

# Visualisation avec ggplot2



# Voitures et cylindrée

**Quelle est la relation entre la cylindrée d'une voiture et son autonomie?**

# À votre tour #1

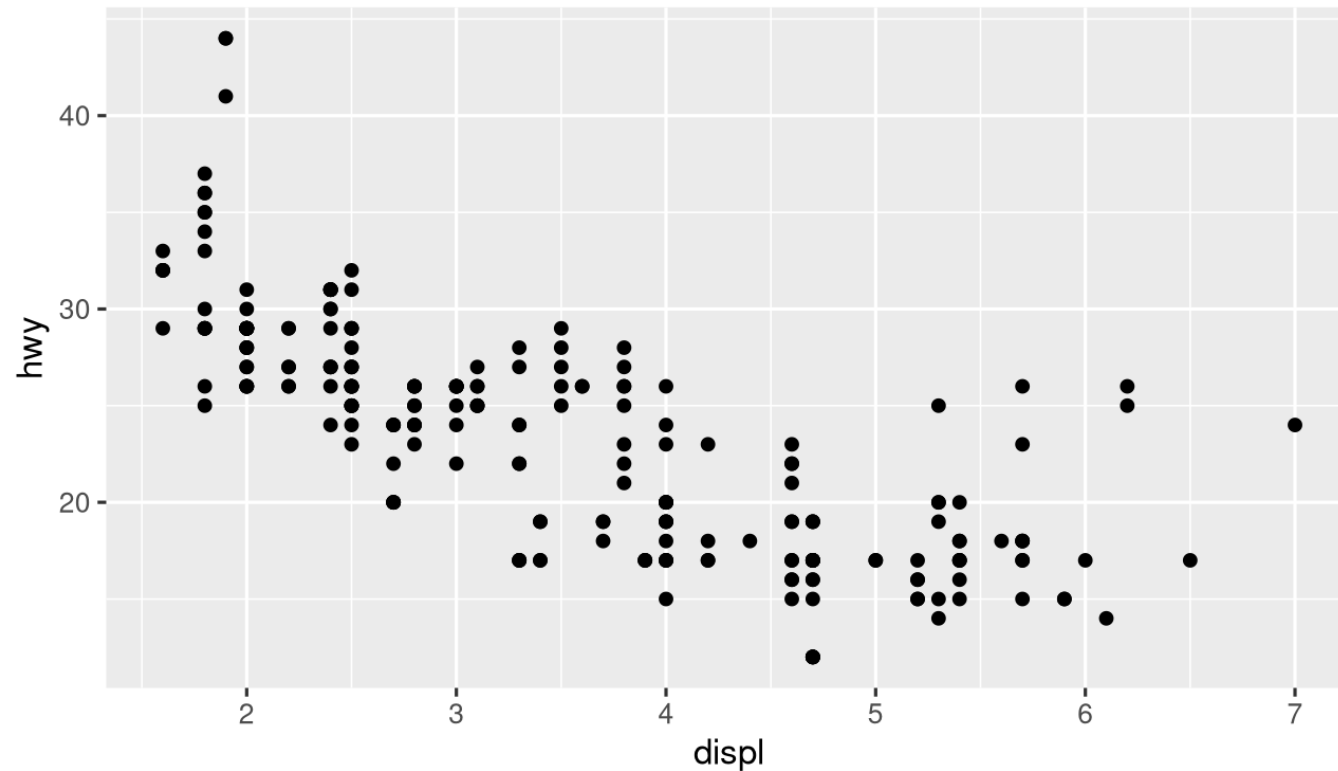
Compilez le code du fichier Rmd pour créer un graphique

Portez attention à l'orthographe, aux majuscules et aux parenthèses!

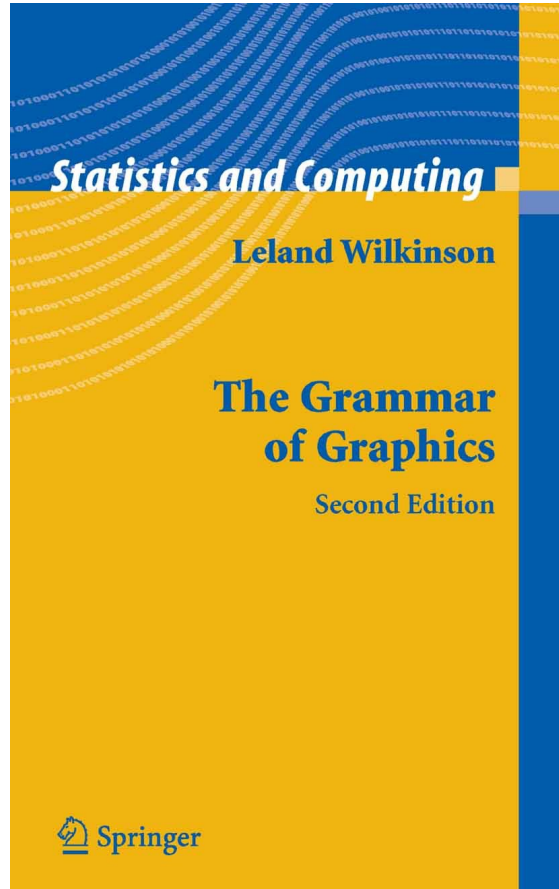
```
library(ggplot2)
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ,
                           y = hwy))
```

