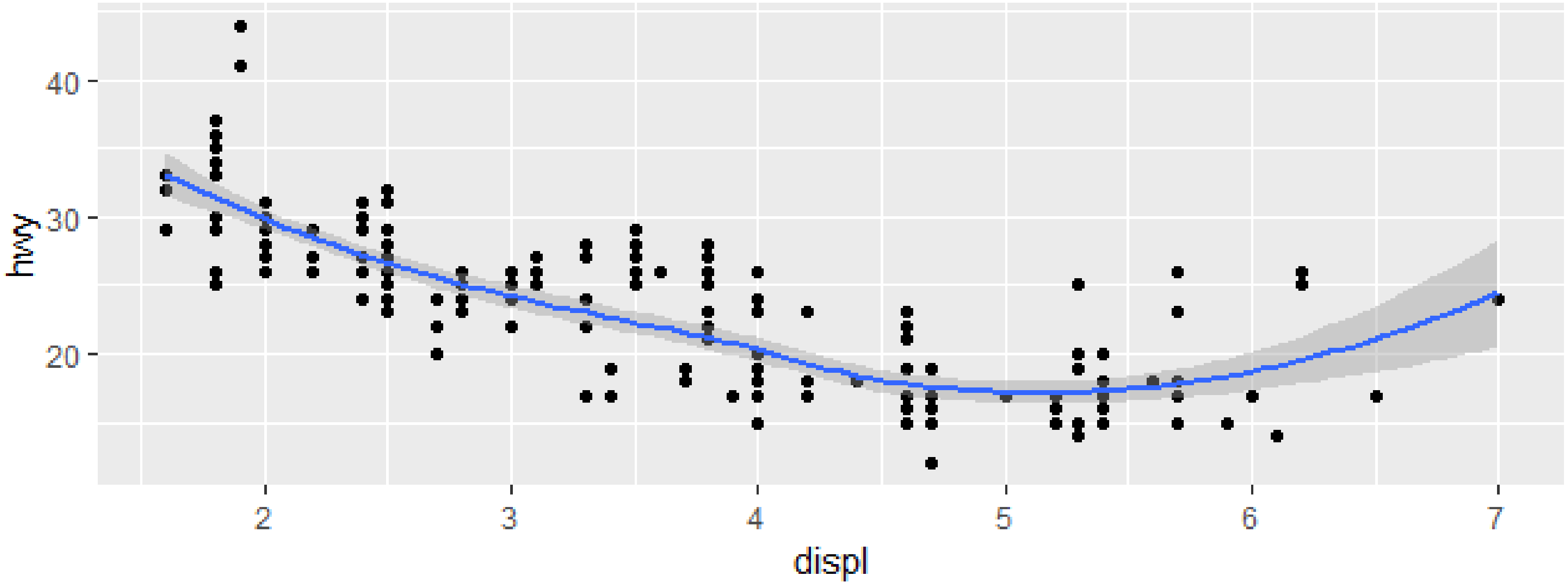
5 geom_bar( )

6 geom_path( )

7 geom_line( )

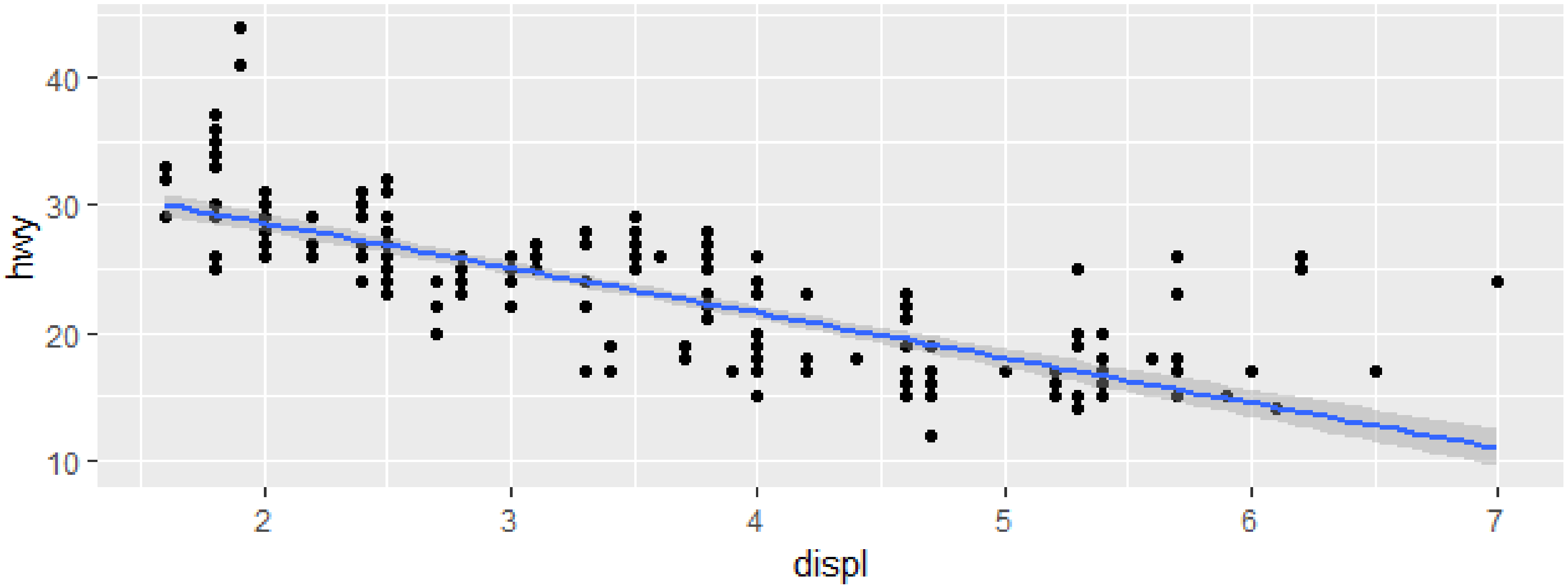
geom_smooth( )

```
ggplot(mpg, aes(displ, hwy)) +
  geom_point() +
  geom_smooth()
```



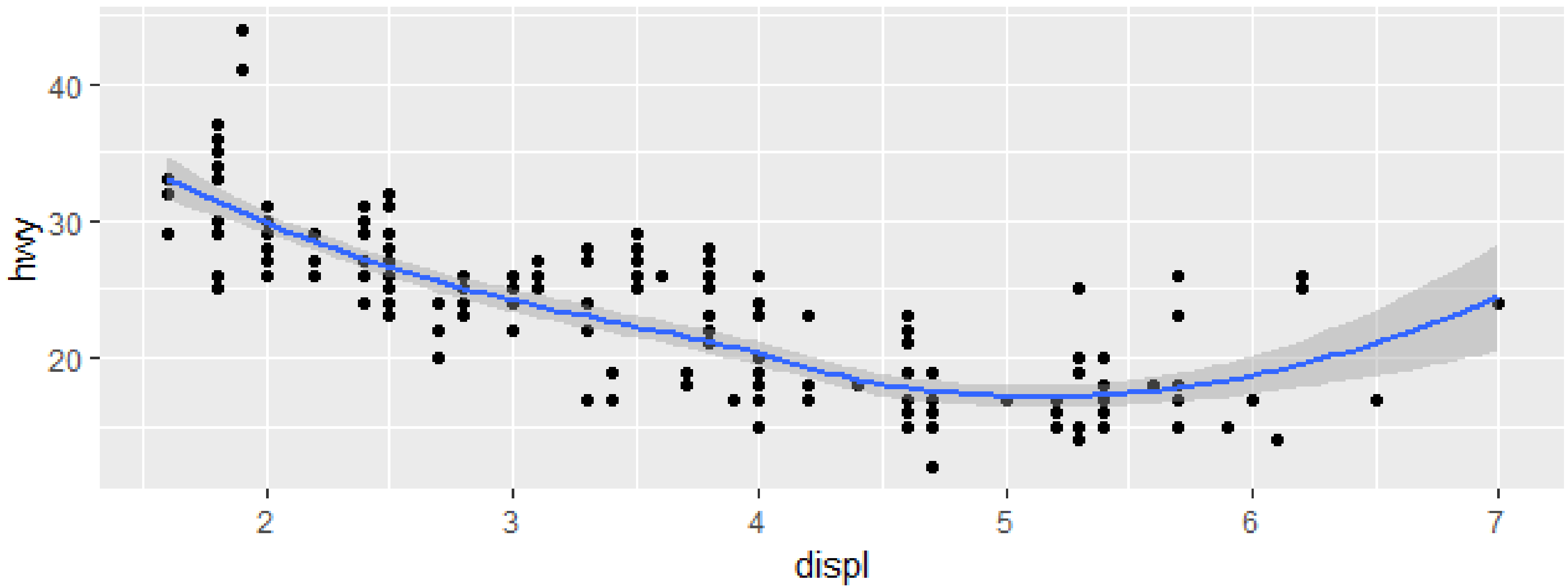
geom_smooth( )

```
ggplot(mpg, aes(displ, hwy)) +
  geom_point() +
  geom_smooth(method = "lm")
```



geom_smooth( )

