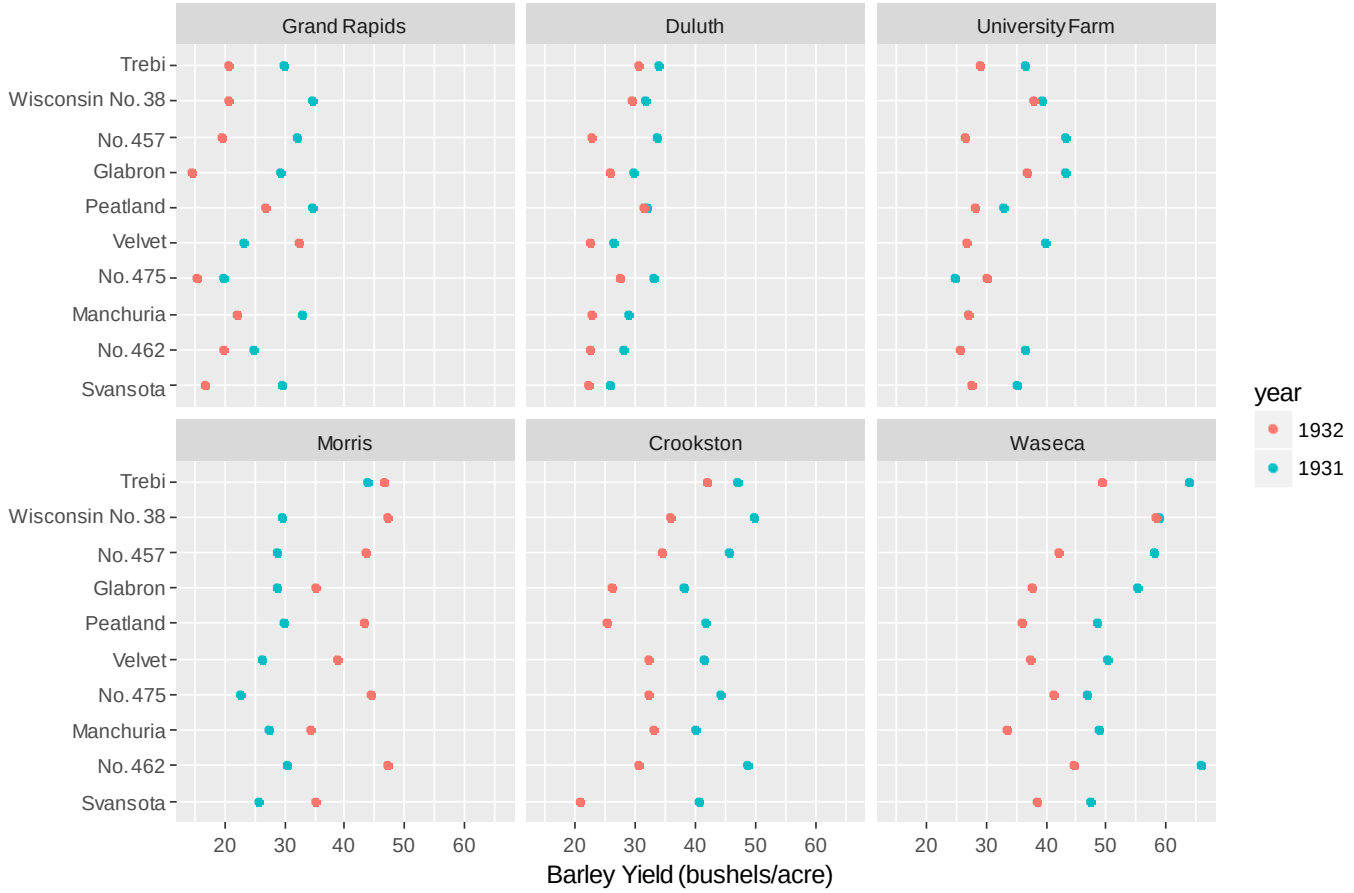
자료의 시각화

박영식(youngsik.park@bsl-lausanne.ch)

그래프 기법의 활용 예제: 보리 자료

- 자료분석과정에서 그래프의 이용이 필수적임을 보여주는 예제
- 1930년대 초 미네소타 주 농경학자들이 보리종류에 따른 수확량의 차이를 비교하기 위해 2년간 경작실험을 실시
- 요인: 6군데 경작지, 10종류의 보리, 2년간의 경작 년도
- 반응변수: 수확량
- 실험계획법에 대한 Fisher의 아이디어가 적용된 최초의 실험자료
- 저명한 학자들에 의해서 여러 번 분석된 자료
- Cleveland가 자료에 있는 문제 발견

보리자료의 문제점


Grand Rapids
Duluth
University Farm
Morris
Crookston
Waseca
Trebi
Wisconsin No. 38
No. 457
Glabron
Peatland
Velvet
No. 475
Manchuria
No. 462
Svansota
20
30
40
50
60
Barley Yield (bushels/acre)
year
1932
1931

그래프의 위력

- 과거 저명한 학자들이 보리자료에 있는 문제점을 파악하지 못한 이유는 그 시대에 명쾌한 그래프가 아직 개발되지 않았기 때문

- 효과적인 분석도구로서 그래픽 기법의 우수성 인식

- Big Data 시대에서 그래픽 기법의 중요성은 더 강조될 것임

- ggplot2: 매우 효과적인 그래프 작성 가능

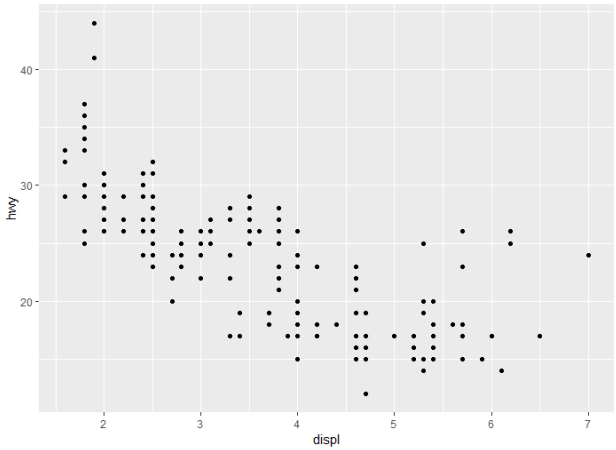
ggplot 2

박영식(youngsikrex@naver.com)

1. ggplot2 시작하기

패키지 ggplot2에 있는 데이터 프레임 mpg의 변수 displ과 hwy의 산점도 작성

```
> library(tidyverse)
  ggplot(data=mpg) + geom_point(mapping=aes(x=displ,
   y=hwy))
```



