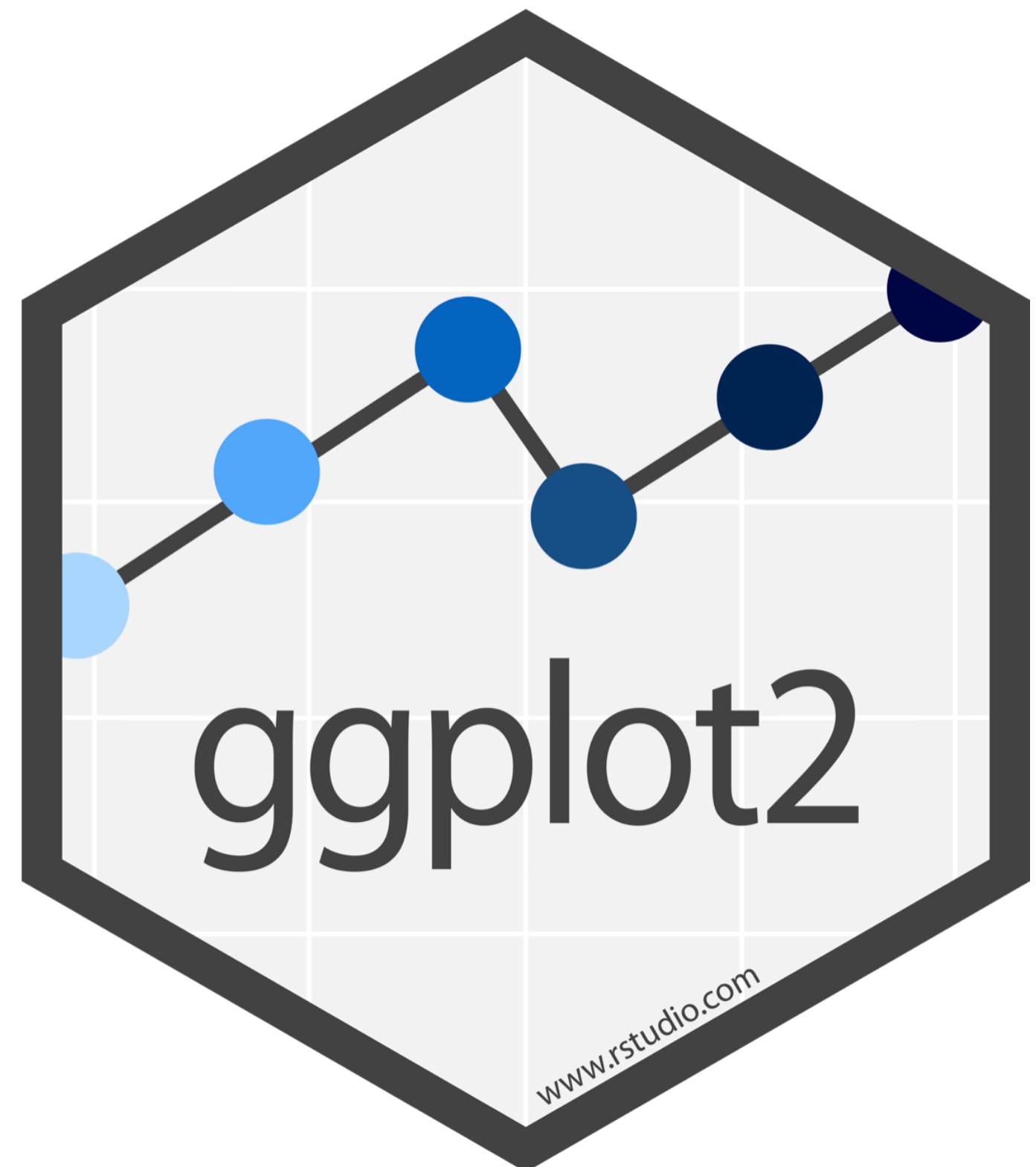
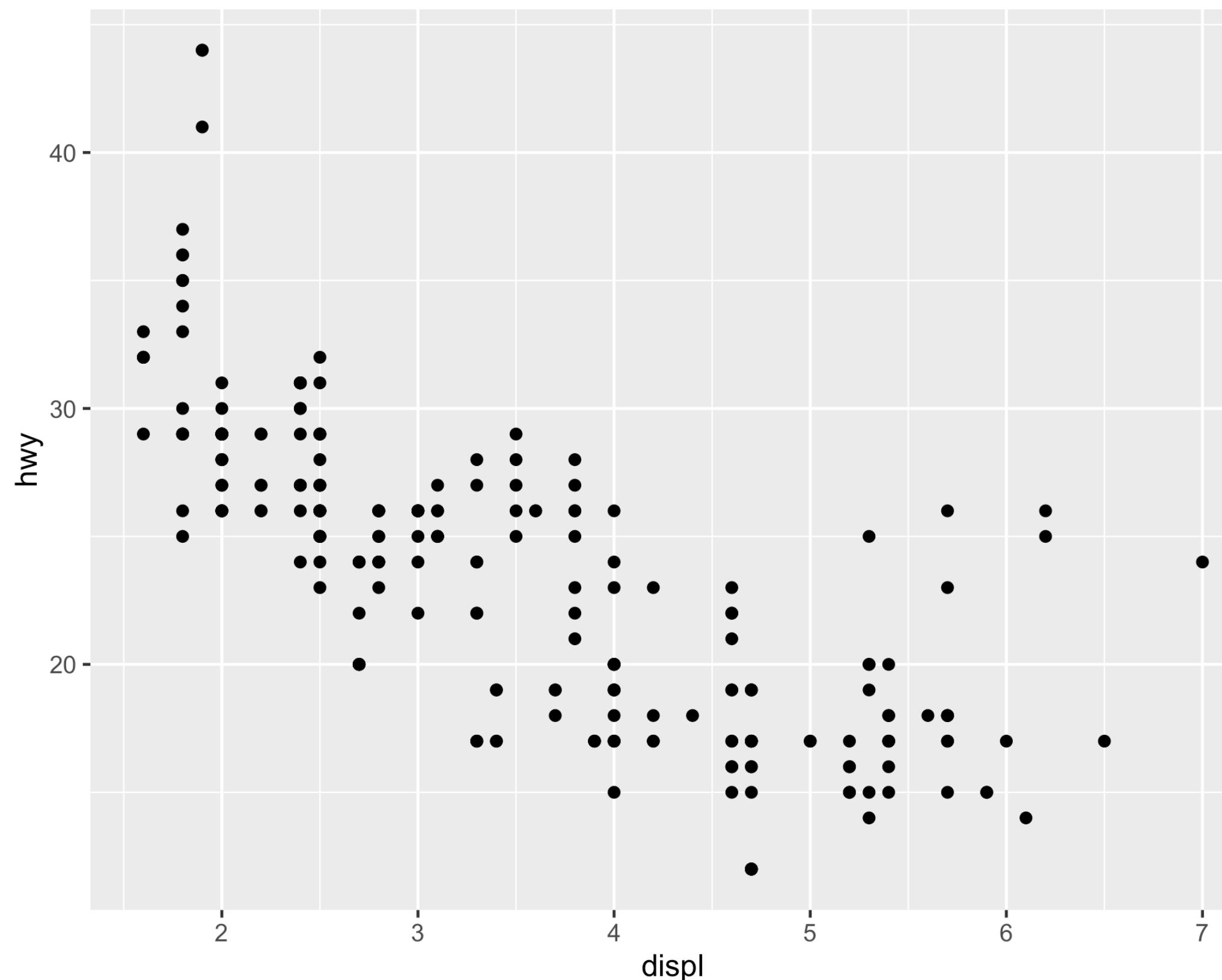


Visualise Data with



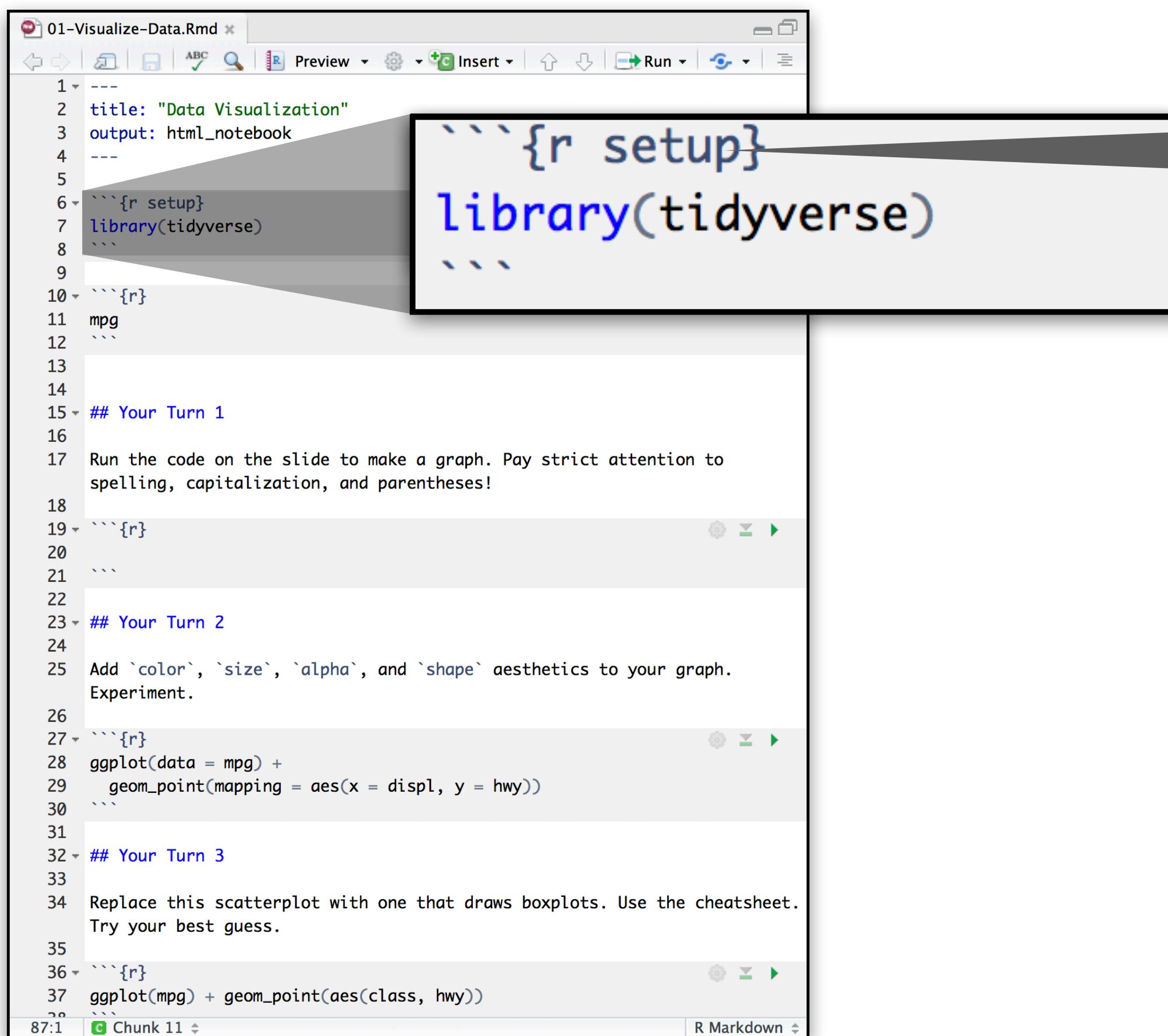
Quale è più facile capire?



	manufacturer	model	displ	year	cyl	trans	drv	cty	hwy	fl	class
16	audi	a6 quattro	2.8	1999	6	auto(l5)	4	15	24	p	midsize
17	audi	a6 quattro	3.1	2008	6	auto(s6)	4	17	25	p	midsize
18	audi	a6 quattro	4.2	2008	8	auto(s6)	4	16	23	p	midsize
19	chevrolet	c1500 suburban 2wd	5.3	2008	8	auto(l4)	r	14	20	r	suv
20	chevrolet	c1500 suburban 2wd	5.3	2008	8	auto(l4)	r	11	15	e	suv
21	chevrolet	c1500 suburban 2wd	5.3	2008	8	auto(l4)	r	14	20	r	suv
22	chevrolet	c1500 suburban 2wd	5.7	1999	8	auto(l4)	r	13	17	r	suv
23	chevrolet	c1500 suburban 2wd	6.0	2008	8	auto(l4)	r	12	17	r	suv
24	chevrolet	corvette	5.7	1999	8	manual(m6)	r	16	26	p	2seater
25	chevrolet	corvette	5.7	1999	8	auto(l4)	r	15	23	p	2seater
26	chevrolet	corvette	6.2	2008	8	manual(m6)	r	16	26	p	2seater
27	chevrolet	corvette	6.2	2008	8	auto(s6)	r	15	25	p	2seater
28	chevrolet	corvette	7.0	2008	8	manual(m6)	r	15	24	p	2seater
29	chevrolet	k1500 tahoe 4wd	5.3	2008	8	auto(l4)	4	14	19	r	suv
30	chevrolet	k1500 tahoe 4wd	5.3	2008	8	auto(l4)	4	11	14	e	suv
31	chevrolet	k1500 tahoe 4wd	5.7	1999	8	auto(l4)	4	11	15	r	suv
32	chevrolet	k1500 tahoe 4wd	6.5	1999	8	auto(l4)	4	14	17	d	suv
33	chevrolet	malibu	2.4	1999	4	auto(l4)	f	19	27	r	midsize
34	chevrolet	malibu	2.4	2008	4	auto(l4)	f	22	30	r	midsize
35	chevrolet	malibu	3.1	1999	6	auto(l4)	f	18	26	r	midsize
36	chevrolet	malibu	3.5	2008	6	auto(l4)	f	18	29	r	midsize
37	chevrolet	malibu	3.6	2008	6	auto(s6)	f	17	26	r	midsize
38	dodge	caravan 2wd	2.4	1999	4	auto(l3)	f	18	24	r	minivan
39	dodge	caravan 2wd	3.0	1999	6	auto(l4)	f	17	24	r	minivan

Iniziamo

La prima parte (“chunk” in inglese) va sempre all’inizio



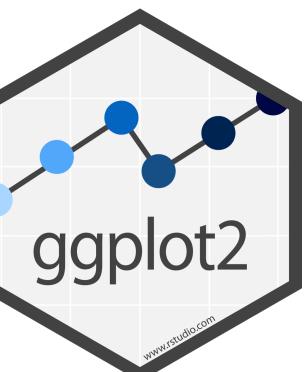
```
1 ---  
2 title: "Data Visualization"  
3 output: html_notebook  
4 ---  
5  
6 ```{r setup}  
7 library(tidyverse)  
8 ```  
9  
10 ```{r}  
11 mpg  
12 ```  
13  
14  
15 ## Your Turn 1  
16  
17 Run the code on the slide to make a graph. Pay strict attention to  
spelling, capitalization, and parentheses!  
18  
19 ```{r}  
20  
21 ```  
22  
23 ## Your Turn 2  
24  
25 Add `color`, `size`, `alpha`, and `shape` aesthetics to your graph.  
Experiment.  
26  
27 ```{r}  
28 ggplot(data = mpg) +  
29   geom_point(mapping = aes(x = displ, y = hwy))  
30 ```  
31  
32 ## Your Turn 3  
33  
34 Replace this scatterplot with one that draws boxplots. Use the cheatsheet.  
Try your best guess.  
35  
36 ```{r}  
37 ggplot(mpg) + geom_point(aes(class, hwy))  
38 ```
```

chunk labels sono opzionali
ora ma necessari per i più
esperti

Il dataset

Dati di risparmio di carburante per 38 modelli di auto.

mpg



Quiz

In gruppi discutete:

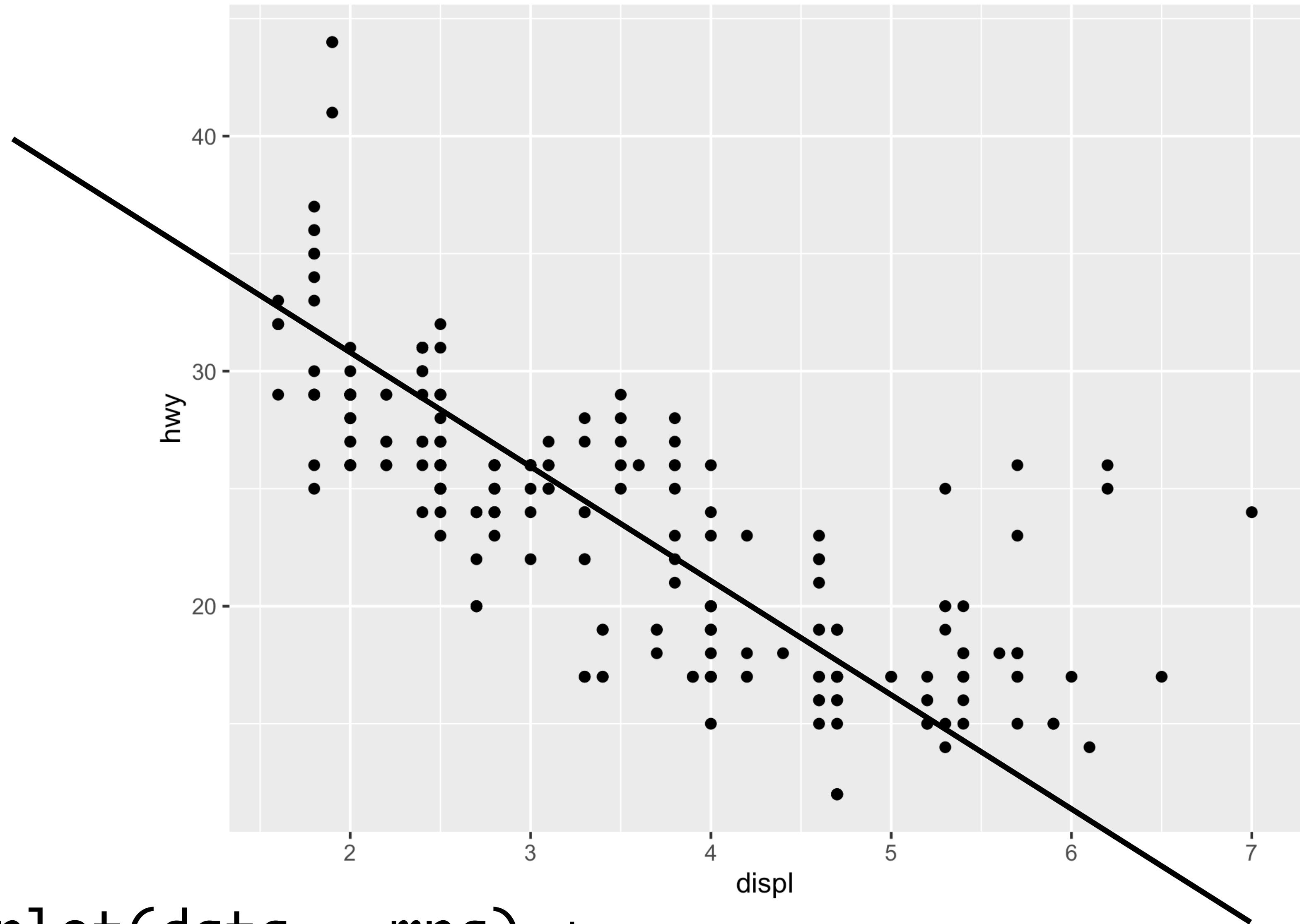
Quale relazione vi aspettate tra le dimensioni del motore (nome variable: **displ**) e l'efficienza del motore (km per litro, nome variable: **hwy**)?

Tocca a te!

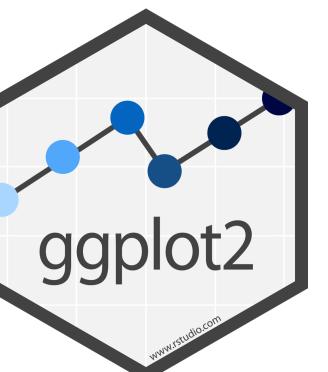
Copiate e “run” the code

Quando si scrive un codice (in qualsiasi linguaggio di programmazione) è importantissimo stare attenti agli errori di ortografia e punteggiatura!

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

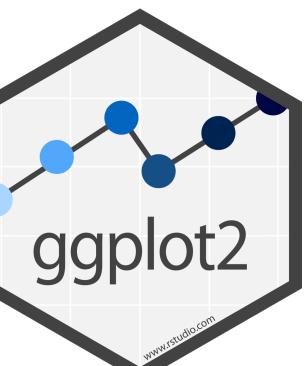


```
ggplot(data = mpg) +  
  geom_point(mapping = aes(x = displ, y = hwy))
```



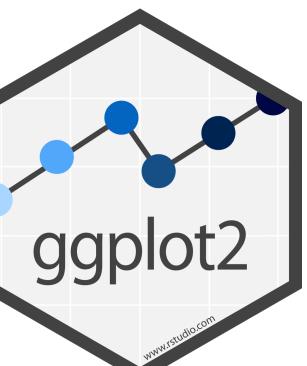
1. Inizia il grafico (plot) con `ggplot()`
2. Aggiungi i diversi strati (layers) con `geom_ functions`

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



Suggerimento: Sempre
aggiungere “+” alla fine della riga
mai all’inizio

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



data

+ prima della nuova riga

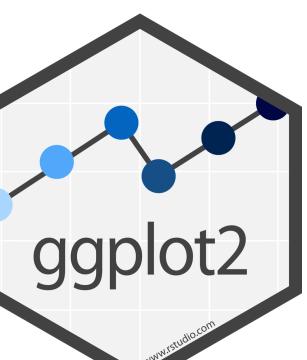
```
ggplot(data = mpg) +  
  geom_point(mapping = aes(x = displ, y = hwy))
```

Tipologia “layer”

aes()

var x

var y

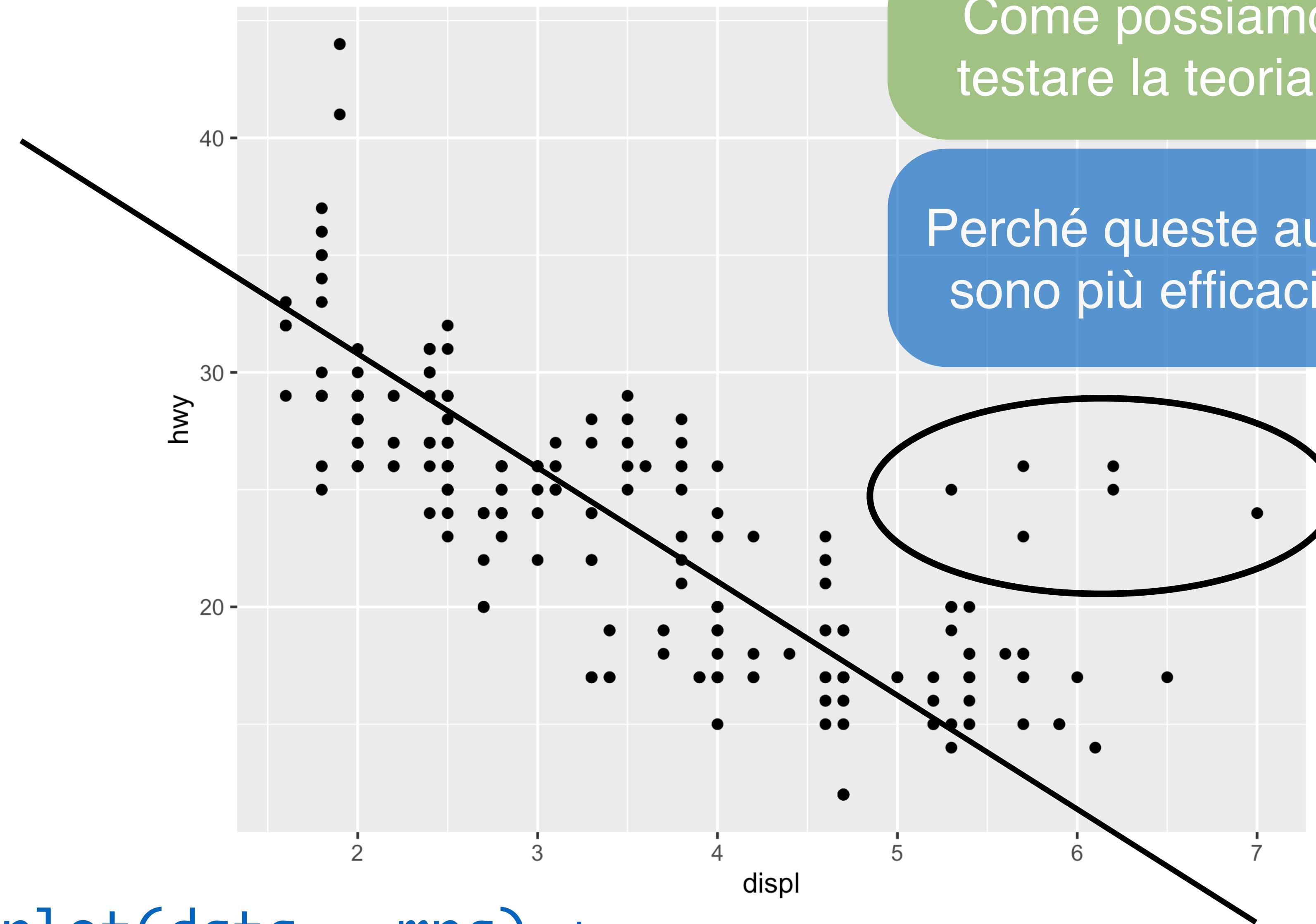


Mappings

R

"The greatest value of a picture
is when it forces us to notice
what we never expected to see."

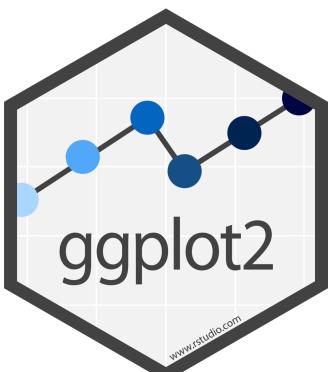
- John Tukey



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

Come possiamo
testare la teoria?

Perché queste auto
sono più efficaci?