02:00

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



# La grammaire des graphiques



**Composante esthétique**

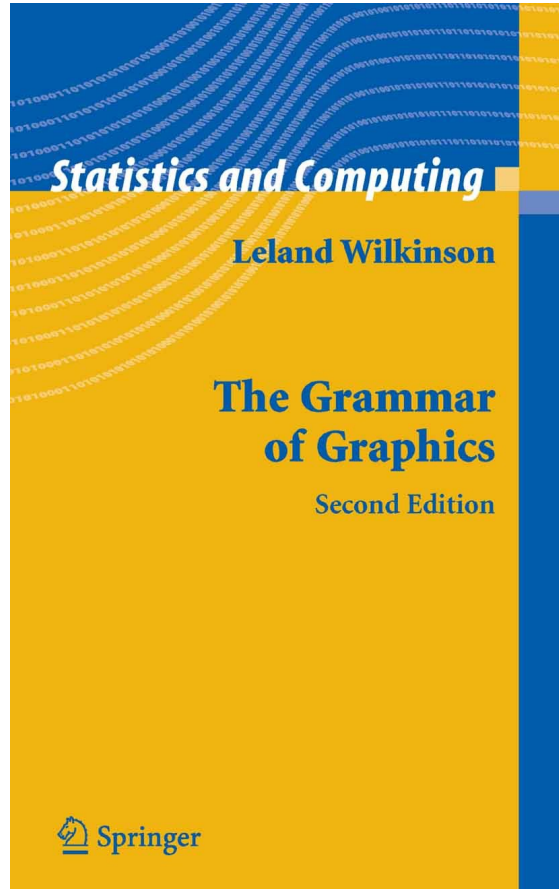
**Propriété visuelle d'un graphique**

**Position, forme, couleur, etc.**

**Variables**

**Une colonne d'une  
base de données**

# La grammaire des graphiques



**Composante esthétique**

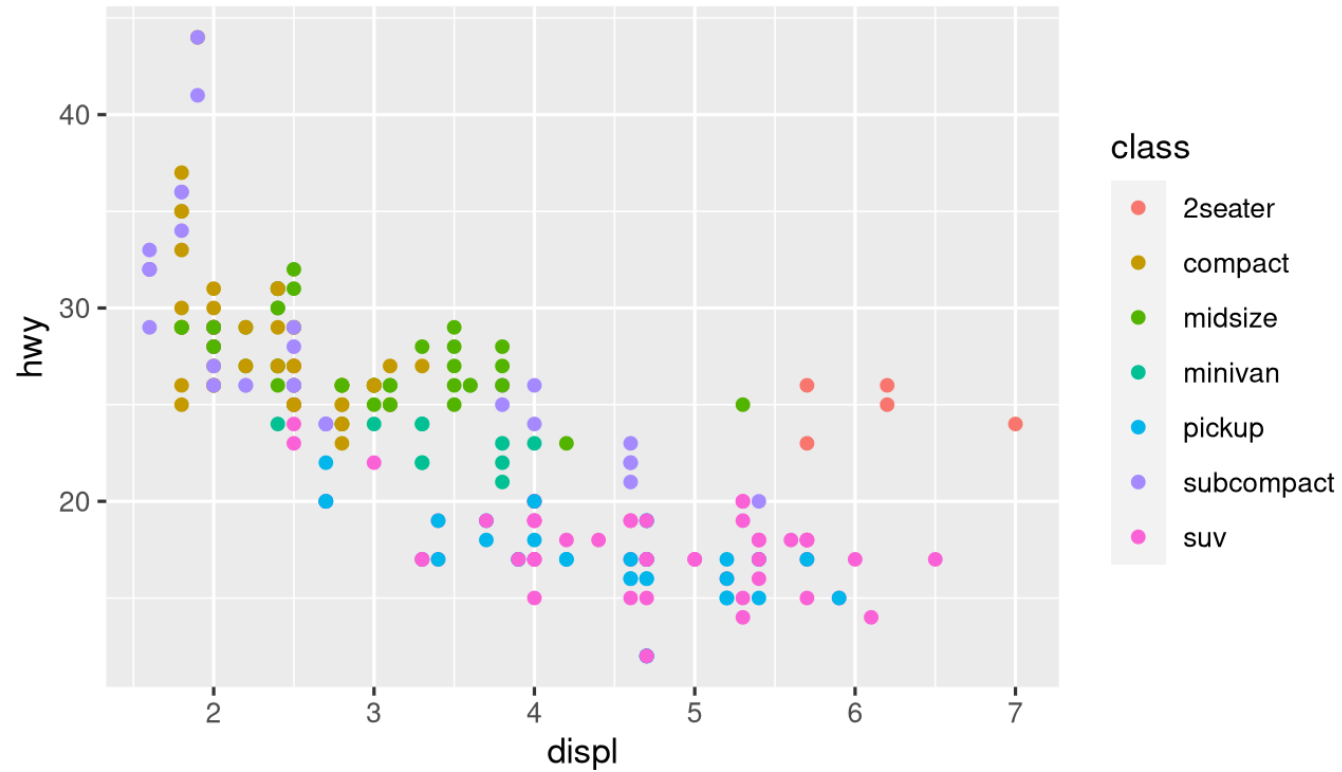
**Propriété visuelle d'un graphique**

**Position, forme, couleur, etc.**

**Variables**

**Une colonne d'une  
base de données**

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



# À votre tour #2

**Modifiez la couleur, la taille, la transparence et la forme des points de votre graphique.**

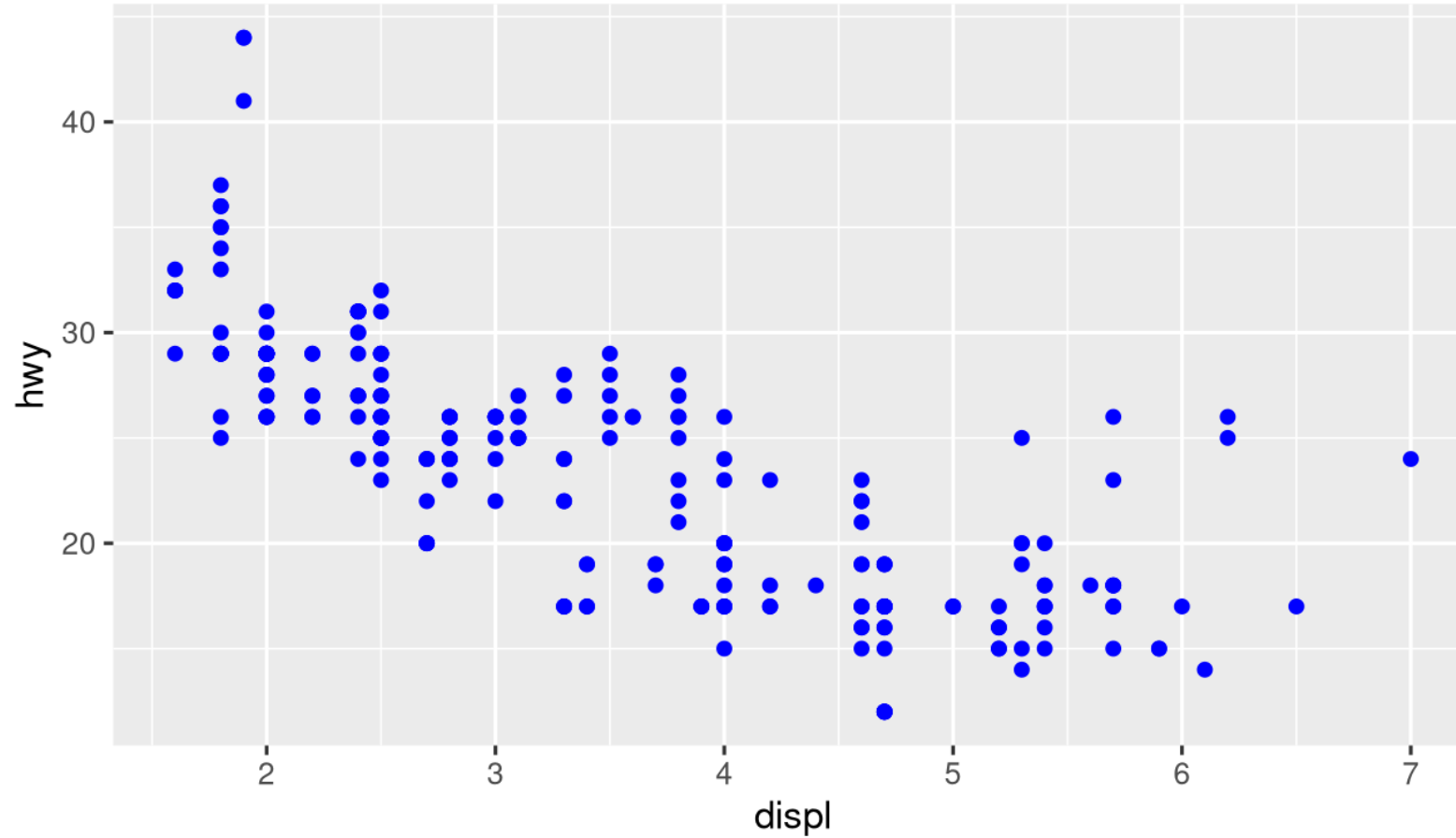
**Expérimentez!**

**Y a-t-il des différences selon qu'on associe une composante esthétique à des variables discrètes ou continues?**

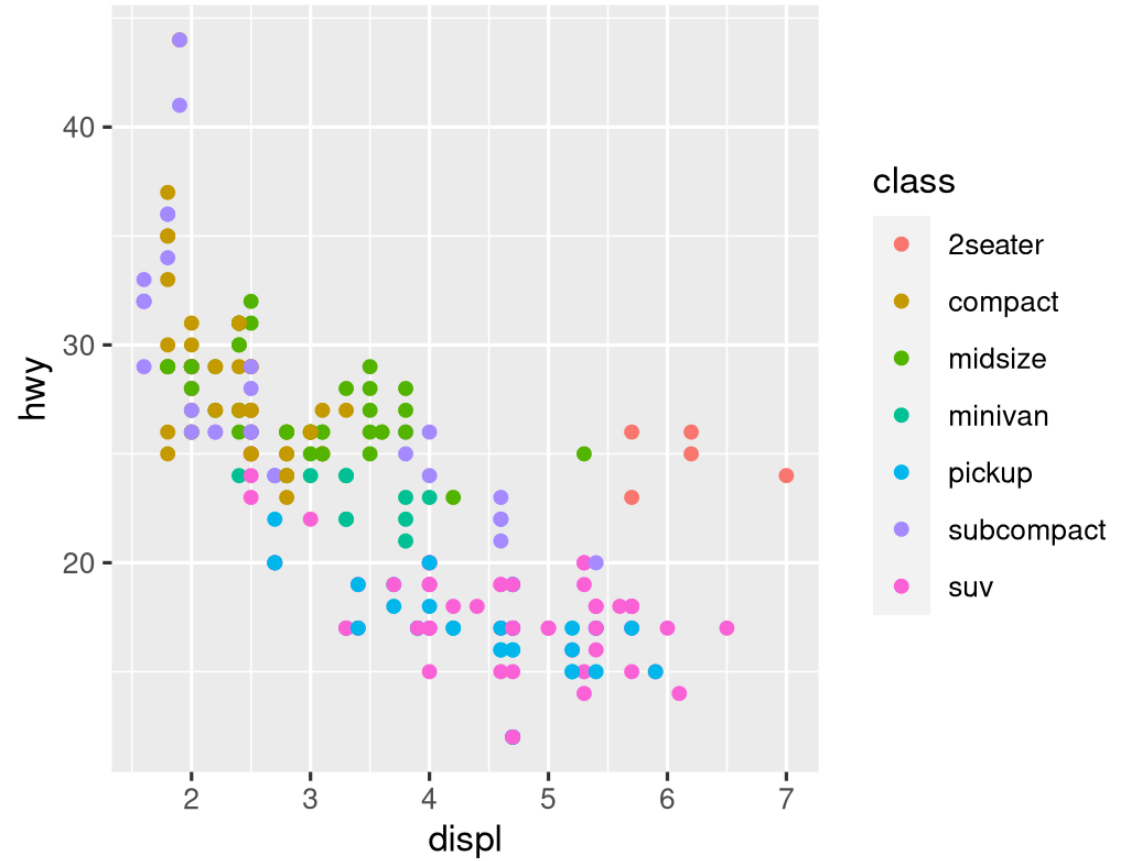
**Quid si on utilise plusieurs composantes esthétiques?**