- 함수 ggplot( ): 데이터 프레임 data 지정. 그래프가 작성될 비어있는 좌표계 작성.
- 함수 geom_point( ): 실질적인 그래프, 레이어(layer)를 작성하는 geom 함수 중 하나
- mapping: geom 함수 내에서 함수 aes( )와 함께 데이터와 시각적 요소를 서로 연결

ggplot2에서 그래프 작성의 최소 요소

- 그래프 작성을 위한 법칙이 있음: 그래프의 문법
- 모든 그래프 작성에 일정하게 적용
- 익숙해지면 복잡한 형태의 그래프도 어렵지 않게 작성 가능

- 그래프 작성을 위한 3가지 최소 요소: <Data>, <Geom_function>, <Mappings>

```
ggplot(data=<Data>) +
                <Geom_function>(mapping=aes(<Mappings>)
```

<Data>: 그래프 작성에 사용될 데이터 프레임 지정

<Geom_function>: geom 함수 중 하나. 레이어(layer) 작성. 여러 개의 레이어를 겹치기 위해서는 여러 개의 geom 함수를 덧셈 기호로 연결

<Mappings>: 시각적 요소(점의 크기, 모양, 색깔, ...)와 데이터 연결

2. 시각적 요소와 연결: Mapping

시각적 요소

- 그래프를 시각적으로 인식할 때 필요한 요소

- 산점도의 경우, 점의 위치, 크기, 모양 및 색깔 등이 시각적 요소

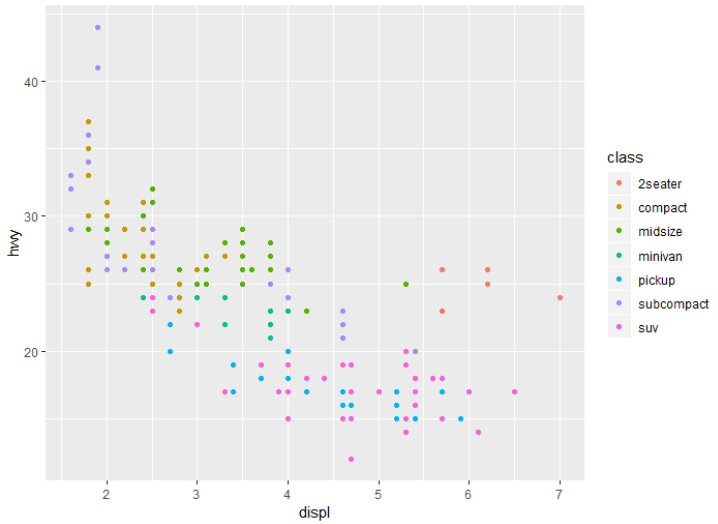
- 시각적 요소의 mapping과 setting
  - mapping: 데이터의 값과 연결되어 결정. 함수 aes( ) 안에서 연결
  - setting: 사용자가 일정한 값을 지정

- 시각적 요소의 mapping
  - 기존의 그래프에 다른 변수의 정보 추가 가능

예제: 데이터 프레임 mpg의 변수 displ과 hwy의 산점도에 시각적 요소와의 mapping으로 다른 변수 정보 추가

- 변수 class를 시각적 요소 color와 mapping

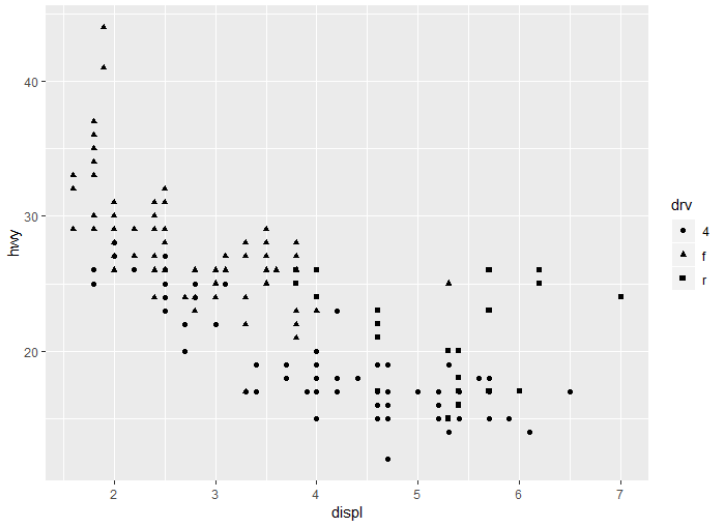
```
> ggplot(data=mpg) +
    geom_point(mapping=aes(x=displ, y=hwy, color=class))
```



- 변수 class: 문자형 벡터
- 변수 class의 값에 따라 다른 색 사용
- 사용된 색깔에 대한 범례는 자동으로 추가

– 변수 drv를 시각적 요소 shape와 mapping

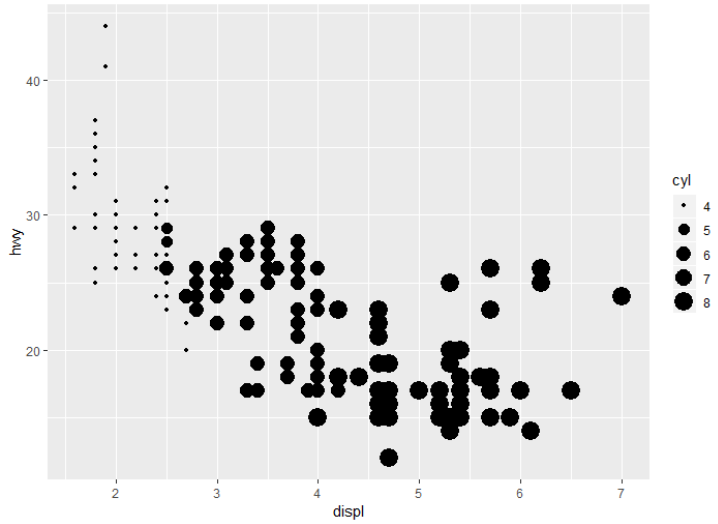
```
> ggplot(data=mpg) +
    geom_point(mapping=aes(x=displ, y=hwy, shape=drv))
```



- shape에 mapping되는 변수는 이산형
- 변수 drv: 문자형 벡터
- 변수 drv의 값에 따라 다른 모양의 점 사용
- 범례 자동 추가

