

R08. ggplot (2)

Grammar of Graphics

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



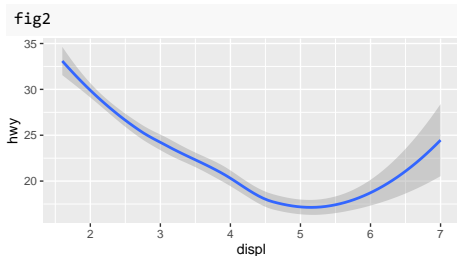
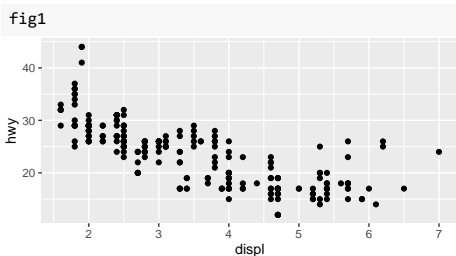
서울과학기술대학교 데이터사이언스학과

- 1 Geometric Objects
- 2 Grammar of Graphics
- 3 Miscellaneous Tips

Geometric Objects

geom_point() vs *geom_smooth()*

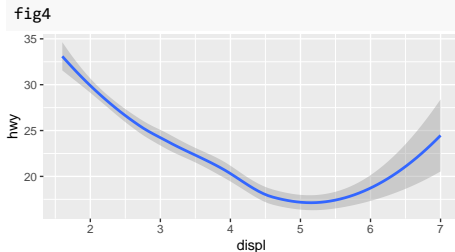
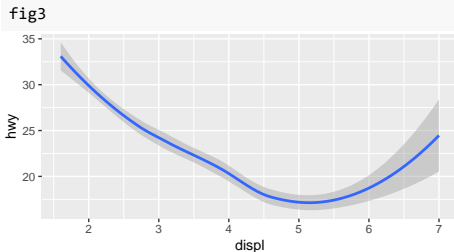
```
fig1 <- ggplot(data = mpg) + geom_point(mapping = aes(x = displ, y = hwy))  
fig2 <- ggplot(data = mpg) + geom_smooth(mapping = aes(x = displ, y = hwy))
```



- *geom_point()*: 점으로 표현
- *geom_smooth()*: Smoothing method를 사용하여 curve 생성
 - 다양한 smoothing method 적용 가능
 - standard error (se) 옵션으로 밴드 생성 가능

mapping object의 상속

```
fig3 <- ggplot(data = mpg) +  
  geom_smooth(mapping = aes(x = displ, y = hwy))  
fig4 <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
  geom_smooth()
```



- fig4에서 ggplot()에 있는 mapping 객체가 geom_smooth()의 인수로 상속되었음
- geom_XXX()에 mapping이 없다면 앞에 있는 ggplot()에서 상속받는다.

geom_point()와 geom_smooth()를 함께 사용

```
fig5 <- ggplot(data = mpg) +  
  geom_point(mapping = aes(x = displ, y = hwy)) +  
  geom_smooth(mapping = aes(x = displ, y = hwy))  
fig6 <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
  geom_point() + geom_smooth()
```

fig5

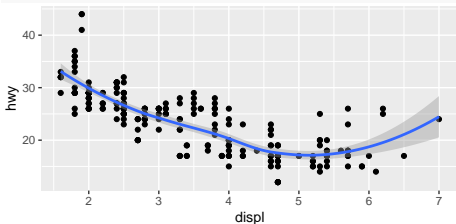
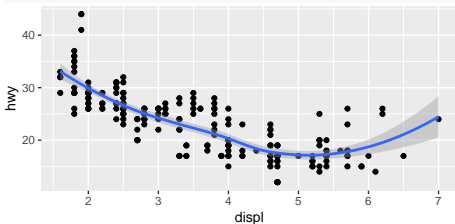