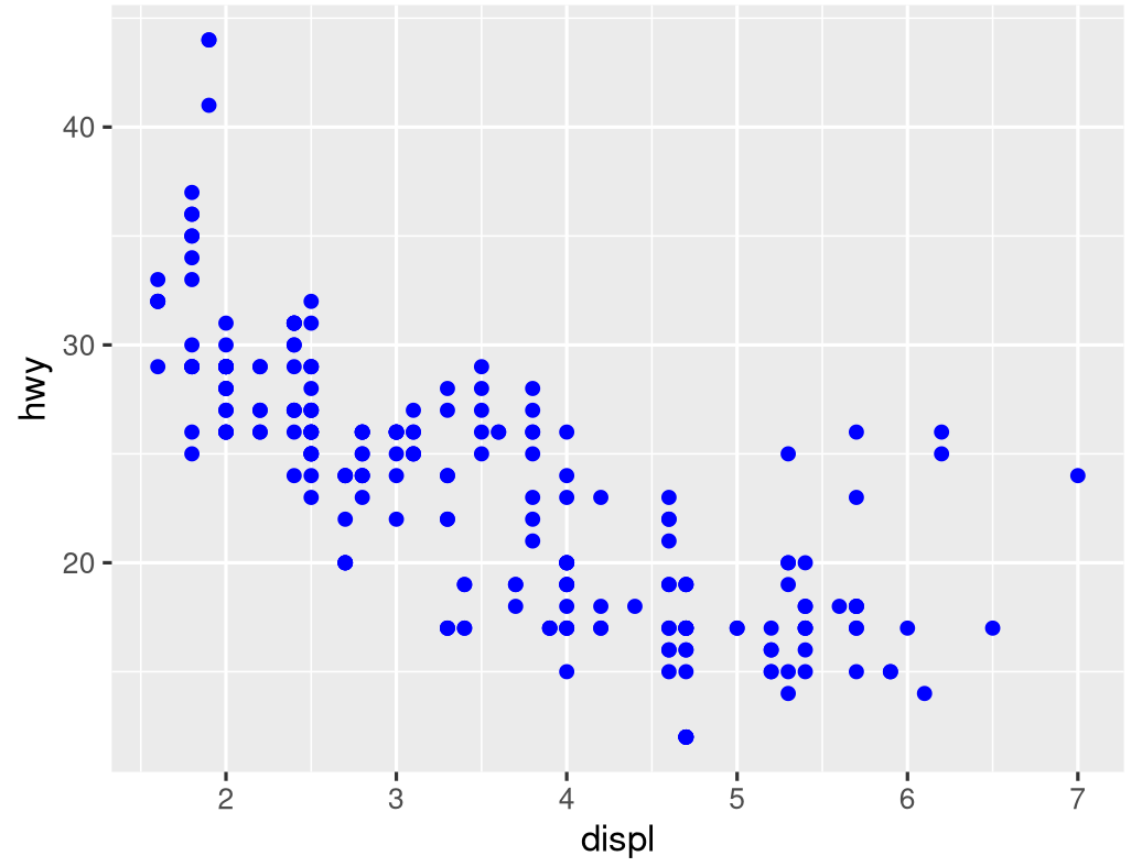
# Comment créer ce graphique?



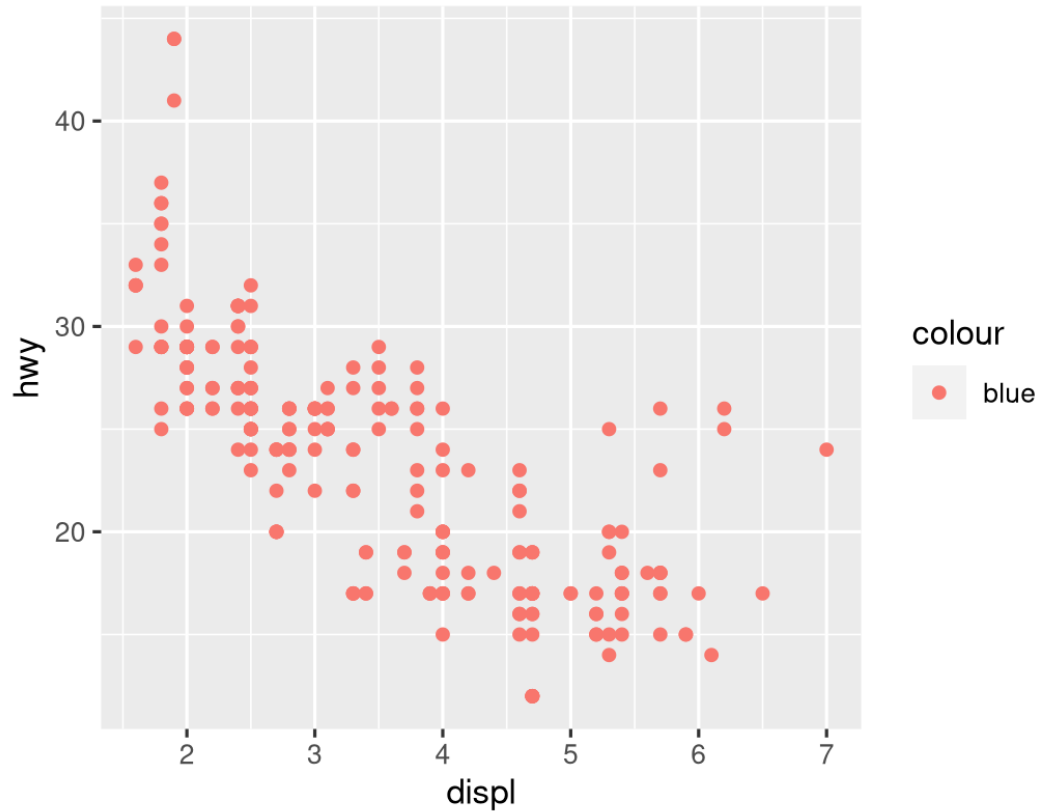
```
ggplot(mpg) +  
  geom_point(aes(x = displ, y = hwy,  
                color = class))
```



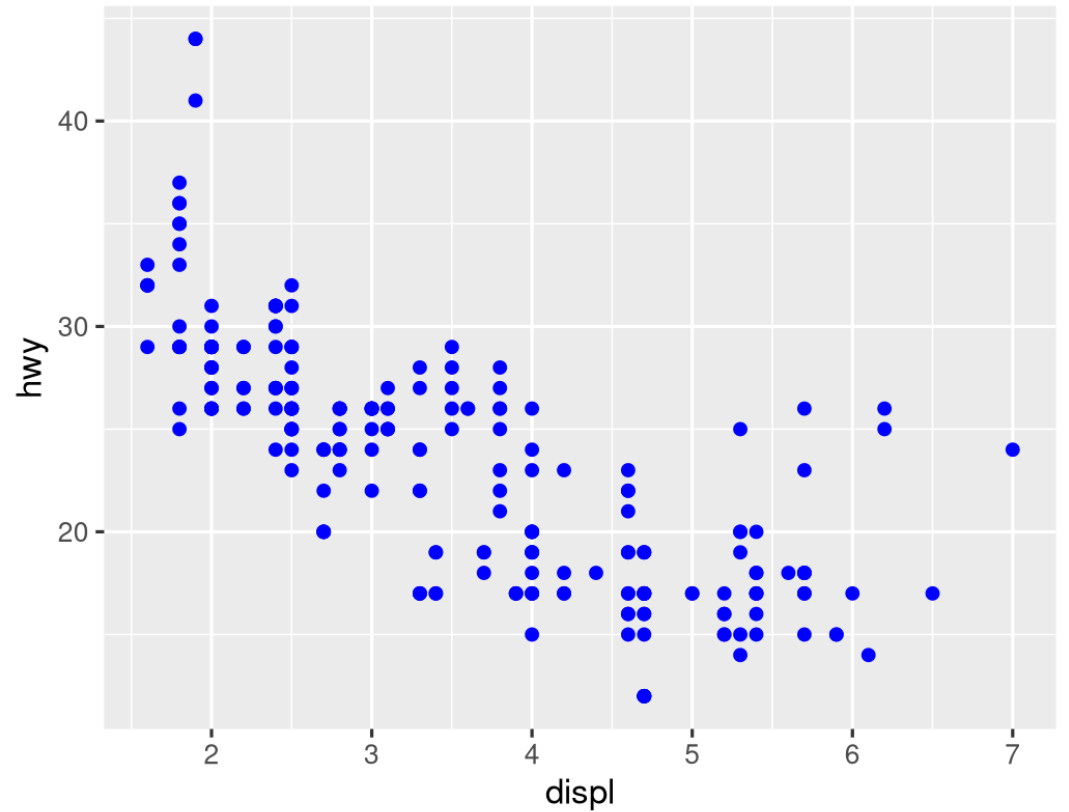
```
ggplot(mpg) +  
  geom_point(aes(x = displ, y = hwy),  
             color = "blue")
```