– 변수 cyl을 시각적 요소 shape와 mapping

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

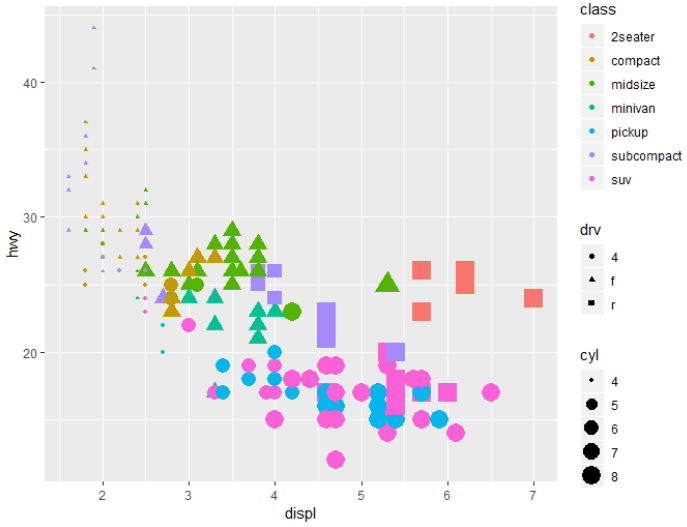


- size에 mapping되는 변수는 연속형이 좋음
- 변수 cyl: 정수형 변수
- cyl의 값에 따라 점의 크기 조절
- 범례 자동 추가

여러 시각적 요소를 동시에 mapping

- 변수 class는 color와, drv는 shape와, cyl은 size와 mapping

```
>   ggplot(data=mpg) + geom_point(mapping=aes(x=displ,
    y=hwy, color=class, shape=drv, size=cyl))
```



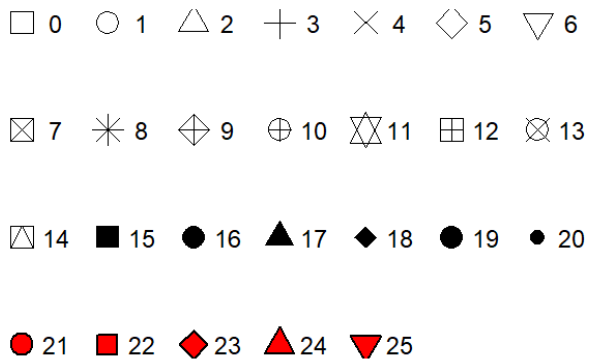
- 너무 많은 정보
- 그래프의 의미가 모호

시각적 요소의 setting

- 함수 aes( ) 밖에서 사용자가 원하는 값으로 지정
- geom 함수의 입력 요소가 됨

- 시각적 요소 color, size, shape에 값 지정 법칙

1) color: 색깔을 나타내는 문자열 지정
2) size: 점 크기를 mm 단위로 지정
3) shape: 점의 형태를 나타내는 0~26 사이의 숫자

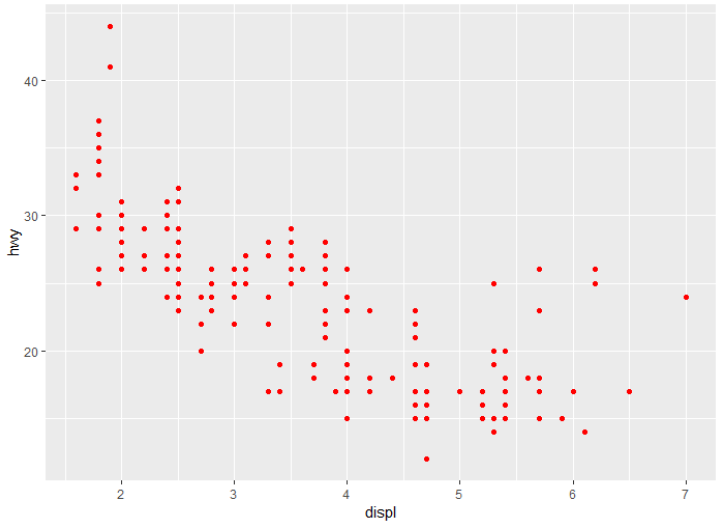


도형에 색깔 지정 방법
1) 0~14의 외곽선 및 15~20의 도형 색: color 사용
2) 21~25의 외곽선: color 사용
3) 21~25의 내부 색: fill 사용

https://www.r-graph-gallery.com/colors

시각적 요소 color의 setting: 모든 점을 빨간 색으로

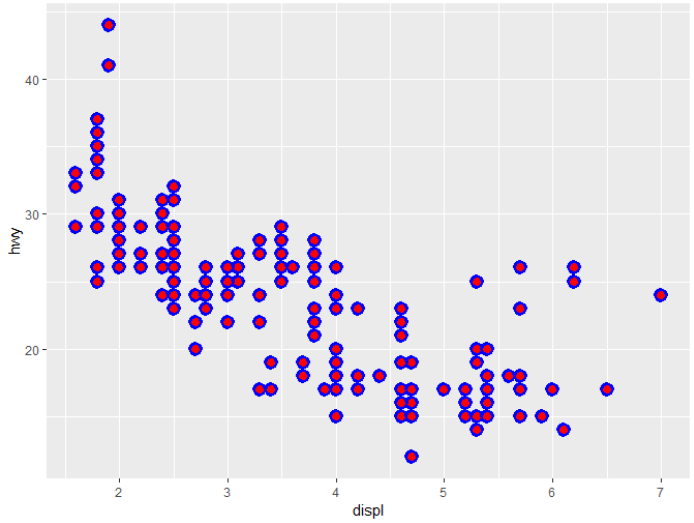
```
> ggplot(data=mpg) +
    geom_point(mapping=aes(x=displ, y=hwy), color="red")
```



- color를 함수 aes( ) 밖에서 지정
- 함수 geom_point( )의 입력 요소

여러 시각적 요소를 동시에 setting

```
> ggplot(data=mpg) +
    geom_point(mapping=aes(x=displ, y=hwy), color="blue",
                  size=3, shape=21, fill="red", stroke=2)
```



- 점의 모양: shape=21
- 점의 내부 색: 빨간색
- 점의 외곽선 색: 파란색
- 점의 크기 확대: size=3
- 점의 외곽선 두께 조절: stroke=2

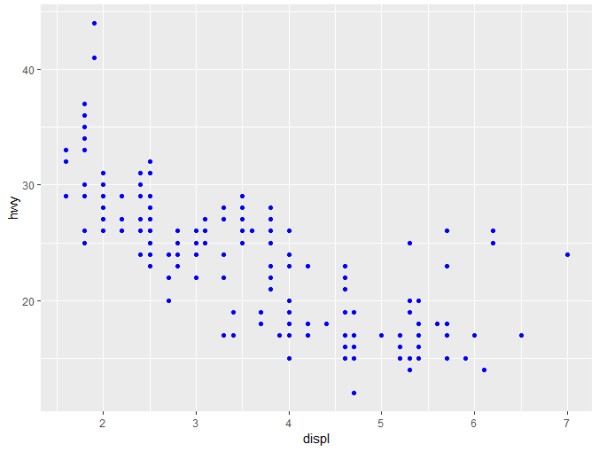
함수 aes( ) 안에서 시각적 요소에 특정 값을 setting한 경우의 결과