Spazio Visivo

colore

Rosso

Marrone

Verde

Acqua

Blu

Viola

Rosa

Spazio Dati

classi

2seater

compact

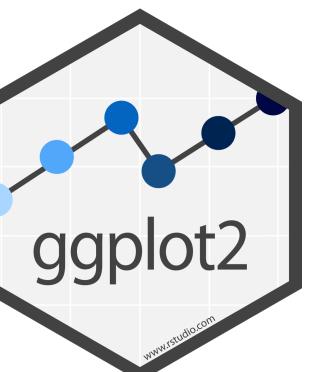
midsize

minivan

pickup

subcompact

suv

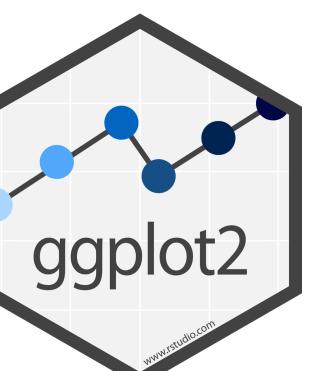


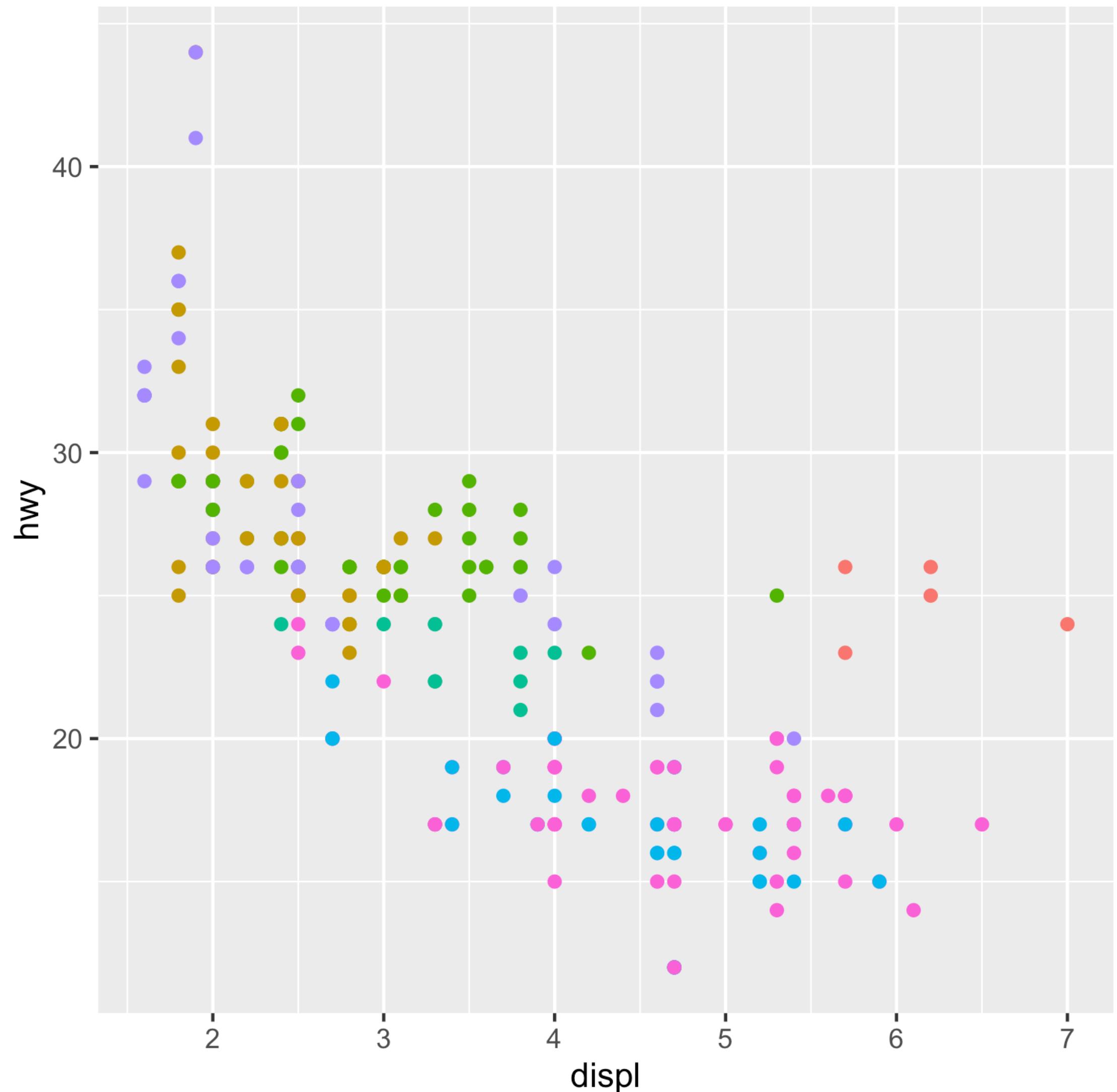
Aesthetics

proprietà estetiche

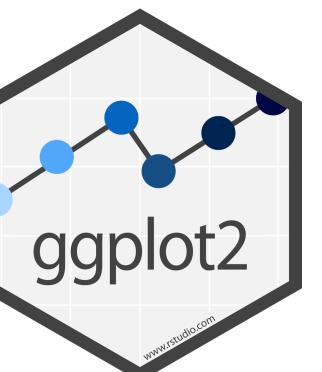
Variabile per mapparlo

```
ggplot(mpg) + geom_point(aes(x = displ, y = hwy, color = class))  
ggplot(mpg) + geom_point(aes(x = displ, y = hwy, size = class))  
ggplot(mpg) + geom_point(aes(x = displ, y = hwy, shape = class))  
ggplot(mpg) + geom_point(aes(x = displ, y = hwy, alpha = class))
```





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



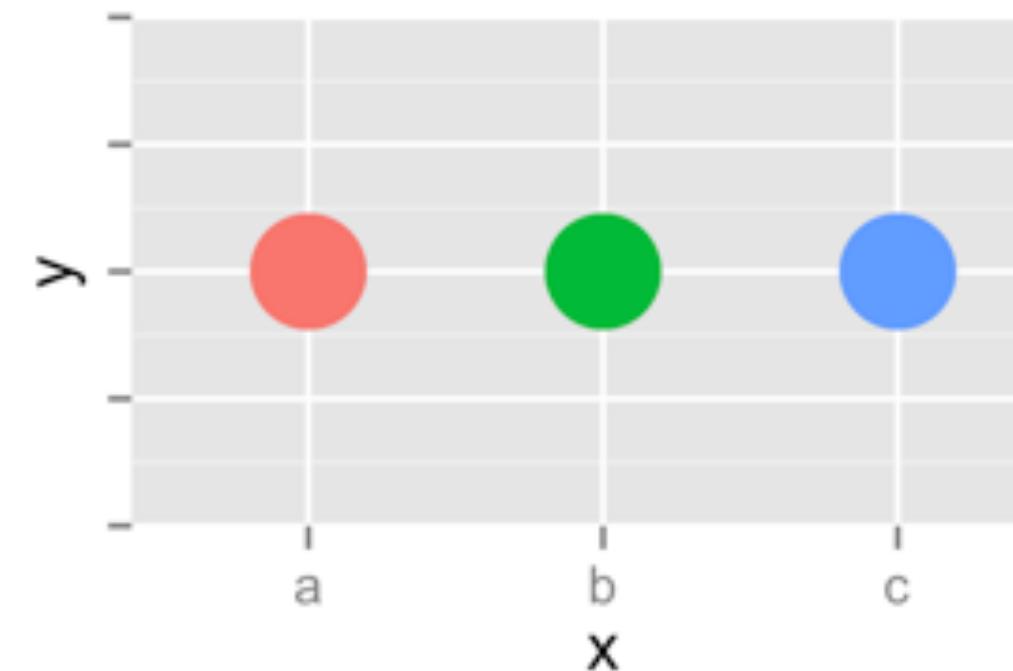
Tocca a te II

Ora aggiungete il colore (color), dimensioni (size), alpha, elementi estetici (aesthetics). Provate a disegnare diversi grafici

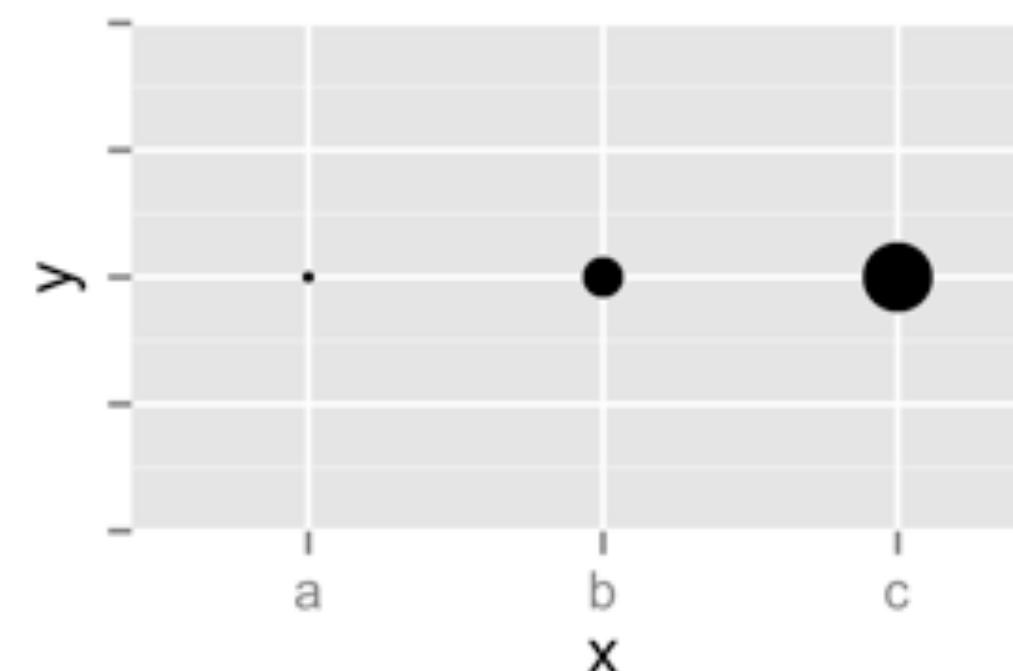
Cosa succede se provate a mappare “aesthetics” con variabili discrete e continue?

Cosa succede se usate più di un elemento “aesthetics”?

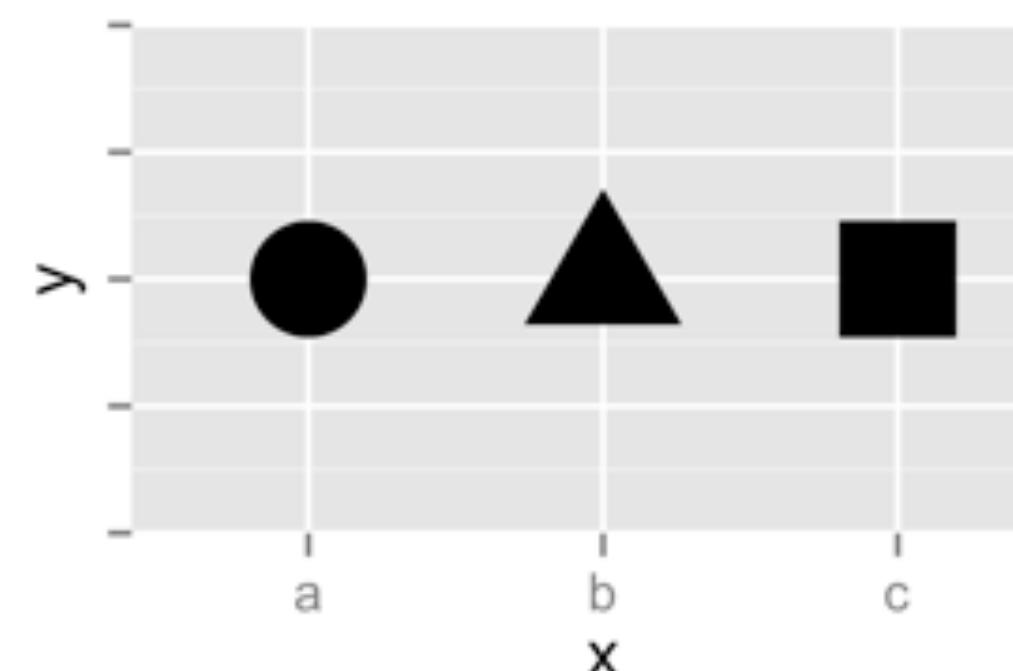
Colore



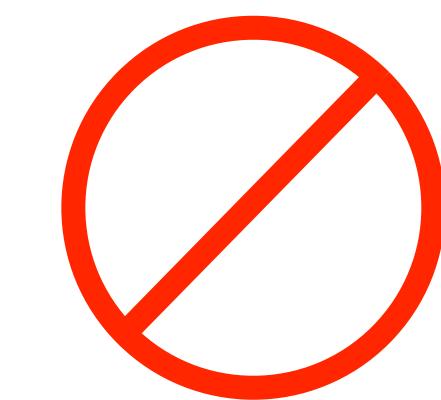
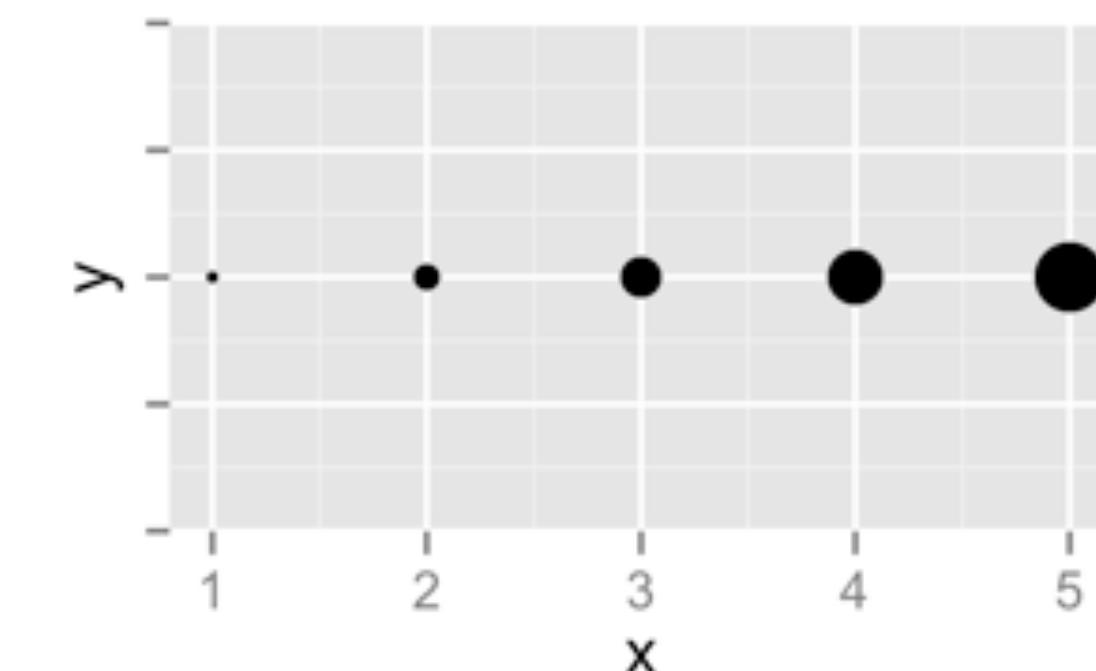
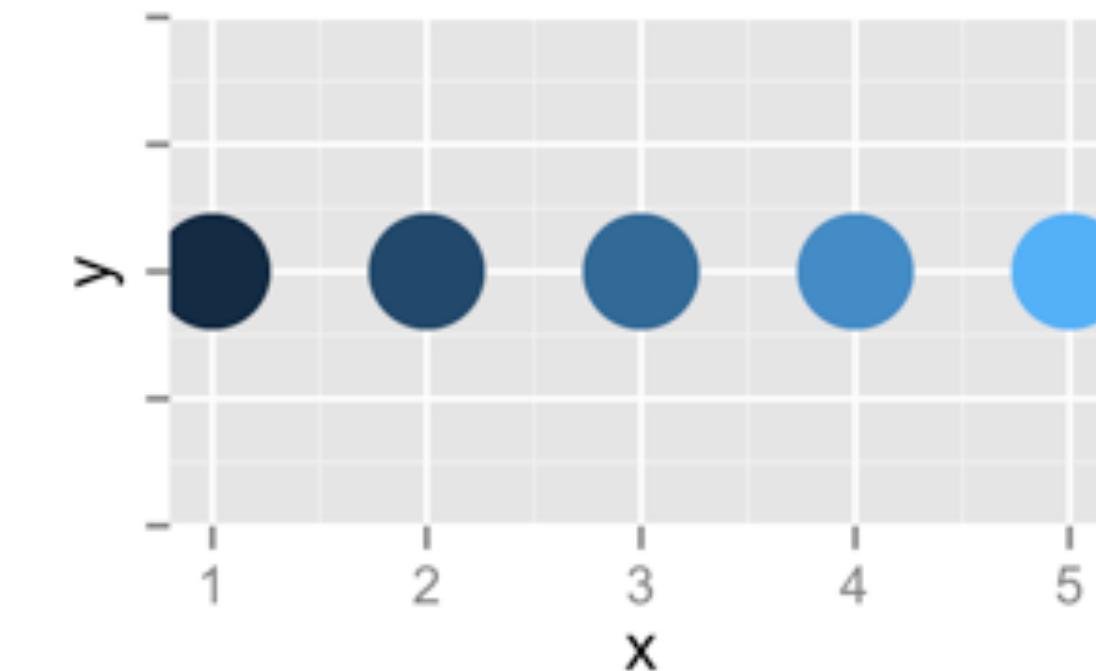
Dimensione
(size)



Forma
(Shape)



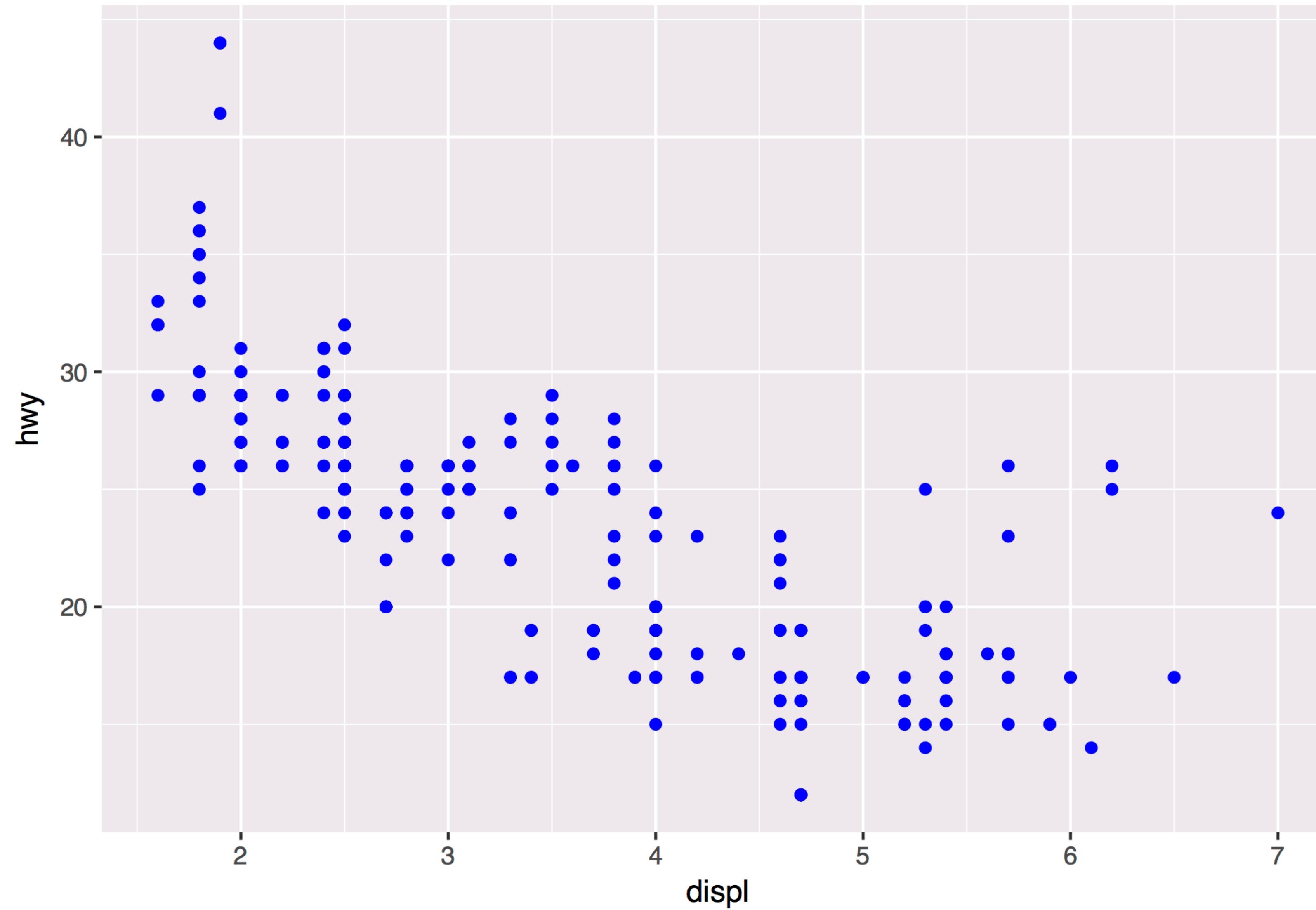
Continuous

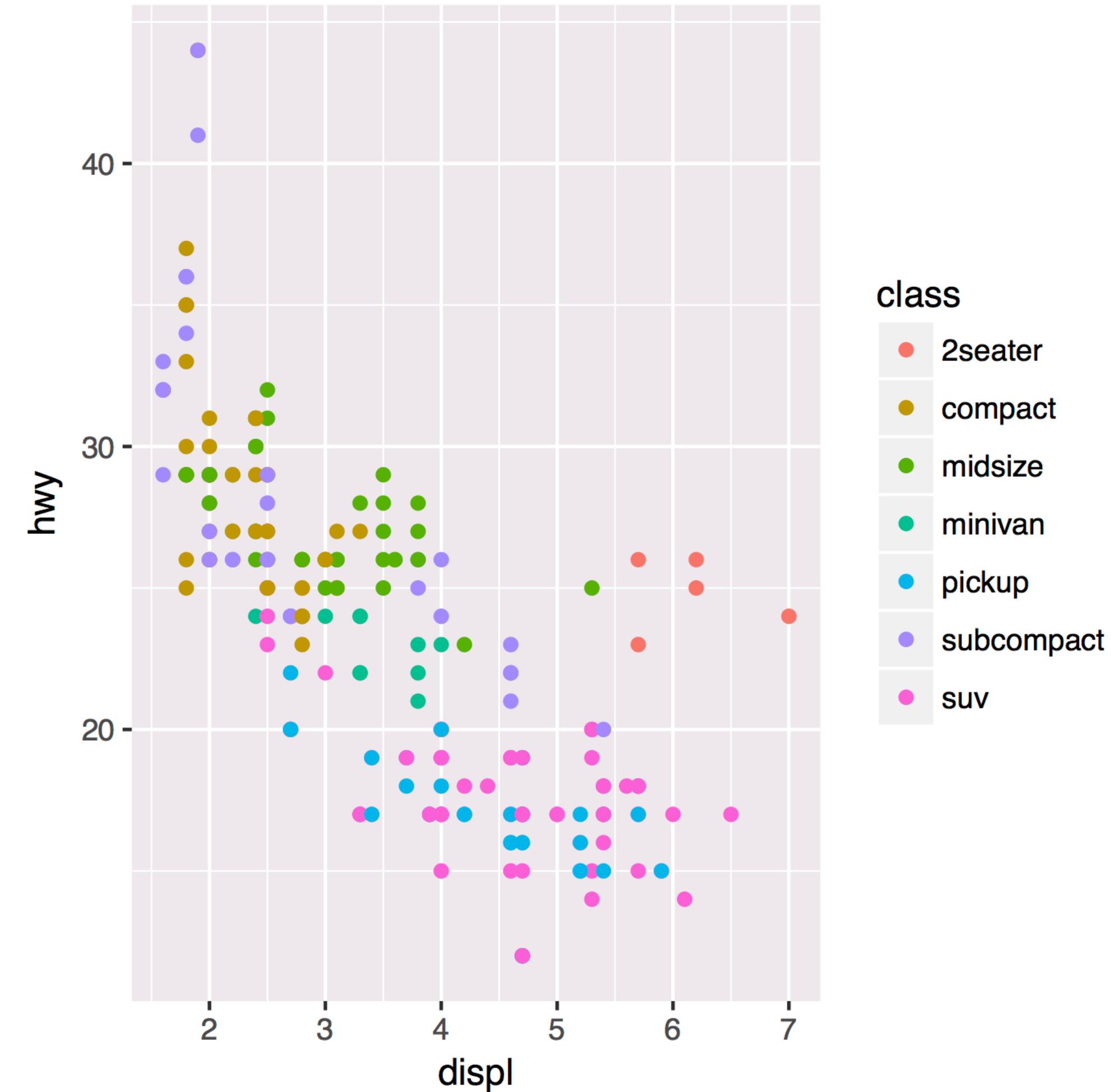


set vs. map

R

Come si può disegnare questo grafico?





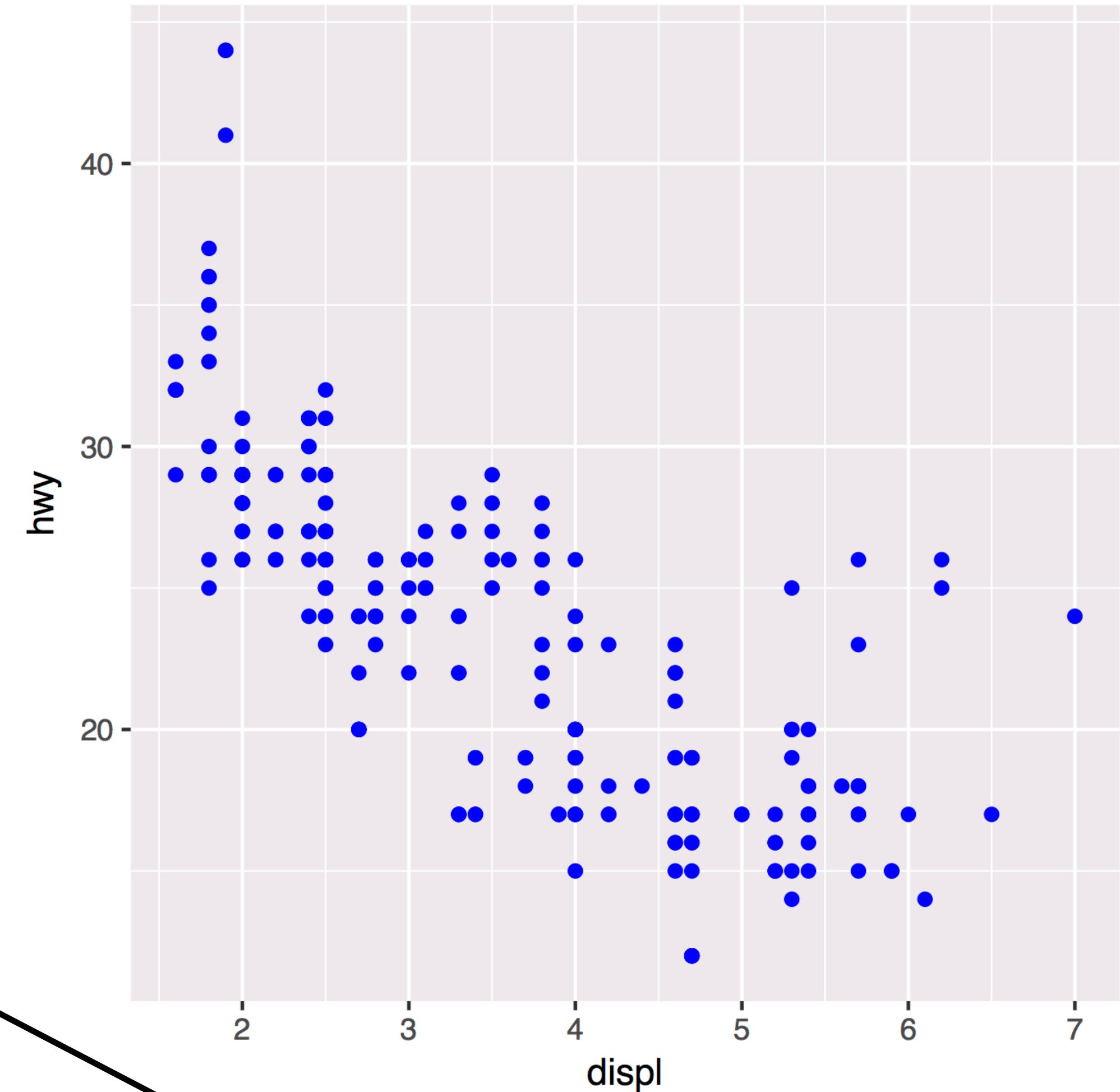
class

- 2seater
- compact
- midsize
- minivan
- pickup
- subcompact
- suv

Dentro aes(): l'elemento estetico mappa una variable

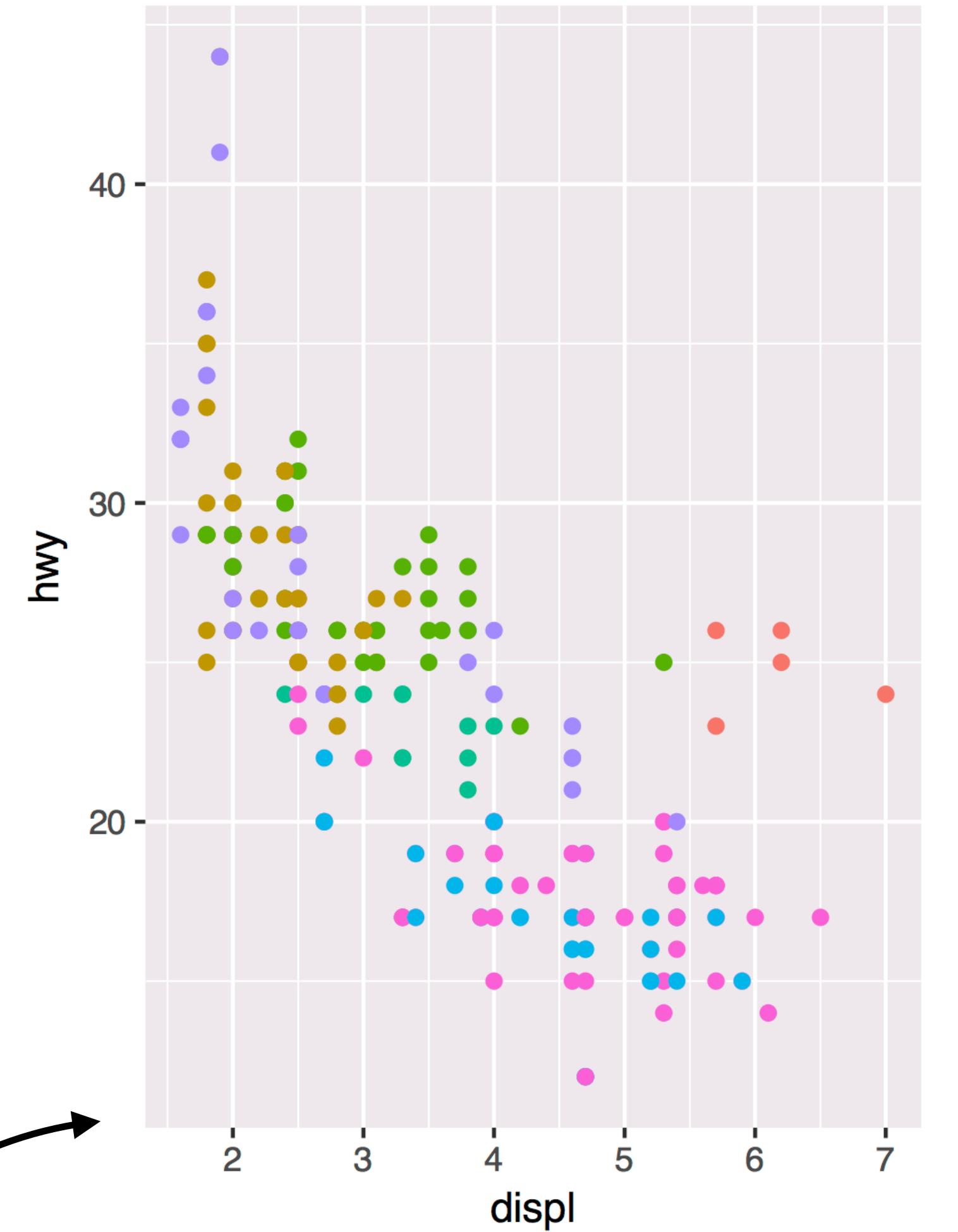
```
ggplot(mpg) + geom_point(aes(x = displ, y = hwy, color = class))
```