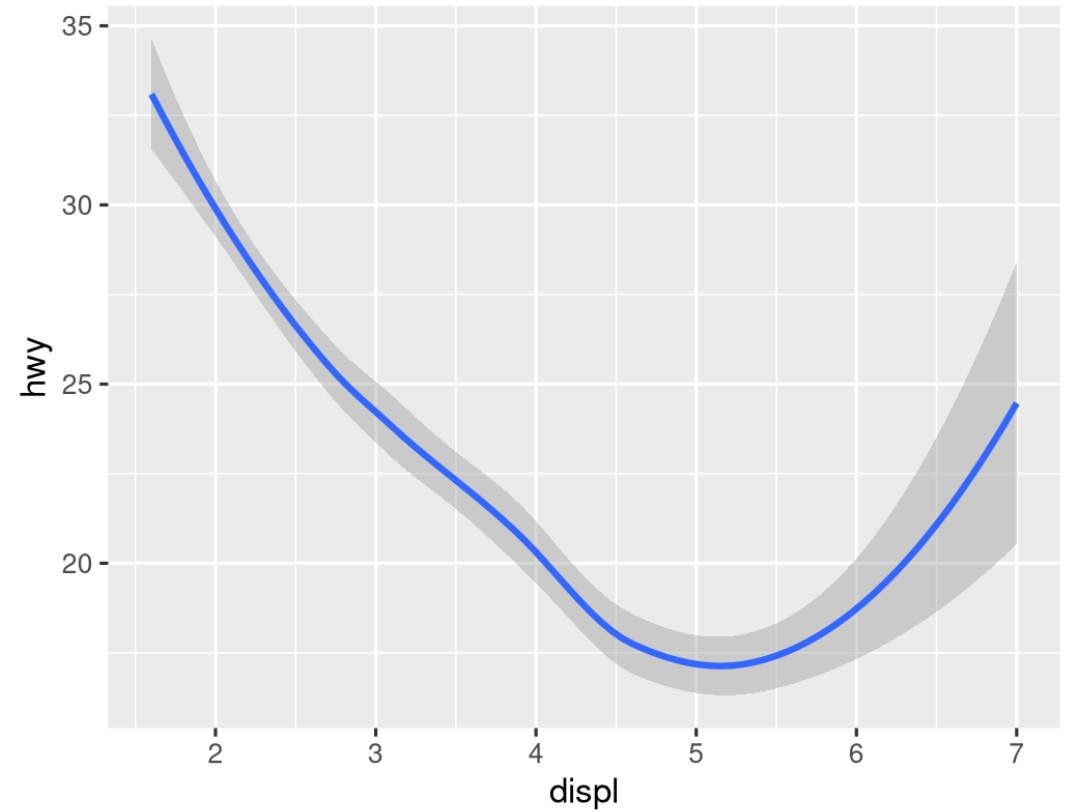
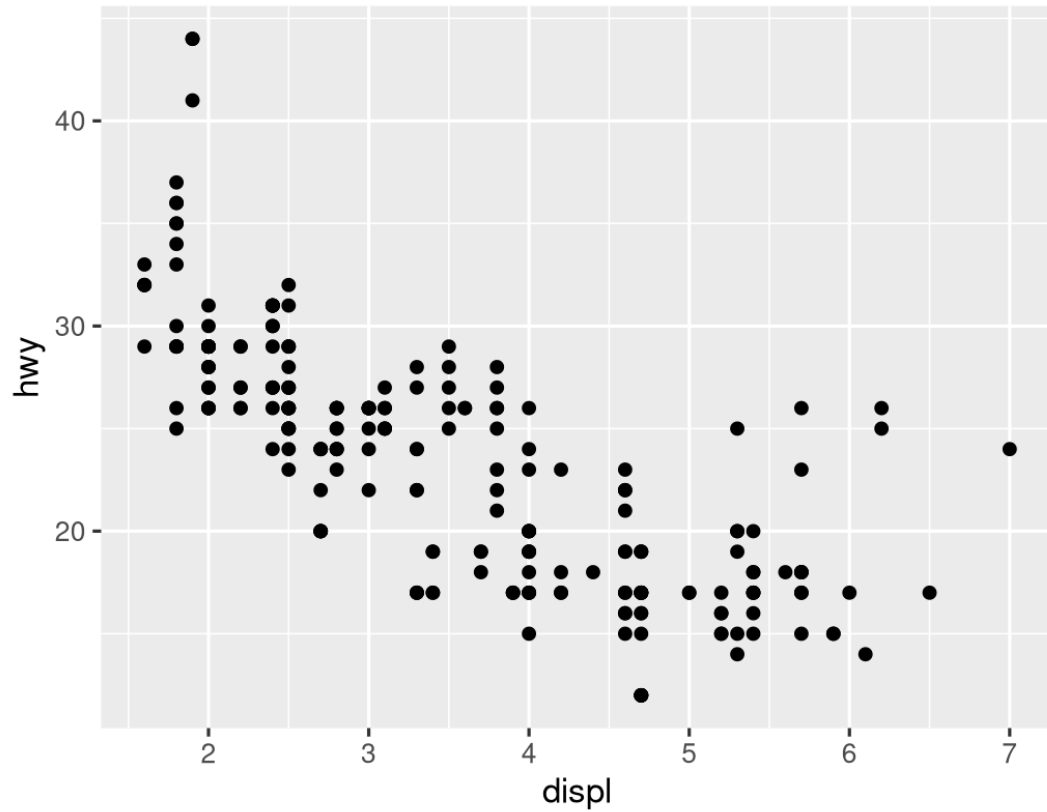
```
ggplot(mpg) +  
  geom_point(aes(x = displ, y = hwy,  
                 color = "blue"))
```



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



# Similarités et différences?



# Géométries

```
ggplot(data = DONNÉES) +  
  FONCTION_GEOM(mapping = aes(APPLICATION ESTHÉTIQUE))
```

# Géométries

## Exemple de géométrie **Sortie**



`geom_col()`

Diagramme à bande

*text*

`geom_text()`

Texte



`geom_point()`

Nuage de points



`geom_boxplot()`

Boîte à moustaches

# Inventaire de géométries

Beaucoup de géométries disponibles!