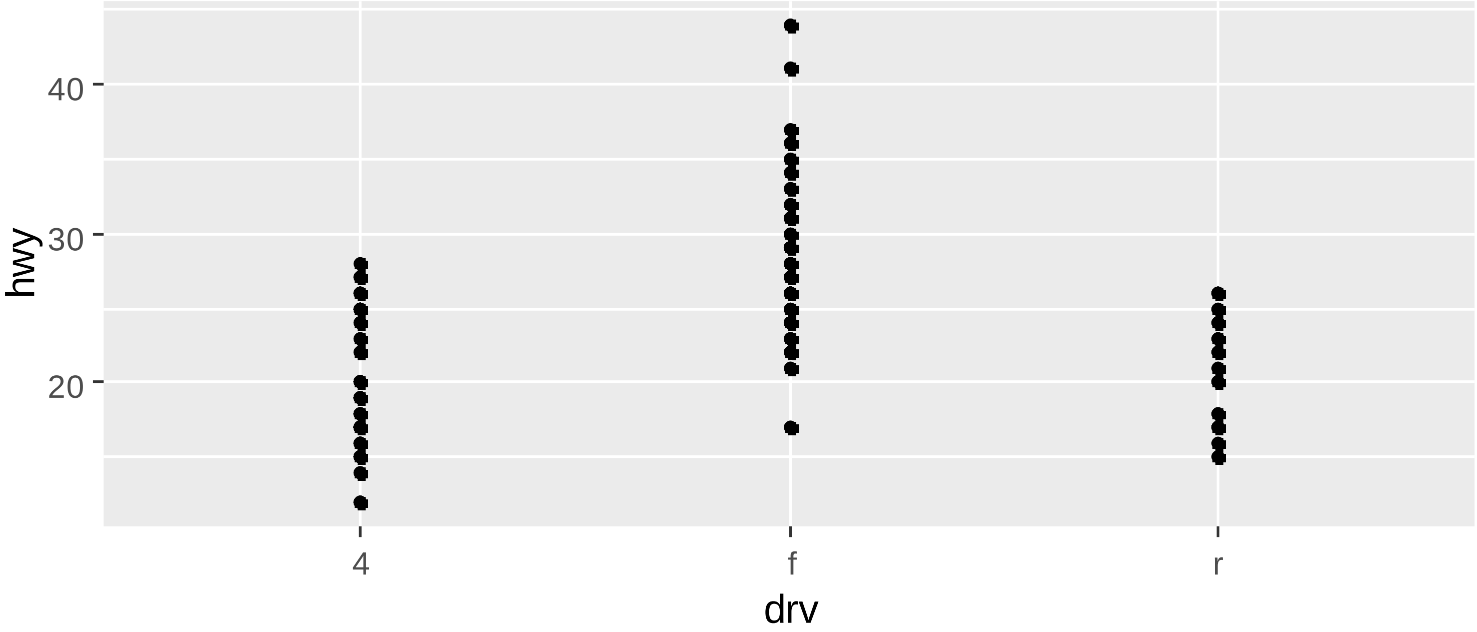
```
ggplot(mpg, aes(displ, hwy))+
  geom_point()+
  geom_smooth(method = 'loess')
```



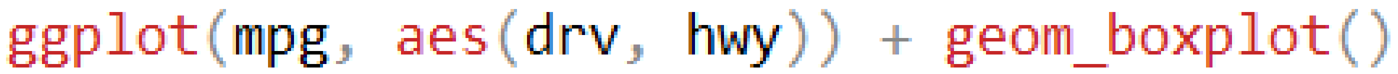
geom_boxplot( )

```
ggplot(mpg, aes(drv, hwy)) +
  geom_point()
```

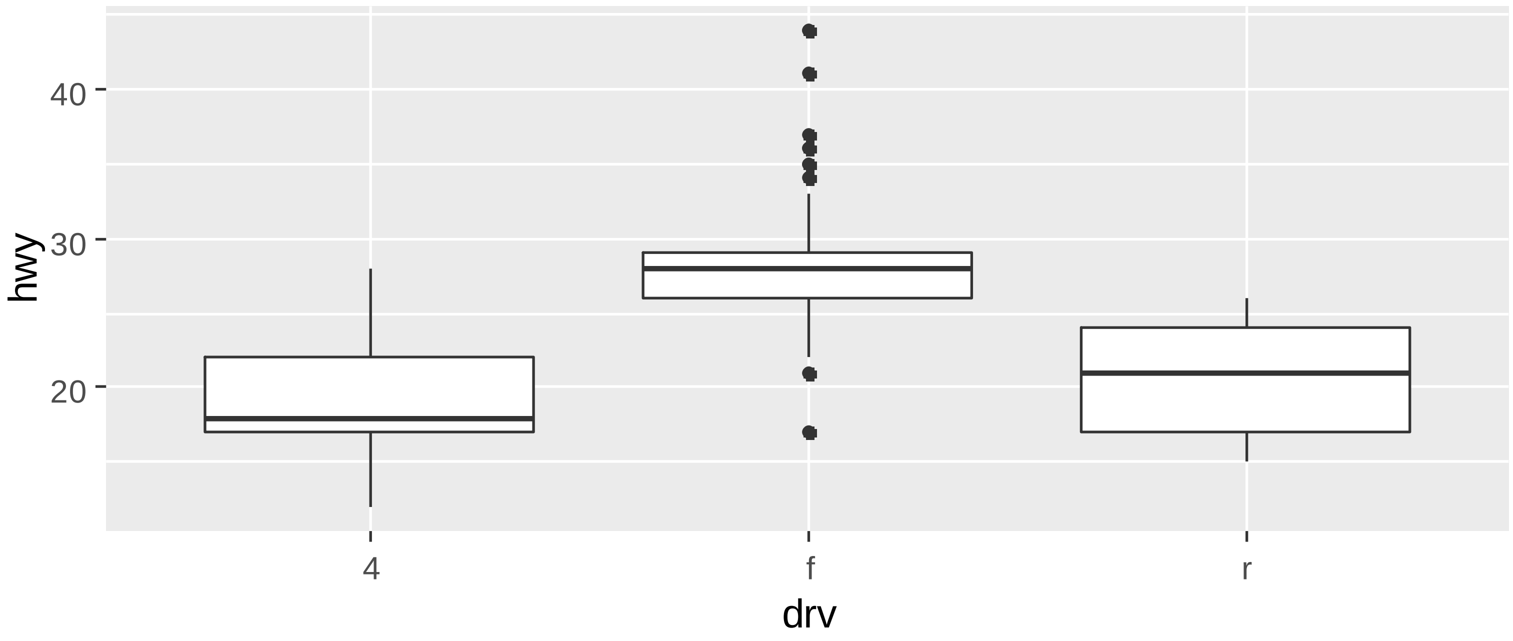
어느 한 변수가 categorical variables(범주형 변수) 일 때 geom_point( ) 를 쓰면 다음과 같은 그림이 나온다.