Outside of aes(): imposta
l'elemento estetico su un valore



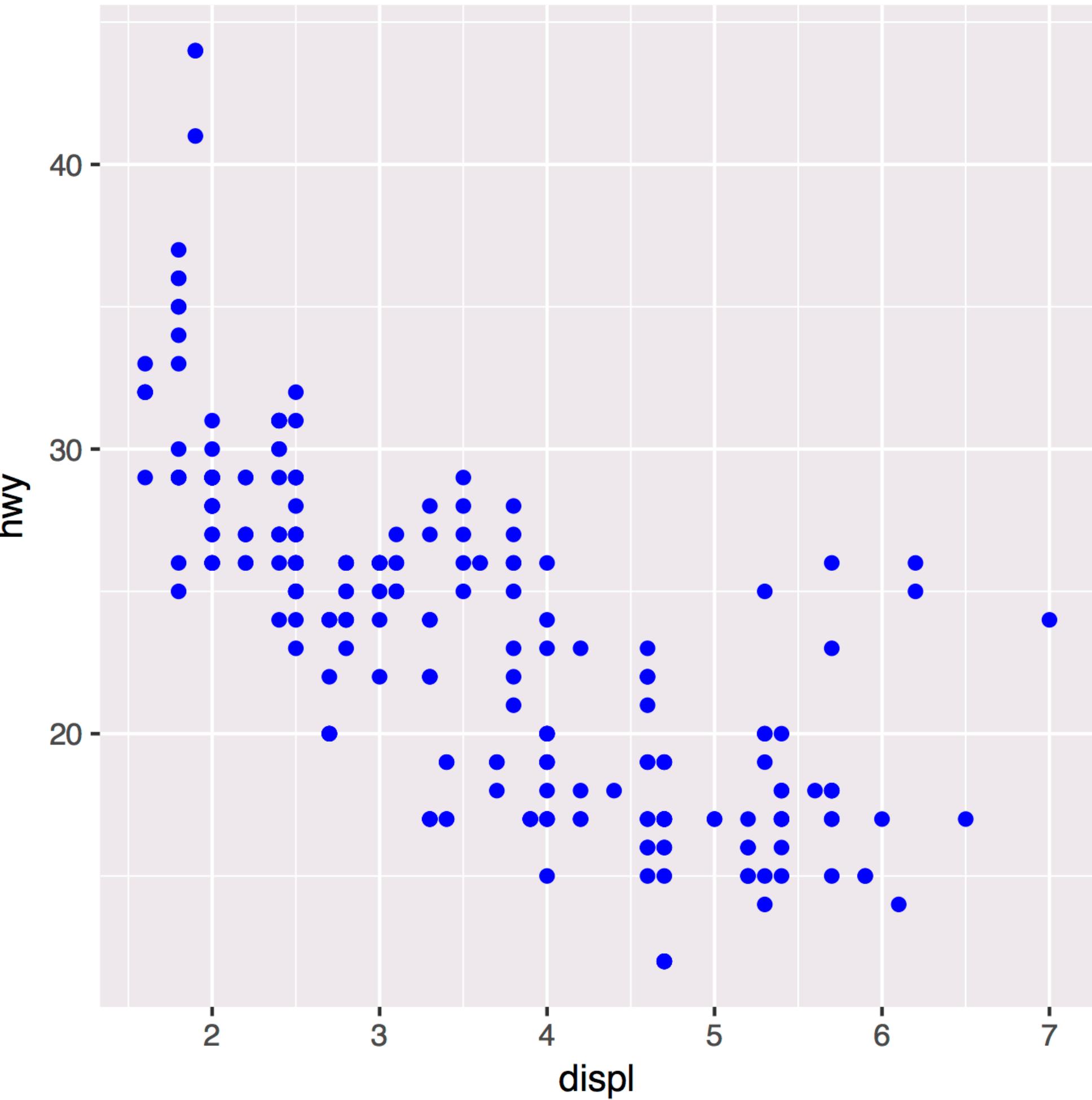
```
ggplot(mpg) + geom_point(aes(x = displ, y = hwy, color = class))
```

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



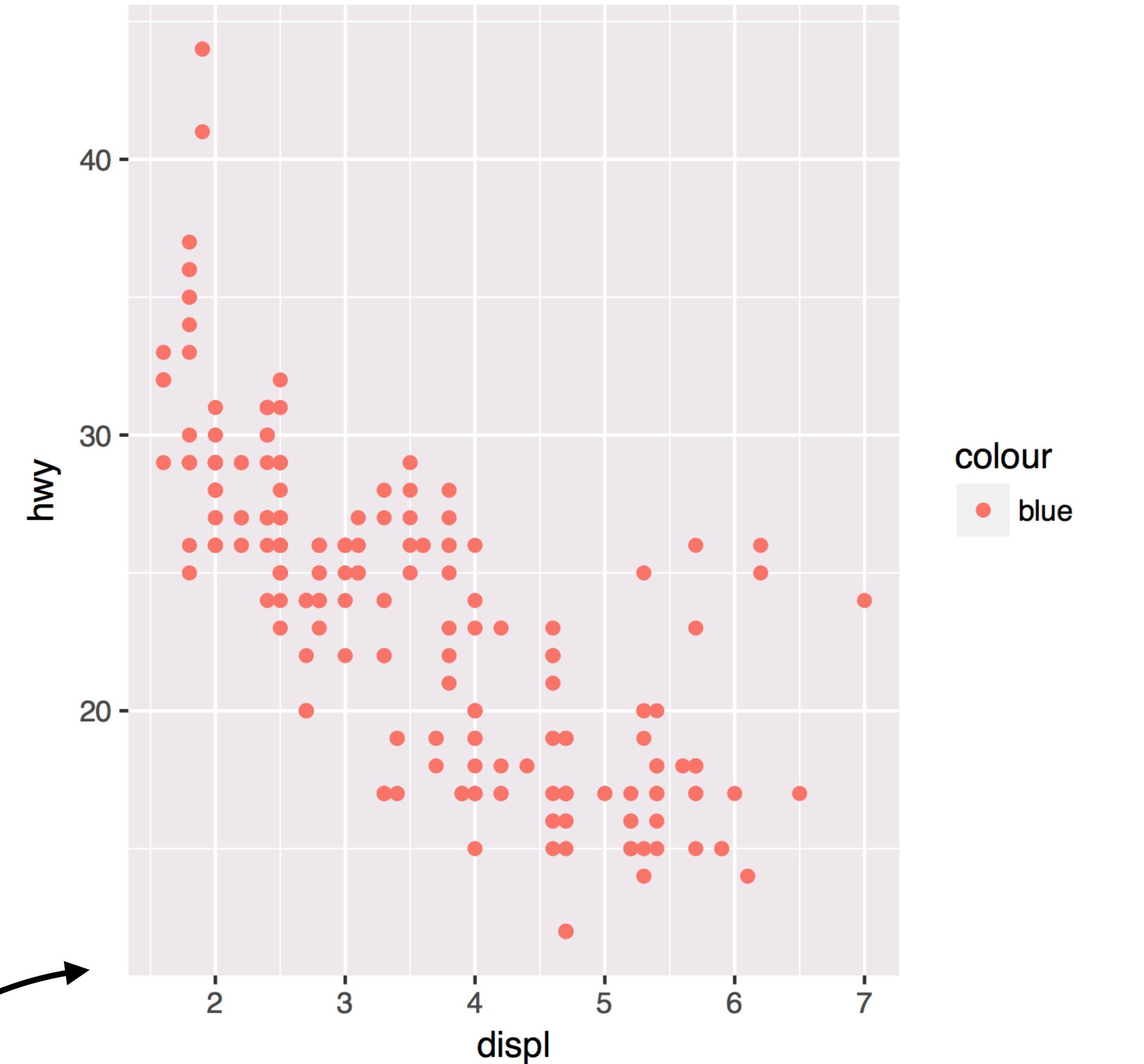
class

- 2seater
- compact
- midsize
- minivan
- pickup
- subcompact
- SUV

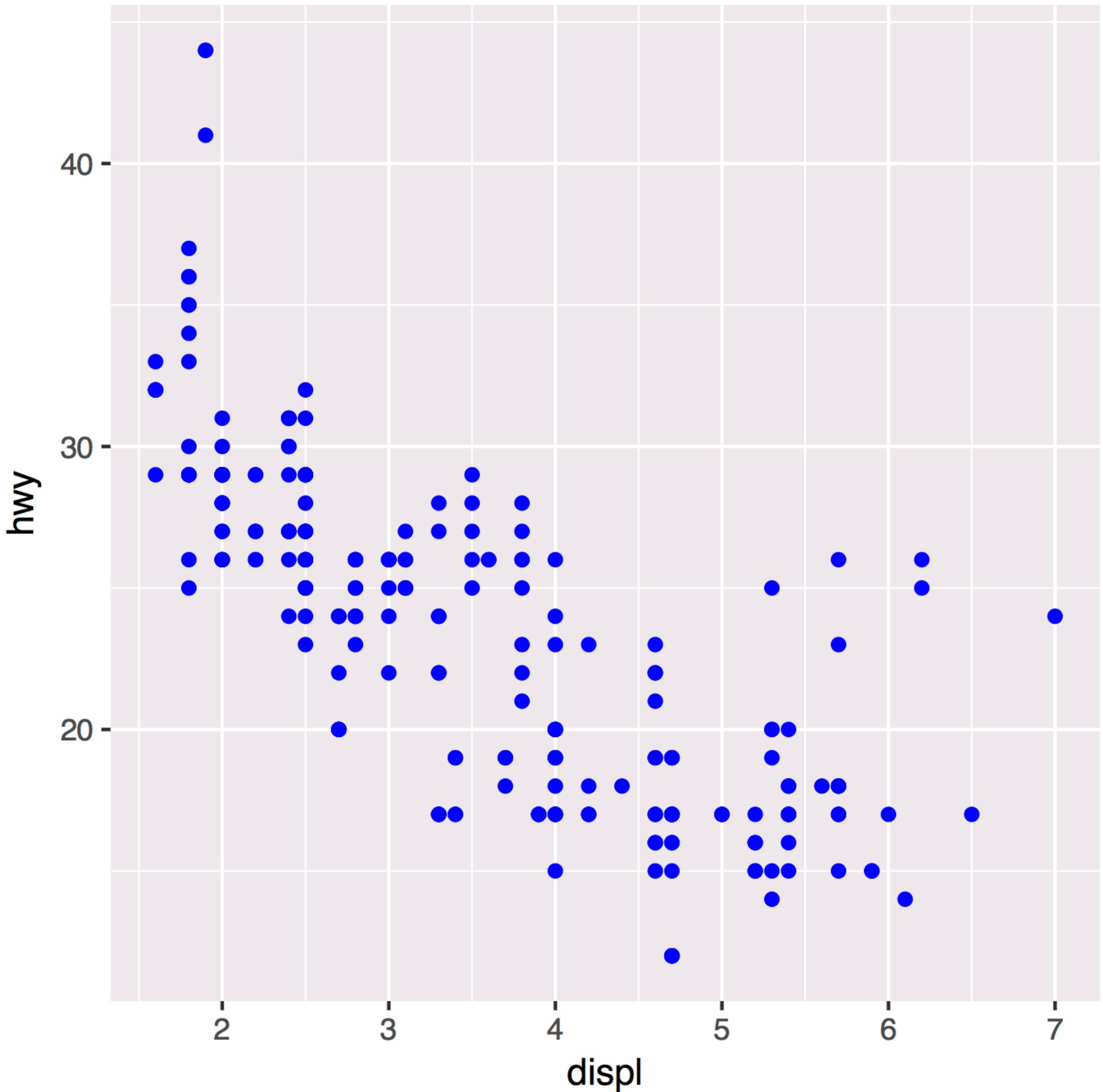


```
ggplot(mpg) + geom_point(aes(x = displ, y = hwy, color = class))
```

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



colour
red



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

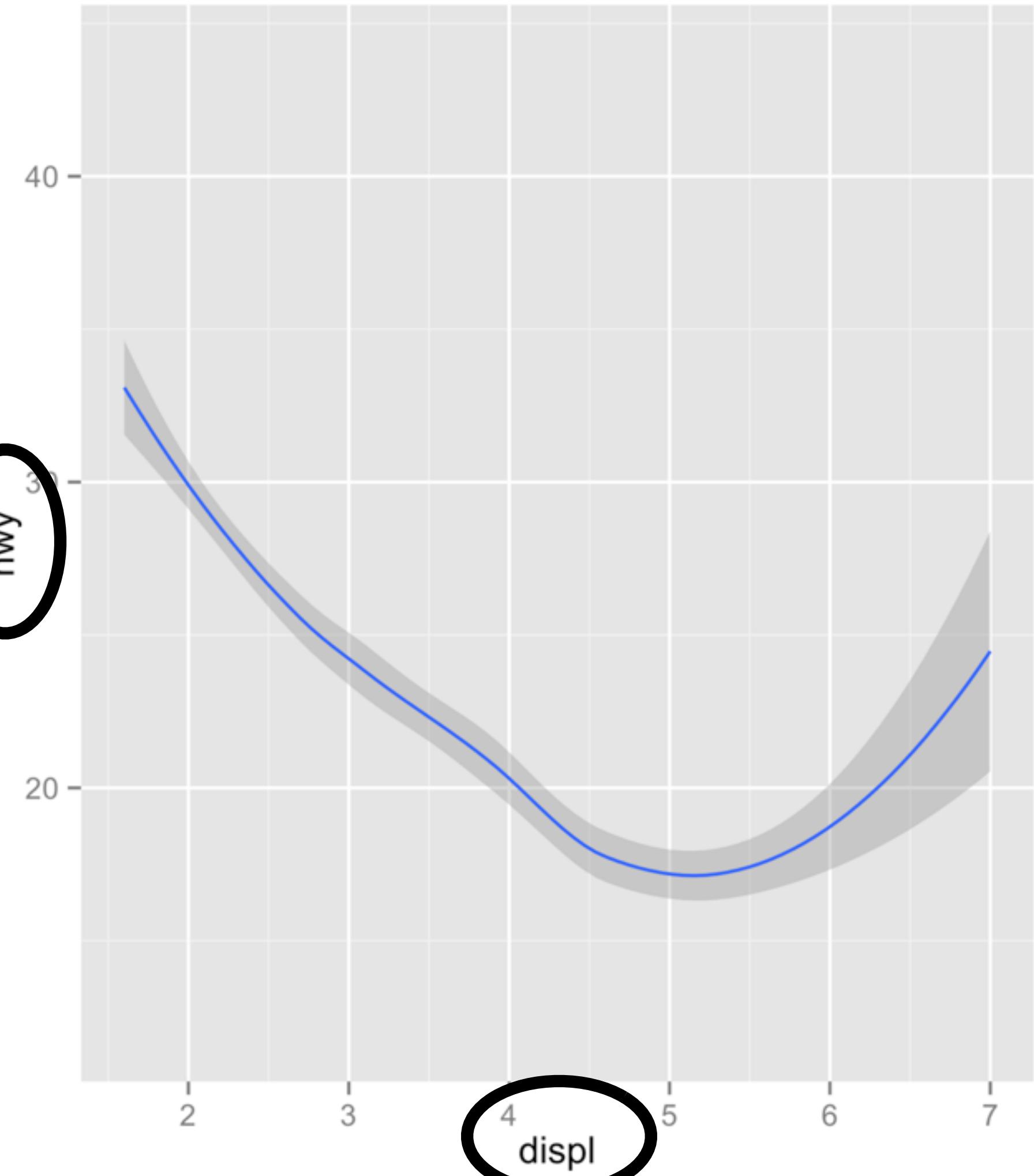
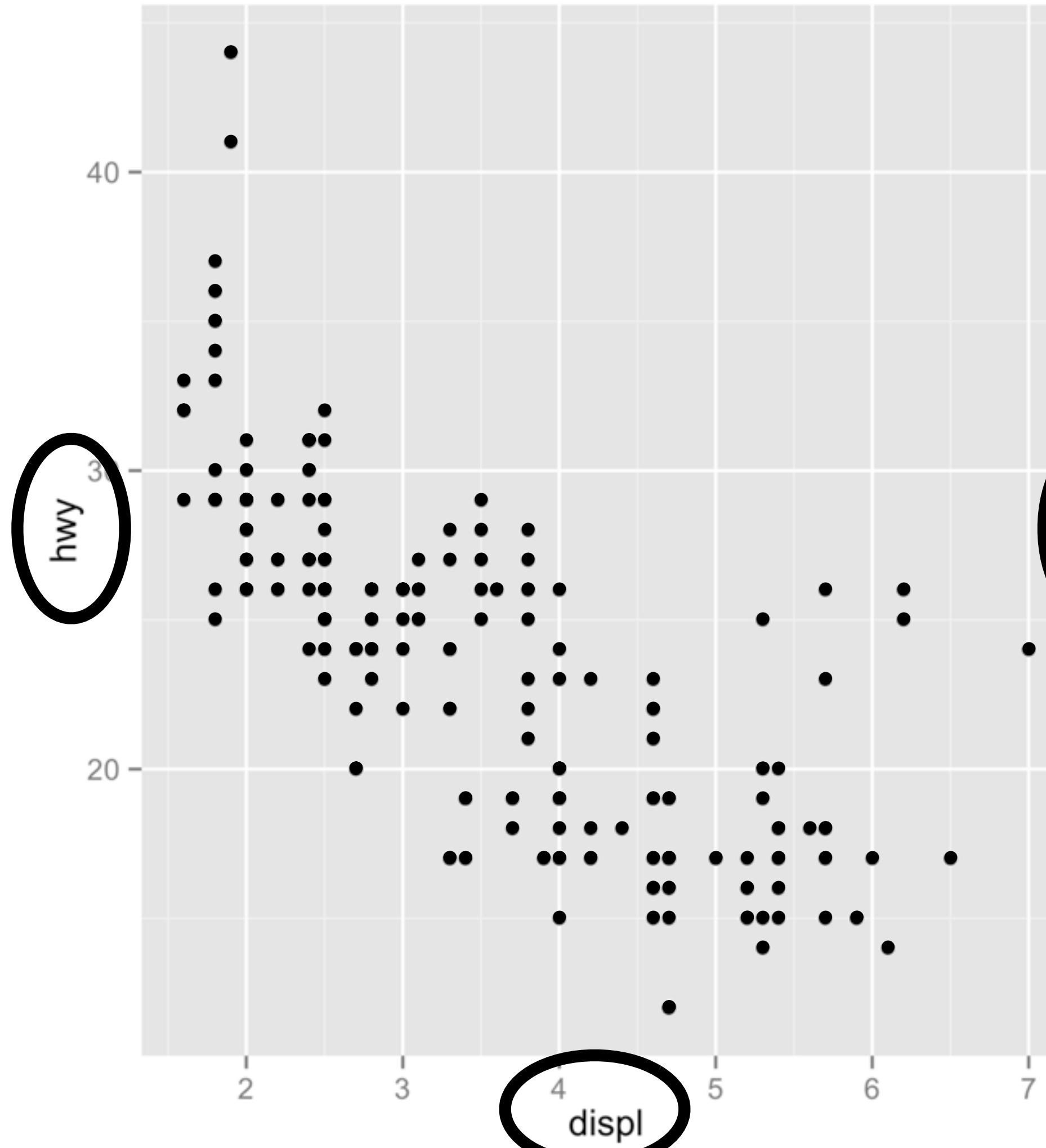
```
ggplot(mpg) + geom_point(aes(x = displ, y = hwy), color = "blue")
```

Geoms



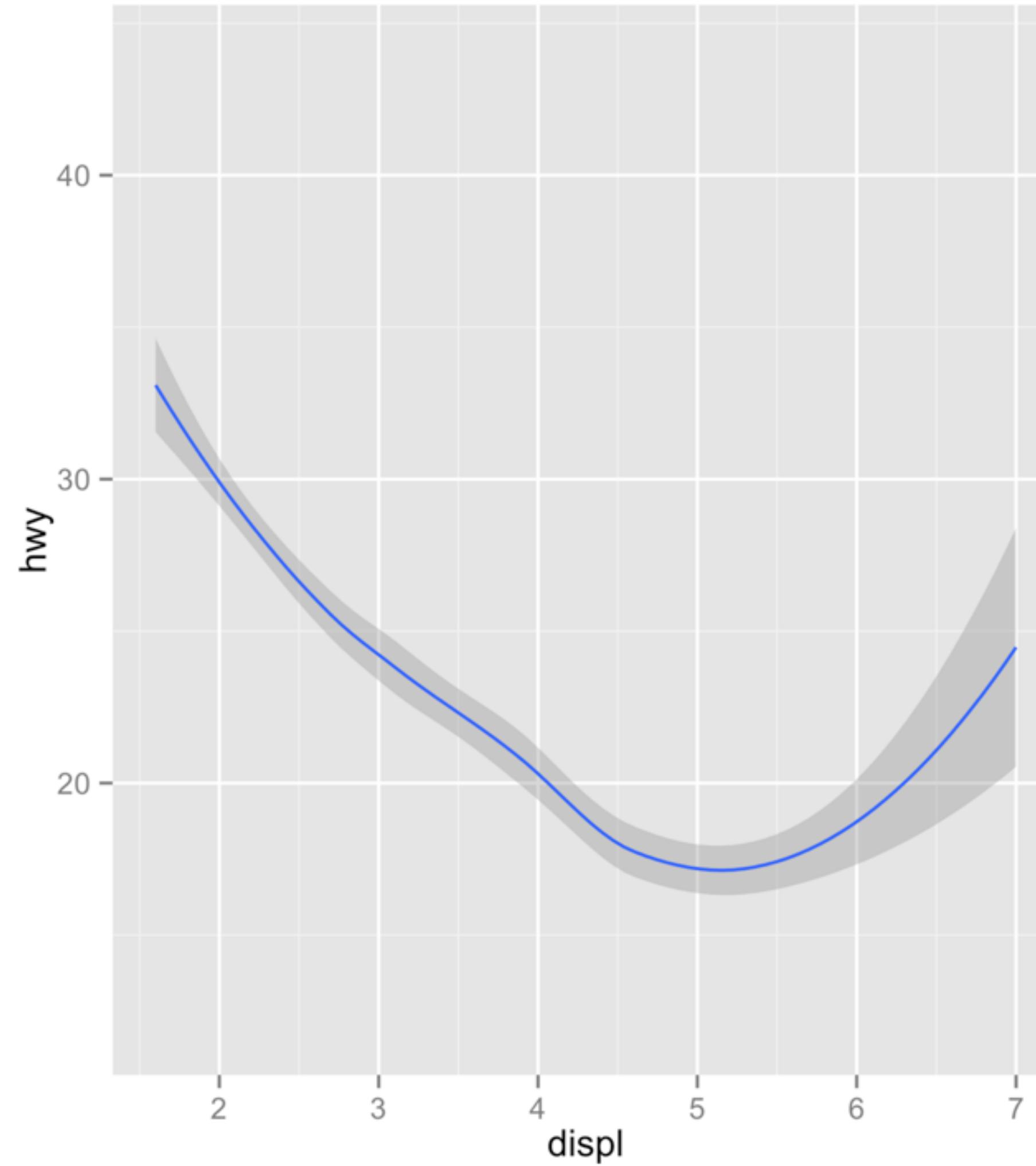
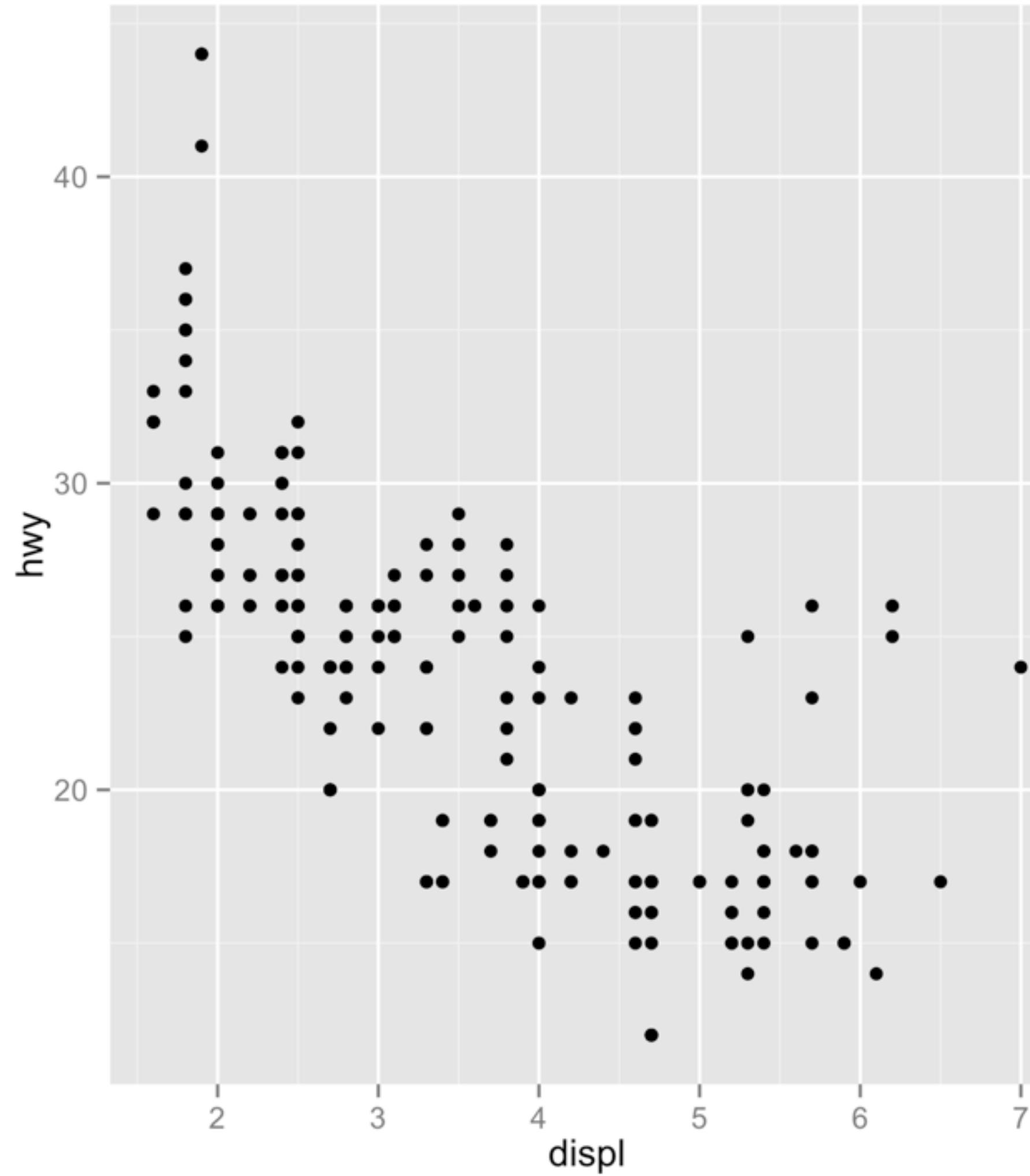
Come è possibile che
questi grafici siano simili?

Stesso: x, y, data



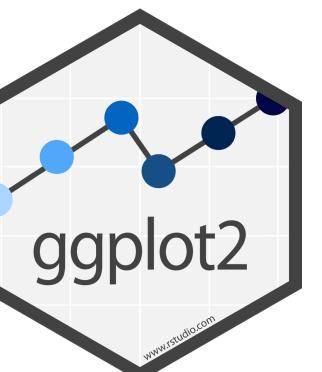
Cosa differenzia questi grafici?

Differenze: elementi geometrici (geom),
E.s. l'oggetto visivo utilizzato per rappresentare i dati



geoms

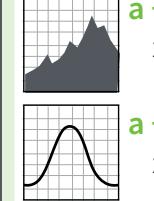
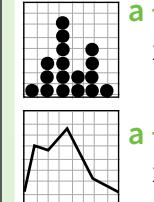
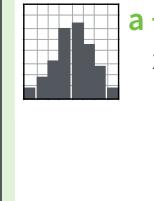
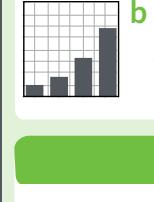
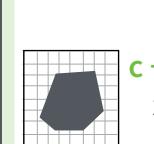
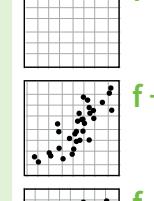
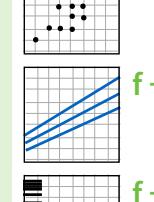
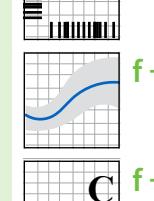
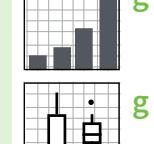
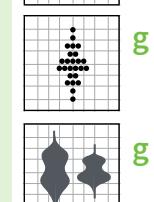
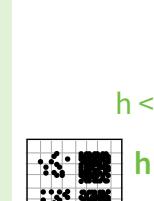
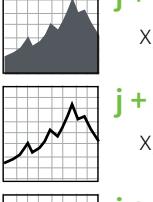
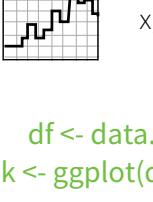
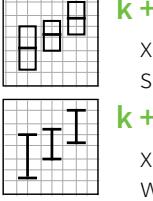
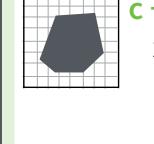
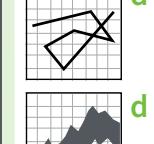
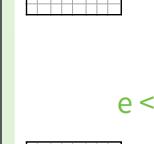
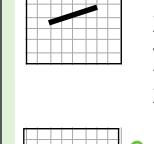
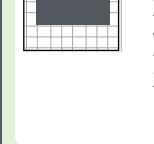
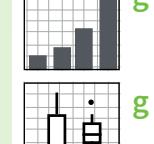
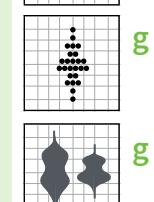
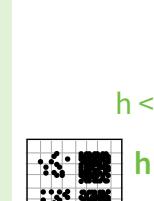
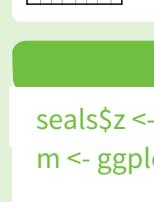
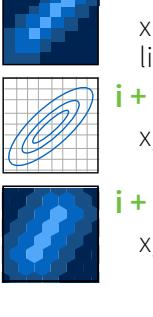
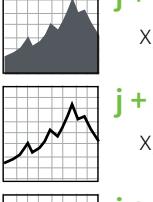
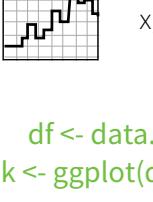
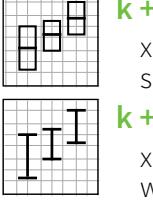
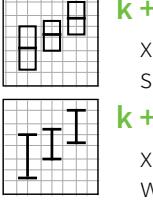
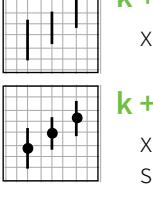
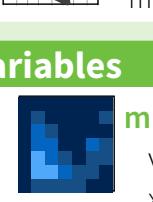
```
ggplot(data = <DATA>) +  
<GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))
```

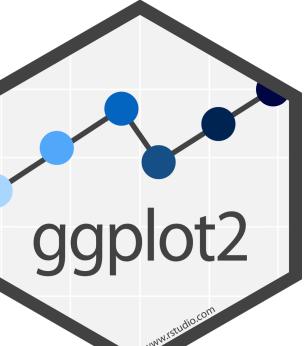


geom_ functions

Ognuno richiede un argomento di mappatura.