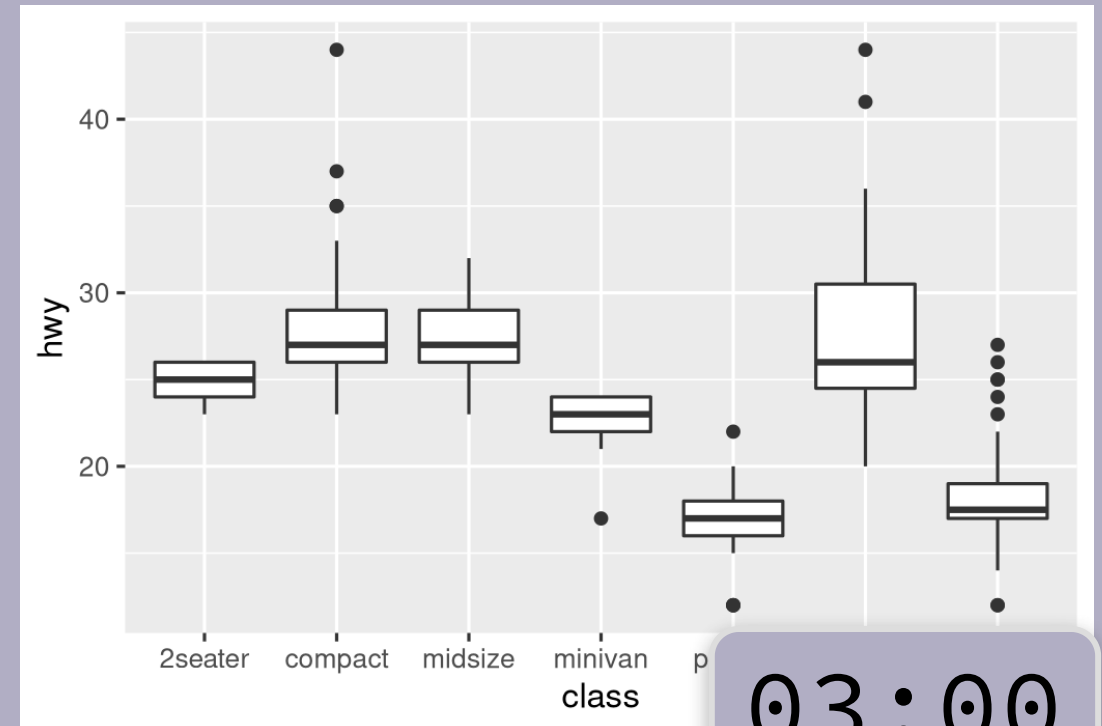
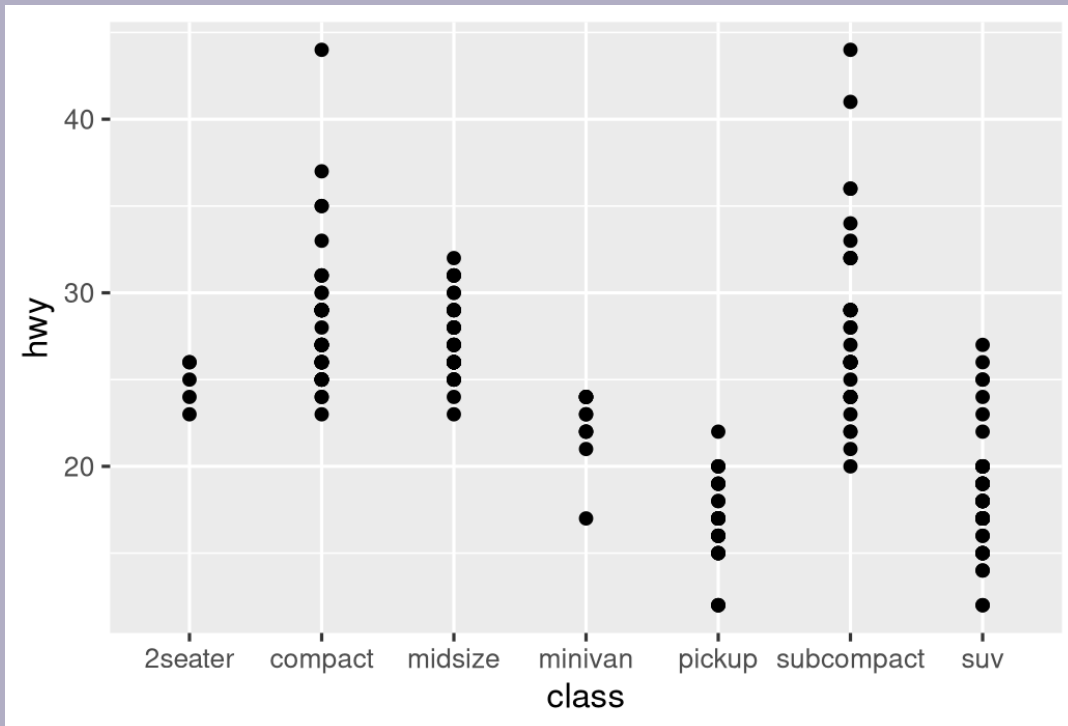
Voir **la documentation `ggplot2`** pour un exemple exhaustif avec toutes les géométries.

Ou la feuille aide-mémoire `ggplot2`



# À votre tour #3

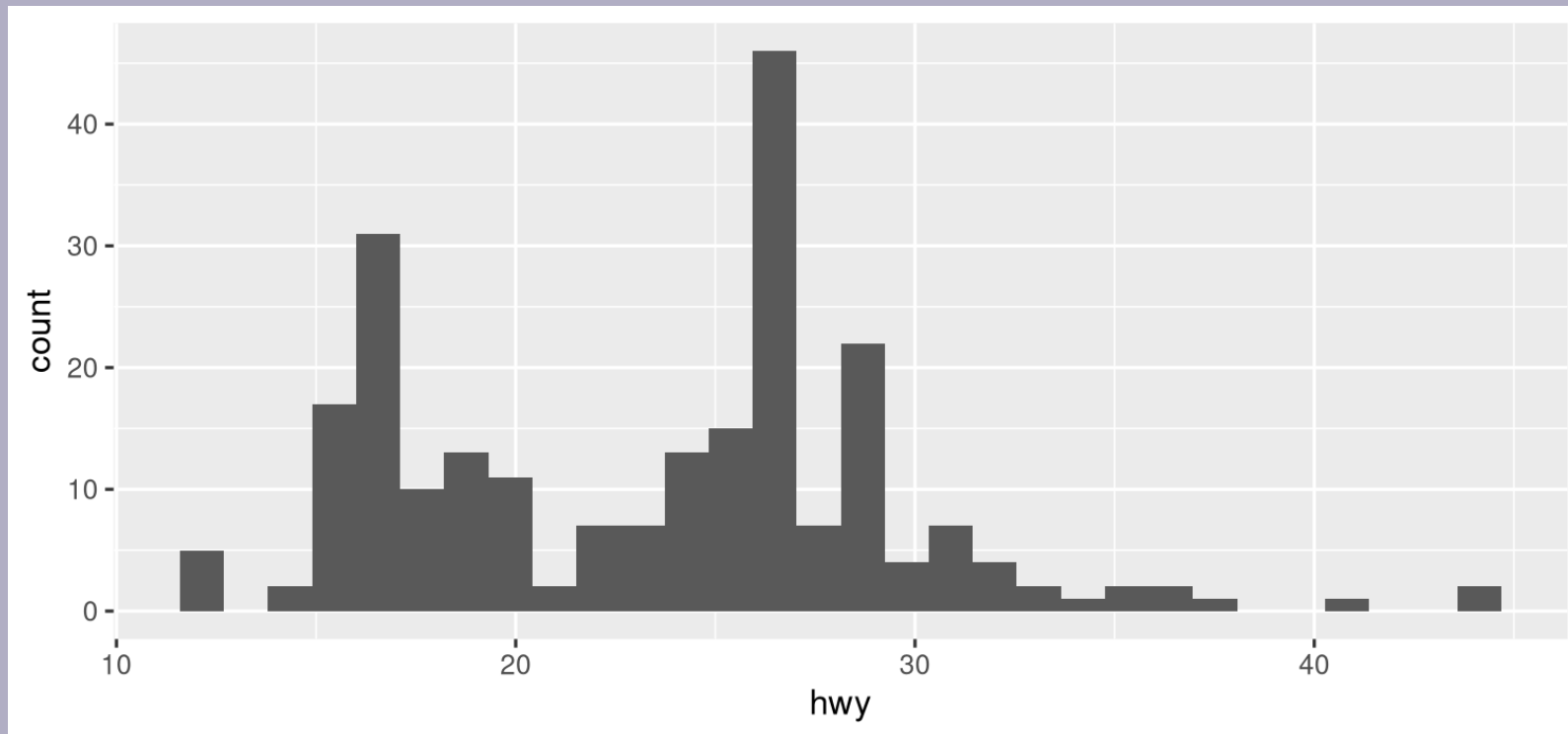
Remplacez le nuage de point par des boîtes à moustache. Utilisez l'aide mémoire.