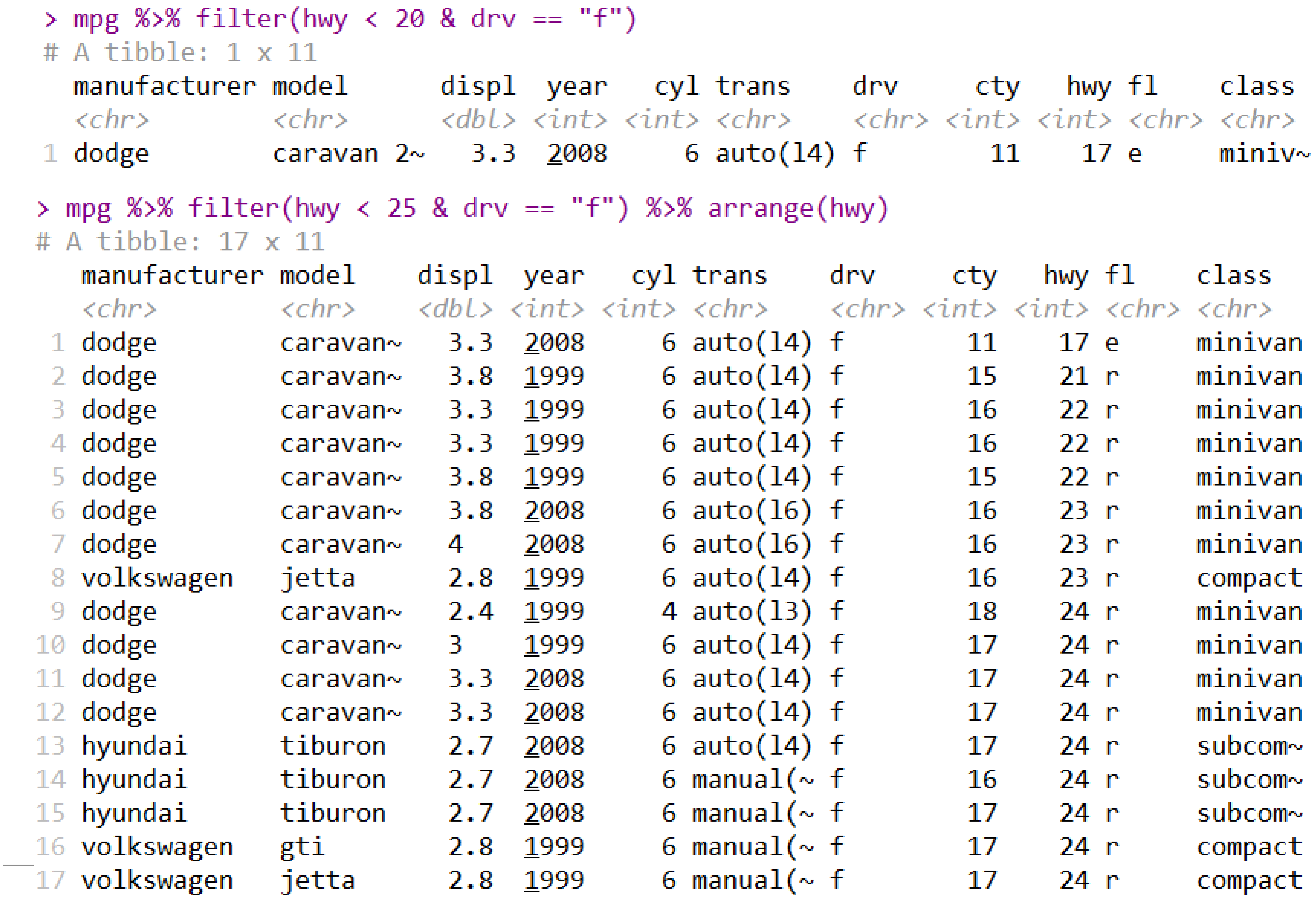


geom_boxplot( )



```
ggplot(mpg, aes(drv, hwy)) + geom_boxplot()
```

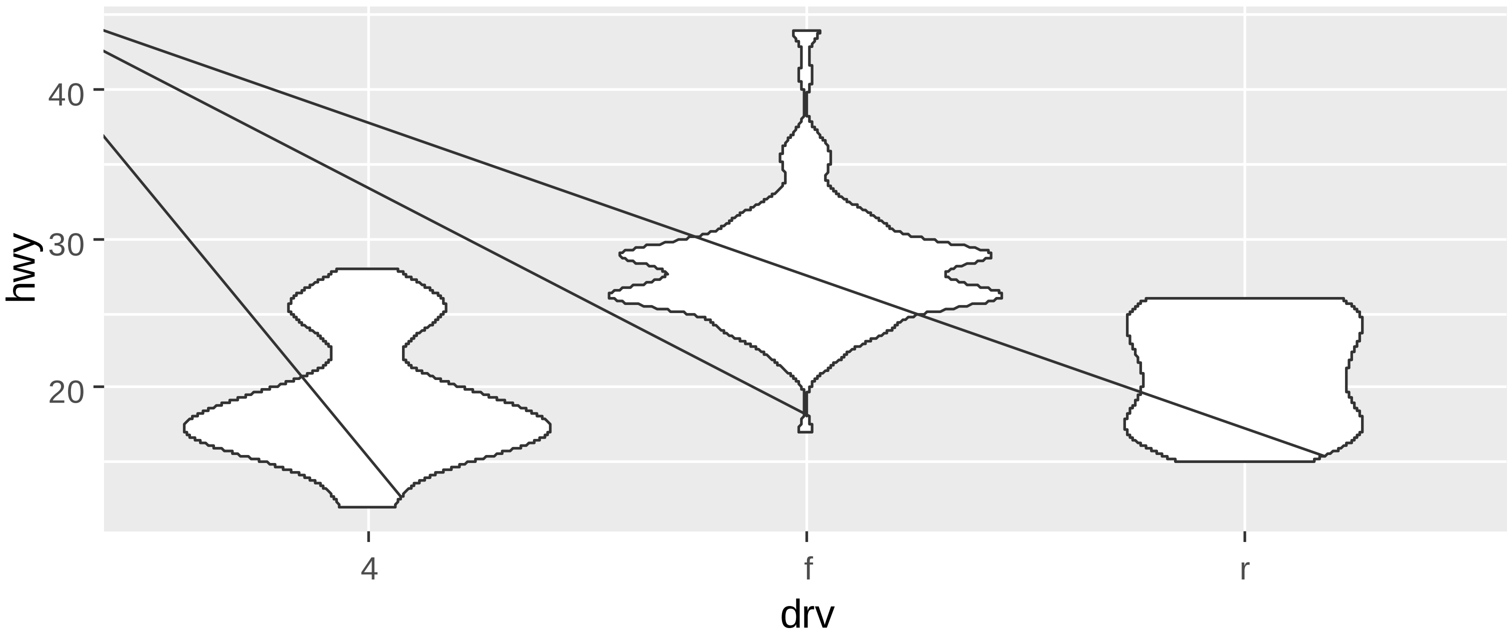




```
> mpg %>% filter(hwy < 20 & drv == "f")
# A tibble: 1 x 11
  manufacturer model      displ  year   cyl trans      drv     cty   hwy fl    class 
  <chr>        <chr>      <dbl> <int> <int> <chr>      <chr> <int> <int> <chr> <chr> 
1 dodge        caravan 2~   3.3  2008     6 auto(l4) f        11    17 e     miniv~

> mpg %>% filter(hwy < 25 & drv == "f") %>% arrange(hwy)
# A tibble: 17 x 11
   manufacturer model    displ  year   cyl trans      drv     cty   hwy fl    class  
   <chr>        <chr>    <dbl> <int> <int> <chr>      <chr> <int> <int> <chr> <chr>  
 1 dodge        caravan~   3.3  2008     6 auto(l4)   f        11    17 e     minivan
 2 dodge        caravan~   3.8  1999     6 auto(l4)   f        15    21 r     minivan
 3 dodge        caravan~   3.3  1999     6 auto(l4)   f        16    22 r     minivan
 4 dodge        caravan~   3.3  1999     6 auto(l4)   f        16    22 r     minivan
 5 dodge        caravan~   3.8  1999     6 auto(l4)   f        15    22 r     minivan
 6 dodge        caravan~   3.8  2008     6 auto(l6)   f        16    23 r     minivan
 7 dodge        caravan~   4    2008     6 auto(l6)   f        16    23 r     minivan
 8 volkswagen   jetta      2.8  1999     6 auto(l4)   f        16    23 r     compact
 9 dodge        caravan~   2.4  1999     4 auto(l3)   f        18    24 r     minivan
10 dodge        caravan~   3    1999     6 auto(l4)   f        17    24 r     minivan
11 dodge        caravan~   3.3  2008     6 auto(l4)   f        17    24 r     minivan
12 dodge        caravan~   3.3  2008     6 auto(l4)   f        17    24 r     minivan
13 hyundai      tiburon    2.7  2008     6 auto(l4)   f        17    24 r     subcom~
14 hyundai      tiburon    2.7  2008     6 manual(~ f        16    24 r     subcom~
15 hyundai      tiburon    2.7  2008     6 manual(~ f        17    24 r     subcom~
16 volkswagen   gti        2.8  1999     6 manual(~ f        17    24 r     compact
17 volkswagen   jetta      2.8  1999     6 manual(~ f        17    24 r     compact
```

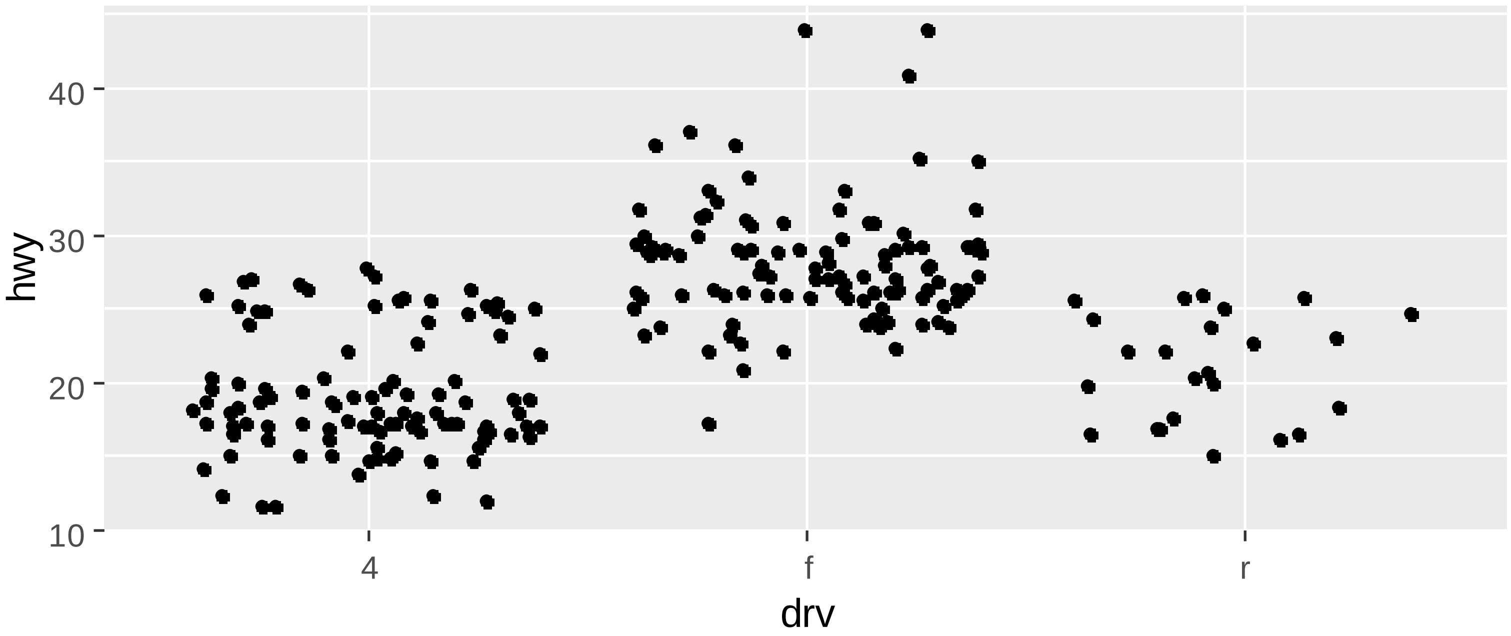
geom_boxplot( )

```
ggplot(mpg, aes(drv, hwy)) + geom_violin()
```