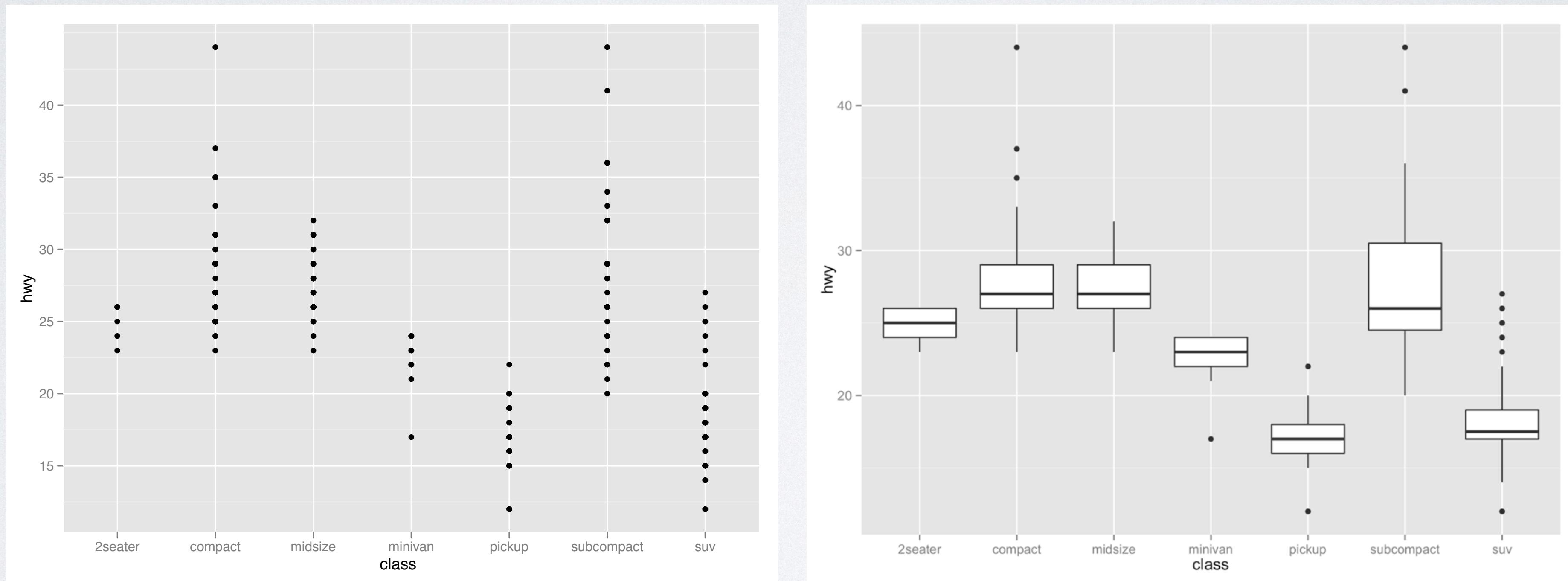


Geoms - Use a geom to represent data points, use the geom's aesthetic properties to represent variables. Each function returns a layer.	
One Variable <ul style="list-style-type: none"> Continuous <pre>a <- ggplot(mpg, aes(hwy))</pre>      Discrete <pre>b <- ggplot(mpg, aes(fl))</pre> <pre>b + geom_bar()</pre>  	Two Variables <ul style="list-style-type: none"> Continuous X, Continuous Y <pre>f <- ggplot(mpg, aes(cty, hwy))</pre>        Continuous Function <pre>j <- ggplot(economics, aes(date, unemploy))</pre>   
Graphical Primitives <ul style="list-style-type: none"> Continuous X, Continuous Y <pre>map <- map_data("state")</pre> <pre>c <- ggplot(map, aes(long, lat))</pre>  Continuous X, Discrete Y <pre>d <- ggplot(economics, aes(date, unemploy))</pre>   Discrete X, Continuous Y <pre>e <- ggplot(seals, aes(x = long, y = lat))</pre>   	Discrete X, Continuous Y <pre>g <- ggplot(mpg, aes(class, hwy))</pre>    
<ul style="list-style-type: none"> Discrete X, Discrete Y <pre>h <- ggplot(diamonds, aes(cut, color))</pre>  Three Variables <pre>seals\$z <- with(seals, sqrt(delta_long^2 + delta_lat^2))</pre> <pre>m <- ggplot(seals, aes(long, lat))</pre> 	Three Variables <ul style="list-style-type: none"> Continuous Bivariate Distribution <pre>i <- ggplot(movies, aes(year, rating))</pre>  Continuous Function <pre>j <- ggplot(economics, aes(date, unemploy))</pre>    Visualizing error <pre>df <- data.frame(grp = c("A", "B"), fit = 4:5, se = 1:2)</pre> <pre>k <- ggplot(df, aes(grp, fit, ymin = fit-se, ymax = fit+se))</pre>     Maps <pre>data <- data.frame(murder = USArrests\$Murder, state = tolower(rownames(USArrests)))</pre> <pre>map <- map_data("state")</pre> <pre>l <- ggplot(data, aes(fill = murder))</pre>

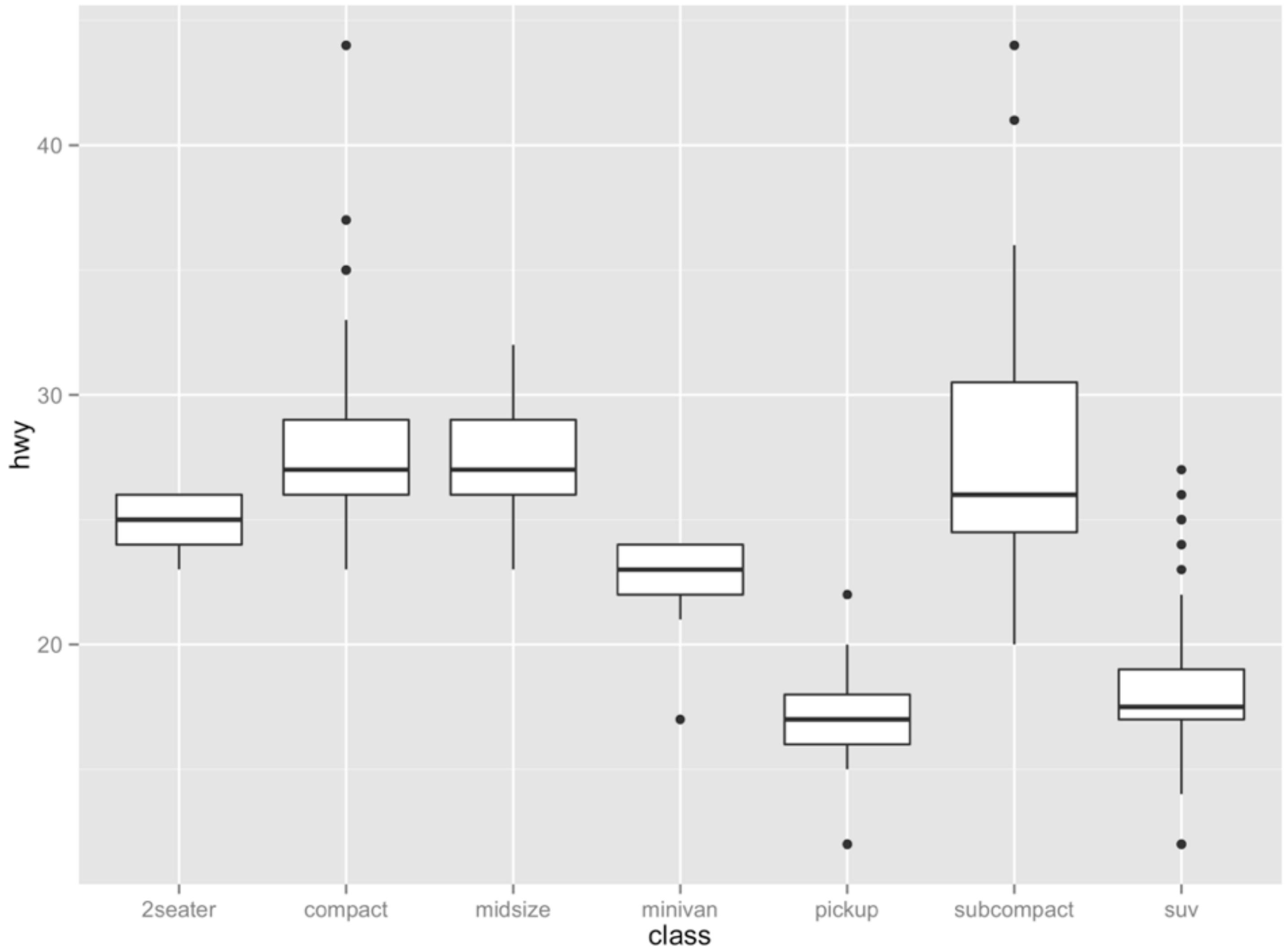


Tocca a te III

Con il tuo partner, pensate come sostituire questo grafico a dispersione (scatterplot) con uno che disegna grafici a scatole (boxplot). Usa il cheatsheet di prima. Prova la tua ipotesi migliore



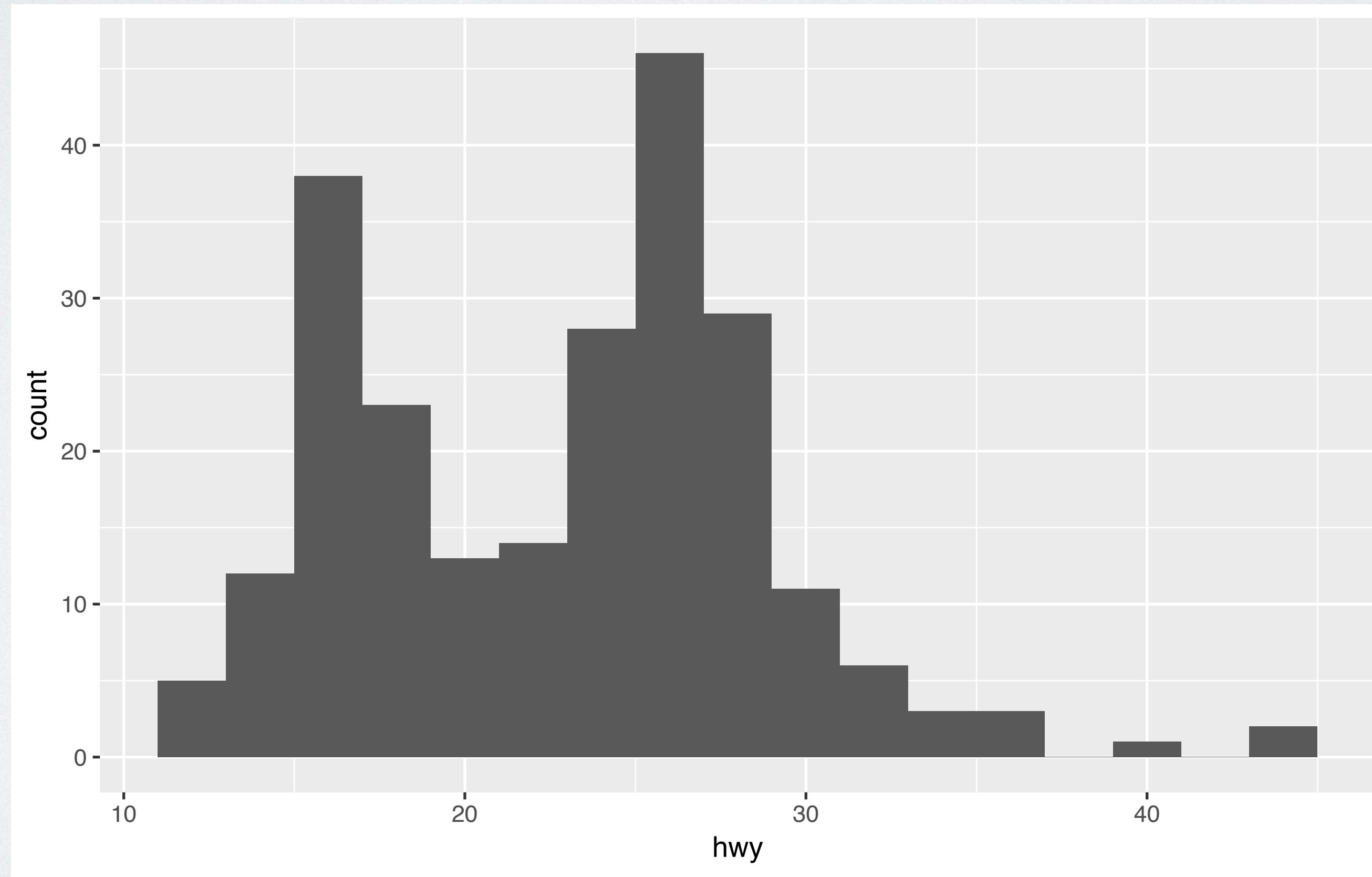
```
ggplot(mpg) + geom_point(aes(x= class, y= hwy))
```

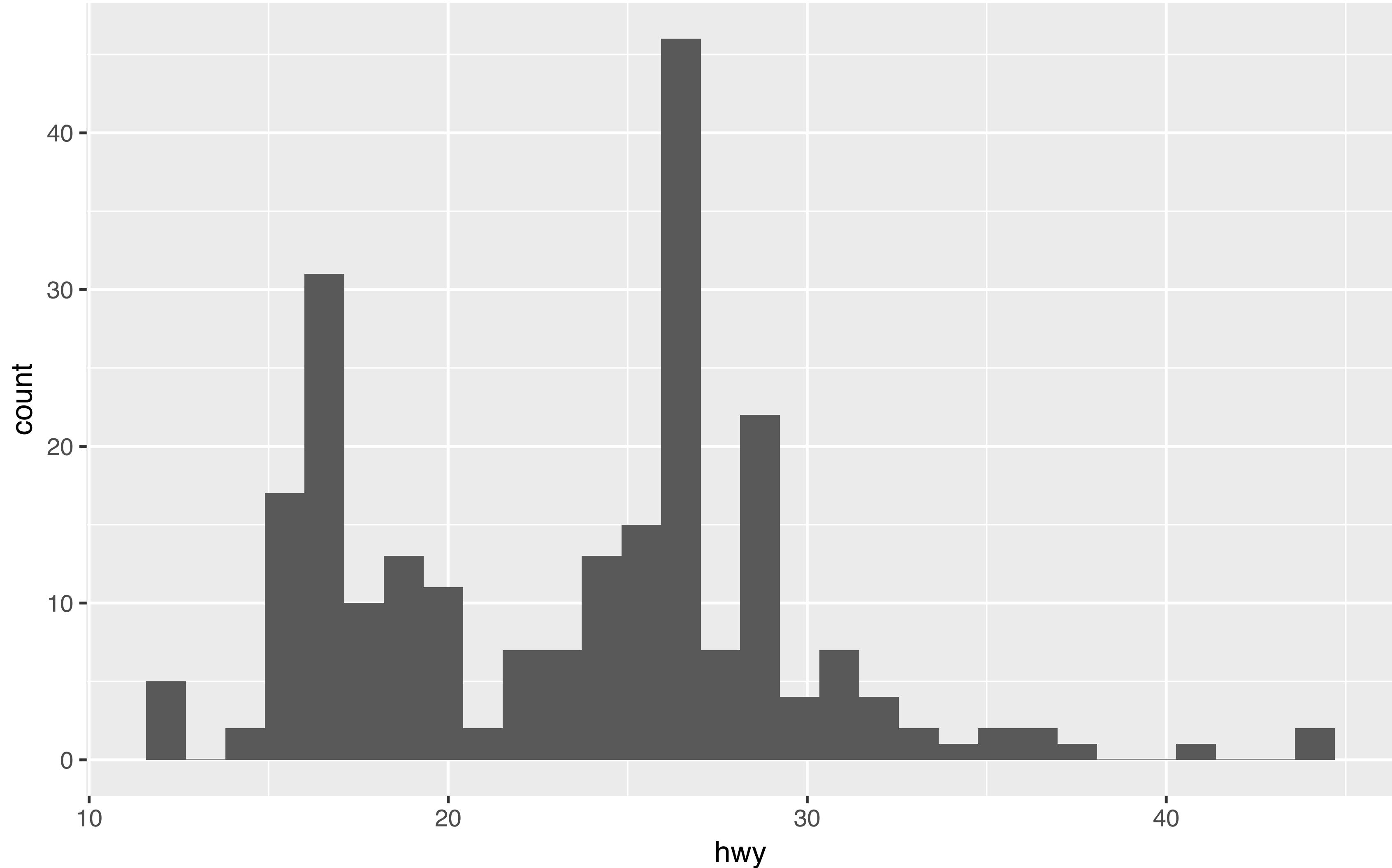


```
ggplot(data = mpg) +  
  geom_boxplot(mapping = aes(x = class, y = hwy))
```

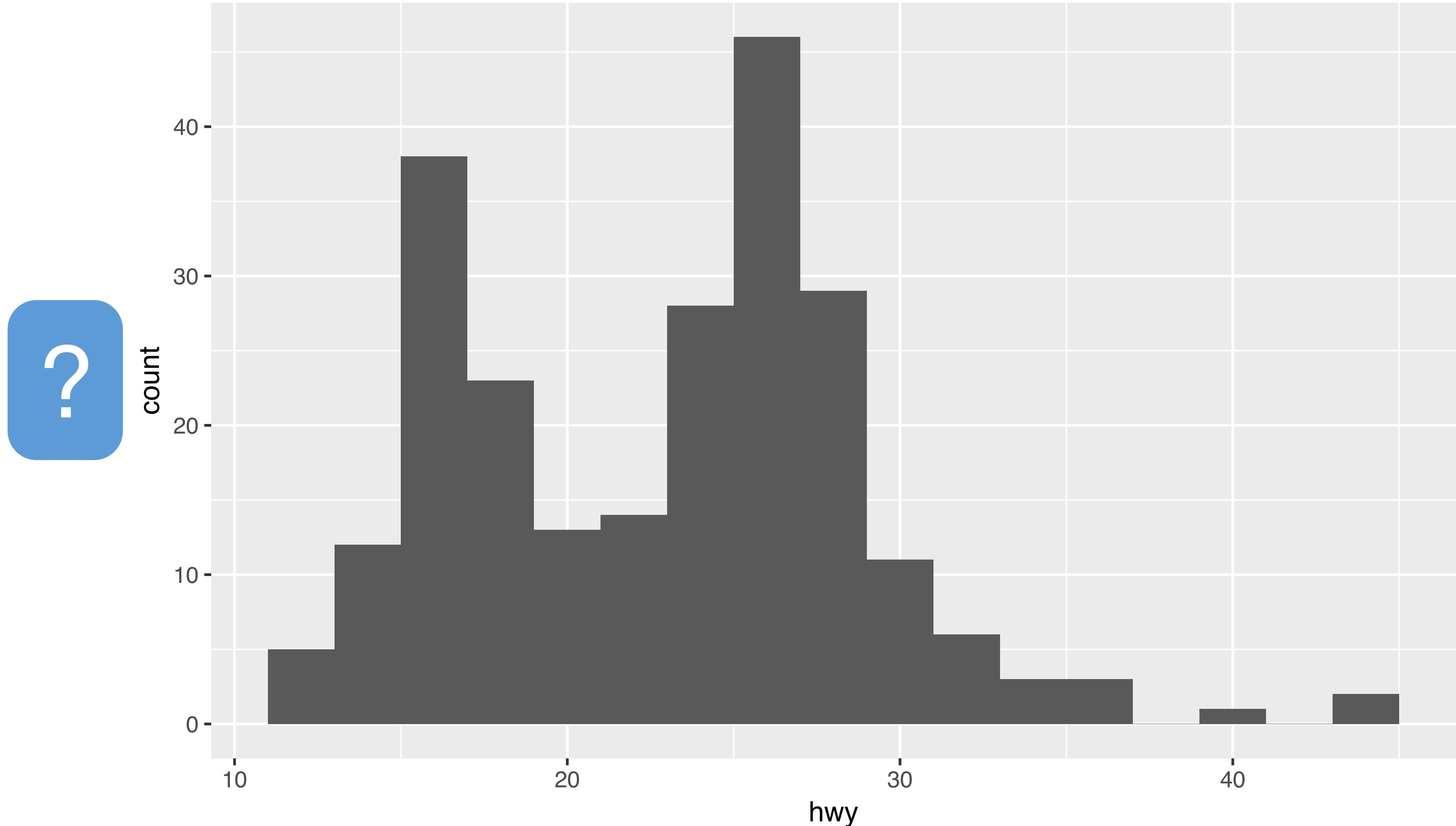
Tocca a te IV

Con il tuo partner, crea l'istogramma di `hwy` di seguito. Usa il cheatsheet. Suggerimento: non fornire alcuna variabile.

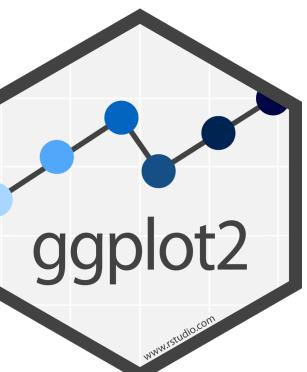




```
ggplot(data = mpg) +  
  geom_histogram(mapping = aes(x = hwy))
```

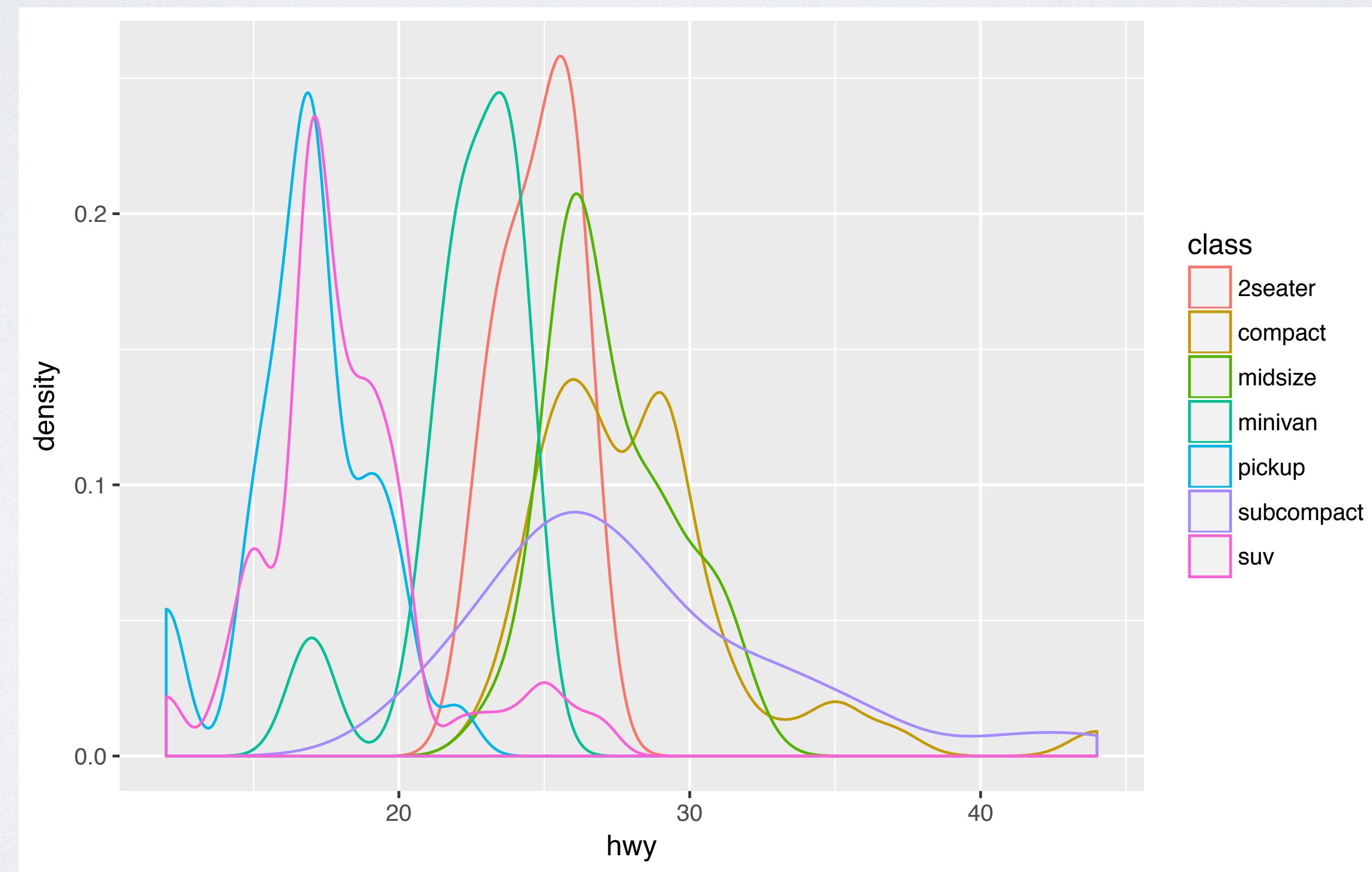


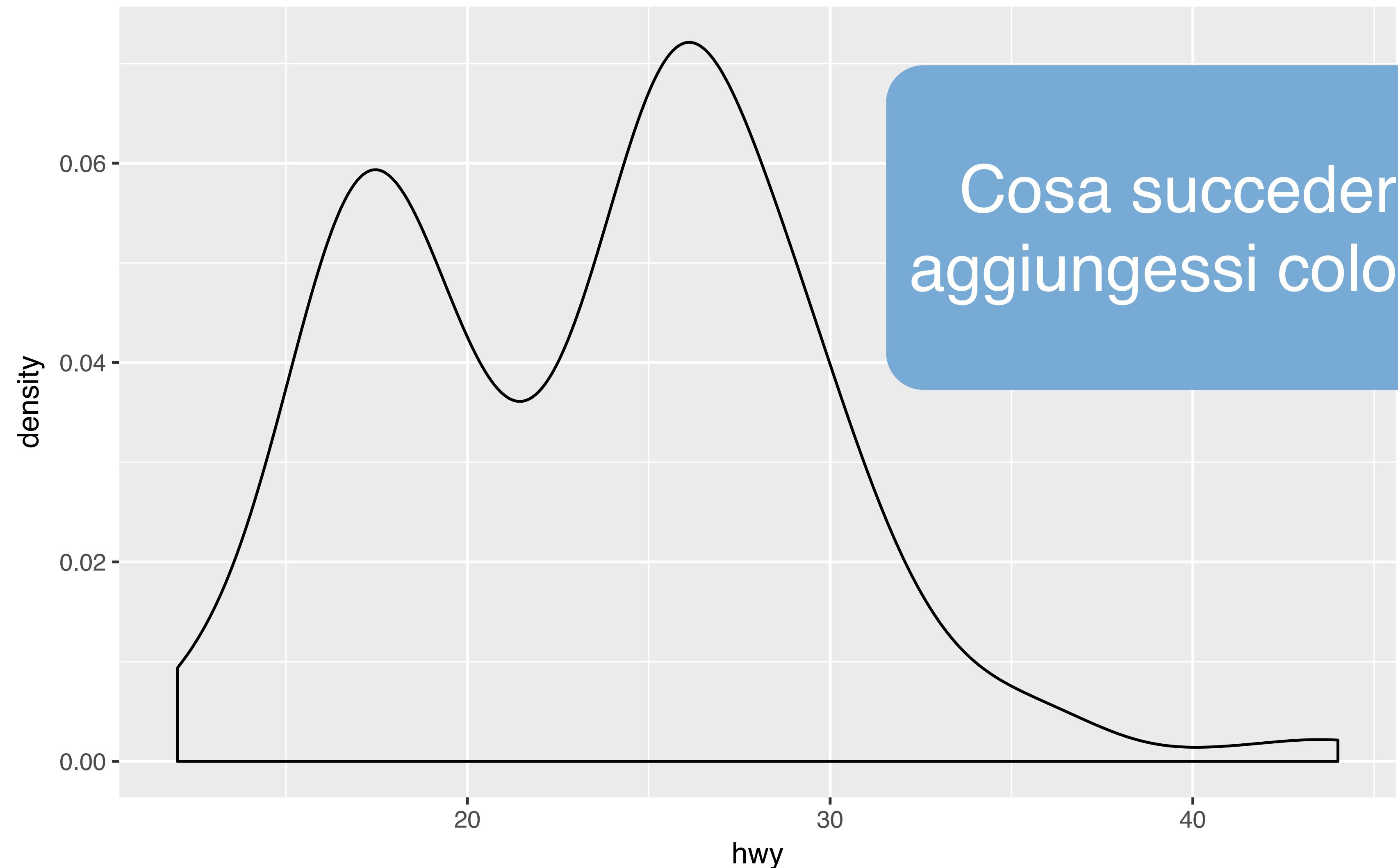
```
ggplot(data = mpg) +  
  geom_histogram(mapping = aes(x = hwy), binwidth = 2)
```



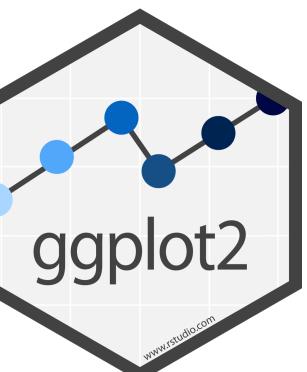
Tocca a te V

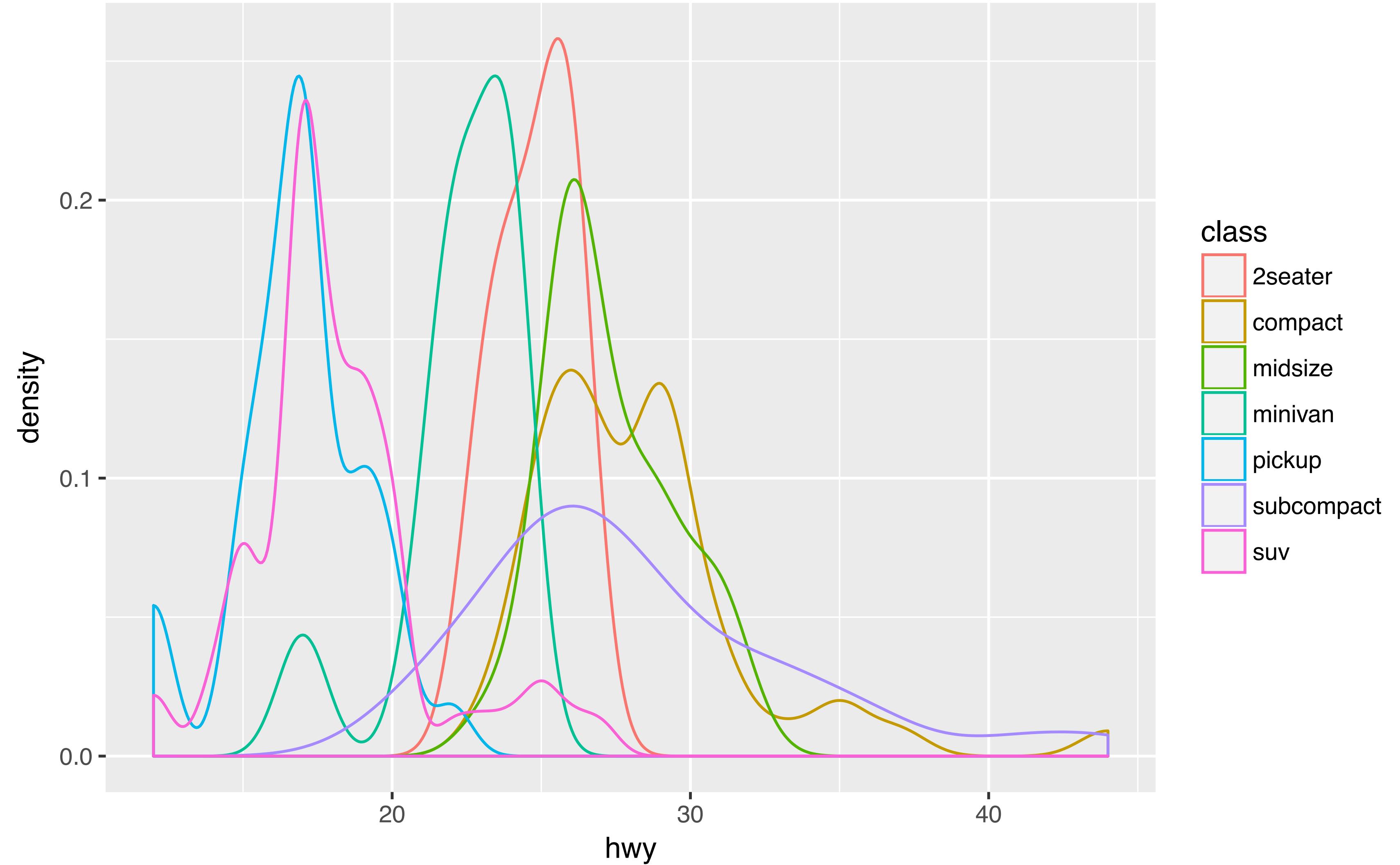
Con il tuo partner, crea la trama della densità di hwy colorata per classe di seguito. Usa il cheatsheet. Prova la tua ipotesi migliore.