03:00

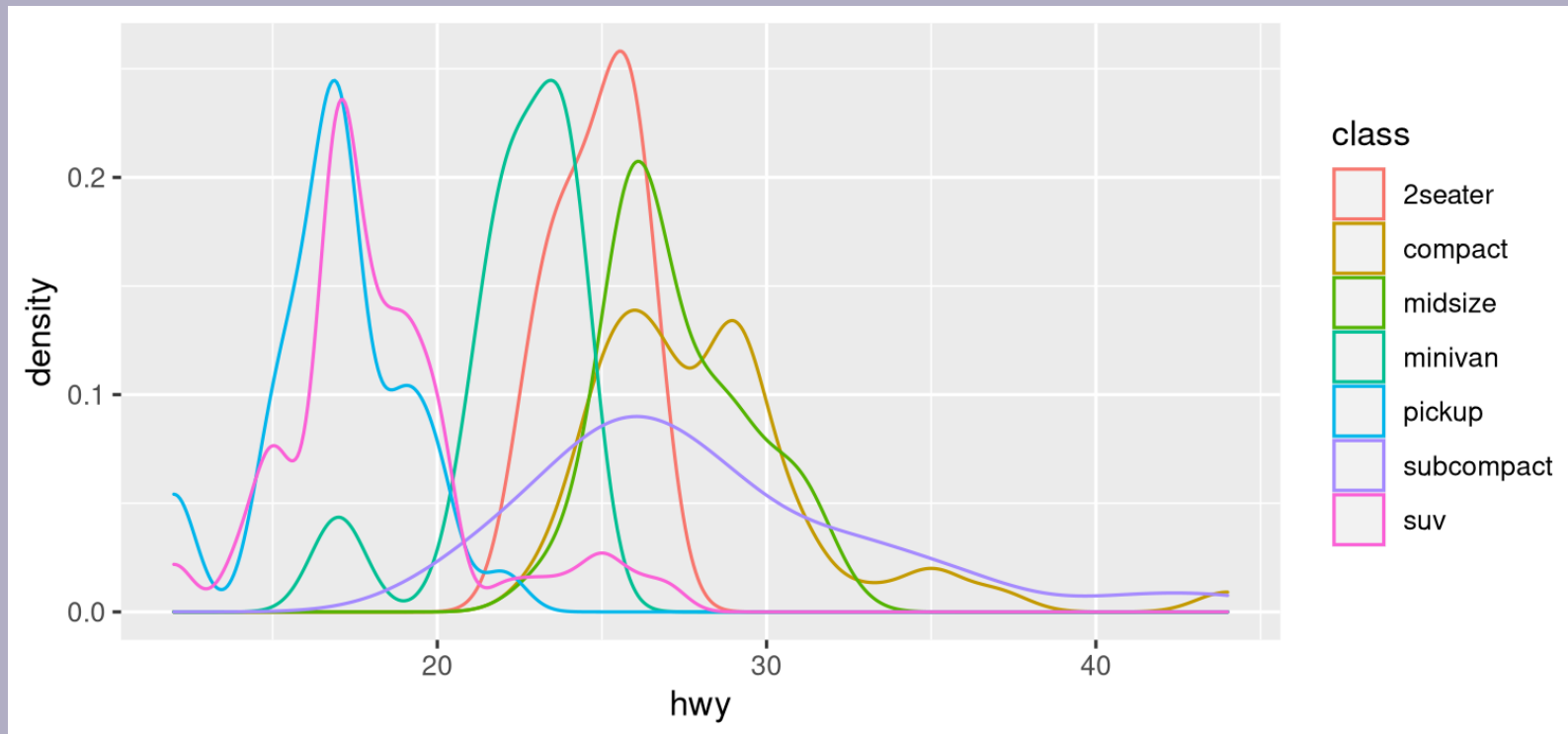
# À votre tour #4

**Produisez un histogramme de `hwy`. Utilisez l'aide mémoire.  
Indice: ne pas fournir de variable `y`.**

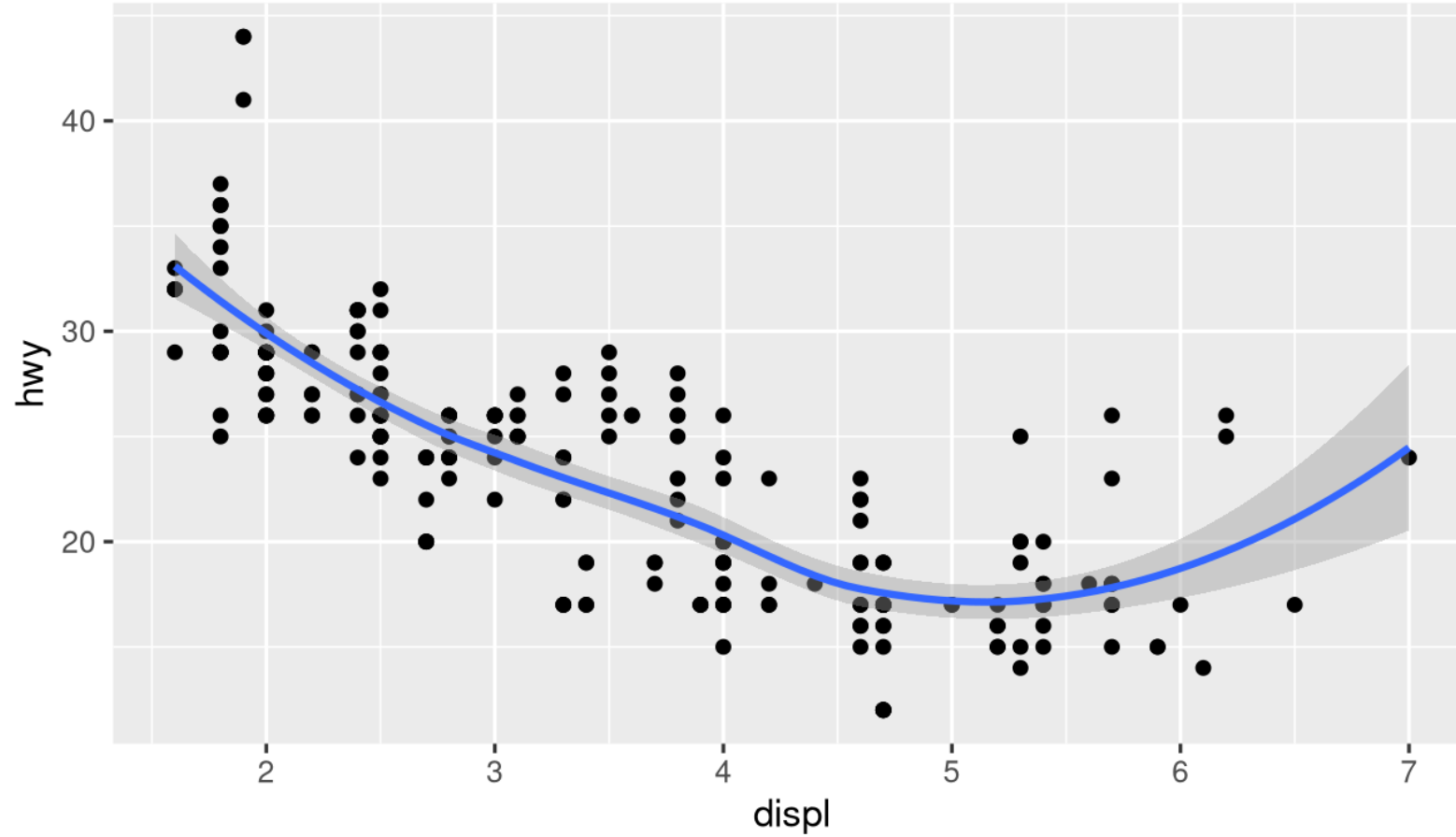


# À votre tour #5

**Colorez la densité de `hwy` par `class`.  
Utilisez l'aide mémoire.  
Indice: ne pas fournir de variable `y`.**



# Graphiques complexes!



# À votre tour #6

**Prédisez ce que le code produit, puis compilez.**

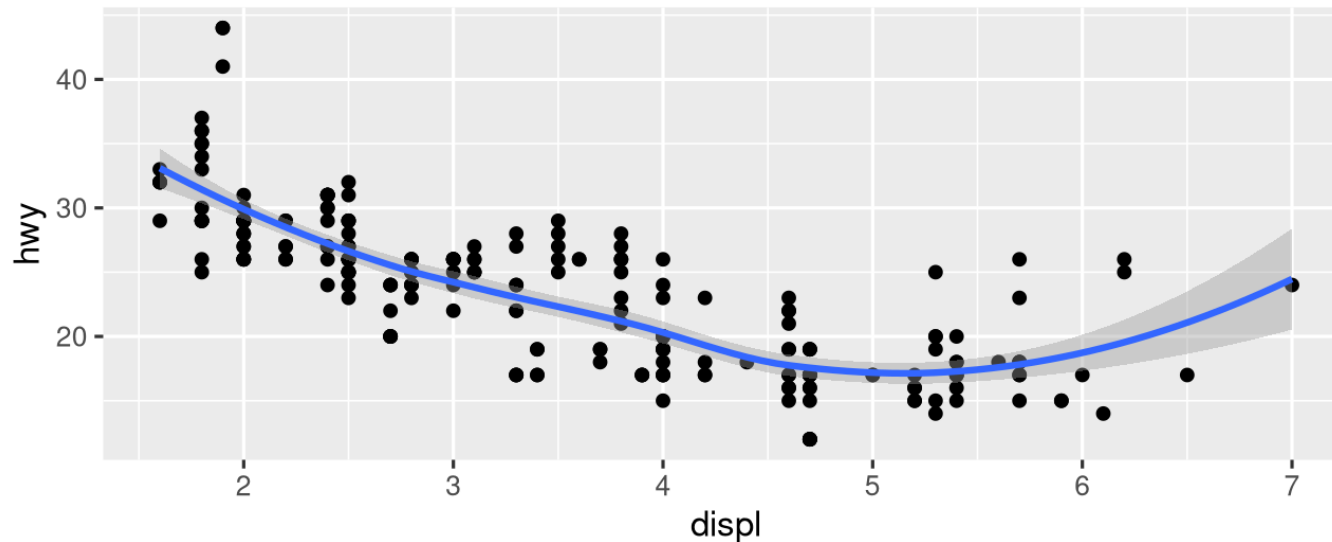
```
ggplot(data = mpg) +  
  geom_point(mapping = aes(x = displ, y = hwy)) +  
  geom_smooth(mapping = aes(x = displ, y = hwy))
```

