

R10. ggplot (4)

Position and coordinates

Sim, Min Kyu, Ph.D.
mksim@seoultech.ac.kr



서울과학기술대학교 데이터사이언스학과

1 Position adjustment

2 Coordinate

Position adjustment

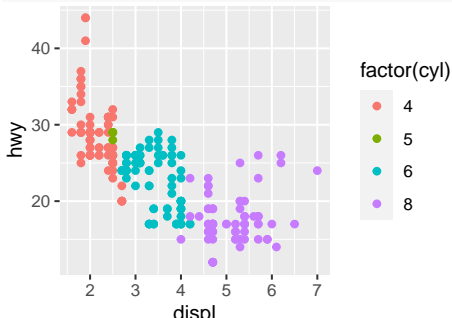
Scatterplot - *position = "jitter"*

- 데이터들이 중첩되는 경우에 노이즈를 넣어주는 기능
 - 왼쪽의 그래프에서는 overplotting 현상이 발생한다.
 - `position="jitter"`로 position에 perturbation을 가한다.
 - `geom_point(position = "jitter")`는 `geom_jitter()`와 같다.

```
fig_a <- ggplot(mpg) + geom_point(aes(displ, hwy, color = factor(cyl)))
```

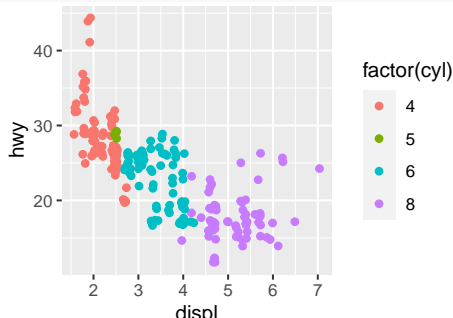
```
fig_b <- ggplot(mpg) + geom_point(aes(displ, hwy, color = factor(cyl)), position = "jitter")
```

fig_a



데이터 시각화

fig_b



R10. ggplot (4)

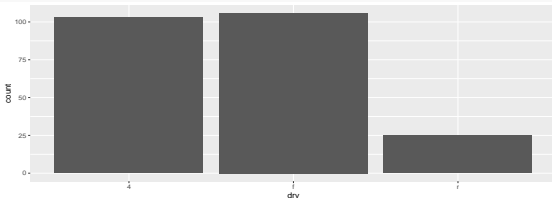
Barchart

- X축은 categorical 변수에 대응되고 Y축은 count 또는 density에 대응된다.
- `geom_bar()`의 aesthetics은 `x`, `alpha`, `color`, `fill`, `linetype`, `size`, `weight`이며 1 변수 혹은 2 변수를 제공하여 그릴수 있다.

1 변수로 그리는 경우

- x변수를 count하여 y의 aesthetic으로 사용한다.
- 즉, `geom_bar()`의 default statistics는 count이다.
- 즉, `geom_bar()`의 인수로 `stat="count"`가 포함되어 있다.

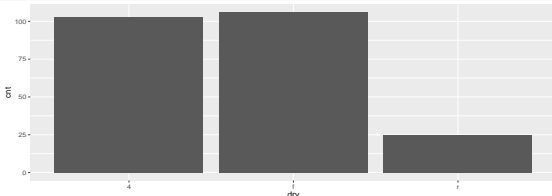
```
fig_c <- ggplot(mpg) + geom_bar(aes(x=drv))  
fig_c
```



2 변수로 그리는 경우

- Frequency table이 있는 상황에서 사용한다.
- y의 aesthetic을 사용하기 때문에 `stat = "identity"`을 명시한다.

```
df <- mpg %>% group_by(drv) %>% summarise(cnt = n())  
fig_d <- ggplot(df) + geom_bar(aes(x=drv, y=cnt), stat = "identity")  
fig_d
```



1 변수 vs 2 변수

1 변수로 그리는 경우

```
fig_c <- ggplot(mpg) + geom_bar(aes(x=drv))
```

2 변수로 그리는 경우

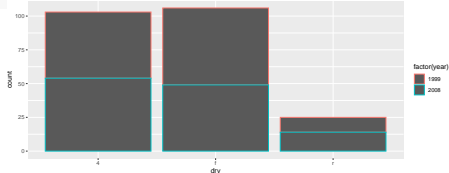
```
fig_d <- ggplot(df) +  
  geom_bar(aes(x=drv, y=cnt), stat = "identity")
```