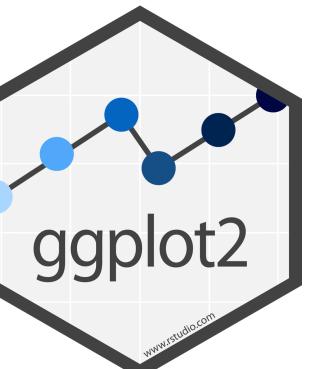


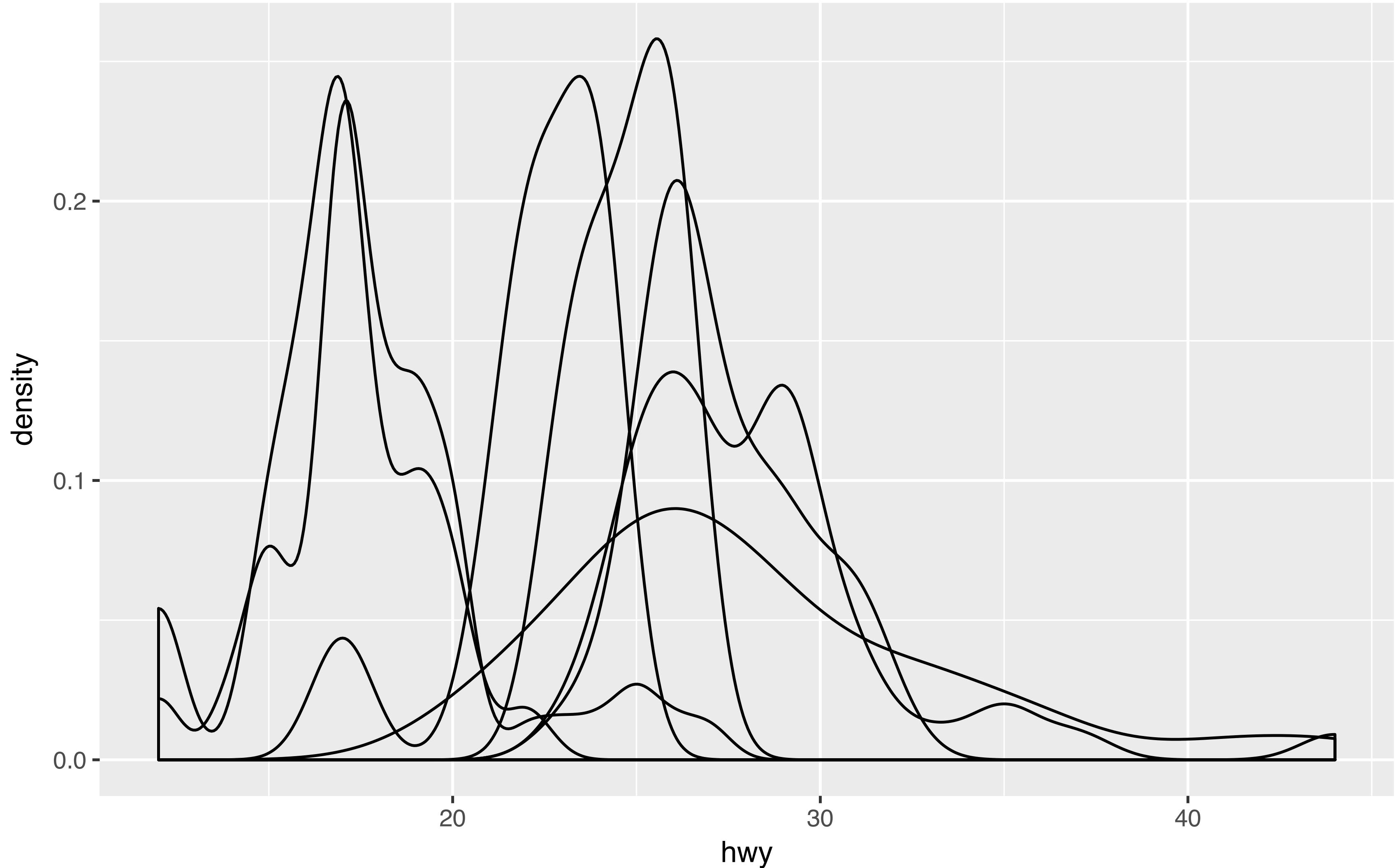
```
ggplot(data = mpg) +  
  geom_density(mapping = aes(x = hwy))
```



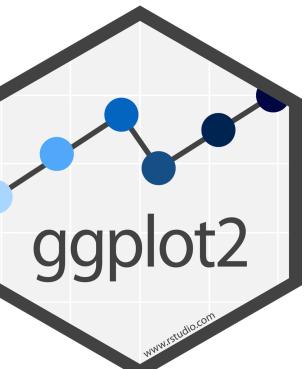


```
ggplot(data = mpg) +  
  geom_density(mapping = aes(x = hwy, color = class))
```



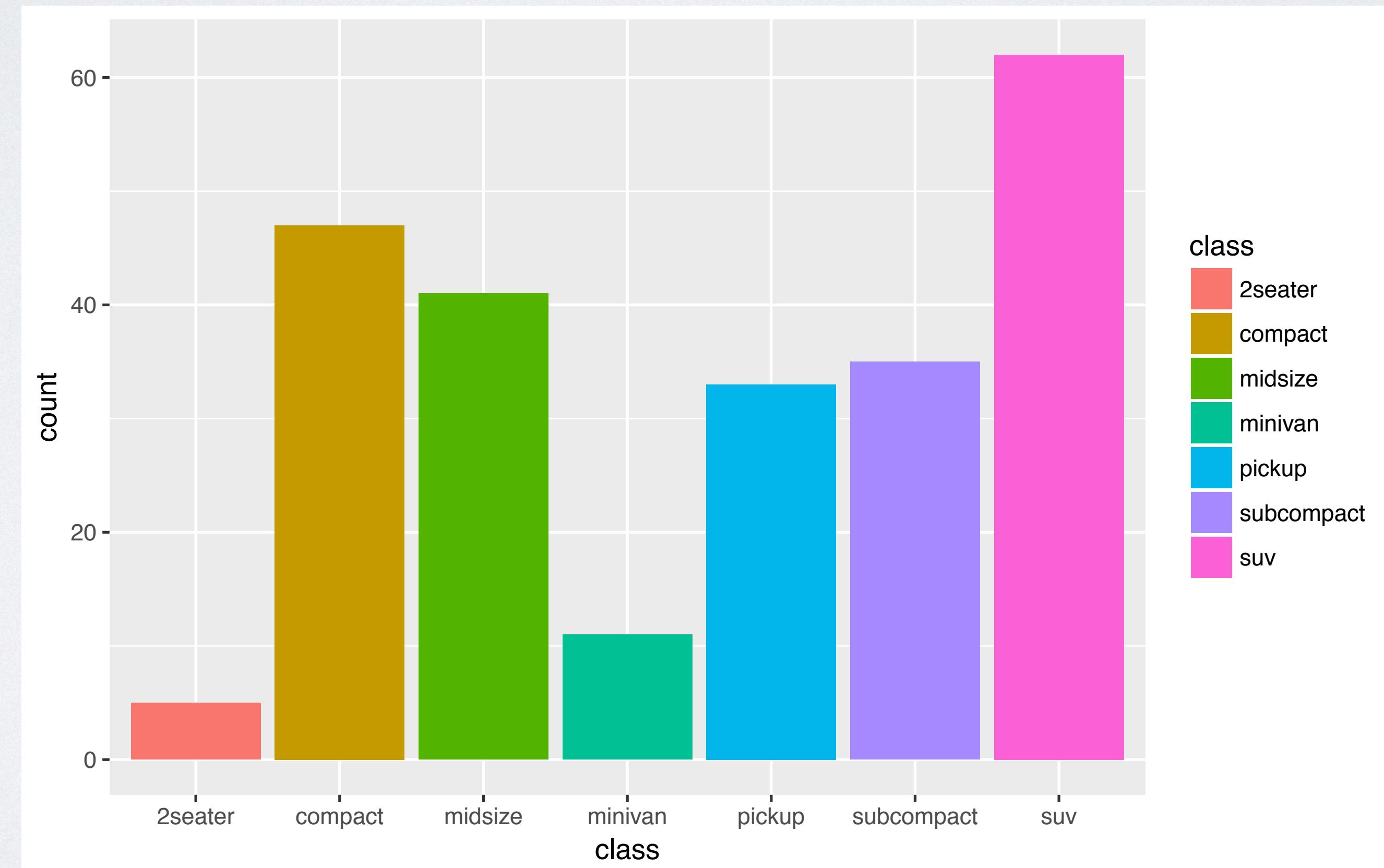


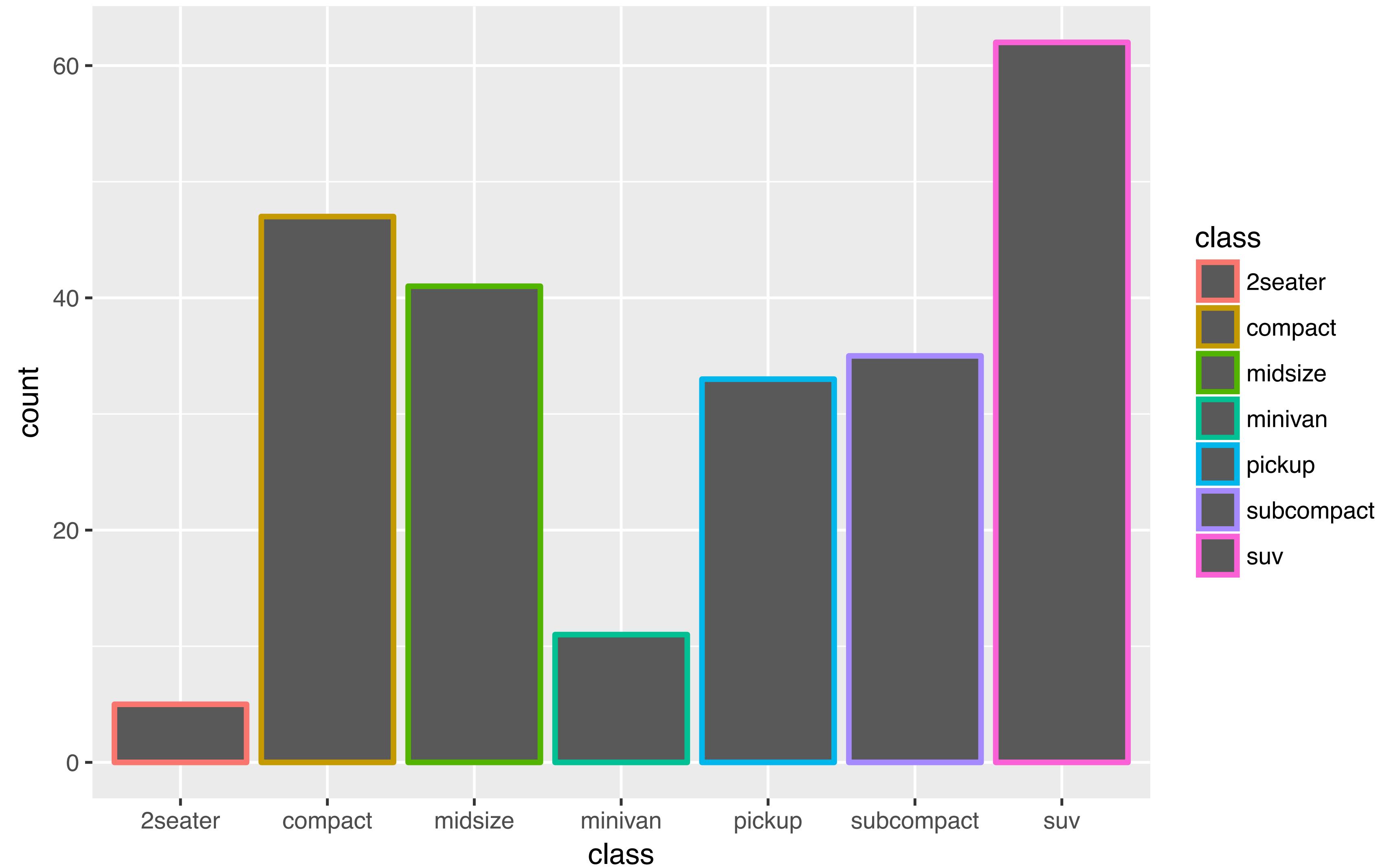
```
ggplot(data = mpg) +  
  geom_density(mapping = aes(x = hwy, group = class))
```



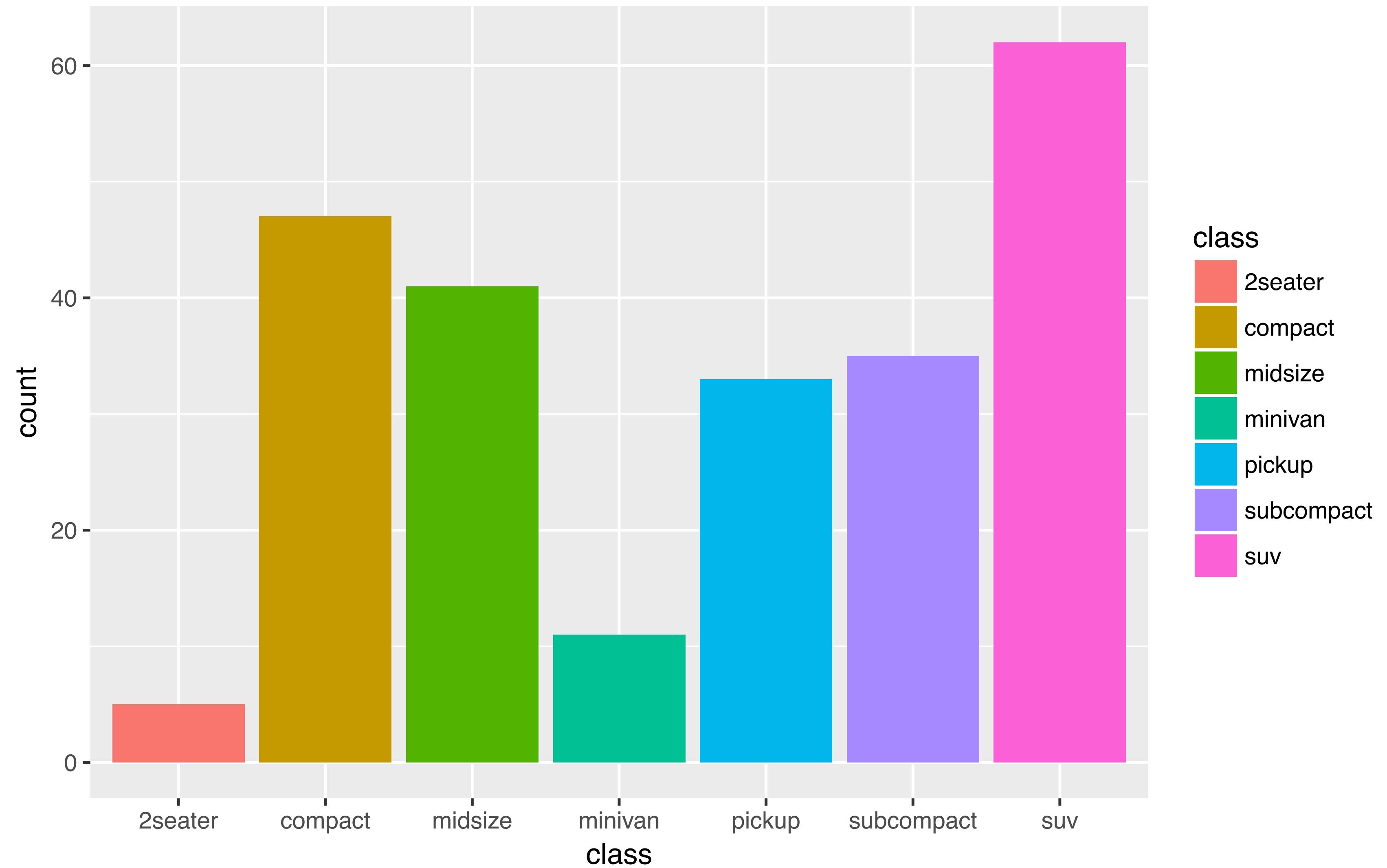
Tocca a te VI

Con il tuo partner, crea il grafico a barre della classe colorato per classe di seguito. Usa il cheatsheet. Prova la tua ipotesi migliore.

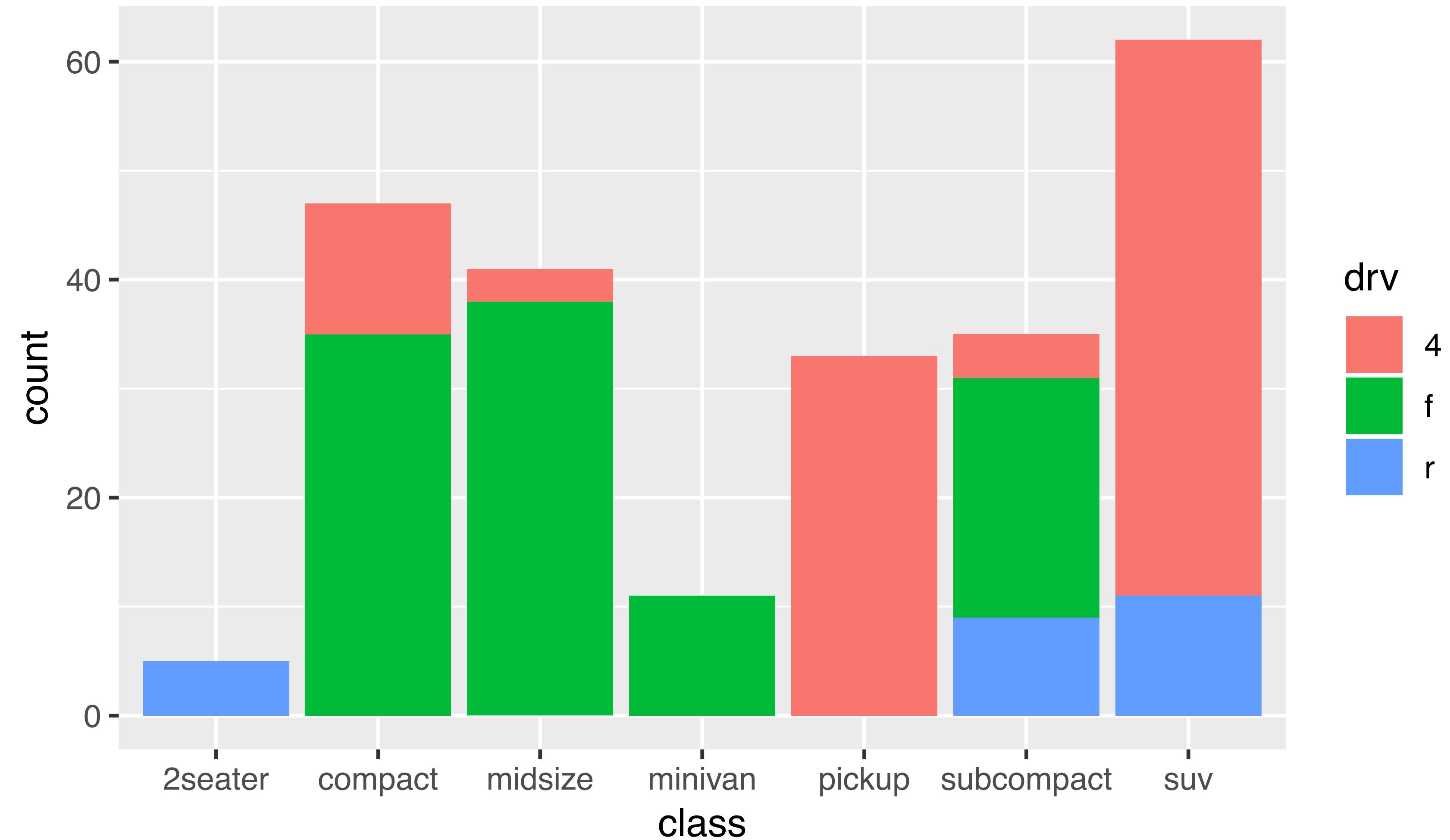




```
ggplot(data = mpg) +  
  geom_bar(mapping = aes(x = class, color = class))
```



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

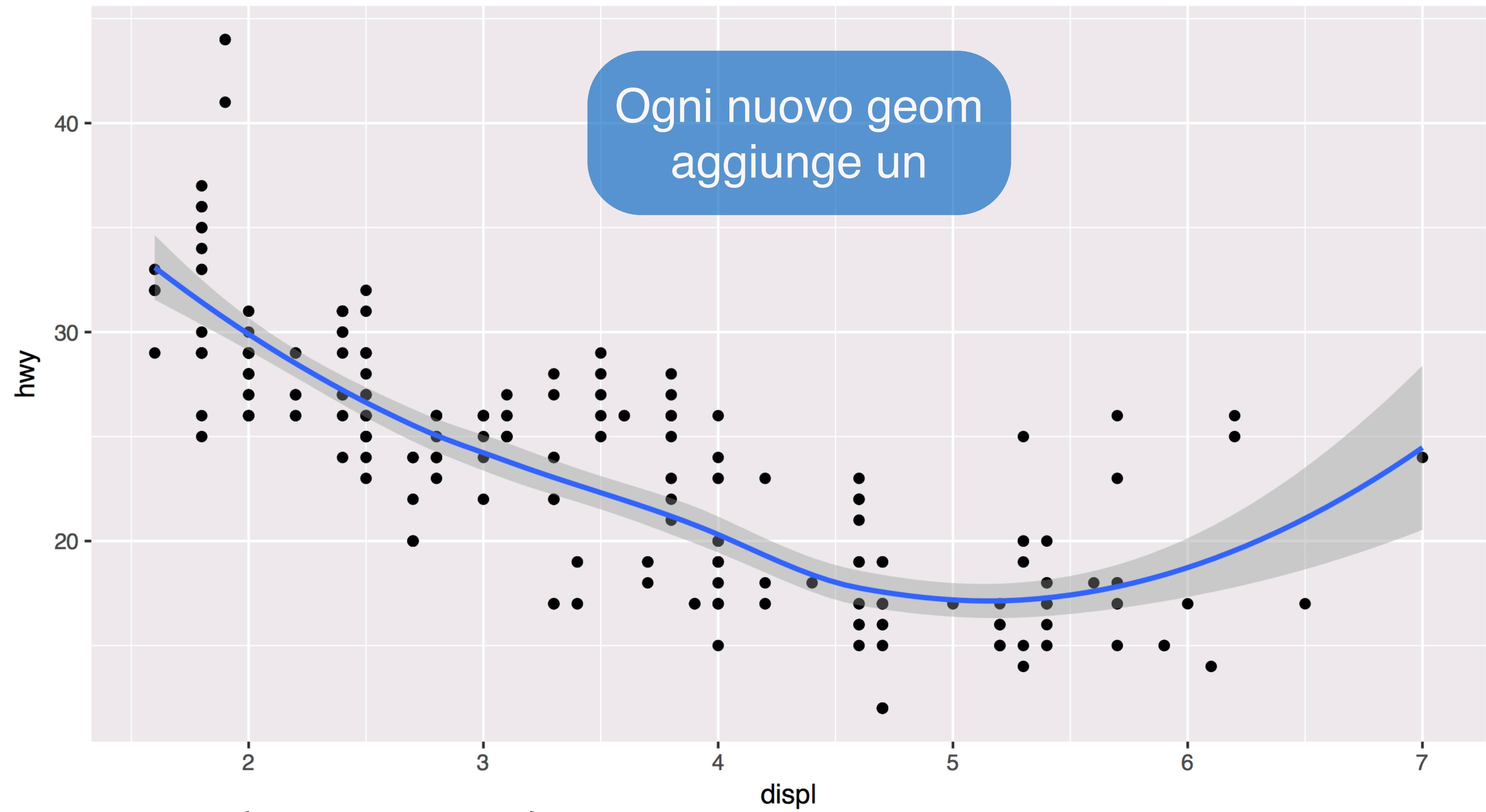


```
ggplot(data = mpg) +  
  geom_bar(mapping = aes(x = class, fill = drv))
```

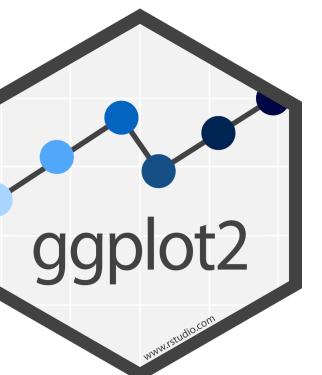
Quiz?