02:00

# Global vs. local

Toute composante esthétique dans `ggplot()`  
est utilisée dans toutes les strates `geom_`

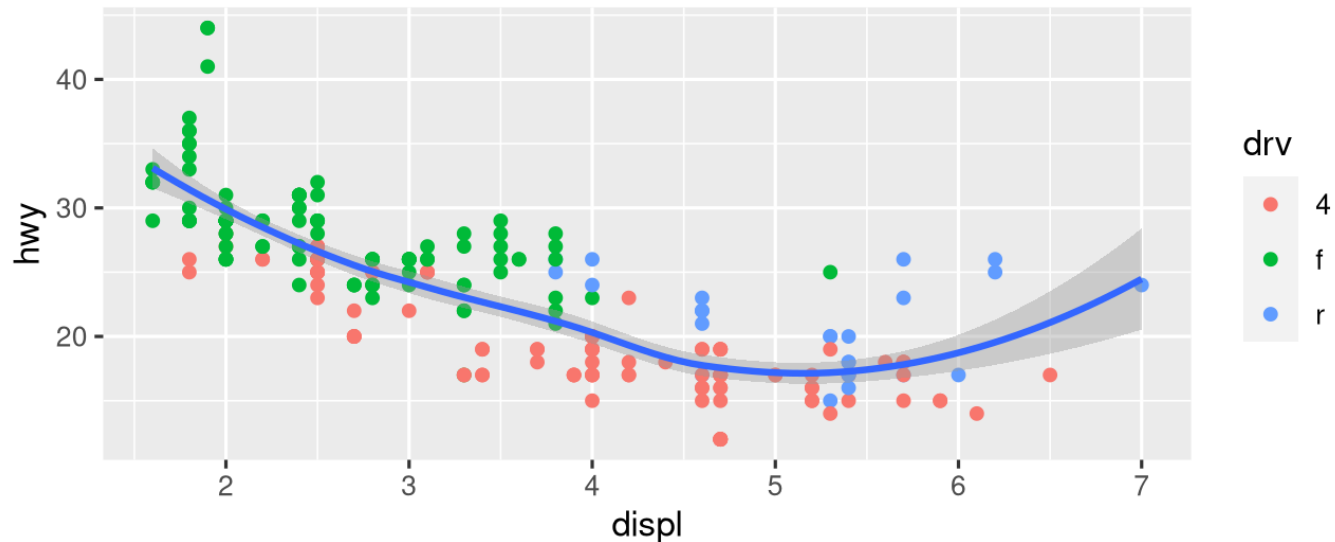
```
ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +  
  geom_point() +  
  geom_smooth()
```



# Global vs. local

Toute composante esthétique fournie dans la strate `geom_` s'applique uniquement à cette dernière

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

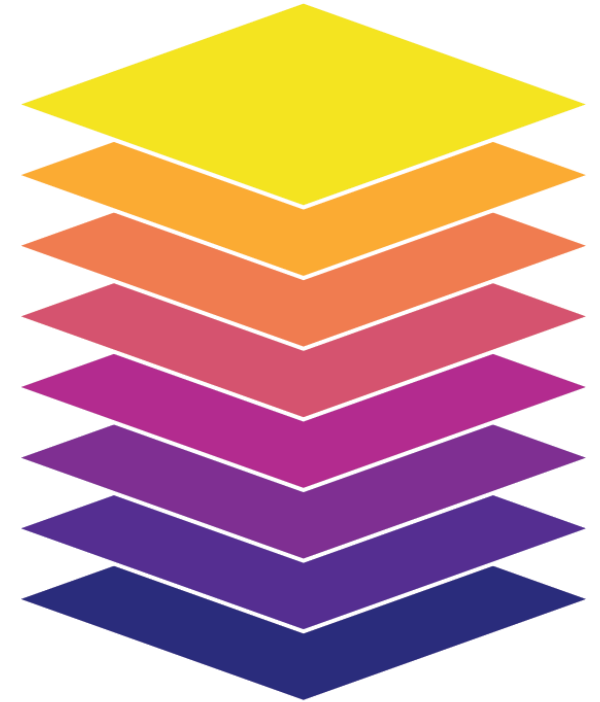


# Bien plus

Plusieurs strates pour améliorer  
l'aspect visuel!