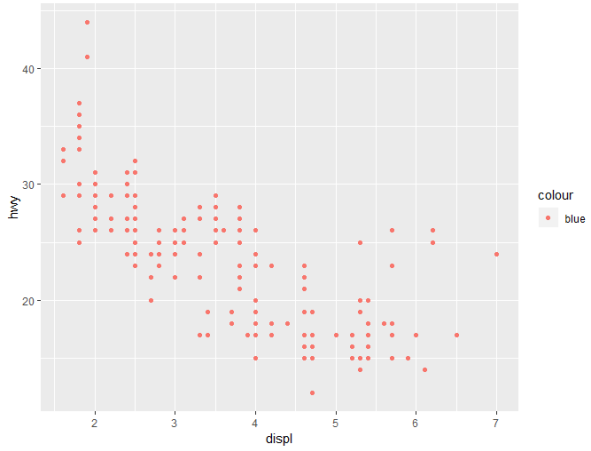
- Setting

```
> ggplot(data=mpg)+
    geom_point(mapping=aes(x=displ, y=hwy), color="blue")
```

- Mapping

```
> ggplot(data=mpg)+
    geom_point(mapping=aes(x=displ, y=hwy, color="blue"))
```





- mapping은 변수와의 연결을 의미
- "blue"라는 값을 갖는 변수 생성

3. 그룹별 그래프 작성: Facet

범주형 변수가 다른 변수에 미치는 영향력을 그래프로 확인하는 방법

1) 시각적 요소에 범주형 변수를 mapping
2) 범주형 변수로 그룹 구성하고, 각 그룹별 그래프 작성: faceting

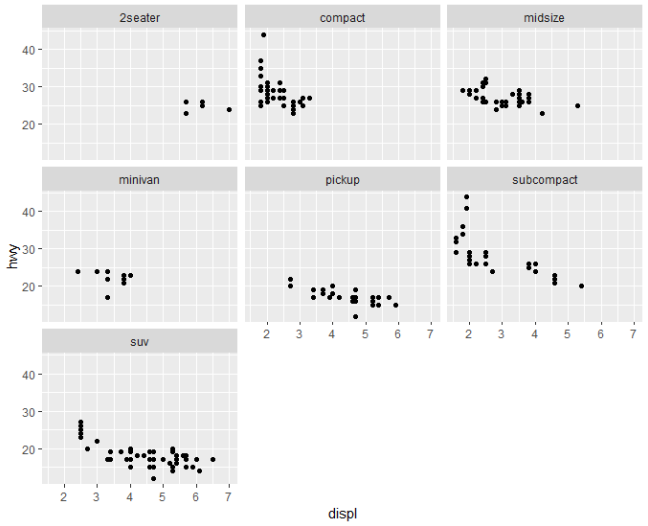
- facet을 적용하기 위한 함수
  ① facet_wrap( ): 한 변수에 의한 facet
  ② facet_grid( ): 한 변수 또는 두 변수에 의한 facet

함수 facet_wrap( )에 의한 faceting

- 데이터를 구분하는 변수가 하나인 경우: facet_wrap(~ x)

- 데이터 프레임 mpg의 변수 displ과 hwy의 산점도를 class의 범주별로 작성

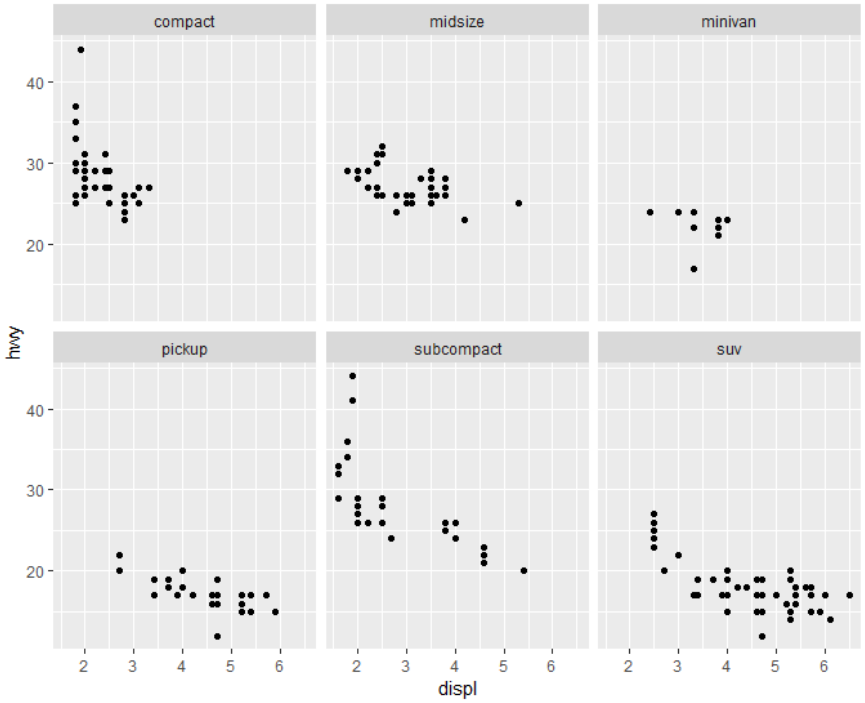
```
> ggplot(data=mpg) + geom_point(mapping=aes(x=displ,
  y=hwy)) + facet_wrap(~ class)
```



- 패널 '2seater'에는 적은 수의 데이터 존재
- class가 '2seater'인 케이스 제거 후 다시 작성

데이터 프레임 mpg의 변수 displ과 hwy의 산점도를 class의 범주별로 작성 (2seater 케이스 제외)

```
> mpg %>%
    filter(class != "2seater") %>%
    ggplot() +
    geom_point(mapping=aes(x=displ,y=hwy))+
    facet_wrap(~ class)
```