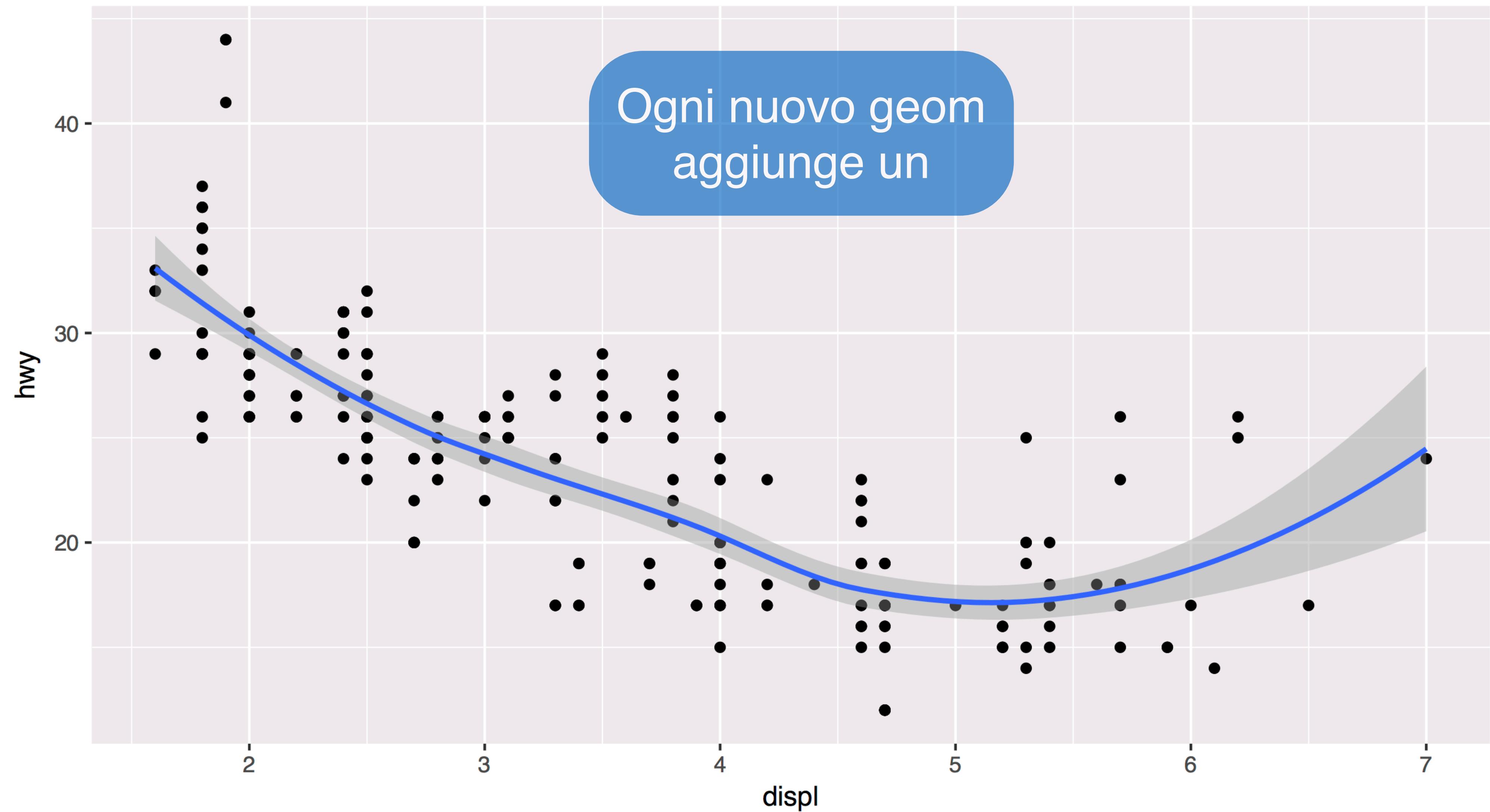
Cosa fa questo codice?

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

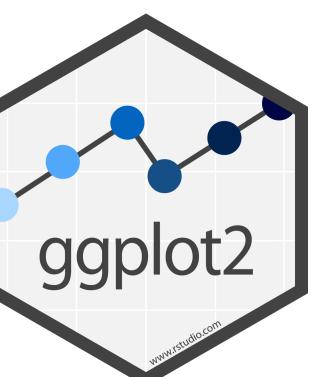


```
ggplot(data = mpg) +  
  geom_point(mapping = aes(x = displ, y = hwy)) +  
  geom_smooth(mapping = aes(x = displ, y = hwy))
```



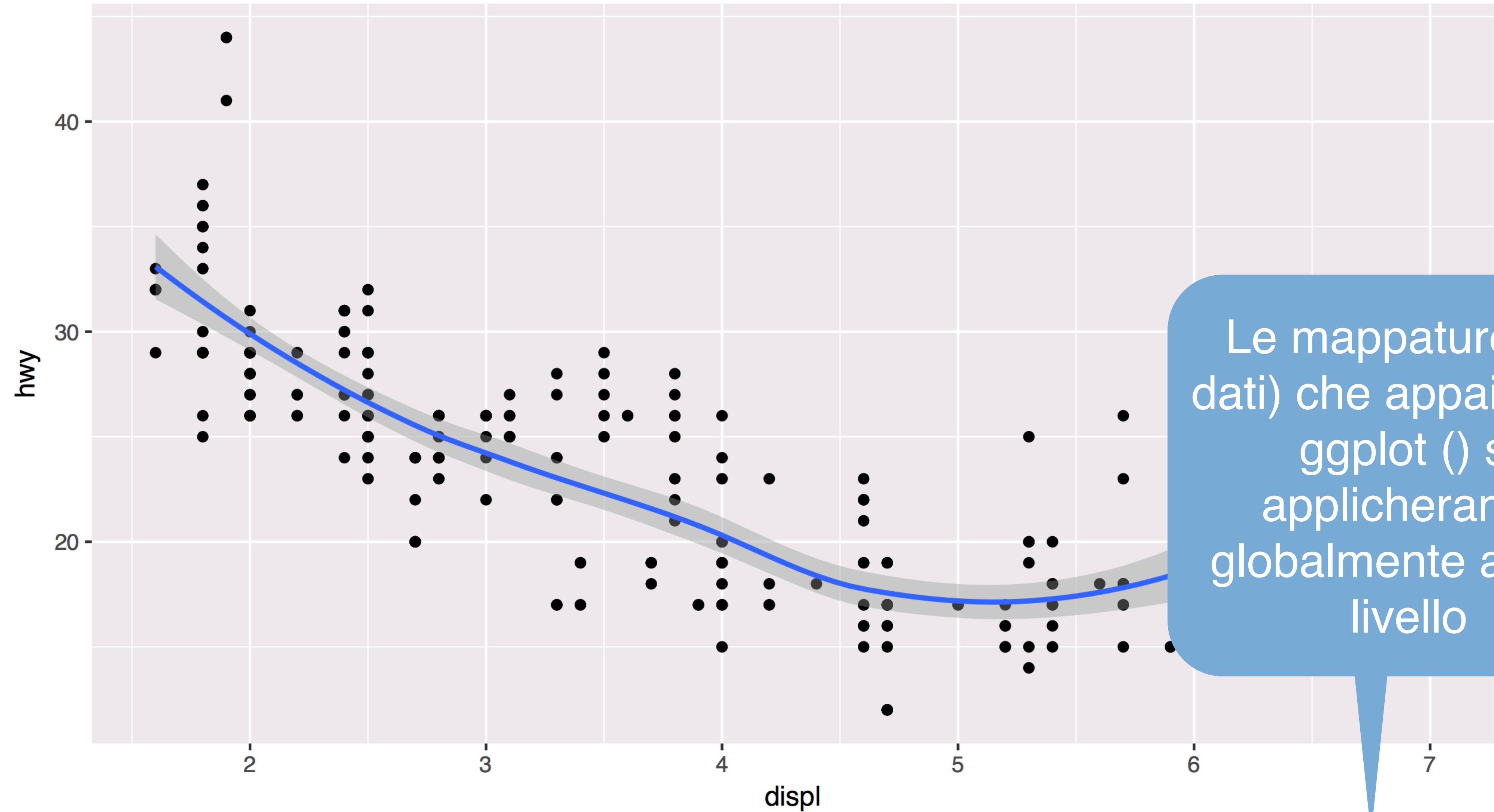


```
ggplot(data = mpg) +  
  geom_point(mapping = aes(x = displ, y = hwy)) +  
  geom_smooth(mapping = aes(x = displ, y = hwy))
```

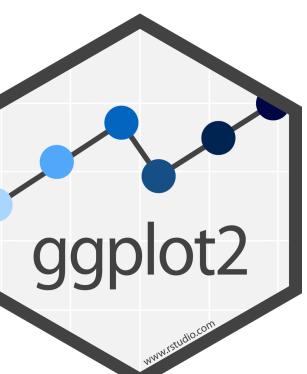


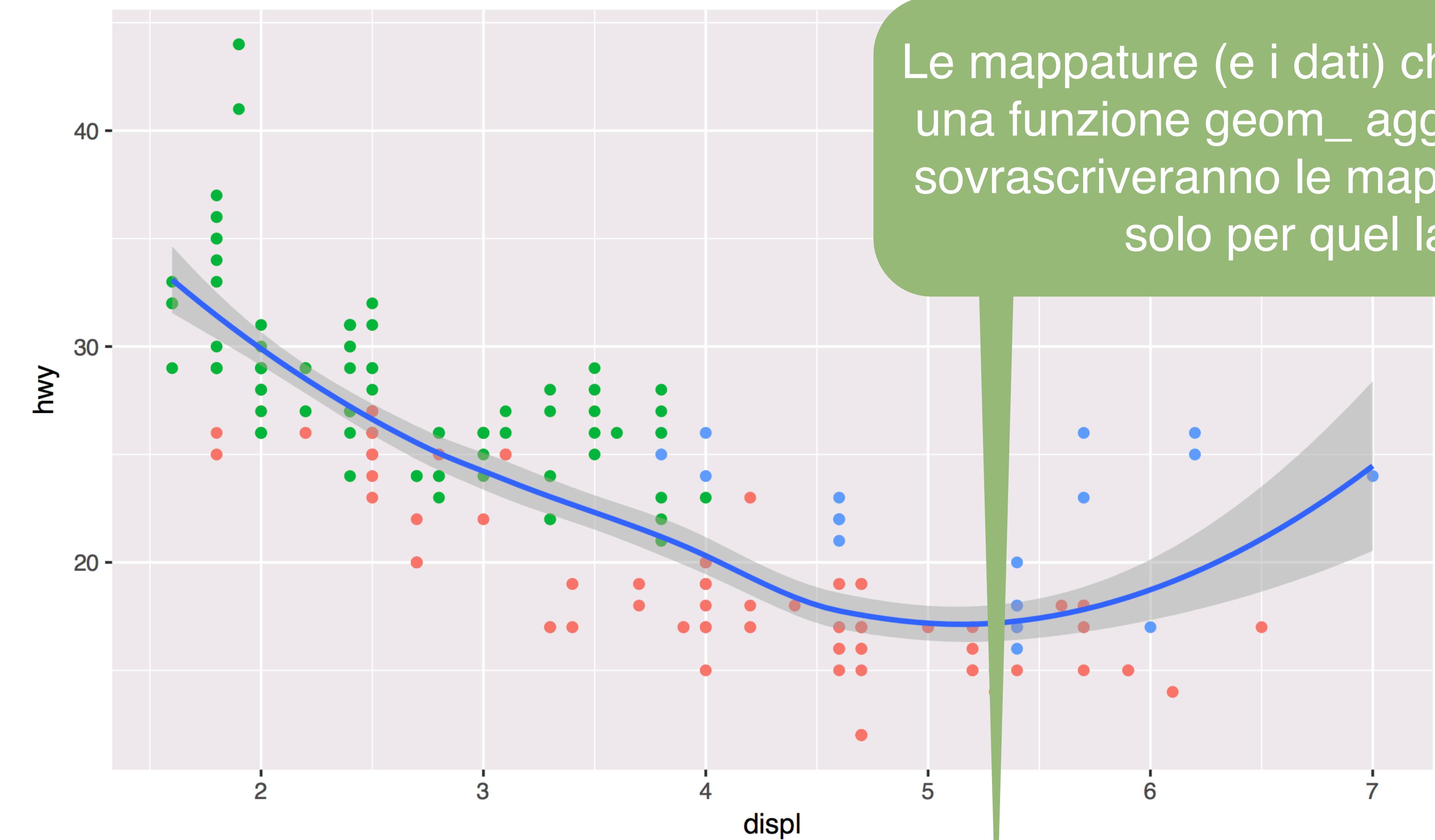
global vs. local

R



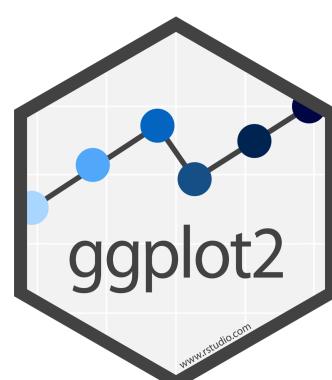
```
ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
  geom_point() +  
  geom_smooth()
```

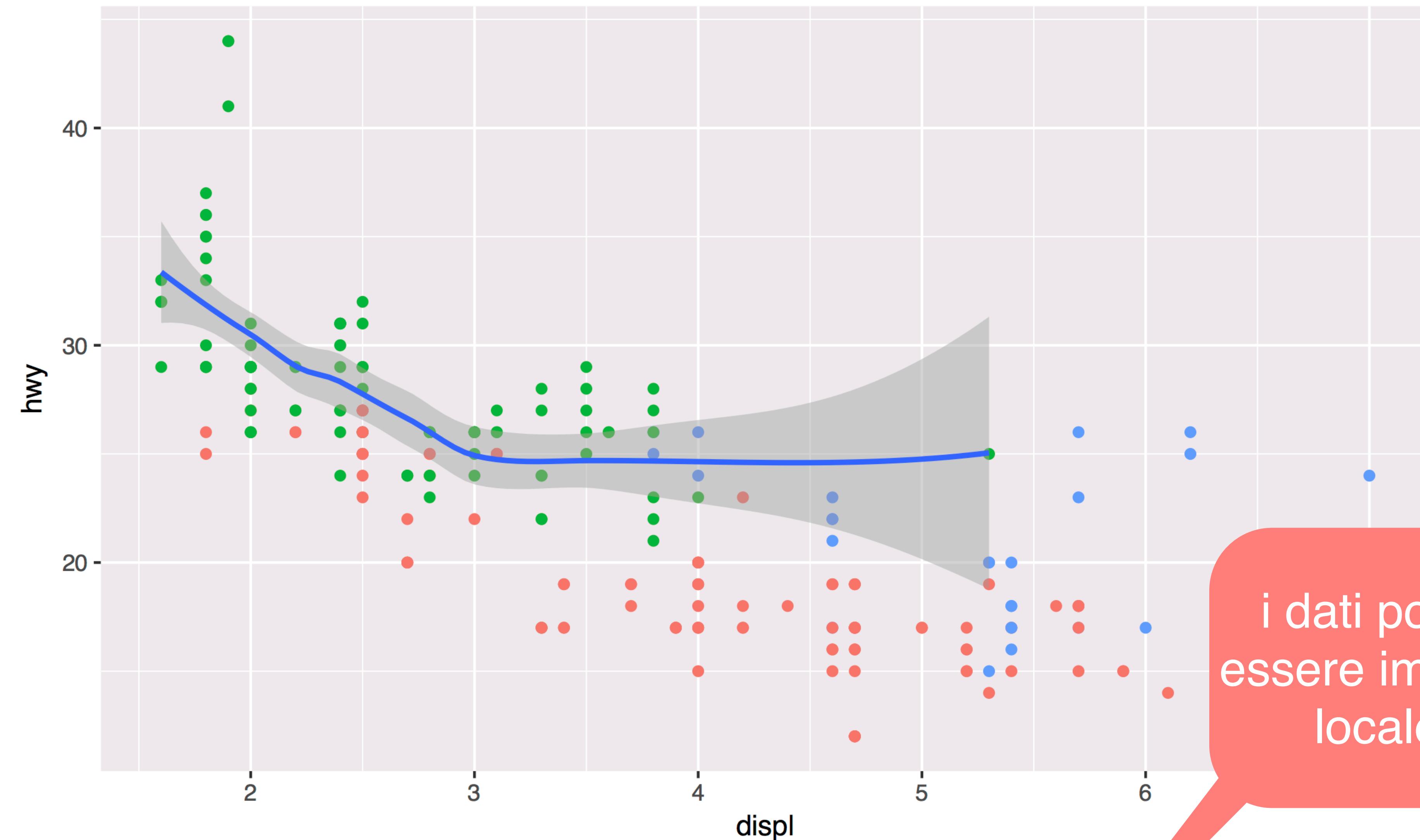




```
ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
  geom_point(mapping = aes(color = drv)) +  
  geom_smooth()
```

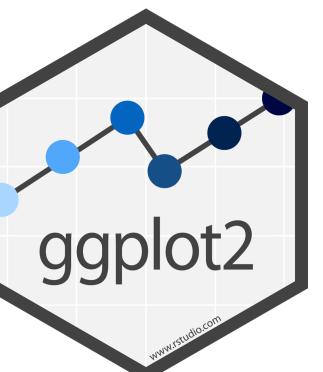
Le mappature (e i dati) che appaiono in una funzione `geom_` aggiungeranno o sovrascriveranno le mappature globali solo per quel layer





i dati possono anche essere impostati a livello locale o globale

```
ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
  geom_point(mapping = aes(color = drv)) +  
  geom_smooth(data = filter(mpg, drv == "f"))
```

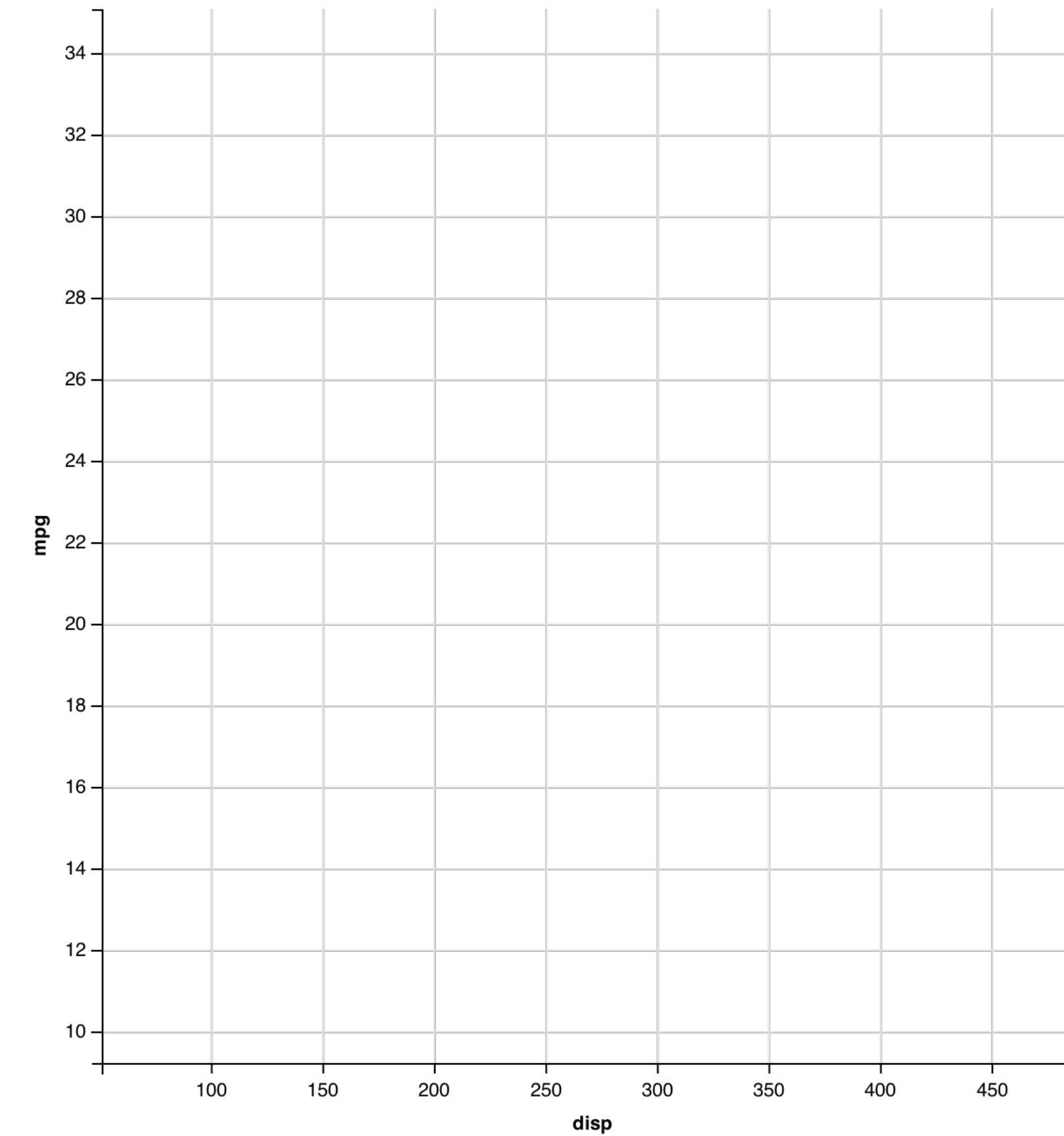


Grammatica di grafica

mpg	cyl	disp	hp
21,0	6	160,0	2
21,0	6	160,0	2
22,8	4	108,0	1
21,4	6	258,0	2
18,7	8	360,0	3
18,1	6	225,0	2
14,3	8	360,0	5
24,4	4	146,7	1
22,8	4	140,8	1
19,2	6	167,6	2
17,8	6	167,6	2
16,4	8	275,8	3
17,3	8	275,8	3
15,2	8	275,8	3
10,4	8	472,0	4
10,4	8	460,0	4
14,7	8	440,0	4
32,4	4	78,7	1
30,4	4	75,7	1
33,9	4	71,1	1

data

geom

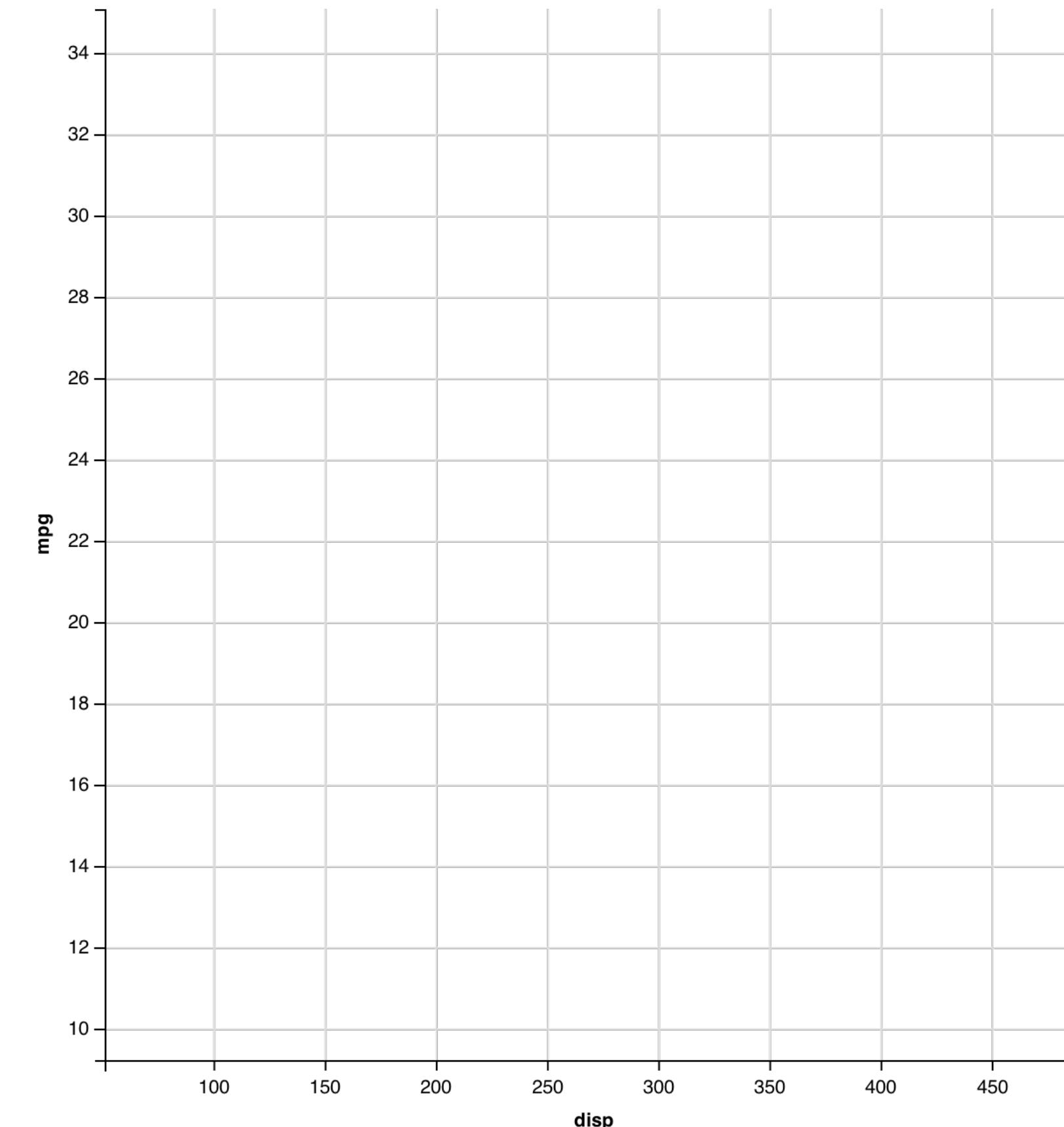


mappings

mpg	cyl	disp	hp	fill
21,0	6	160,0	2	blue
21,0	6	160,0	2	blue
22,8	4	108,0	1	light green
21,4	6	258,0	2	blue
18,7	8	360,0	3	red
18,1	6	225,0	2	blue
14,3	8	360,0	5	purple
24,4	4	146,7	1	light green
22,8	4	140,8	1	light green
19,2	6	167,6	2	blue
17,8	6	167,6	2	blue
16,4	8	275,8	3	red
17,3	8	275,8	3	red
15,2	8	275,8	3	red
10,4	8	472,0	4	yellow-green
10,4	8	460,0	4	yellow-green
14,7	8	440,0	4	yellow-green
32,4	4	78,7	1	light green
30,4	4	75,7	1	light green
33,9	4	71,1	1	light green

data

geom

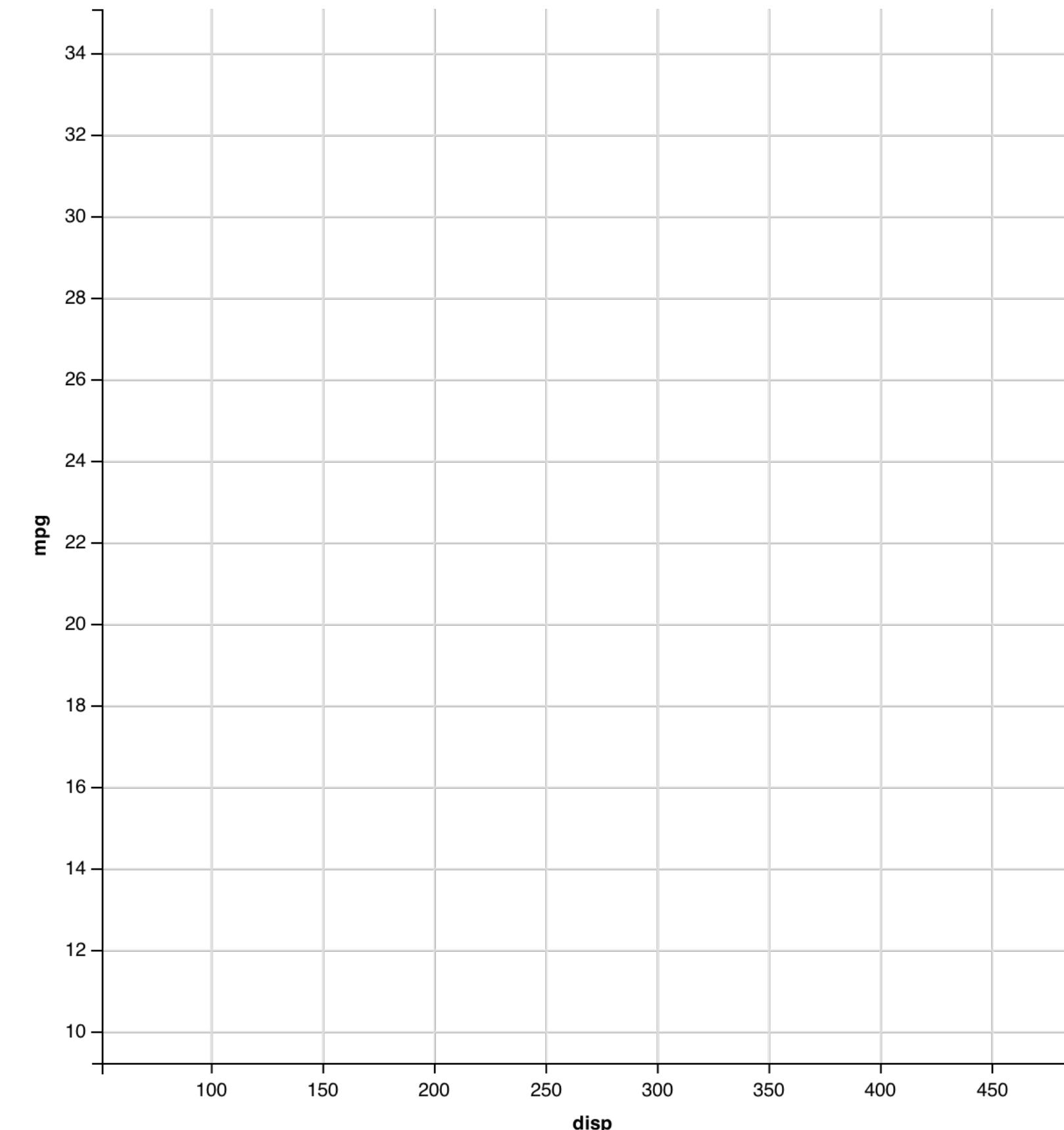


mappings

	shape	fill
mpg	cyl	hp
21,0	6 +	160,0
21,0	6 +	160,0
22,8	4 ●	108,0
21,4	6 +	258,0
18,7	8 ♦	360,0
18,1	6 +	225,0
14,3	8 ♦	360,0
24,4	4 ●	146,7
22,8	4 ●	140,8
19,2	6 +	167,6
17,8	6 +	167,6
16,4	8 ♦	275,8
17,3	8 ♦	275,8
15,2	8 ♦	275,8
10,4	8 ♦	472,0
10,4	8 ♦	460,0
14,7	8 ♦	440,0
32,4	4 ●	78,7
30,4	4 ●	75,7
33,9	4 ●	71,1