Barchart에 다른 Categorical 변수 추가

color vs fill

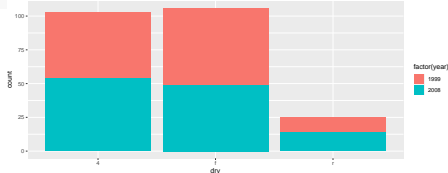
color는 테두리를 결정

```
ggplot(mpg, aes(x=drv)) +  
  geom_bar(aes(color=factor(year)))
```



fill은 색을 채움

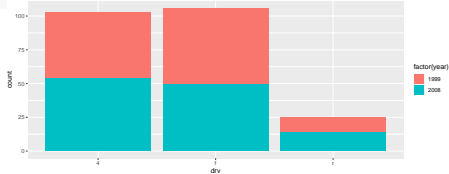
```
ggplot(mpg, aes(x=drv)) +  
  geom_bar(aes(fill=factor(year)))
```



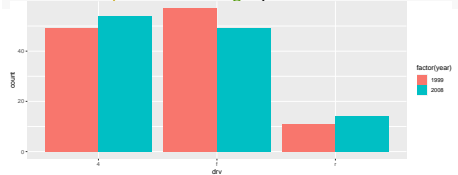
position="dodge"

- `position="dodge"`로 비껴서 나란히 배치한다.

```
ggplot(mpg, aes(x=drv)) +  
  geom_bar(aes(fill=factor(year)))
```



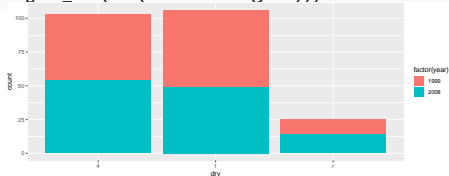
```
ggplot(mpg, aes(x=drv)) +  
  geom_bar(aes(fill=factor(year)),  
           position="dodge")
```



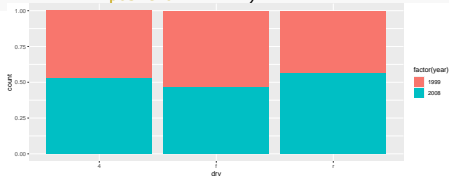
position = "fill"

- `position="fill"`로 세로 포지션을 가득채운다. 갯수가 아닌 비율에 초점을 맞춘다.

```
ggplot(mpg, aes(x=drv)) +  
  geom_bar(aes(fill=factor(year)))
```



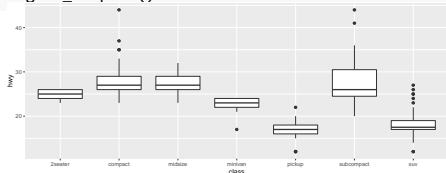
```
ggplot(mpg, aes(x=drv)) +  
  geom_bar(aes(fill=factor(year)),  
           position="fill")
```



geom_boxplot() vs geom_violin()

geom_boxplot()

```
ggplot(mpg, aes(x = class, y = hwy)) +  
  geom_boxplot()
```

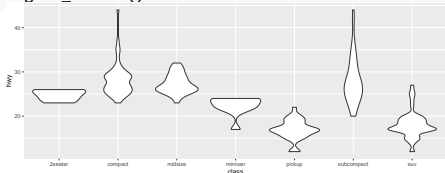


1. 박스의 가운데 직선은 “median”
2. 박스의 상하단은 “hinges”
3. 선의 상하단은 “whiskers”
4. 점으로 표현된 것들은 “outliers”

- 두 차트에서 글자의 크기를 키우면 X축의 공간이 부족하게 될 것이다. 이런 경우에는 어떻게 할까?

geom_violin()

```
ggplot(mpg, aes(x = class, y = hwy)) +  
  geom_violin()
```



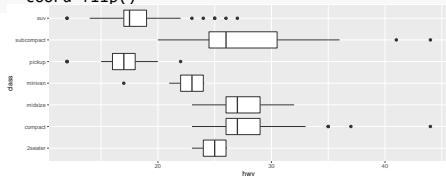
- 바이올린 차트는 density curve를 드러나게 해주는 모던한 차트이다.
- 때로는 boxplot을 대체하여 유용할수도 있다.

Coordinate

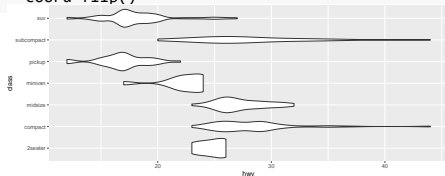
coord_flip()

coord_flip()은 x축과 y축을 바꾸어준다.

```
ggplot(mpg, aes(x = class, y = hwy)) +  
  geom_boxplot() +  
  coord_flip()
```