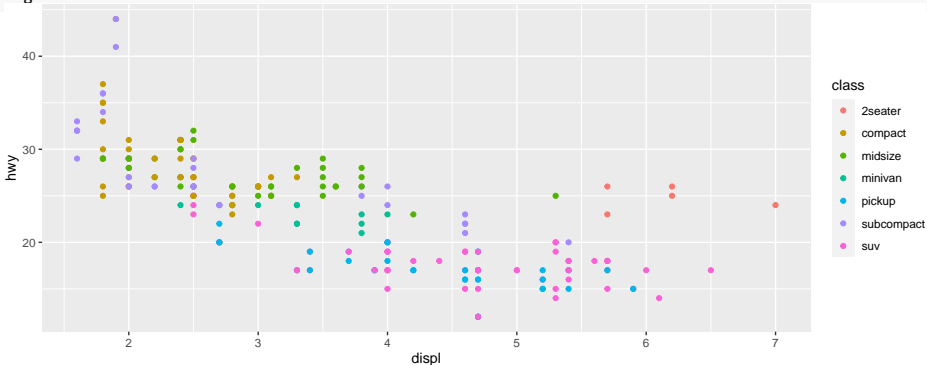
fig6



- fig5와 fig6는 동일한 vis 이지만,
- fig6에서는 ggplot() 함수에서 mapping을 정하고 geom_xxx()에서는 이를 사용함
 - 코드가 간결해짐
 - geom_xxx()를 간단하게 추가할 수 있음

Geometric object에 따라서 서로 종류의 *mapping*을 가질수 있음 (1)

```
fig7 <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
  geom_point(mapping = aes(color = class))  
fig7
```



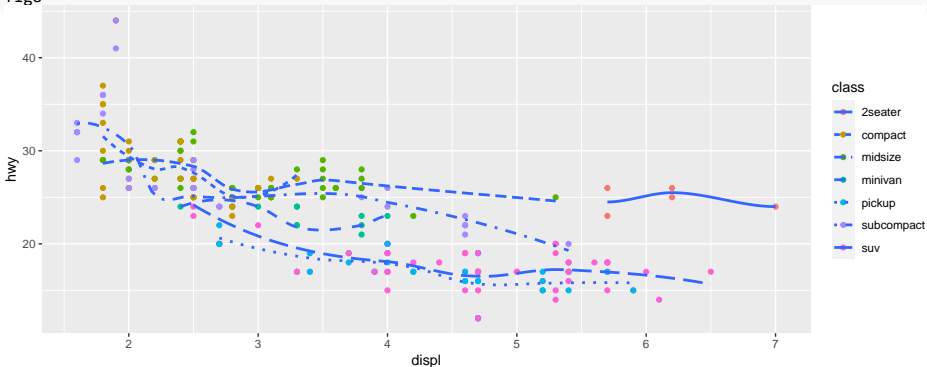
● geom_point()의 mapping

- 부모(ggplot())로 부터 상속받은 mapping = aes(x = displ, y = hwy)
- 스스로 가지고 있는 mapping = aes(color=class)

Geometric object에 따라서 서로 종류의 *mapping*을 가질수 있음 (2)

```
fig8 <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
  geom_point(aes(color=class)) +  
  geom_smooth(aes(linetype = class), se = FALSE)
```

fig8



- `geom_point()`의 mapping: `aes(x = displ, y = hwy, color=class)`
- `geom_smooth()`의 mapping: `aes(x = displ, y = hwy, linetype=class)`

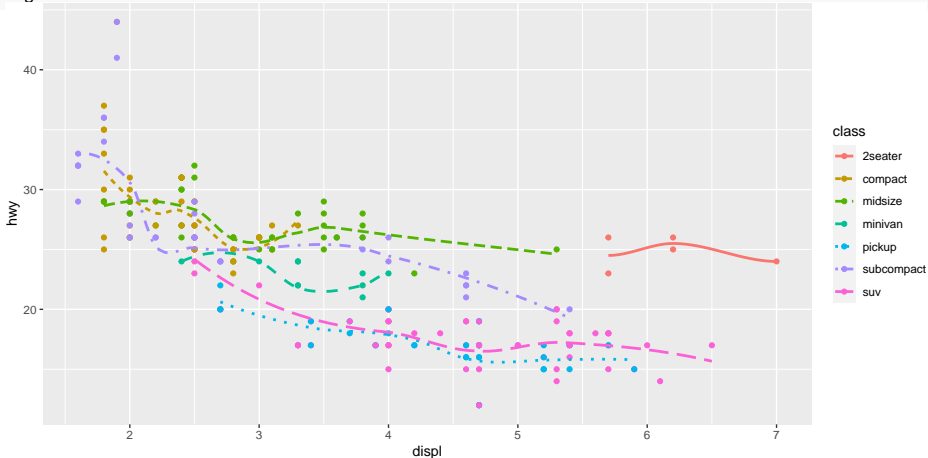

```
fig8 <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
  geom_point(aes(color = class)) +  
  geom_smooth(aes(linetype = class), se = FALSE)
```