data

geom

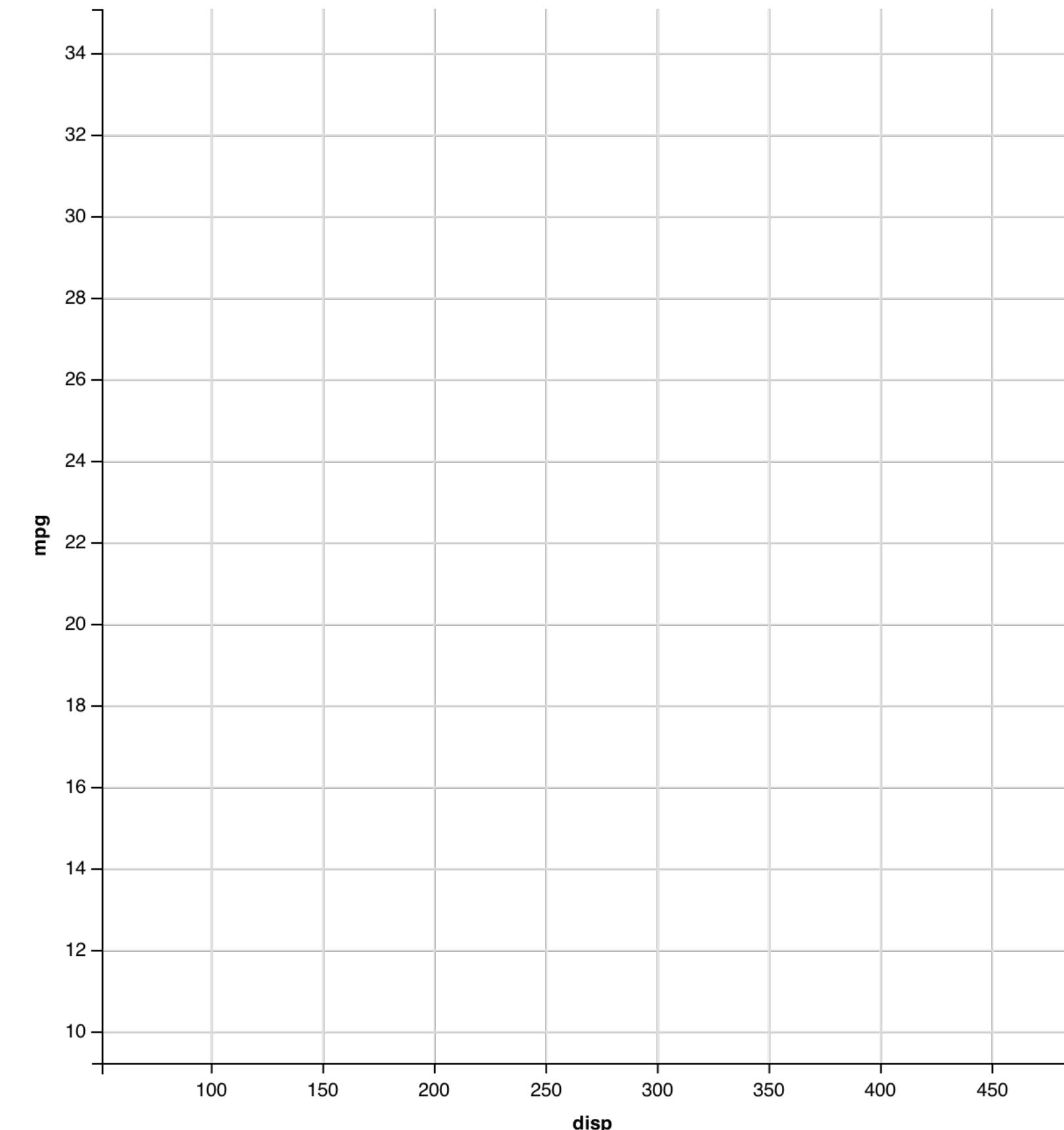


mappings

	shape	x	fill
mpg	cyl	disp	hp
21,0	6	160,0	2
21,0	6	160,0	2
22,8	4	108,0	1
21,4	6	258,0	2
18,7	8	360,0	3
18,1	6	225,0	2
14,3	8	360,0	5
24,4	4	146,7	1
22,8	4	140,8	1
19,2	6	167,6	2
17,8	6	167,6	2
16,4	8	275,8	3
17,3	8	275,8	3
15,2	8	275,8	3
10,4	8	472,0	4
10,4	8	460,0	4
14,7	8	440,0	4
32,4	4	78,7	1
30,4	4	75,7	1
33,9	4	71,1	1

data

geom

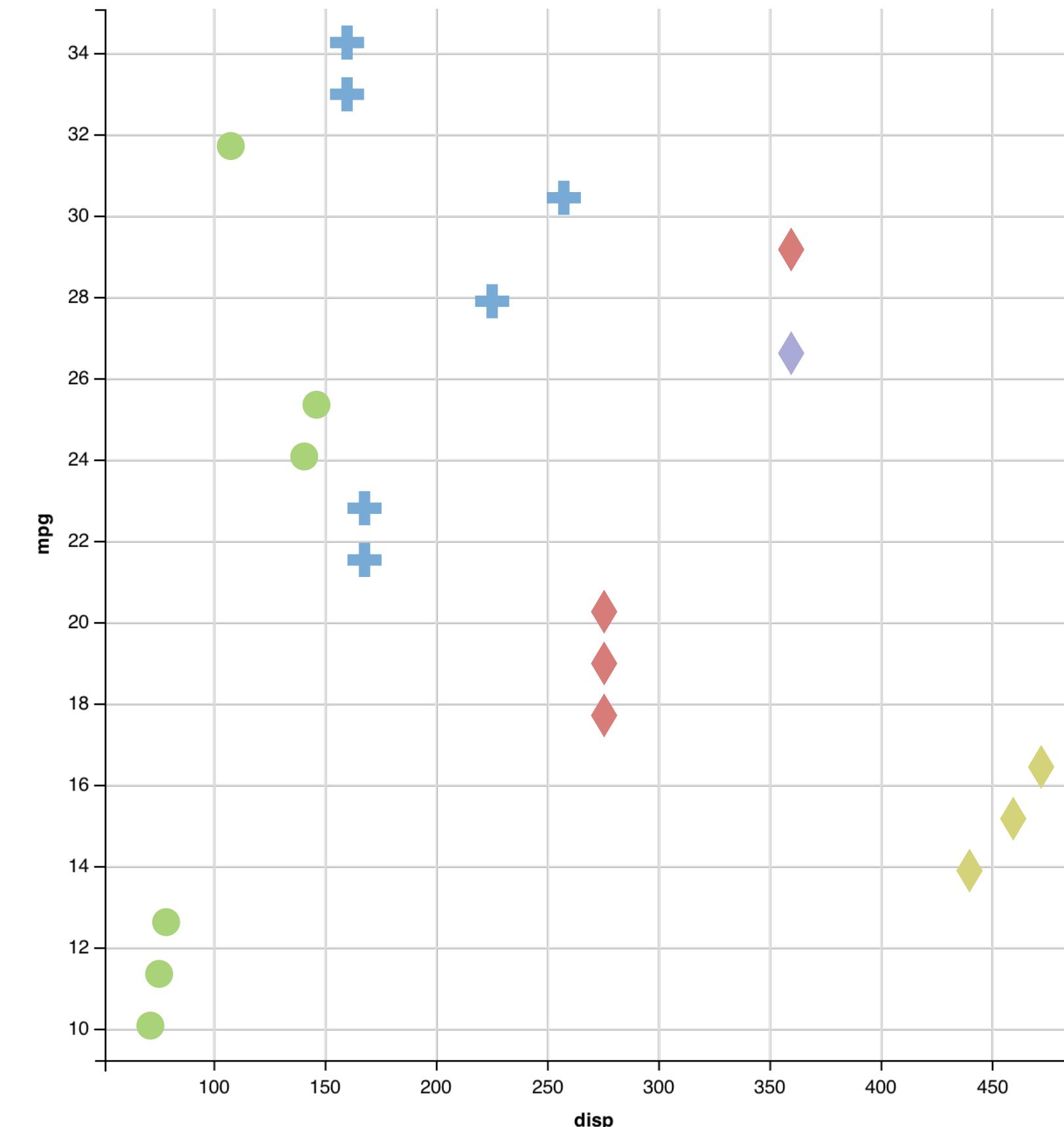


mappings

	y	shape	x	fill
	mpg	cyl	disp	hp
21,0	6	160,0	2	
21,0	6	160,0	2	
22,8	4	108,0	1	
21,4	6	258,0	2	
18,7	8	360,0	3	
18,1	6	225,0	2	
14,3	8	360,0	5	
24,4	4	146,7	1	
22,8	4	140,8	1	
19,2	6	167,6	2	
17,8	6	167,6	2	
16,4	8	275,8	3	
17,3	8	275,8	3	
15,2	8	275,8	3	
10,4	8	472,0	4	
10,4	8	460,0	4	
14,7	8	440,0	4	
32,4	4	78,7	1	
30,4	4	75,7	1	
33,9	4	71,1	1	

data

geom

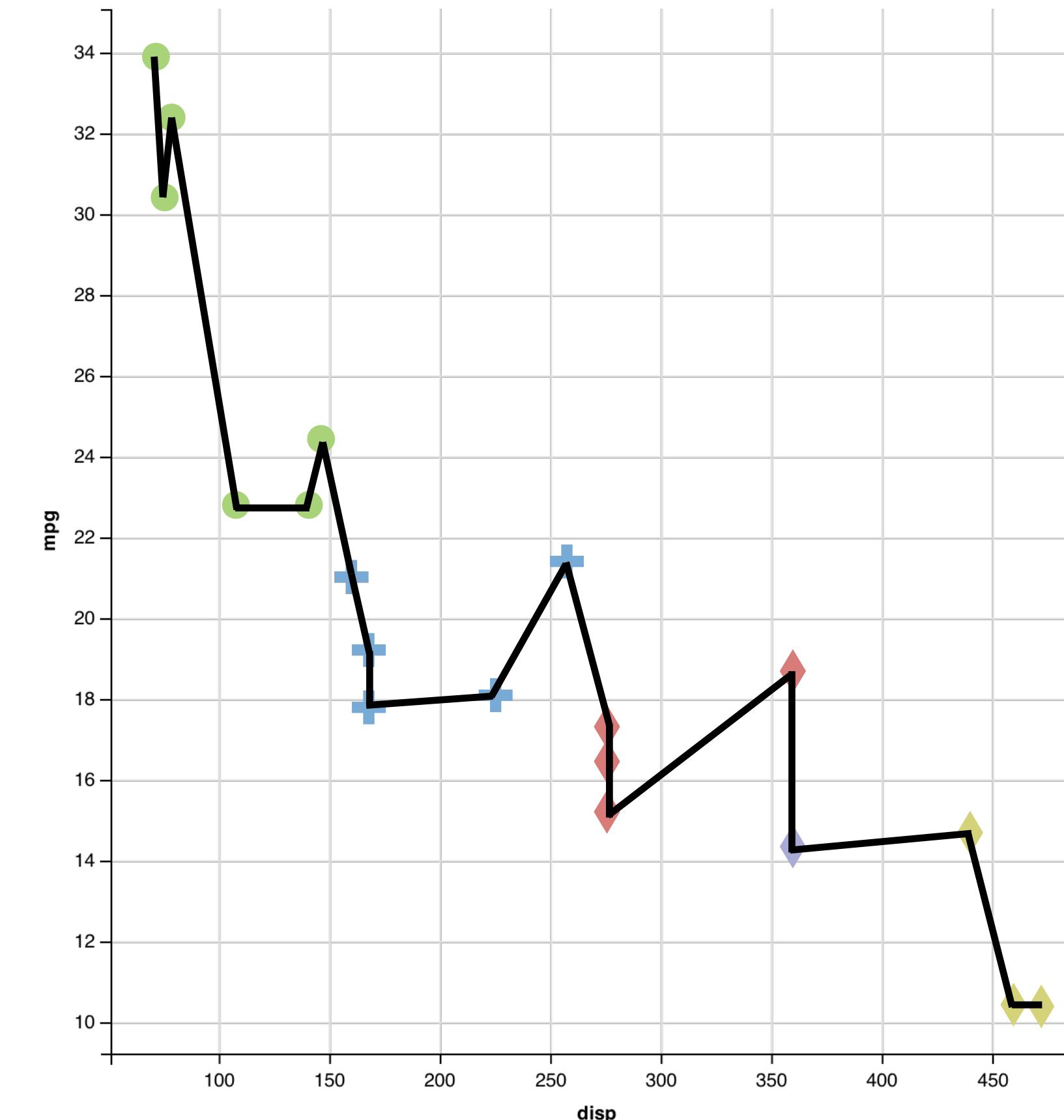


mappings

	y	shape	x	fill
	mpg	cyl	disp	hp
21,0	6	160,0	2	
21,0	6	160,0	2	
22,8	4	108,0	1	
21,4	6	258,0	2	
18,7	8	360,0	3	
18,1	6	225,0	2	
14,3	8	360,0	5	
24,4	4	146,7	1	
22,8	4	140,8	1	
19,2	6	167,6	2	
17,8	6	167,6	2	
16,4	8	275,8	3	
17,3	8	275,8	3	
15,2	8	275,8	3	
10,4	8	472,0	4	
10,4	8	460,0	4	
14,7	8	440,0	4	
32,4	4	78,7	1	
30,4	4	75,7	1	
33,9	4	71,1	1	

data

geom
points
lines

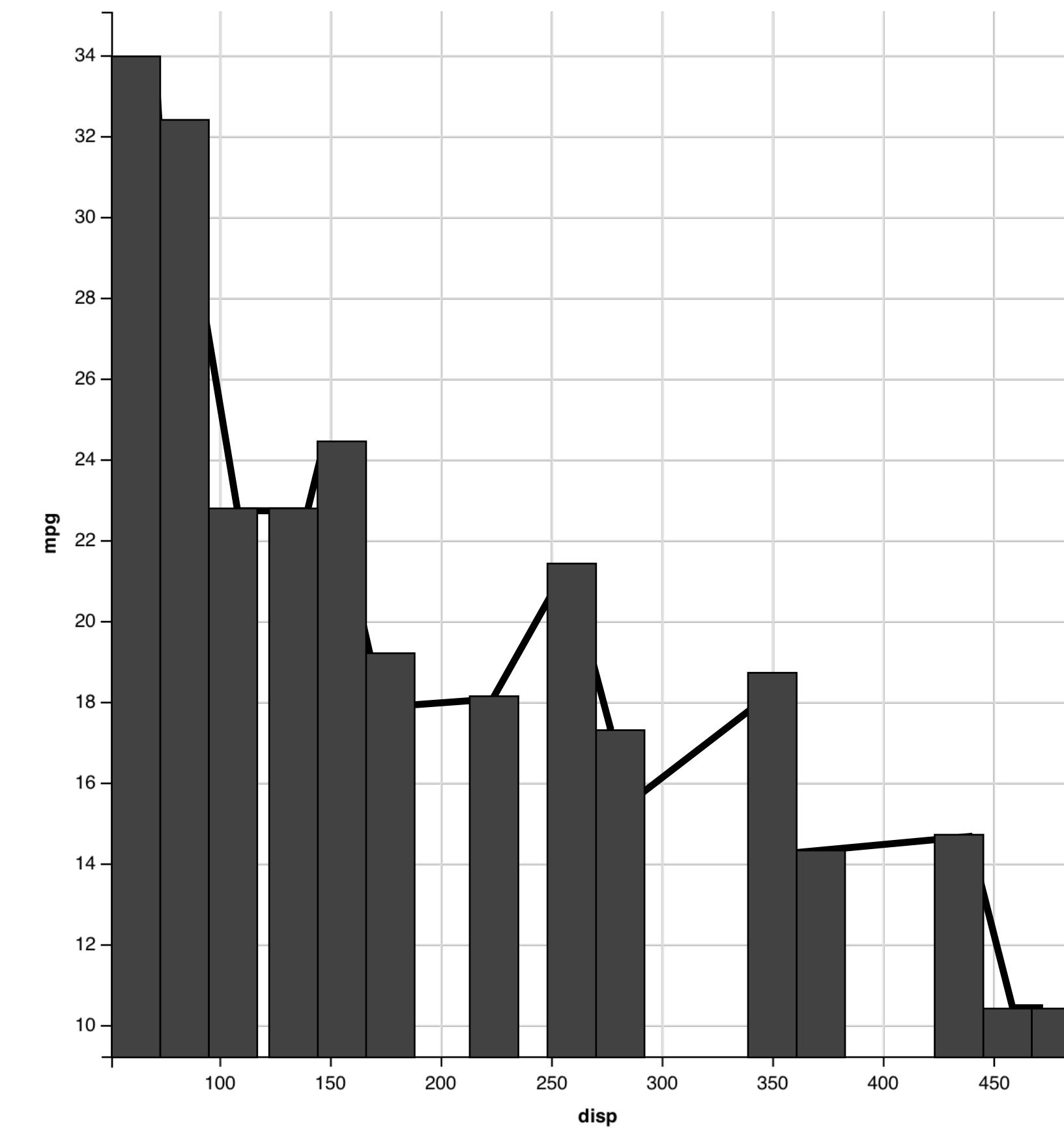


mappings

	<i>y</i>		<i>x</i>	
	mpg	cyl	disp	hp
21,0	6	160,0	2	
21,0	6	160,0	2	
22,8	4	108,0	1	
21,4	6	258,0	2	
18,7	8	360,0	3	
18,1	6	225,0	2	
14,3	8	360,0	5	
24,4	4	146,7	1	
22,8	4	140,8	1	
19,2	6	167,6	2	
17,8	6	167,6	2	
16,4	8	275,8	3	
17,3	8	275,8	3	
15,2	8	275,8	3	
10,4	8	472,0	4	
10,4	8	460,0	4	
14,7	8	440,0	4	
32,4	4	78,7	1	
30,4	4	75,7	1	
33,9	4	71,1	1	

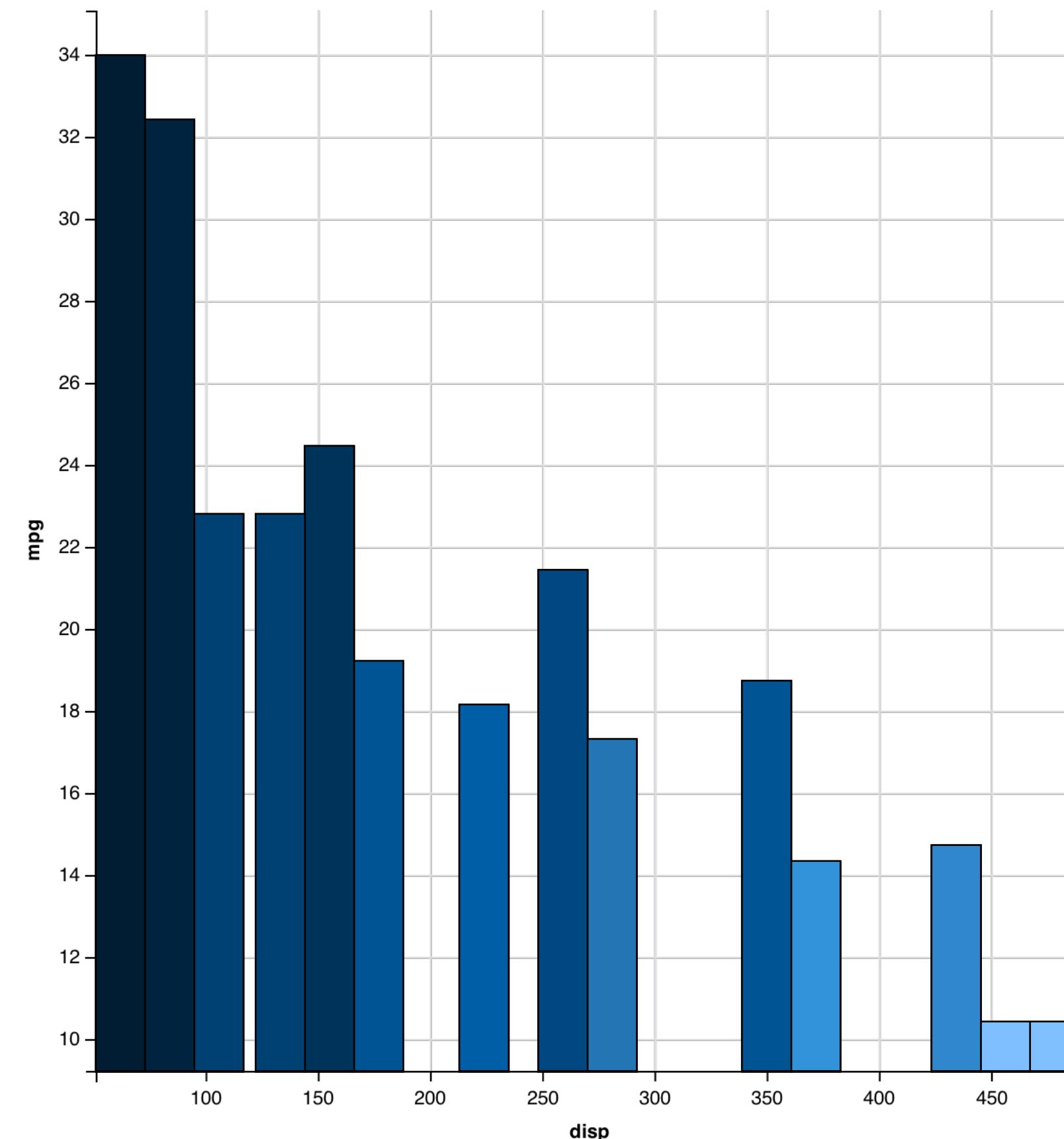
data

geom
points
lines
bars



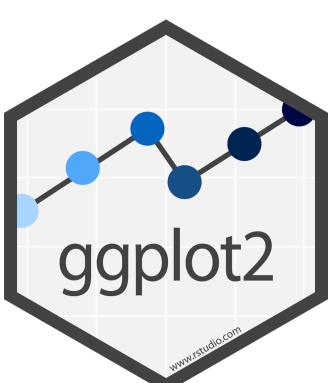
mappings

	<i>y</i>		<i>x</i> <i>fill</i>	
	mpg	cyl	disp	hp
21,0	6	160,0	2	
21,0	6	160,0	2	
22,8	4	108,0	1	
21,4	6	258,0	2	
18,7	8	360,0	3	
18,1	6	225,0	2	
14,3	8	360,0	5	
24,4	4	146,7	1	
22,8	4	140,8	1	
19,2	6	167,6	2	
17,8	6	167,6	2	
16,4	8	275,8	3	
17,3	8	275,8	3	
15,2	8	275,8	3	
10,4	8	472,0	4	
10,4	8	460,0	4	
14,7	8	440,0	4	
32,4	4	78,7	1	
30,4	4	75,7	1	
33,9	4	71,1	1	



data

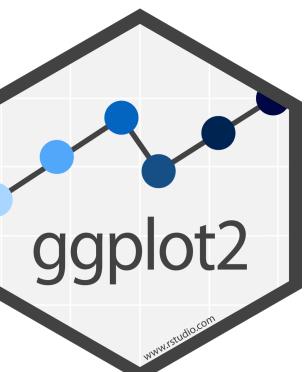
geom
points
lines
bars



[modello]

To make a graph

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



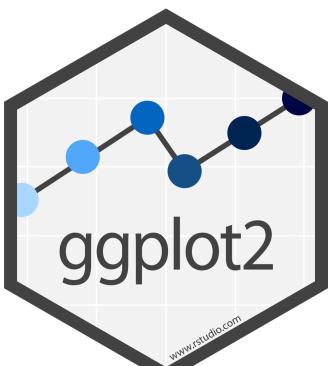
To make a graph

mpg	cyl	disp	hp
21,0	6	160,0	2
21,0	6	160,0	2
22,8	4	108,0	1
21,4	6	258,0	2
18,7	8	360,0	3
18,1	6	225,0	2
14,3	8	360,0	5
24,4	4	146,7	1
22,8	4	140,8	1
19,2	6	167,6	2
17,8	6	167,6	2
16,4	8	275,8	3
17,3	8	275,8	3
15,2	8	275,8	3
10,4	8	472,0	4
10,4	8	460,0	4
14,7	8	440,0	4
32,4	4	78,7	1
30,4	4	75,7	1
33,9	4	71,1	1

data

1. Scegli un set di dati

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



To make a graph

mpg	cyl	disp	hp	
21,0	6	160,0	2	●
21,0	6	160,0	2	●
22,8	4	108,0	1	●
21,4	6	258,0	2	●
18,7	8	360,0	3	●
18,1	6	225,0	2	●
14,3	8	360,0	5	●
24,4	4	146,7	1	●
22,8	4	140,8	1	●
19,2	6	167,6	2	●
17,8	6	167,6	2	●
16,4	8	275,8	3	●
17,3	8	275,8	3	●
15,2	8	275,8	3	●
10,4	8	472,0	4	●
10,4	8	460,0	4	●
14,7	8	440,0	4	●
32,4	4	78,7	1	●
30,4	4	75,7	1	●
33,9	4	71,1	1	●

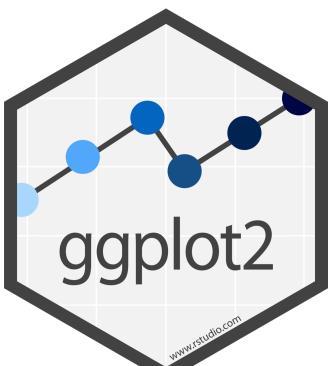
data

geom

1. Scegli un set di dati

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

2. Scegli un geom per
visualizzare i casi



To make a graph

mappings

mpg	cyl	disp	hp	fill
21,0	6	160,0	2	blue
21,0	6	160,0	2	blue
22,8	4	108,0	1	light green
21,4	6	258,0	2	blue
18,7	8	360,0	3	red
18,1	6	225,0	2	blue
14,3	8	360,0	5	purple
24,4	4	146,7	1	light green
22,8	4	140,8	1	light green
19,2	6	167,6	2	blue
17,8	6	167,6	2	blue
16,4	8	275,8	3	red
17,3	8	275,8	3	red
15,2	8	275,8	3	red
10,4	8	472,0	4	yellow-green
10,4	8	460,0	4	yellow-green
14,7	8	440,0	4	yellow-green
32,4	4	78,7	1	light green
30,4	4	75,7	1	light green
33,9	4	71,1	1	light green

data

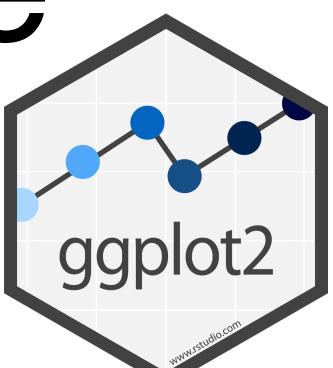
geom

1. Scegli un set di dati

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

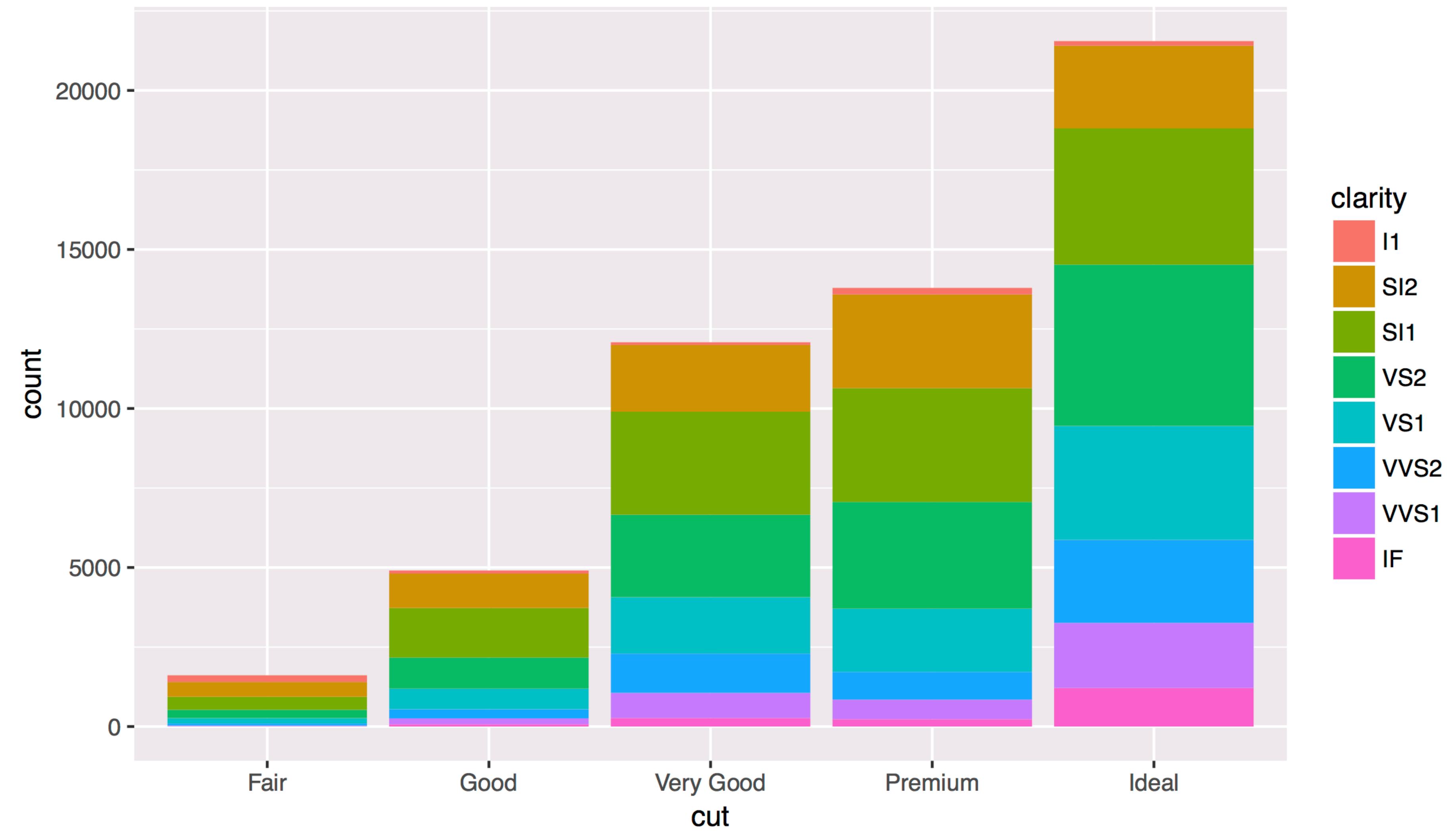
2. Scegli un geom per visualizzare i casi

3. Mappa le proprietà estetiche alle variabili



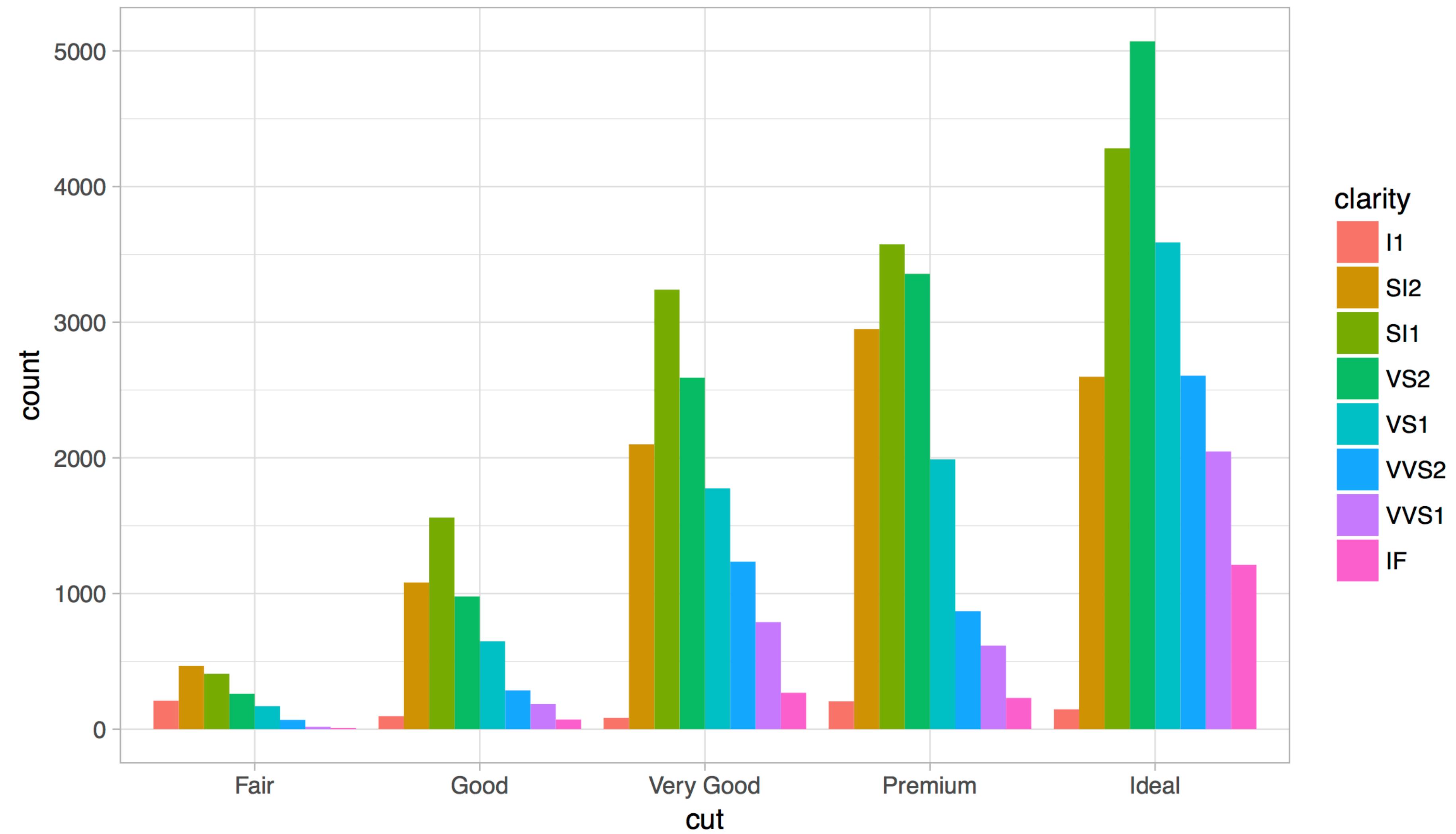
Altro?