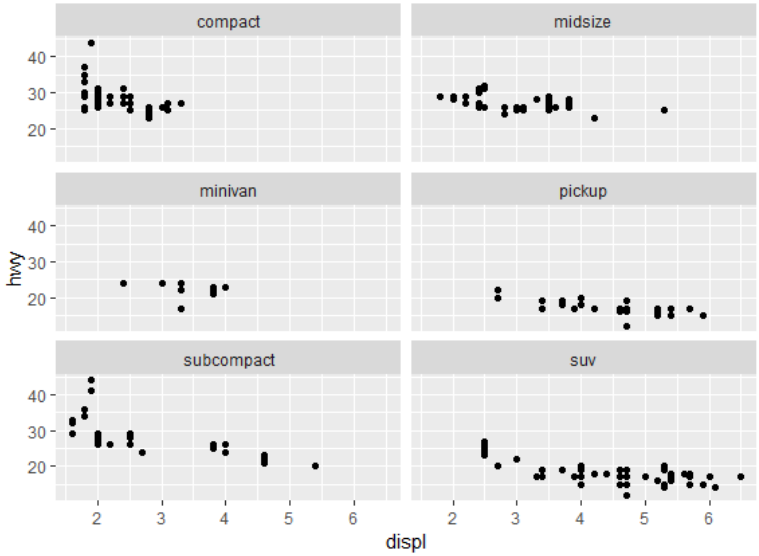


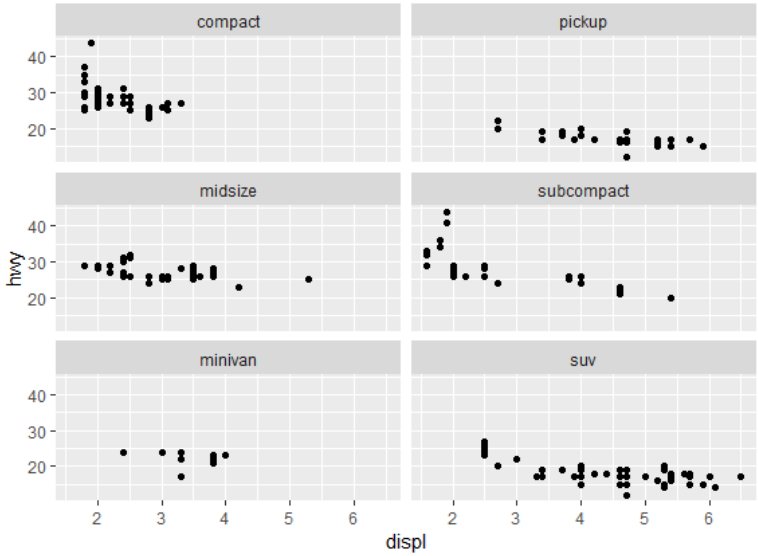
패널 배치 조절

- 2×3 패널 패치를 3×2 배치로 수정: ncol=2
- 패널에 그래프 배치 순서를 열 단위로 수정: dir="v"

```
> pp <- mpg %>%
    filter(class != "2seater") %>%  ggplot() +
    geom_point(mapping=aes(x=displ, y=hwy))
```

```
> pp + facet_wrap(~ class, ncol=2)
> pp + facet_wrap(~ class, ncol=2, dir="V")
```





함수 facet_grid( )에 의한 faceting

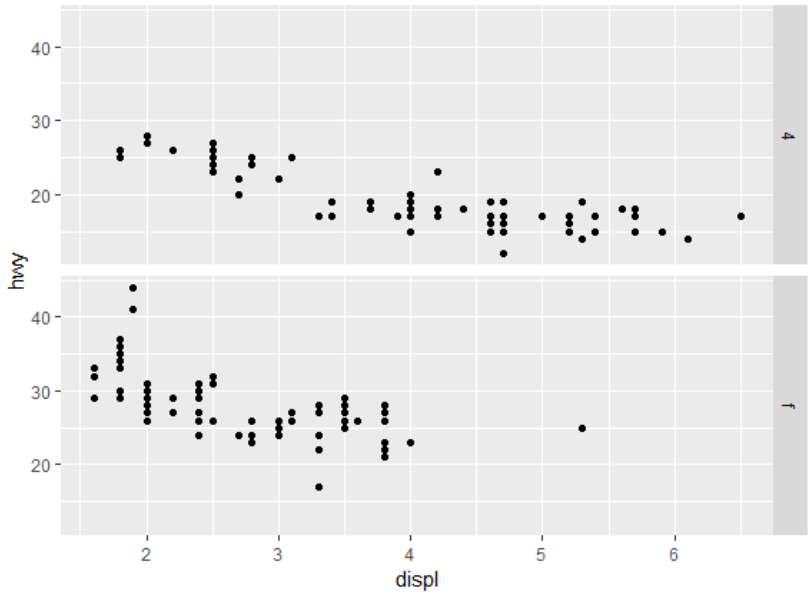
- 한 변수에 의한 faceting:
  하나의 행으로 패널 배치: `facet_grid(. ~ x)`
  하나의 열로 패널 배치: `facet_grid(x ~ .)`

- 두 변수에 의한 faceting: `facet_grid(y ~ x)`
  행 범주: 변수 y의 범주
  열 범주: 변수 x의 범주

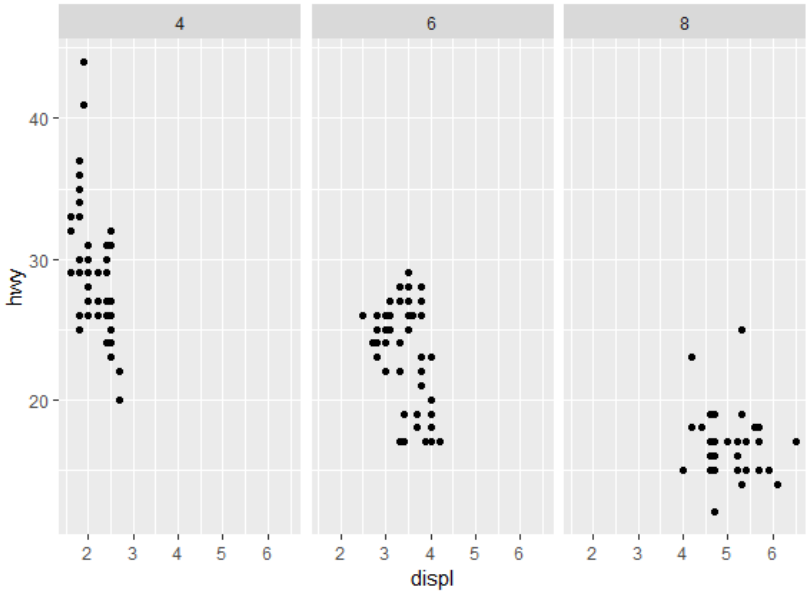
데이터 프레임 mpg에서 변수 drv와 cyl의 범주별로 displ과 hwy의 산점도 작성. 단, drv가 "r"인 자료와 cyl이 5인 자료는 제외