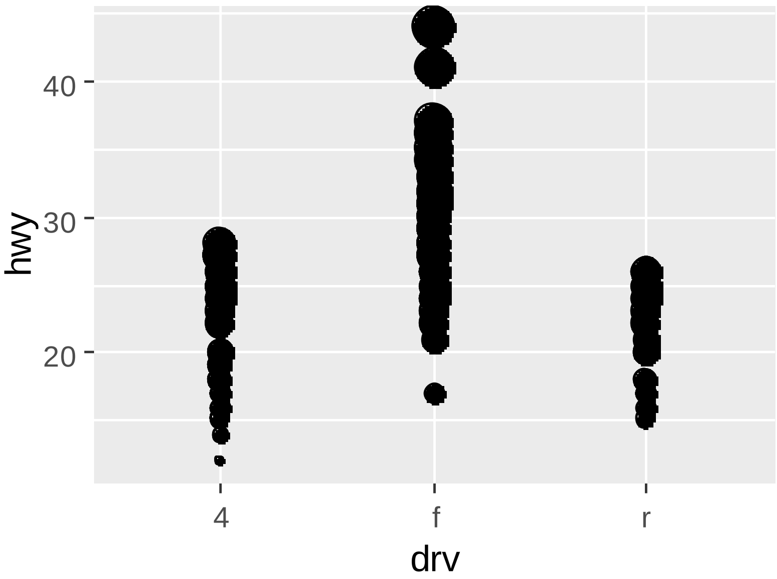
geom_boxplot( )

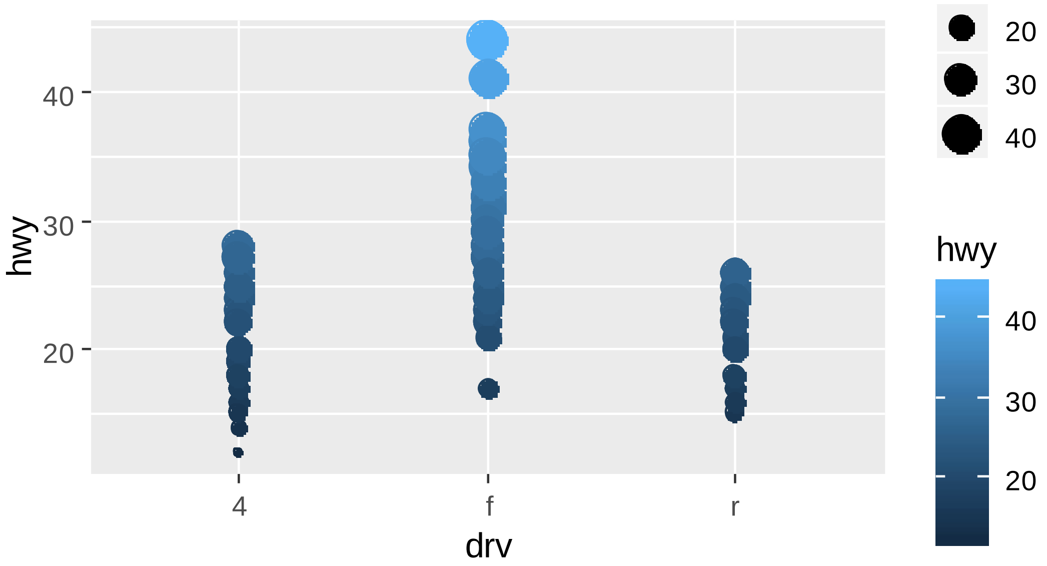
```
ggplot(mpg, aes(drv, hwy)) + geom_jitter()
```



geom_boxplot( )

```
ggplot(mpg, aes(drv, hwy, size = hwy)) + geom_point()

ggplot(mpg, aes(drv, hwy, size = hwy, color = hwy)) +
  geom_point()
```

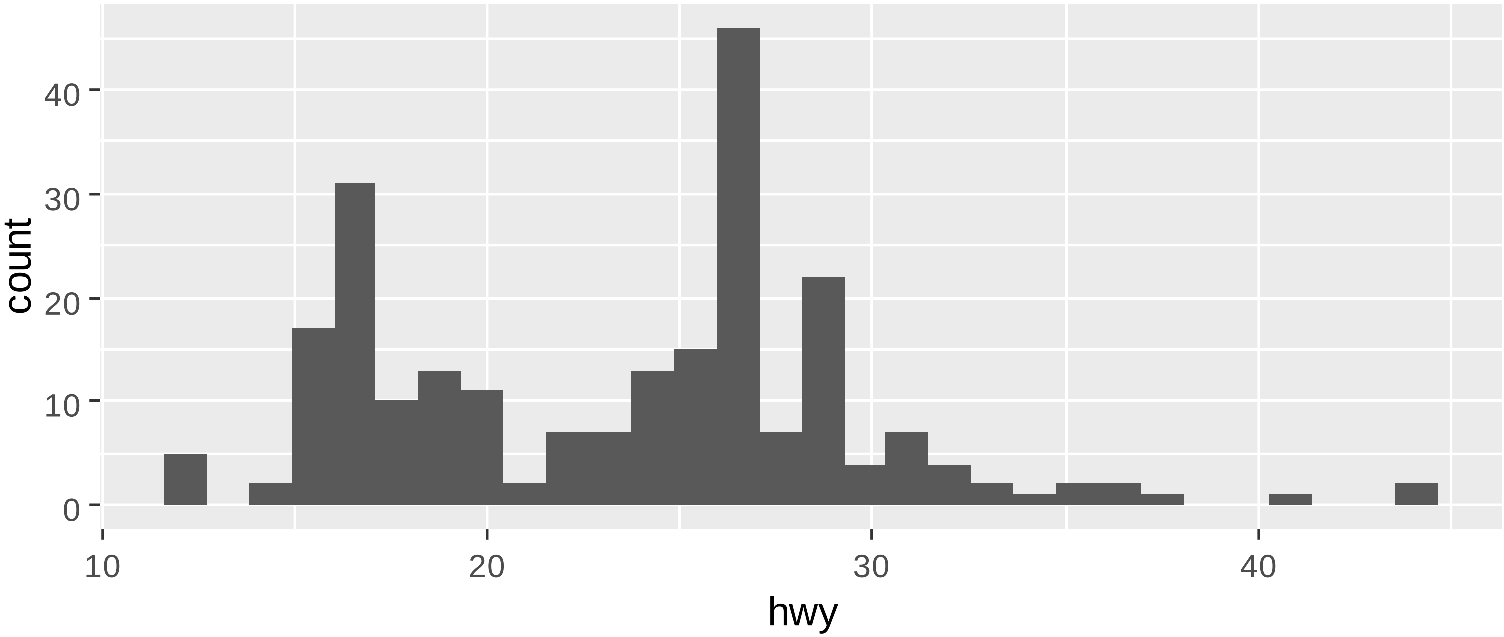




geom_histogram( )

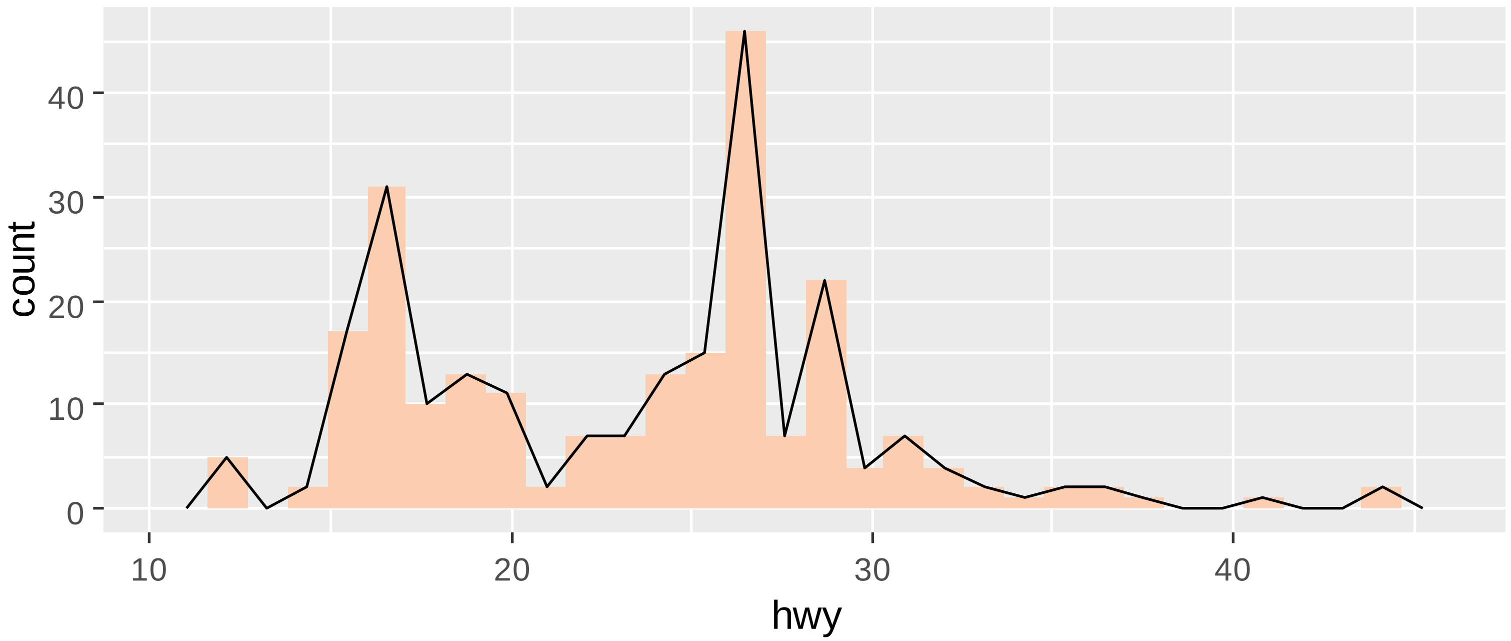
```
ggplot(mpg, aes(hwy)) + geom_histogram()
#> `stat_bin()` using `bins = 30`. Pick better value with
#> `binwidth`.
```