- linetype=class는 geom_point()에 포함될 수 없다.
- color=class는 geom_smooth()에 포함될 수 있으며, 이 경우가 더 좋다.
- 그러므로 fig8은 아래의 fig9처럼 바뀌어야 한다.

```
fig9 <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy, color=class)) +  
  geom_point() +  
  geom_smooth(aes(linetype = class), se = FALSE)
```

Perhaps the best one so far?

```
fig9 <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy, color=class)) +  
  geom_point() +  
  geom_smooth(aes(linetype = class), se = FALSE)  
fig9
```



Exercise

- 앞의 fig9의 그래프를 3요소(what-why-how)의 요인으로 분석하라.
- `ggplot()`의 `labs()` 함수를 활용하여 제목, 부제목, x축 레이블, y축 레이블을 추가한 그래프를 그려라.
- color legend의 위치를 우측에서 상단으로 옮겨라.

Grammar of Graphics

“gg” in “ggplot” stands for Grammar of Graphics

- Motivation
 - 그래픽에 관련된 함수와 패키지는 단지 특수경우의 모음일 뿐일까?
 - 그렇지 않게 하려면 **체계적인 문법**이 존재해야 한다.
- Requirement to be “grammar”
 - 주어-동사-목적어의 문법처럼 체계와 어울림이 있어야 한다.
 - 필요에 따라서 요소들을 더하고 뺄 수 있어야 한다.
 - 생각의 흐름, 이야기의 흐름을 반영하는 그래픽의 변형이 간단한 코드로 구현되어야 한다.

단계적인 구현과 전달의 예시

```
fig <- ggplot(data = Carseats, aes(x = Advertising, y = Sales)) +  
  geom_point(aes(color = Urban))
```

fig

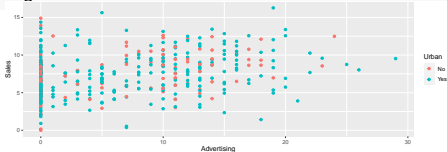


fig + facet_wrap(~ Urban)

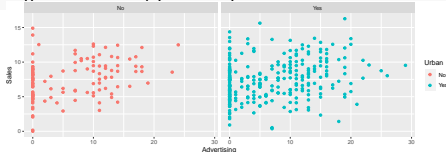
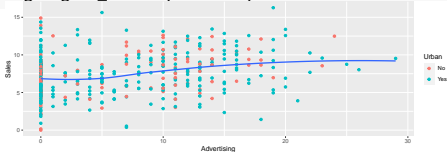
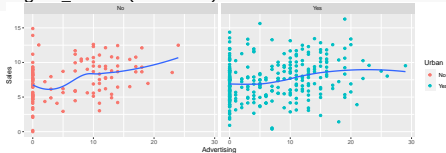


fig + geom_smooth(se=FALSE)

fig + facet_wrap(~ Urban) +
geom_smooth(se=FALSE)

- (Bottom to Top) (geom_smooth()) 추상화 하며 내용 전달
- (Top to Bottom) (facet_wrap()) 분할하여 내용 전달

Interactive flow

- The interactive flow for the above example
 - https://acemk.shinyapps.io/S01_ggplot2_shiny/
- Will be cover in the second half of this course
 - `flexdashboard`
 - `shiny`

ggplot2의 문법 구조

- Skeleton of the syntax

```
ggplot(data = <DATA>) + <GEOM_FUNCTION>(mapping = aes(<MAPPING>))
```

- A minimal example

```
ggplot(data = mpg,  
       mapping = aes(x = displ, y = hwy, color = factor(cyl))) +  
  geom_point()
```

- Data: tidy dataset (각 컬럼이 변수를 의미)
- Aesthetics mapping
 - aes() - 각 변수를 aesthetic attribute에 어떻게 mapping할지 결정
 - x, y, position, size, color, shape, linetype, ...
- Geometric object
 - 차트의 종류가 결정됨 (산점도, 막대그래프 등)
 - 실제로 플랏에 보이게 될 것이 결정됨 (points, lines, polygons, and etc.)