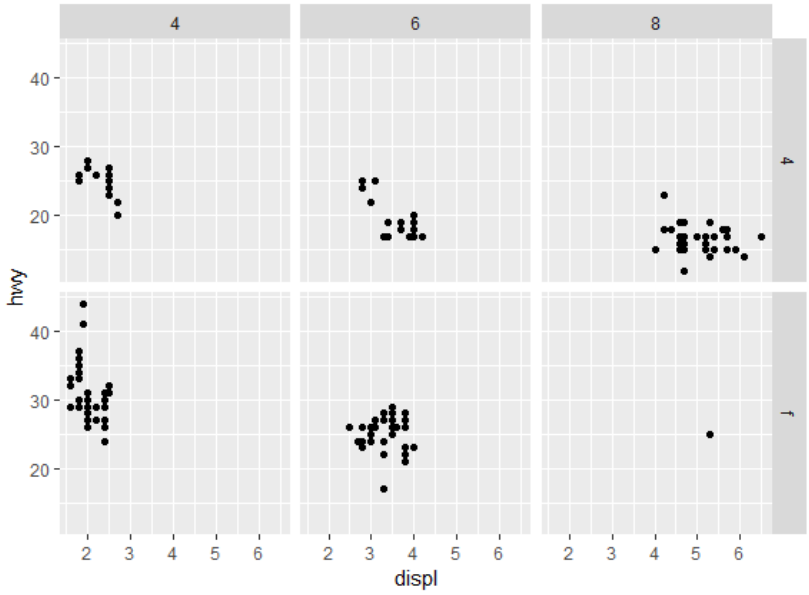
```
>  my_plot <- mpg %>%  filter(cyl!=5,
   drv!="r") %>%  ggplot() +
     geom_point(mapping=aes(x=displ, y=hwy))
```

```
 > my_plot + facet_grid(drv ~ .)
```



```
> my_plot + facet_grid(. ~ cyl)

> my_plot + facet_grid(drv ~ cyl)
```





연속형 변수에 의한 faceting

- 연속형 변수를 범주형 변수로 변환 후 faceting

- 유용한 함수

  ① `cut_interval(x, n, length)`: 벡터 x를 길이가 length인 n개의 구간으로 구분

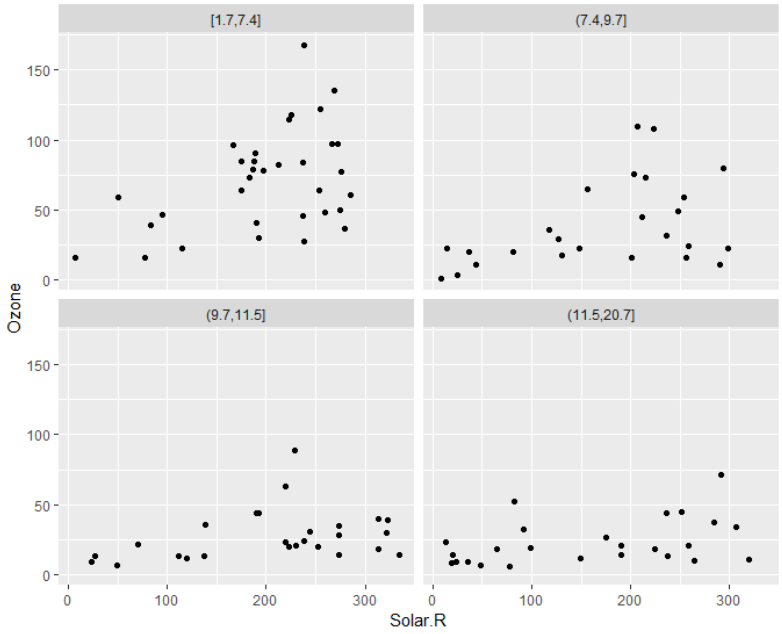
  ② `cut_width(x, width, boundary)`: 벡터 x를 길이가 width인 구간으로 구분. 옵션 boundary는 구간의 시작점 지정.

  ③ `cut_number(x, n)`: 벡터 x를 n개의 구간으로 구분하되 각 구간에 속한 데이터의 개수가 대략 동일하도록 구분

데이터 프레임 airquality에서 변수 Ozone, Solar.R, Wind의 관계 탐색

1) 변수 Wind를 4개의 구간으로 구분하되 속한 자료의 개수가 비슷하도록
2) 4개의 구간에서 Ozone과 Solor.R의 산점도 작성

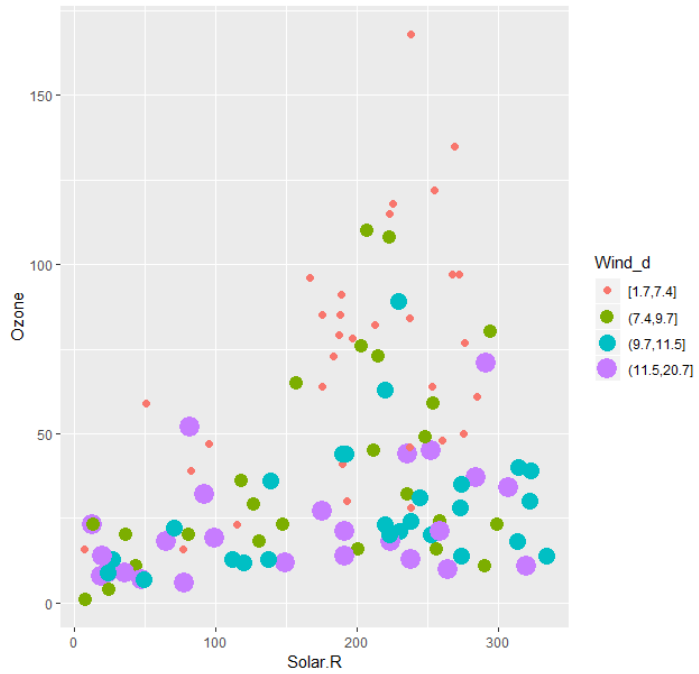
```
> pp <- airquality %>%
    mutate(Wind_d = cut_number(Wind, n=4)) %>%
    ggplot(mapping=aes(x=Solar.R, y=Ozone)) +
    geom_point()

> pp + facet_wrap(~ Wind_d)
```



- 변수 Wind가 큰 값을 가질수록 두 변수의 관계는 점점 미약해지고 있음
- 세 연속형 변수의 관계 탐색 방법 중 하나

한 그래프에 함께 작성

```
> pp + geom_point(mapping=aes(color=Wind_d, size=Wind_d))
```



4. 기하 객체: Geometric object

- Base graphics에서 그래프 작성 방식: pen on paper
  - 높은-수준의 그래프 함수: 좌표축과 주요 그래프 작성
  - 낮은-수준의 그래프 함수: 점, 선, 문자 등을 추가하여 원하는 그래프 작성