- 히스토그램은 1개의 연속형 변수에 대하여 사용 (boxplot은 2개 이상 가능)
- bins 개수는 30개. bins 또는 binwidth 로 조정

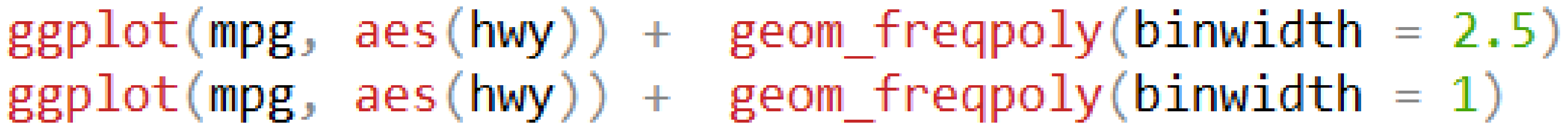


geom_freploy( )

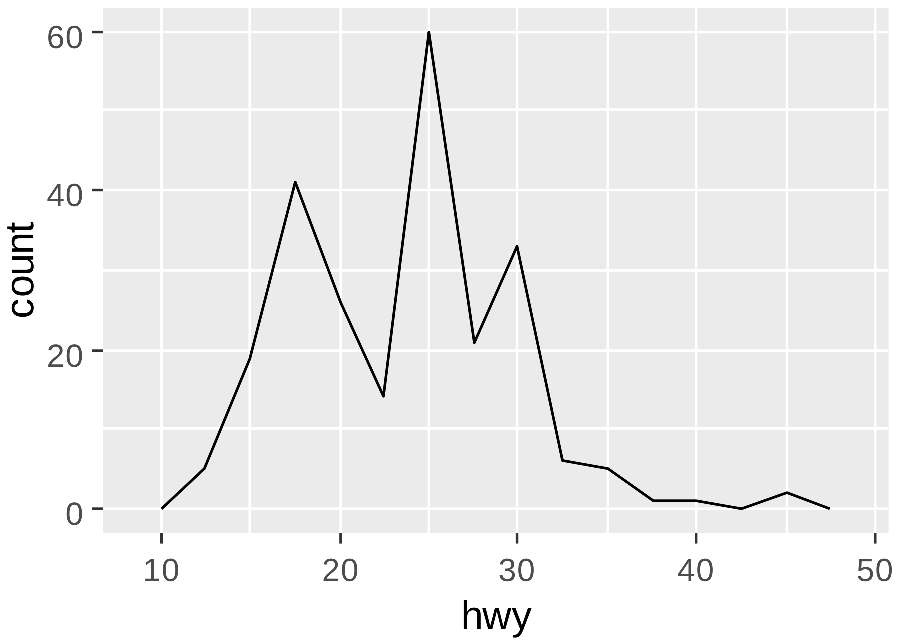
```
ggplot(mpg, aes(hwy)) + geom_freqpoly()
#> `stat_bin()` using `bins = 30`. Pick better value with
#> `binwidth`.
```

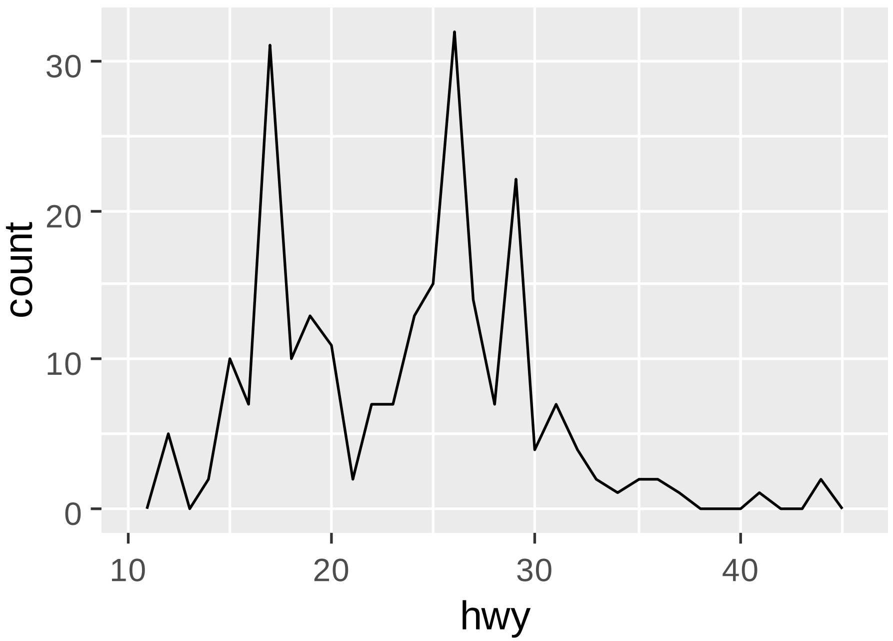


geom_histogram( )

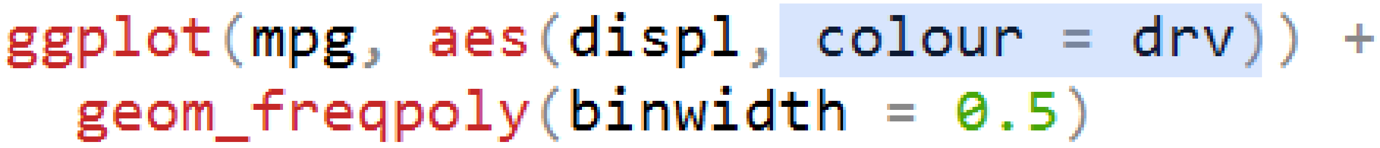


```
ggplot(mpg, aes(hwy)) +  geom_freqpoly(binwidth = 2.5)
ggplot(mpg, aes(hwy)) +  geom_freqpoly(binwidth = 1)
```

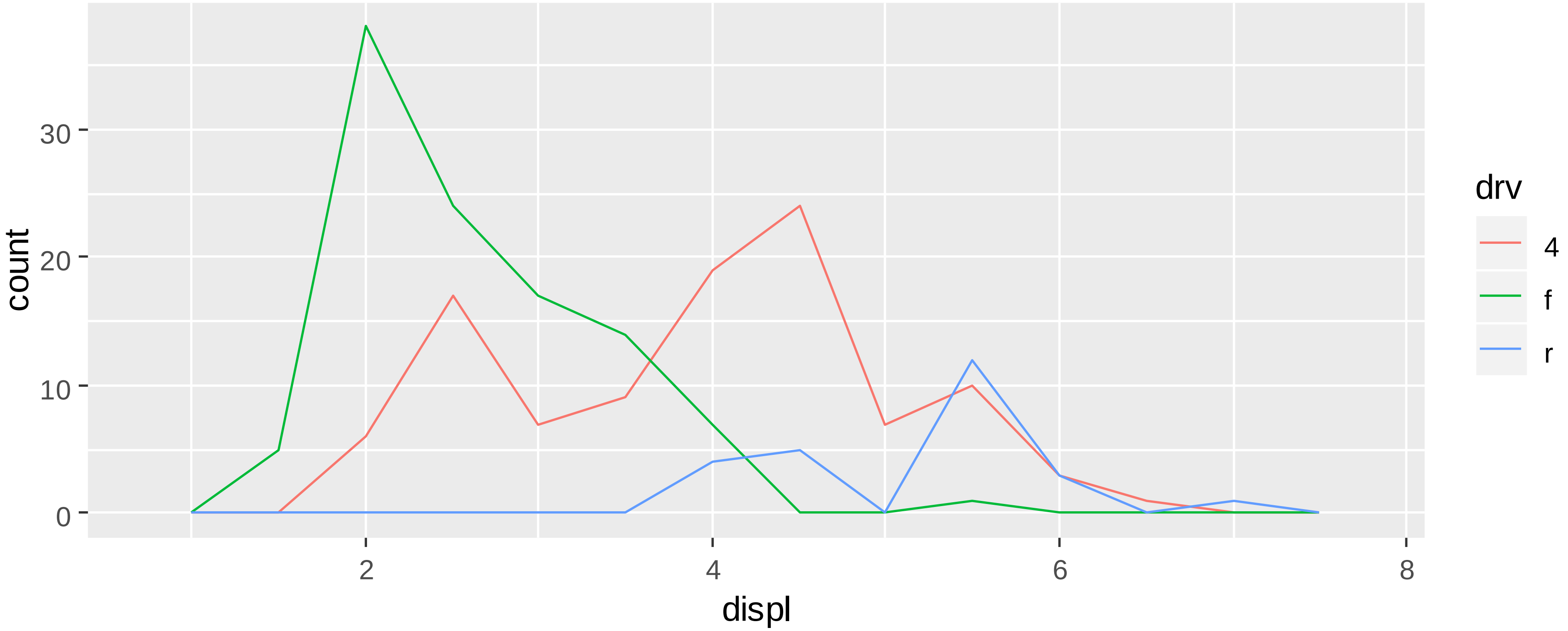




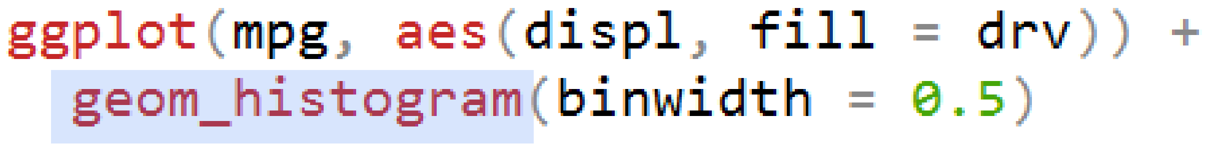
geom_histogram( )