Behind the scene

```
ggplot(data = mpg,  
       mapping = aes(x = displ, y = hwy, color = factor(cyl))) +  
  geom_point()
```

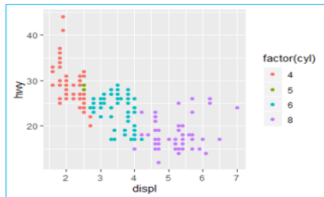
data = mpg

```
> mpg[,c("displ", "hwy", "cyl")] %>% head(8)  
# A tibble: 8 x 3  
  displ hwy  cyl  
  <dbl> <int> <int>  
1  1.8   29    4  
2  1.8   29    4  
3  2     31    4  
4  2     30    4  
5  2.8   26    6  
6  2.8   26    6  
7  3.1   27    6  
8  1.8   26    4
```

mapping = aes(x = displ, y = hwy, color = factor(cyl))

x	y	colour
1.8	29	4
1.8	29	4
2.0	31	4
2.0	30	4
2.8	26	6
2.8	26	6
3.1	27	6
1.8	26	4

geom_point()



x	y	colour	size	shape
0.037	0.531	#F8766D	1	19
0.037	0.531	#F8766D	1	19
0.074	0.594	#F8766D	1	19
0.074	0.562	#F8766D	1	19
0.222	0.438	#00BFC4	1	19
0.222	0.438	#00BFC4	1	19
0.278	0.469	#00BFC4	1	19
0.037	0.438	#F8766D	1	19

Miscellaneous Tips

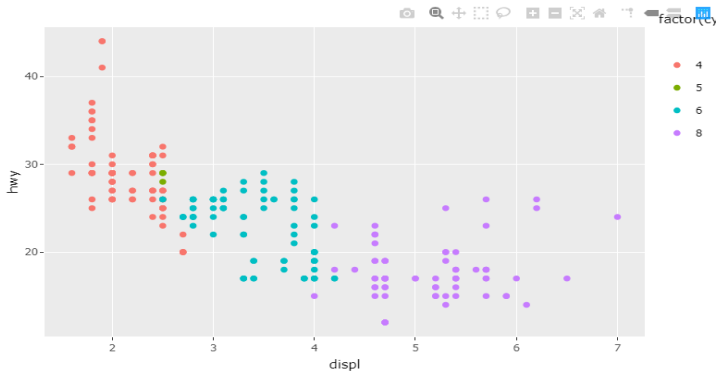
1. Export to a image file

- .eps, .png, .pdf, .tiff 등의 그림파일로 저장 가능
- dpi: dot per inch, 300 추천
- Aspect ratio: 일반적으로 알려진 황금비율(0.612)을 고려하여 width와 height를 설정

```
ggsave(filename = "fig.png", plot = fig1, dpi = 300, width = 8, height = 5)
```

2. Transform to interactive graphic object of *htmlwidget*

```
library(plotly)  
ggplotly(fig)
```



3. Place multiple plots in a grid structure (*gridExtra::grid.arrange()*)

- 이 강의노트의 4페이지에 해당하는 Rmd 코드는 아래와 같음

```
##  
### `geom_point()` vs `geom_smooth()`  
```\r  
fig1 <- ggplot(data = mpg) + geom_point(mapping = aes(x = displ, y = hwy))
fig2 <- ggplot(data = mpg) + geom_smooth(mapping = aes(x = displ, y = hwy))
```\br/>  
\lc  
```\r, eval=FALSE}  
fig1
```\br/>  
\rc  
```\r, eval=FALSE}  
fig2
```\br/>  
\ec  
```\r, echo=FALSE}  
gridExtra::grid.arrange(fig1, fig2, nrow = 1, ncol = 2)
```\br/>  
+ `geom_point()`: 점으로 표현  
+ `geom_smooth()`: Smoothing method를 사용하여 curve 생성  
  + 다양한 smoothing method 적용 가능  
  + standard error (`se`) 옵션으로 밴드 생성 가능|
```

```
a <- "The simple graph has brought more information to the data analyst's mind than any other device
- John Tukey"
```

```
a
```

```
## [1] "The simple graph has brought more information to the data analyst's mind than any other device \n-
John Tukey"
```

```
cat(a)
```

```
## The simple graph has brought more information to the data analyst's mind than any other device
## - John Tukey
```