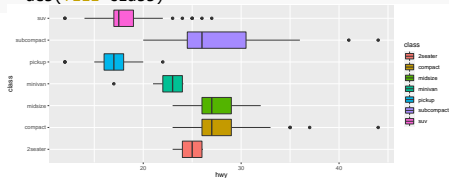


```
ggplot(mpg, aes(x = class, y = hwy)) +  
  geom_violin() +  
  coord_flip()
```

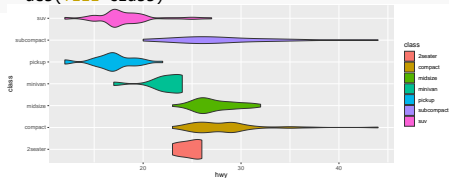


*aes()*를 나중에 더해도 된다.

```
ggplot(mpg, aes(x = class, y = hwy)) +  
  geom_boxplot() +  
  coord_flip() +  
  aes(fill=class)
```

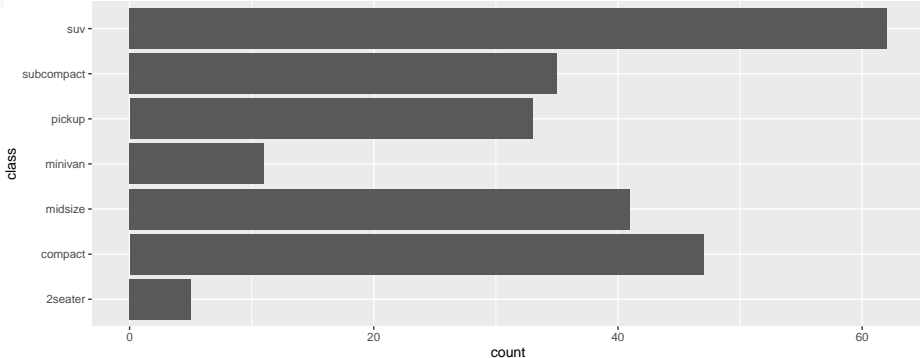


```
ggplot(mpg, aes(x = class, y = hwy)) +  
  geom_violin() +  
  coord_flip() +  
  aes(fill=class)
```



Barchart에서도 때로는 *coord_flip()*이 유용하다.

```
ggplot(mpg, aes(x = class)) + geom_bar() + coord_flip()
```



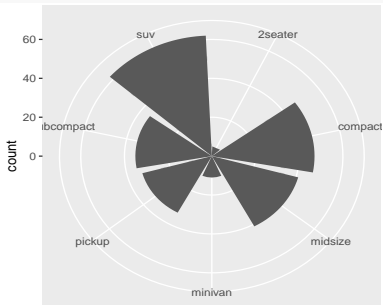
- 파이차트는 어떻게 만들어야 할까? 바차트와 파이차트의 차이는 무엇인가??

`coord_polar()`

- 직각좌표계가 아닌 극좌표계를 사용한다.
- 파이차트를 만드는 과정은 아래와 같다.

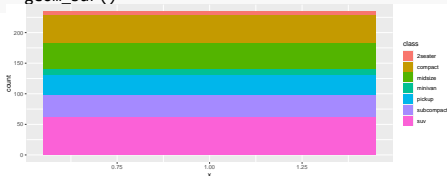
1. `coord_polar()`에서 aesthetic인 `theta`가 일반적인 의미의 y축에 대응된다.

```
ggplot(mpg, aes(x = class)) +  
  geom_bar() + coord_polar()
```



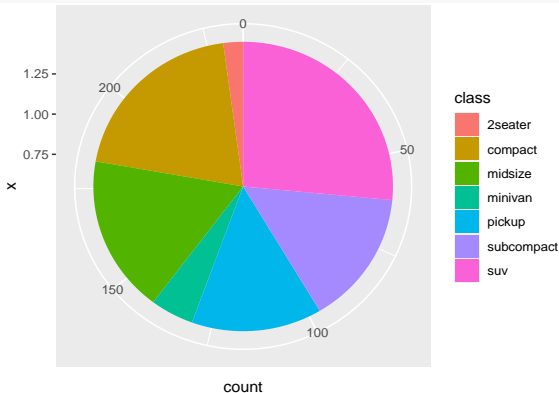
2. 그러므로 아래의 그림을 먼저 그린다.

```
ggplot(mpg, aes(x = 1, fill = class)) +  
  geom_bar()
```



3. Finally,

```
ggplot(mpg, aes(x = 1, fill = class)) +  
  geom_bar() + coord_polar(theta = "y")
```



- axis를 지우면 파이차트가 완성된다.
- 막대그래프와 파이차트의 유일한 차이는 좌표계이다.
- Wickham은 파이차트가 일반적으로 poor한 choice라고 주장한다. Why?


```
"Tantum videmus quantum scimus."
```

```
## [1] "Tantum videmus quantum scimus."
```