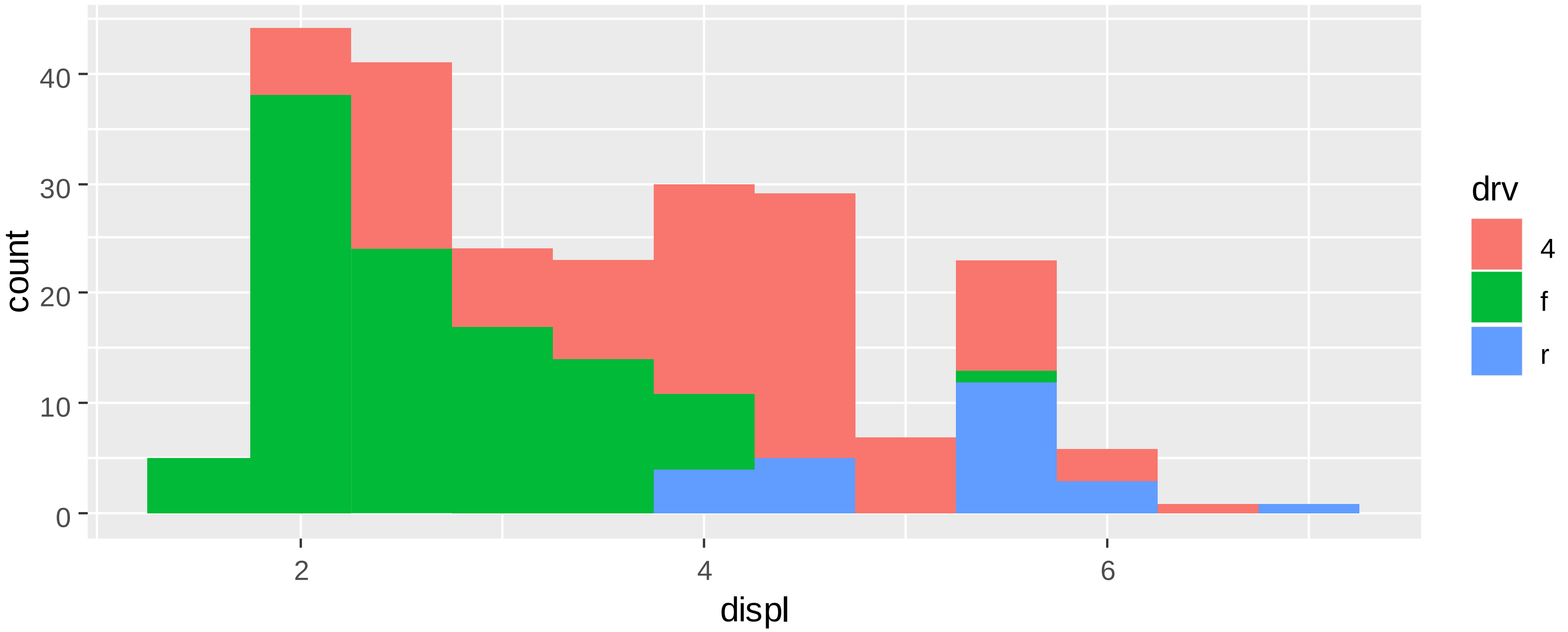
```
ggplot(mpg, aes(displ, colour = drv)) +
  geom_freqpoly(binwidth = 0.5)
```



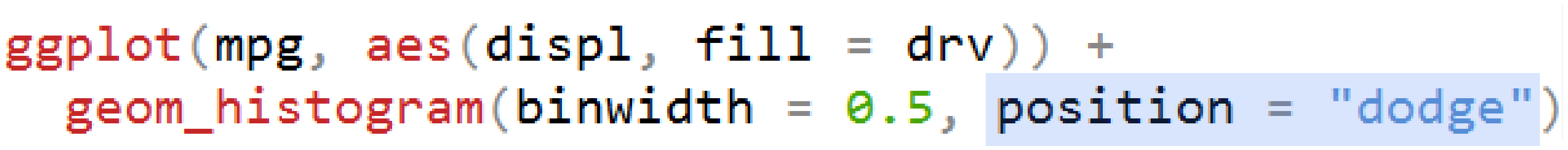
geom_histogram( )



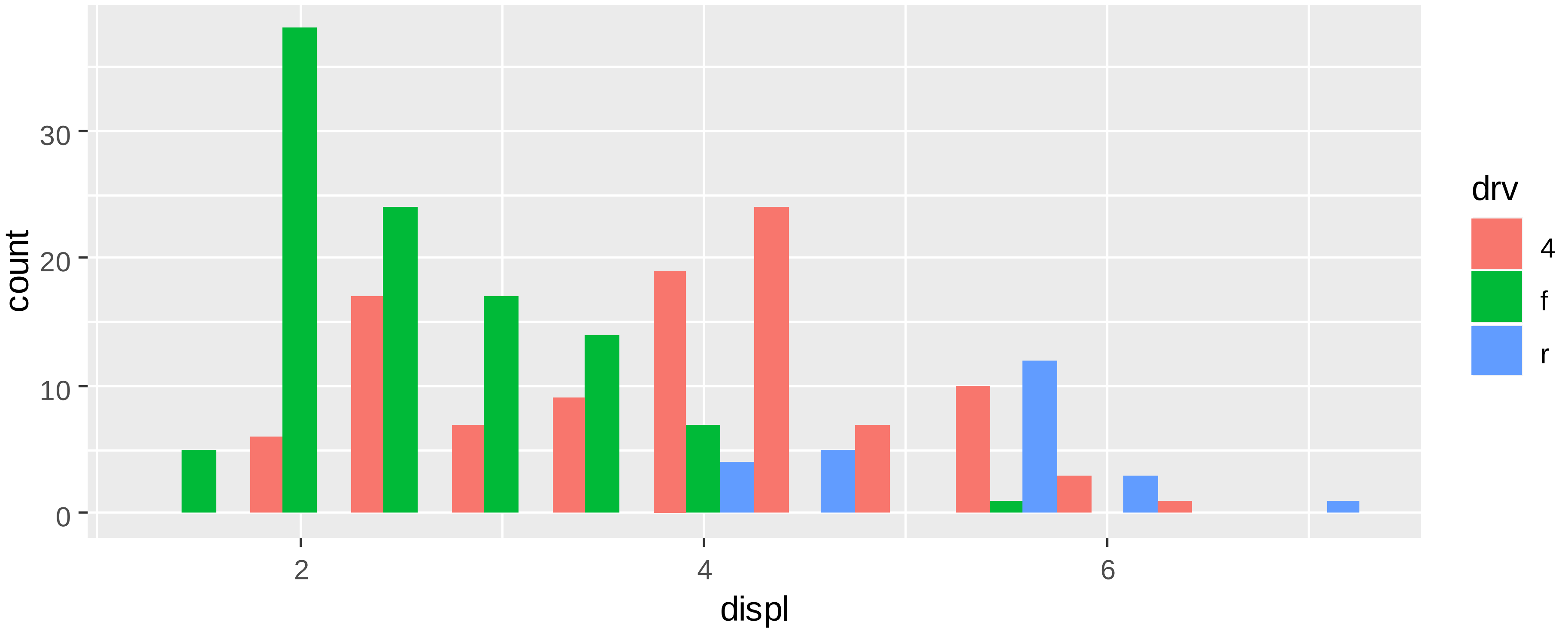
```
ggplot(mpg, aes(displ, fill = drv)) +
  geom_histogram(binwidth = 0.5)
```



geom_histogram( )