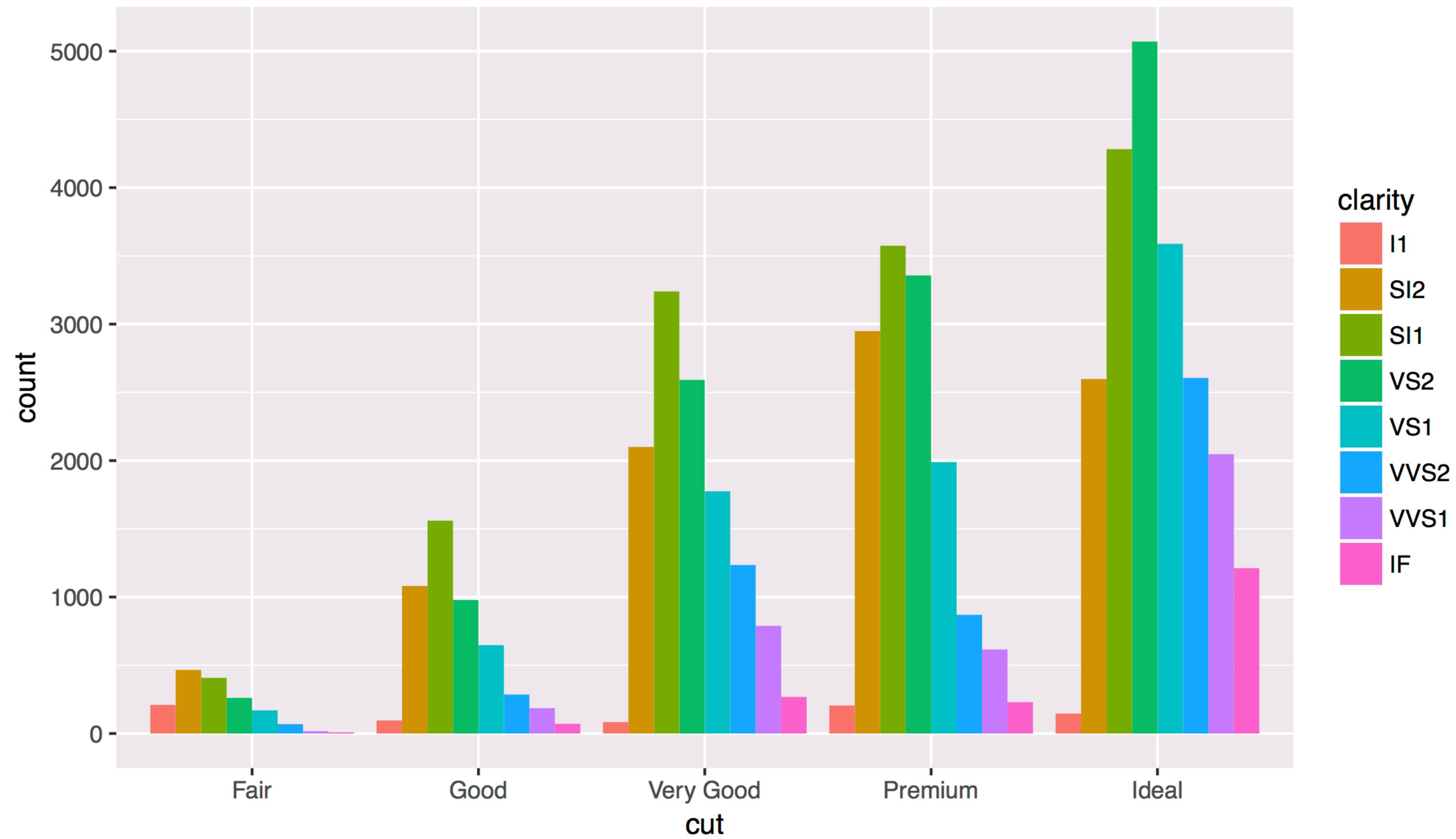
Themes

Aspetto visivo di elementi non di dati



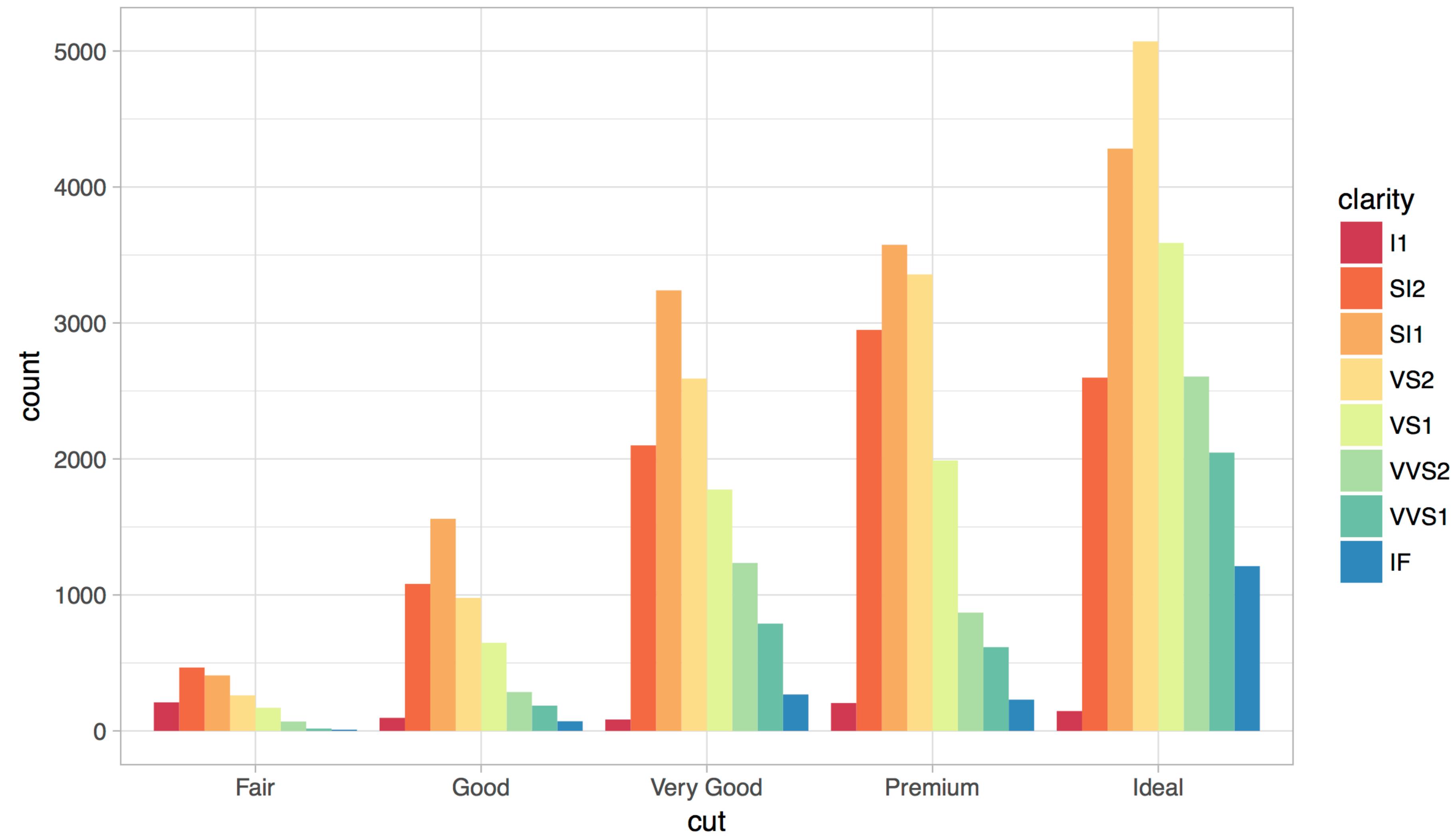
Aggiustare la posizione

Come sono disposti gli oggetti sovrapposti



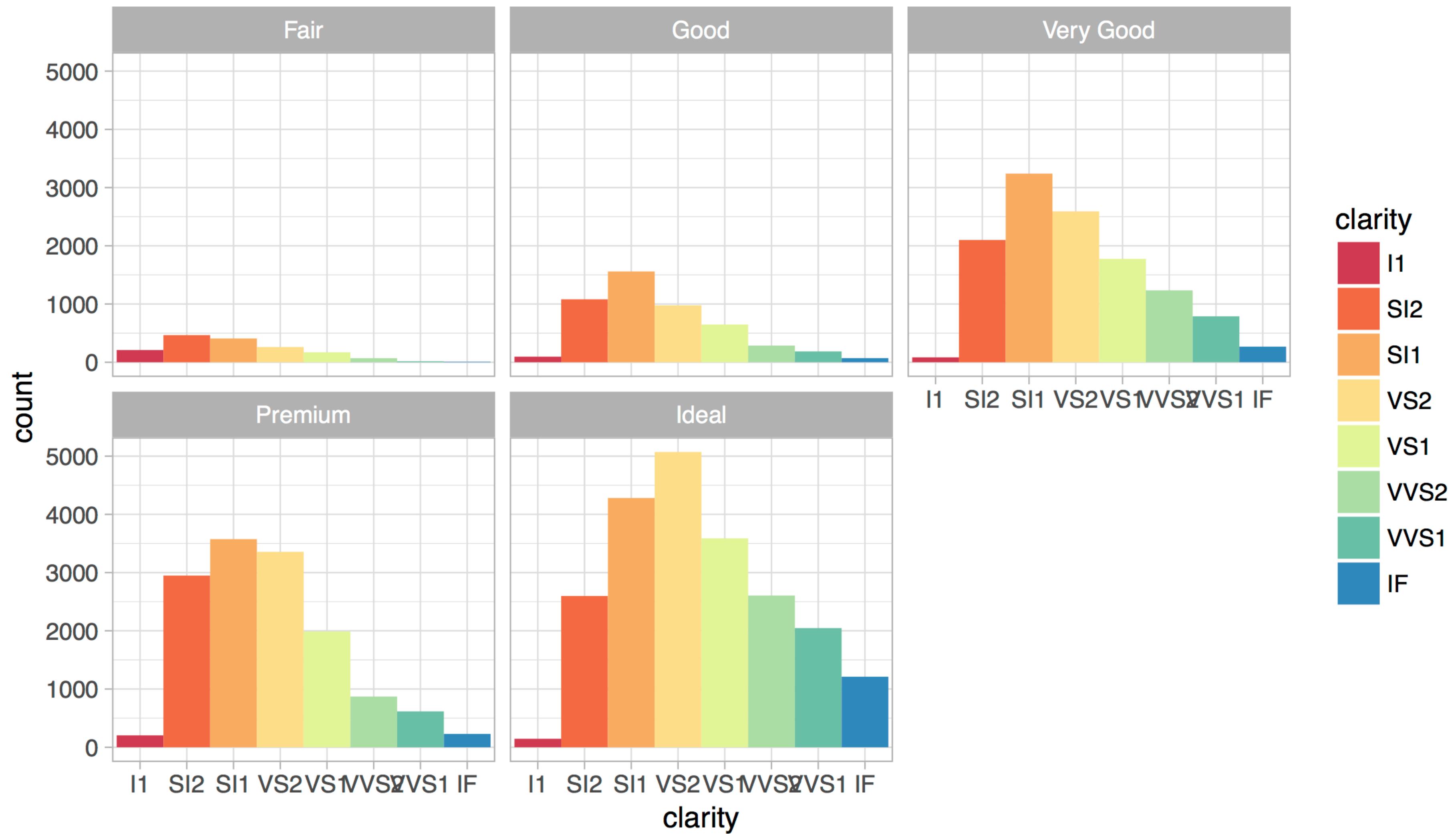
Scales

Personalizza le scale di colore, altre mappature

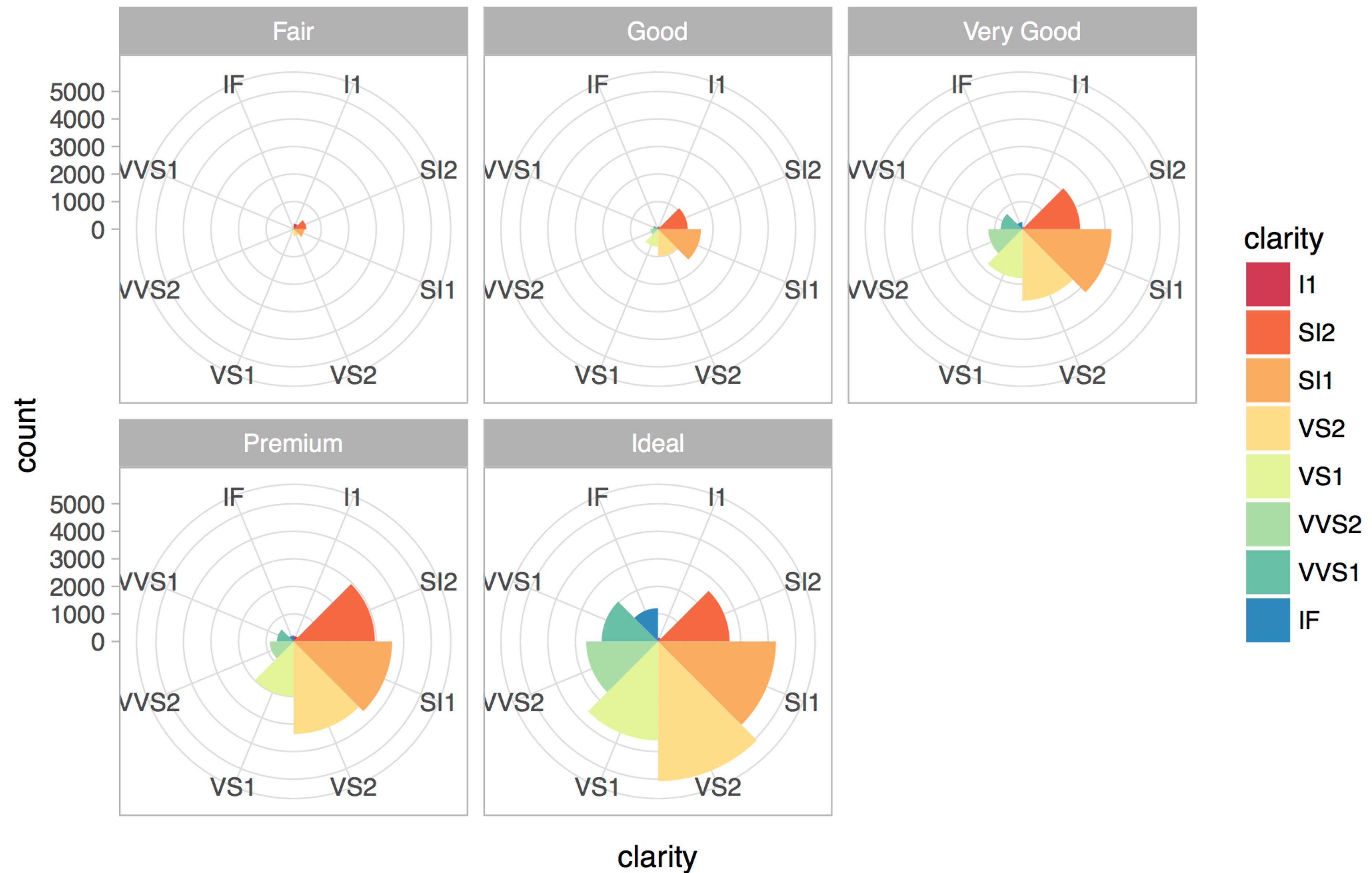


Facets

Sottotrame che visualizzano sottoinsiemi di dati.



Sistemi di coordinate



Titoli e sottotitoli

Diamonds data

The data set is skewed towards ideal cut diamonds



Data by Hadley Wickham

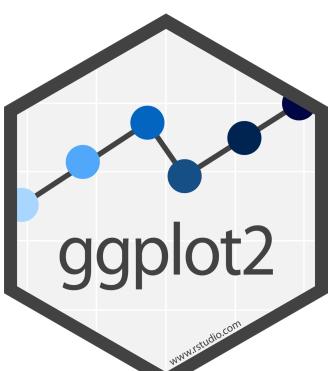
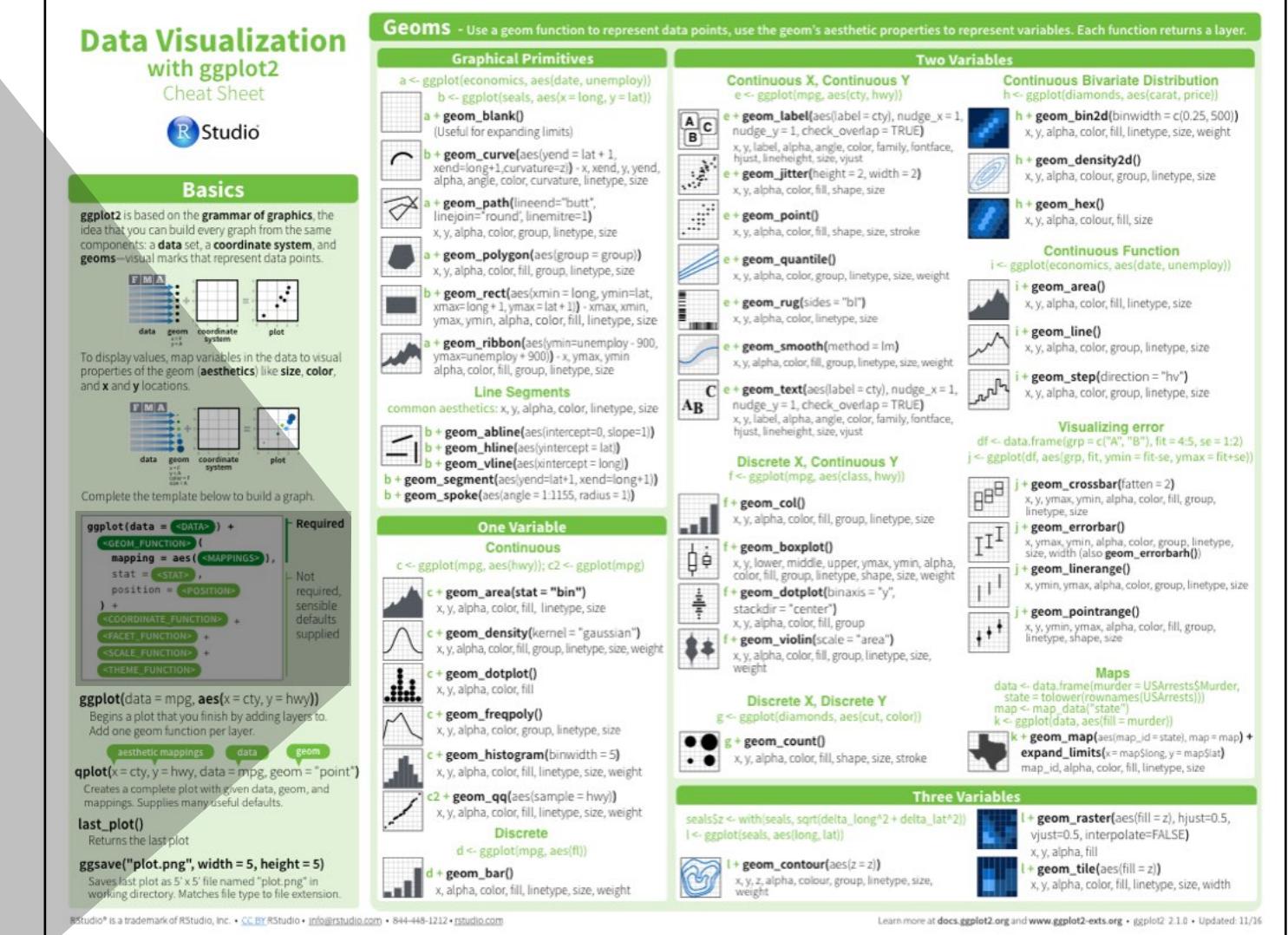
Il modello ggplot2

Crea qualsiasi grafico compilando i parametri di questo modello

```
ggplot(data = <DATA>) +
  <GEOM_FUNCTION>(
    mapping = aes(<MAPPINGS>),
    stat = <STAT>,
    position = <POSITION>
  ) +
  <COORDINATE_FUNCTION> +
  <FACET_FUNCTION> +
  <SCALE_FUNCTION> +
  <THEME_FUNCTION>
```

Required

Not required,
sensible
defaults
supplied



ggplot2.tidyverse.org

The screenshot shows a web browser window with the title "Create Elegant Data Visualisati x" and the user "Garrett". The address bar displays "ggplot2.tidyverse.org". The page content includes the ggplot2 logo and the text "part of the tidyverse". A main heading "Usage" is followed by a paragraph explaining the philosophy of ggplot2 and how to use it. Below this is a code block showing R code to create a scatter plot. To the right, there are sections for "Links", "License", and "Developers", along with a ggplot2 logo at the bottom right.

Usage

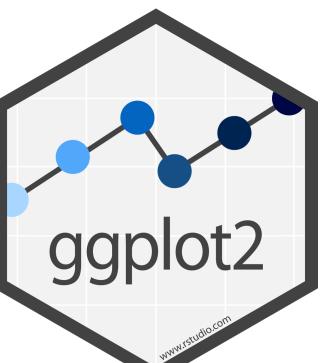
It's hard to succinctly describe how ggplot2 works because it embodies a deep philosophy of visualisation. However, in most cases you start with `ggplot()`, supply a dataset and aesthetic mapping (with `aes()`). You then add on layers (like `geom_point()` or `geom_histogram()`), scales (like `scale_colour_brewer()`), faceting specifications (like `facet_wrap()`) and coordinate systems (like `coord_flip()`).

```
library(ggplot2)

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

A scatter plot showing fuel efficiency (mpg) on the y-axis versus engine displacement (displ) on the x-axis. The plot uses a color scale where purple represents the "2seater" class. The data points show a general trend where fuel efficiency decreases as engine displacement increases, with some outliers like the Toyota Corolla (purple dot) having very low displacement and high fuel efficiency.

class	displ	mpg
2seater	1.6	44
2seater	1.8	40
2seater	2.0	33
2seater	2.2	32
2seater	2.4	30
2seater	2.6	28
2seater	2.8	27
2seater	3.0	26
2seater	3.2	25
2seater	3.4	23
2seater	3.6	22
2seater	3.8	21
2seater	4.0	20
2seater	4.2	18
2seater	4.4	17
2seater	4.6	16
2seater	4.8	15
2seater	5.0	14
2seater	5.2	13
2seater	5.4	12
2seater	5.6	11
2seater	5.8	10
2seater	6.0	9
2seater	6.2	8
2seater	6.4	7
2seater	6.6	6
2seater	6.8	5
2seater	7.0	4
2seater	7.2	3
2seater	7.4	2
2seater	7.6	1
2seater	7.8	0
4cyl	1.6	33
4cyl	1.8	30
4cyl	2.0	28
4cyl	2.2	27
4cyl	2.4	26
4cyl	2.6	25
4cyl	2.8	24
4cyl	3.0	23
4cyl	3.2	22
4cyl	3.4	21
4cyl	3.6	20
4cyl	3.8	19
4cyl	4.0	18
4cyl	4.2	17
4cyl	4.4	16
4cyl	4.6	15
4cyl	4.8	14
4cyl	5.0	13
4cyl	5.2	12
4cyl	5.4	11
4cyl	5.6	10
4cyl	5.8	9
4cyl	6.0	8
4cyl	6.2	7
4cyl	6.4	6
4cyl	6.6	5
4cyl	6.8	4
4cyl	7.0	3
4cyl	7.2	2
4cyl	7.4	1
4cyl	7.6	0
6cyl	1.6	22
6cyl	1.8	20
6cyl	2.0	18
6cyl	2.2	17
6cyl	2.4	16
6cyl	2.6	15
6cyl	2.8	14
6cyl	3.0	13
6cyl	3.2	12
6cyl	3.4	11
6cyl	3.6	10
6cyl	3.8	9
6cyl	4.0	8
6cyl	4.2	7
6cyl	4.4	6
6cyl	4.6	5
6cyl	4.8	4
6cyl	5.0	3
6cyl	5.2	2
6cyl	5.4	1
6cyl	5.6	0
8cyl	1.6	15
8cyl	1.8	14
8cyl	2.0	13
8cyl	2.2	12
8cyl	2.4	11
8cyl	2.6	10
8cyl	2.8	9
8cyl	3.0	8
8cyl	3.2	7
8cyl	3.4	6
8cyl	3.6	5
8cyl	3.8	4
8cyl	4.0	3
8cyl	4.2	2
8cyl	4.4	1
8cyl	4.6	0
8cyl	4.8	0
V6	1.6	15
V6	1.8	14
V6	2.0	13
V6	2.2	12
V6	2.4	11
V6	2.6	10
V6	2.8	9
V6	3.0	8
V6	3.2	7
V6	3.4	6
V6	3.6	5
V6	3.8	4
V6	4.0	3
V6	4.2	2
V6	4.4	1
V6	4.6	0
V6	4.8	0
V8	1.6	15
V8	1.8	14
V8	2.0	13
V8	2.2	12
V8	2.4	11
V8	2.6	10
V8	2.8	9
V8	3.0	8
V8	3.2	7
V8	3.4	6
V8	3.6	5
V8	3.8	4
V8	4.0	3
V8	4.2	2
V8	4.4	1
V8	4.6	0
V8	4.8	0
V10	1.6	15
V10	1.8	14
V10	2.0	13
V10	2.2	12
V10	2.4	11
V10	2.6	10
V10	2.8	9
V10	3.0	8
V10	3.2	7
V10	3.4	6
V10	3.6	5
V10	3.8	4
V10	4.0	3
V10	4.2	2
V10	4.4	1
V10	4.6	0
V10	4.8	0
V12	1.6	15
V12	1.8	14
V12	2.0	13
V12	2.2	12
V12	2.4	11
V12	2.6	10
V12	2.8	9
V12	3.0	8
V12	3.2	7
V12	3.4	6
V12	3.6	5
V12	3.8	4
V12	4.0	3
V12	4.2	2
V12	4.4	1
V12	4.6	0
V12	4.8	0
V16	1.6	15
V16	1.8	14
V16	2.0	13
V16	2.2	12
V16	2.4	11
V16	2.6	10
V16	2.8	9
V16	3.0	8
V16	3.2	7
V16	3.4	6
V16	3.6	5
V16	3.8	4
V16	4.0	3
V16	4.2	2
V16	4.4	1
V16	4.6	0
V16	4.8	0



Visualize Data with