- ggplot2에서 그래프 작성 방식
  - 작성하고자 하는 그래프: 몇몇 유형의 그래프(점 그래프, 선 그래프 등등)를 겹쳐 놓은 것
  - 몇몇 유형의 그래프를 각기 따로 작성
  - 작성된 그래프를 겹쳐 놓음으로써 원하는 그래프 작성

ggplot2 시스템

원하는 유형의 그래프(점 그래프, 선 그래프 등등) 작성

↔ 해당되는 기하 객체(geom)를 사용하여 그래프 작성

- 기하 객체의 사용

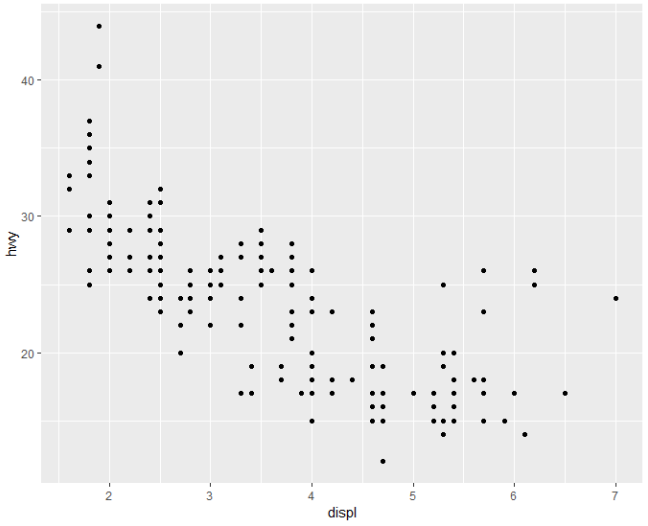
  - 해당되는 geom 함수의 실행
  - geom 함수 실행 → 해당 유형의 그래프가 작성된 layer 생성
  - 여러 개의 geom 함수 실행: 여러 layer 생성되고 이것들이 겹쳐져서 원하는 그래프 완성

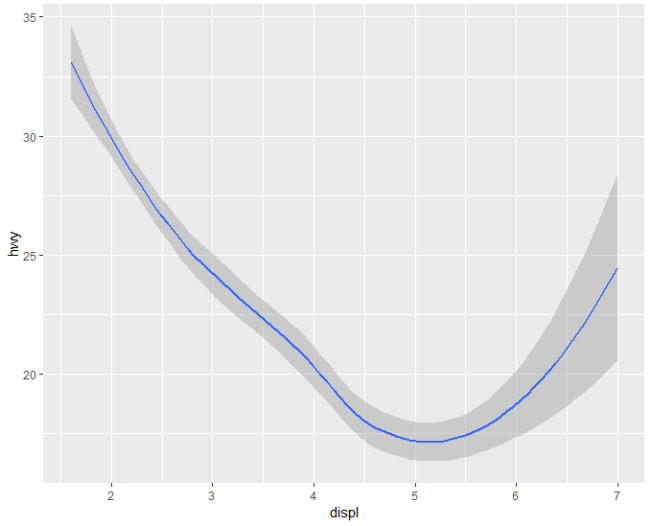
동일 자료에 다른 geom 적용

- mpg의 변수 displ과 hwy를 대상으로 point geom과 smooth geom 적용
  - point geom: 점 그래프 작성
  - smooth geom: 비모수 회귀곡선 작성

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

```
> ggplot(data=mpg) +
  geom_smooth(mapping=aes(x=displ,y=hwy))
```





geom 함수 리스트

- 현재 대략 30개 이상의 geom 함수가 있음
- 한 변수에 대한 함수: geom_bar( ), geom_histogram( ), geom_density( ), geom_dotplot( ) 등등
- 두 변수에 대한 함수: geom_point( ), geom_smooth( ), geom_text( ), geom_line( ), geom_boxplot( ) 등등
- 세 변수에 대한 함수: geom_contour( ), geom_tile( ) 등등
- geom 함수의 리스트: R studio의 메뉴에서 'Help > Cheatsheets > Data Visualization with ggplot2' 에서 확인 가능

글로벌 매핑과 로컬 매핑

- 글로벌 매핑: 함수 ggplot( )에서의 매핑. 해당 그래프 작성에 참여한 모든 geom 함수에 적용

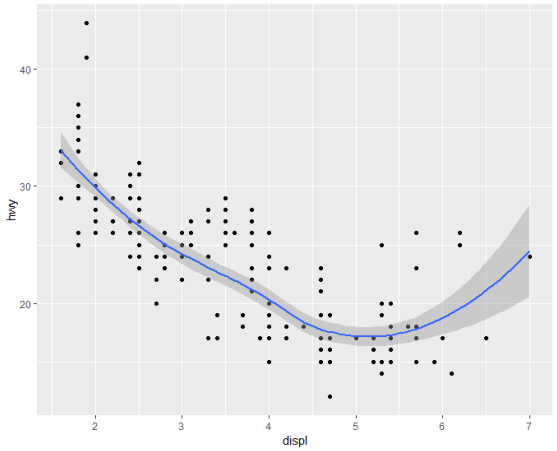
- 로컬 매핑: geom 함수에서의 매핑. 해당 geom 함수로 작성되는 layer에만 적용. 해당 layer에서는 글로벌 매핑보다 우선해서 적용됨.

```
ggplot(data, mapping=aes( ) ) +
      geom_*(mapping=aes( ) ) +
      geom_*(mapping=aes( ) )
```

글로벌 매핑

로컬 매핑

예: mpg의 변수 displ과 hwy의 산점도에 비모수 회귀곡선 추가



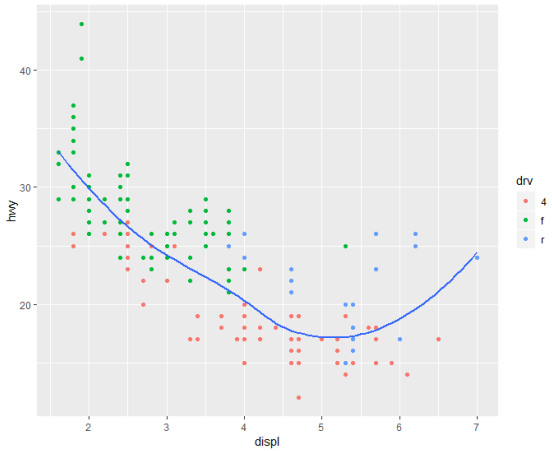
- 두 geom 함수에 동일한 내용의 매핑이 중복되어 입력

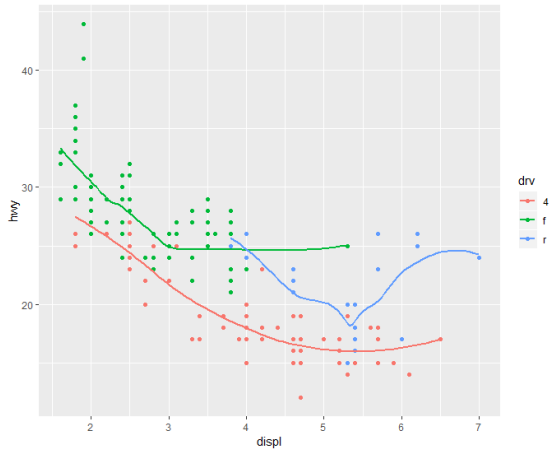
```
> ggplot(data=mpg) +
  geom_point(mapping=aes(x=displ,y=hwy)) +
  geom_smooth(mapping=aes(x=displ,y=hwy))
```