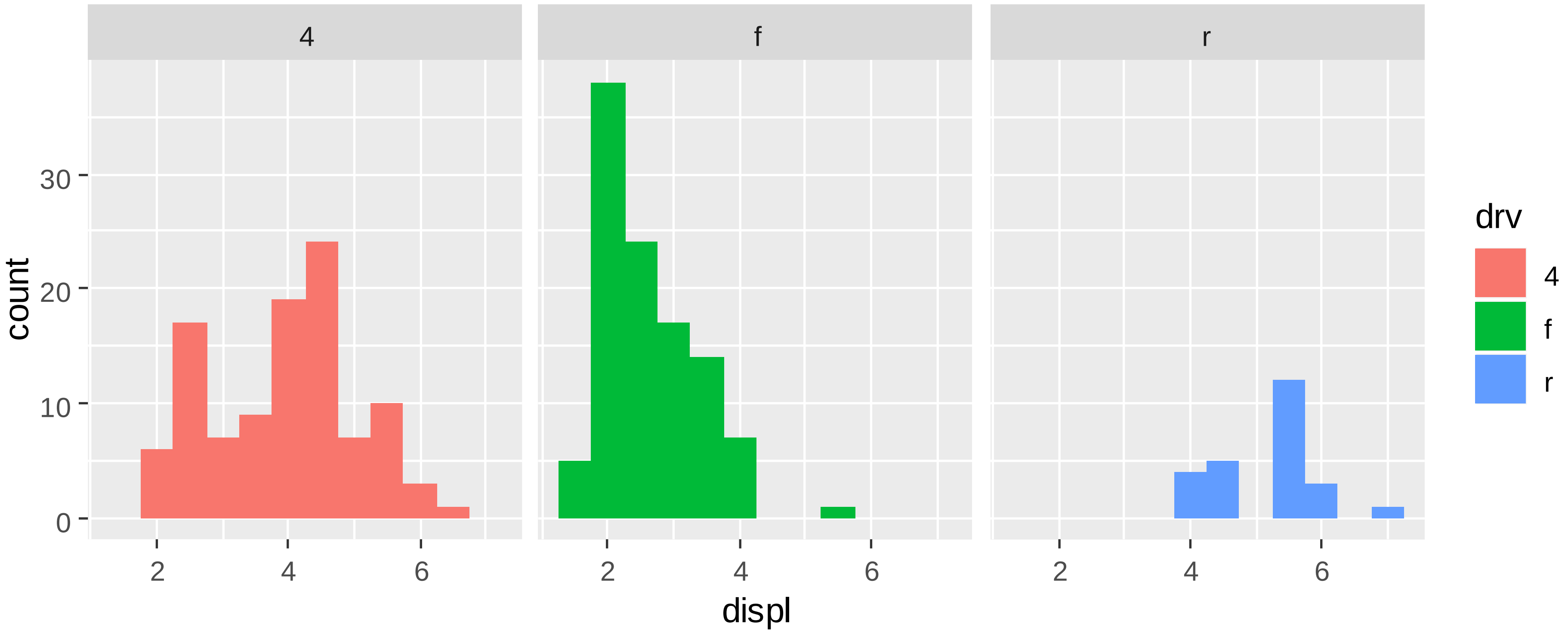


```
ggplot(mpg, aes(displ, fill = drv)) +
  geom_histogram(binwidth = 0.5, position = "dodge")
```



geom_histogram( )

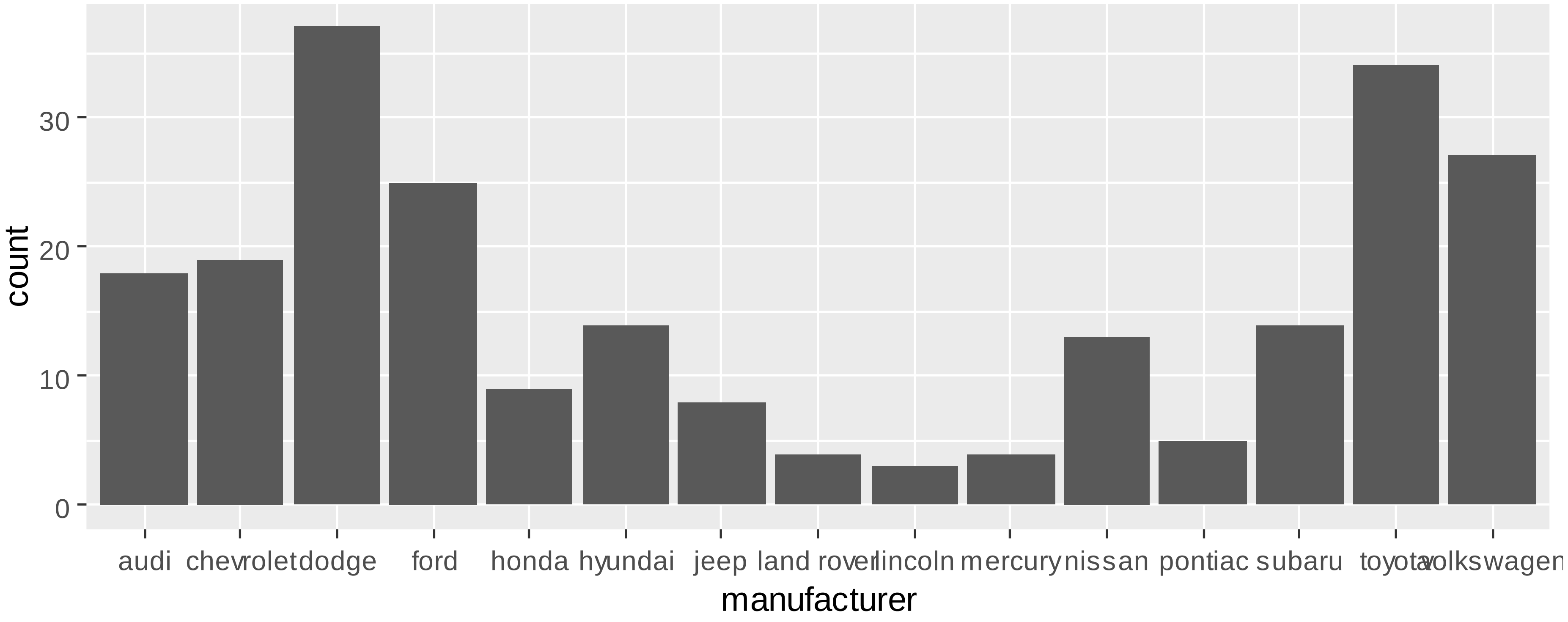
```
ggplot(mpg, aes(displ, fill = drv)) +
  geom_histogram(binwidth = 0.5) +
  facet_wrap(~drv)
```



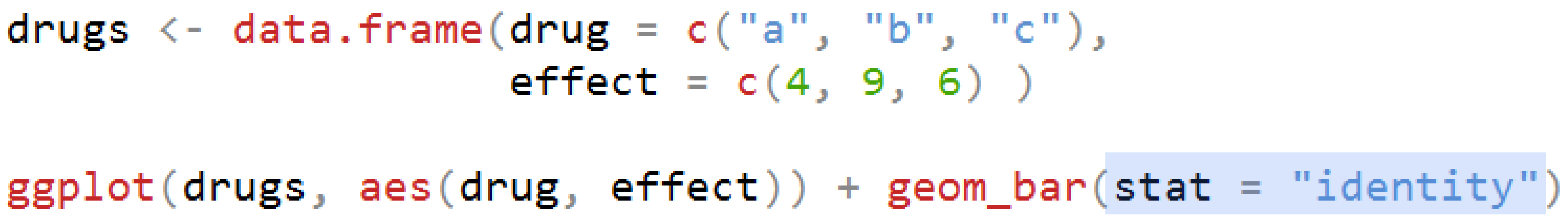
geom_bar( )

```
ggplot(mpg, aes(manufacturer)) +
  geom_bar()
```

- $manufacturer 안에 나오는 제조사 개수와 횟수를 자동count한다. 변수가 1개.

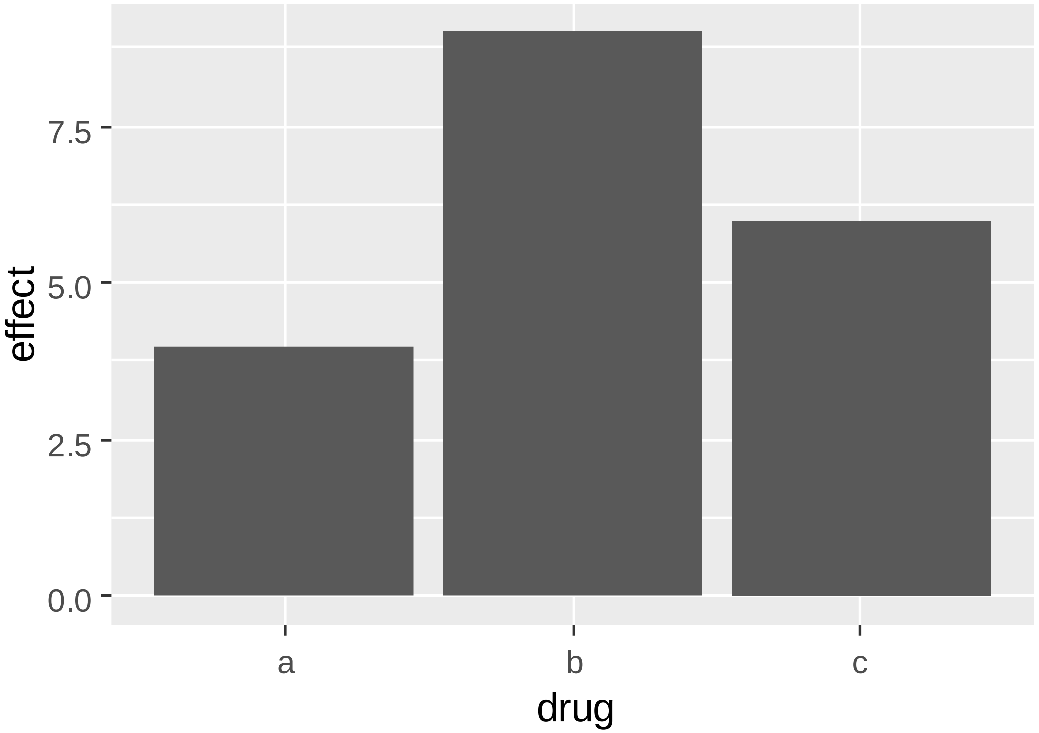


geom_bar( )



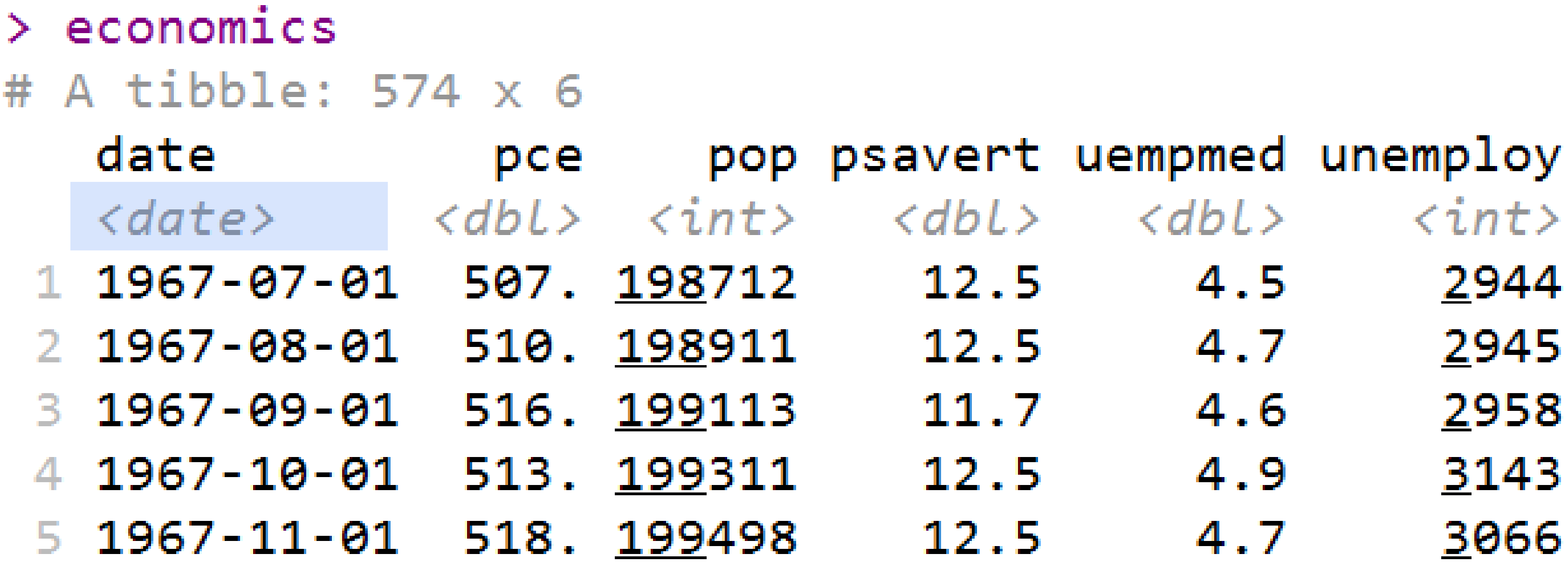
```
drugs <- data.frame(drug = c("a", "b", "c"),
                    effect = c(4, 9, 6) )

ggplot(drugs, aes(drug, effect)) + geom_bar(stat = "identity")
```

- 변수 2개. bins 개수와 회수를 count 하지 않고 그대로 표현한다.



geom_line( ) with Time Series

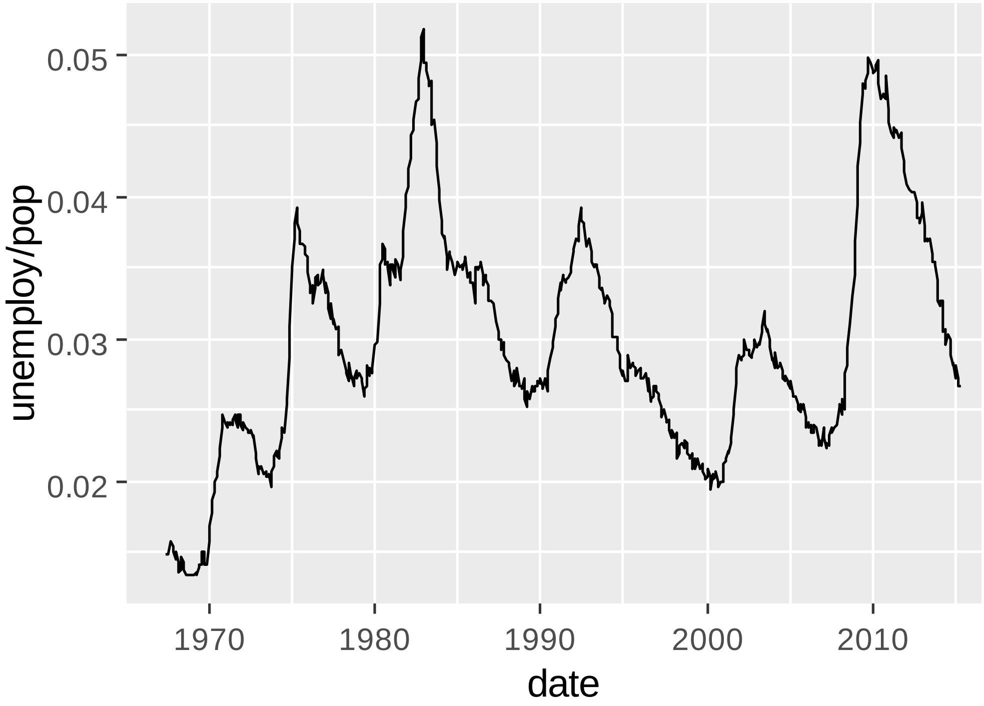
```
ggplot(economics, aes(date, unemploy / pop)) +
  geom_line()
ggplot(economics, aes(date, uempmed)) +
  geom_line()
```

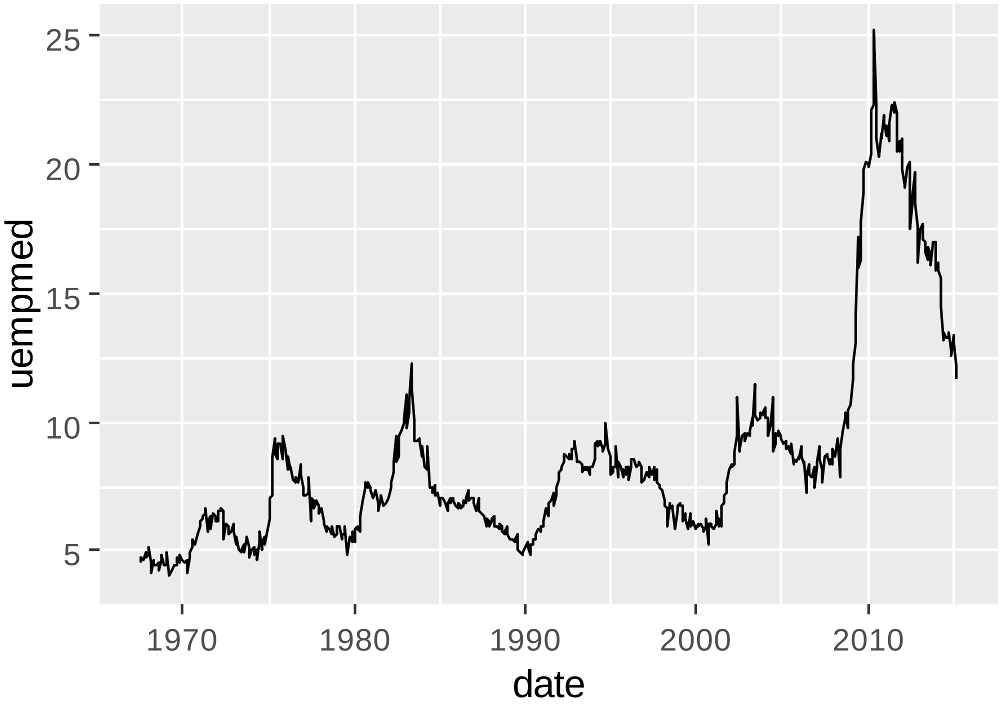


```
> economics
# A tibble: 574 x 6
   date         pce    pop psavert uempmed unemploy
   <date>     <dbl>  <int>   <dbl>   <dbl>    <int>
 1 1967-07-01  507. 198712    12.5     4.5     2944
 2 1967-08-01  510. 198911    12.5     4.7     2945
 3 1967-09-01  516. 199113    11.7     4.6     2958
 4 1967-10-01  513. 199311    12.5     4.9     3143
 5 1967-11-01  518. 199498    12.5     4.7     3066
```

geom_line( ) with Time Series

```
ggplot(economics, aes(date, unemploy / pop)) +
  geom_line()
ggplot(economics, aes(date, uempmed)) +
  geom_line()
```


unemploy/pop
date


uempmed
date