- 글로벌 매핑으로 중복 입력 문제 해결

```
> ggplot(data=mpg, mapping=aes(x=displ,y=hwy)) +
    geom_point() +
    geom_smooth()
```

예: mpg의 변수 displ과 hwy의 비모수 회귀곡선 작성. 그 위에 산점도 추가하되 drv의 값에 따라 점의 색을 구분.






```
> ggplot(data=mpg,mapping=aes(x=displ, y=hwy)) +
  geom_point(mapping=aes(color=drv)) +
  geom_smooth(se=FALSE)
```

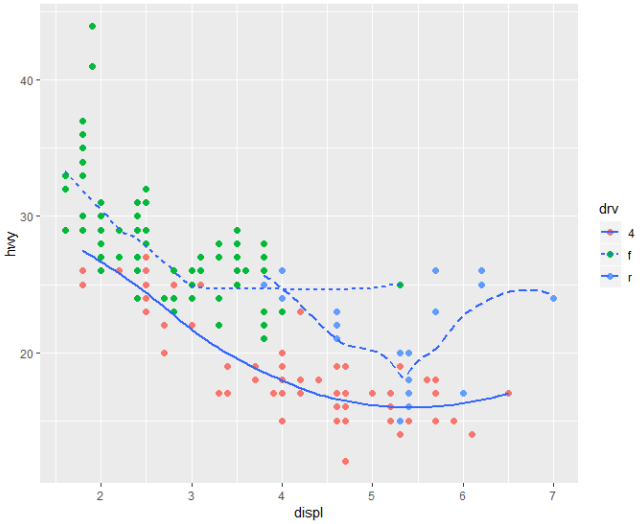
x, y: 글로벌 매핑
color: 로컬 매핑

```
> ggplot(data=mpg, mapping=aes(x=displ, y=hwy, color=drv))+
    geom_point()+
    geom_smooth(se=FALSE)
```

x, y, color: 글로벌 매핑

예: mpg의 변수 displ과 hwy의 비모수 회귀곡선 작성하되 drv에 의해 구분되는 그룹별 각각 추정하여 선의 종류를 다르게 표시. 그 위에 산점도 추가하되 drv의 값에 따라 점의 색을 구분, 점의 크기 확대.

```
> ggplot(data=mpg, mapping=aes(x=displ, y=hwy)) +
  geom_point(mapping=aes(color=drv), size=2) +
  geom_smooth(mapping=aes(linetype=drv), se=FALSE)
```

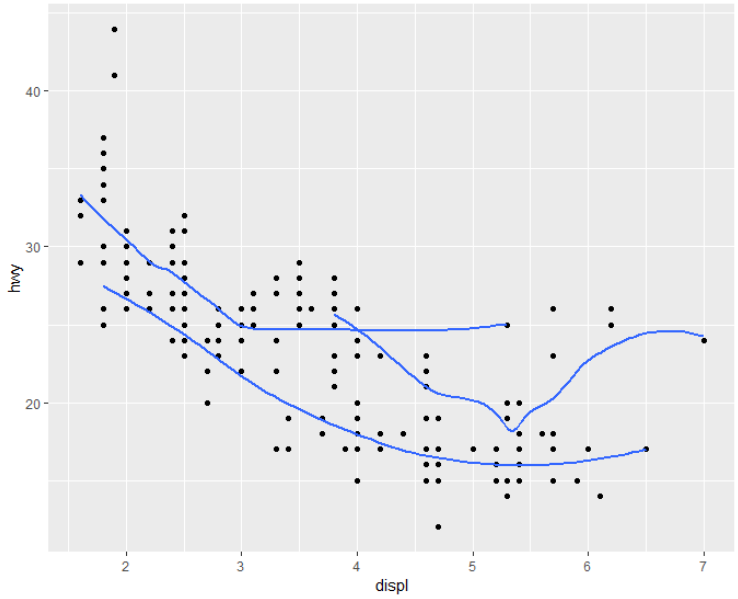


linetype: 선의 종류를 나타내는 시각적 요소

예: 다음의 그래프 작성

- 변수 drv의 그룹별로 따로 비모수 회귀곡선 작성하되, 선의 색과 종류는 같은 것을 사용

```
> ggplot(data=mpg, mapping=aes(x=displ, y=hwy)) +
  geom_point() +
  geom_smooth(mapping=aes(group=drv), se=FALSE)
```



group: 그룹을 구성하는 시각적 요소

5. 통계적 변환: Statistical transformation

그래프 작성에 사용되는 자료

1) 입력된 자료: 산점도
2) 입력된 자료를 대상으로 통계적 변환 과정을 거쳐 생성된 자료: 비모수 회귀곡선 그래프

- 통계적 변환(stat)
  - 입력된 데이터 프레임 자료의 변환을 의미
  - 각 그래프 유형별 대응되는 stat 존재
    - 산점도: stat="identity"
    - 비모수 회귀곡선: stat="smooth"
    - 막대 그래프: stat="count"
  - 각 geom 함수마다 대응되는 디폴트 stat 존재
    - geom_point( ) → geom_point(stat="identity")
    - geom_smooth( ) → geom_smooth(stat="smooth")
    - geom_bar( ) → geom_bar(stat="count")

주요 이력
現) ㈜RTMC 전략기획실장
前) ㈜B사 웹로그분석 및 DP사업 完
前) ㈜H금속사 회계팀
前) ㈜B건설사 회계팀
前) K문고 CRM VIP 군집전략 CRM프로젝트 보조연구원
前) L백화점 CRM Alert 전략 CRM프로젝트 보조연구원

BSL(스위스 로잔 비즈니스 스쿨) MBA
ASSIST 빅데이터경영통계 MBA

국가공인 ADSP(빅데이터 준전문가)
現 **코리아IT아카데미 빅데이터 R 강사**
現 **코리아IT아카데미 빅데이터 기초 파이썬 강사**
現 **코리아IT아카데미 빅데이터 기초통계 전담강사**

"자료는 대가이신 박동련 교수님께 도움을 받았음을 밝힙니다."

[박영식] 완성에 이르기까지