On ajoute des strates  
séquentiellement pour créer des  
figures complexes

Theme  
Labels  
Coordinates  
Facets  
Scales  
Geometries  
Aesthetics  
Data



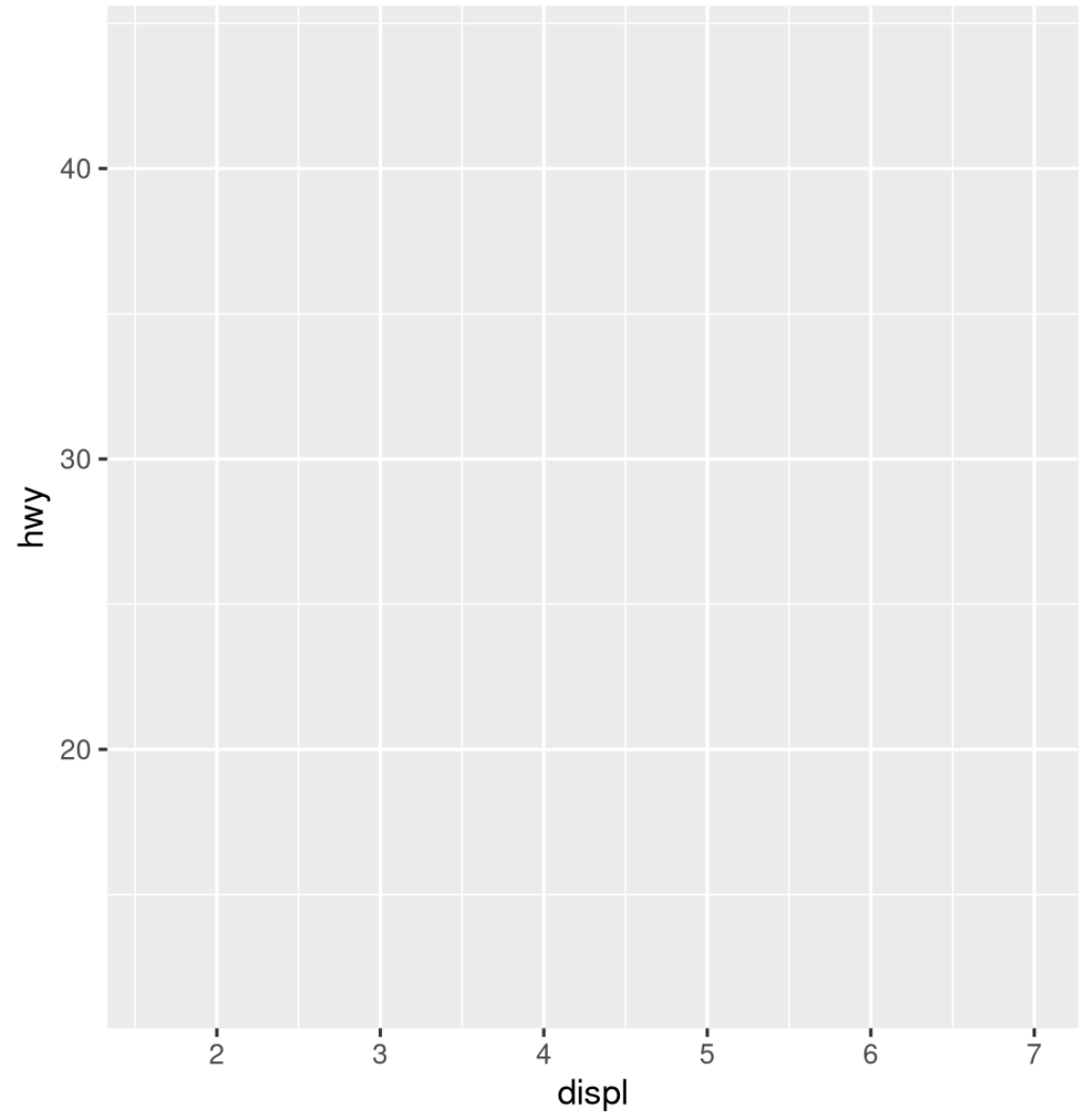


# Mettre tout ensemble

**On construit le graphique étape par étape  
pour voir l'impact des strates  
sur l'apparence.**

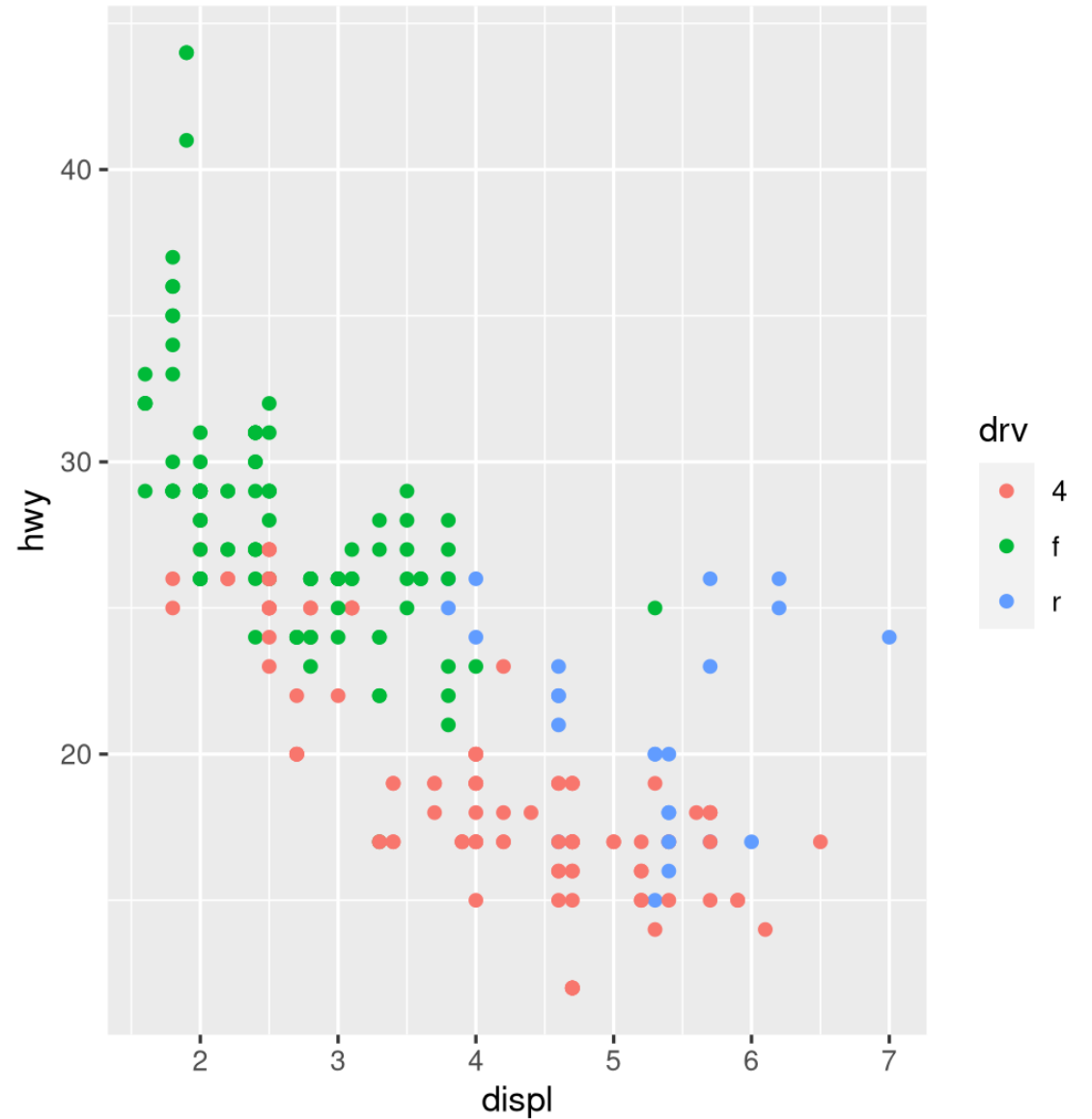
# Données et composantes esthétiques globales

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



# Ajouter une strate géométrique (nuage de points)

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



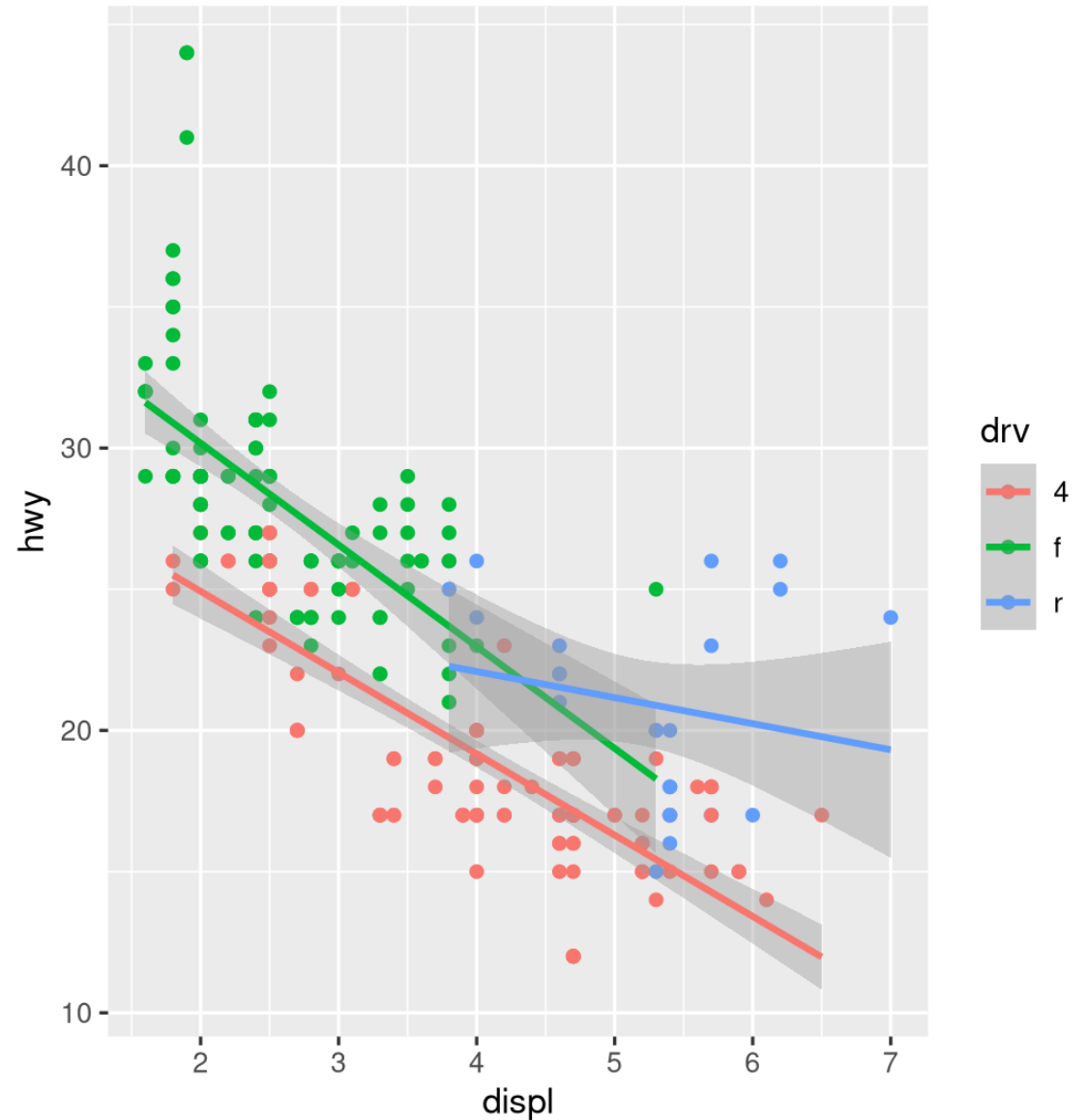
# Ajouter une composante de lissage

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



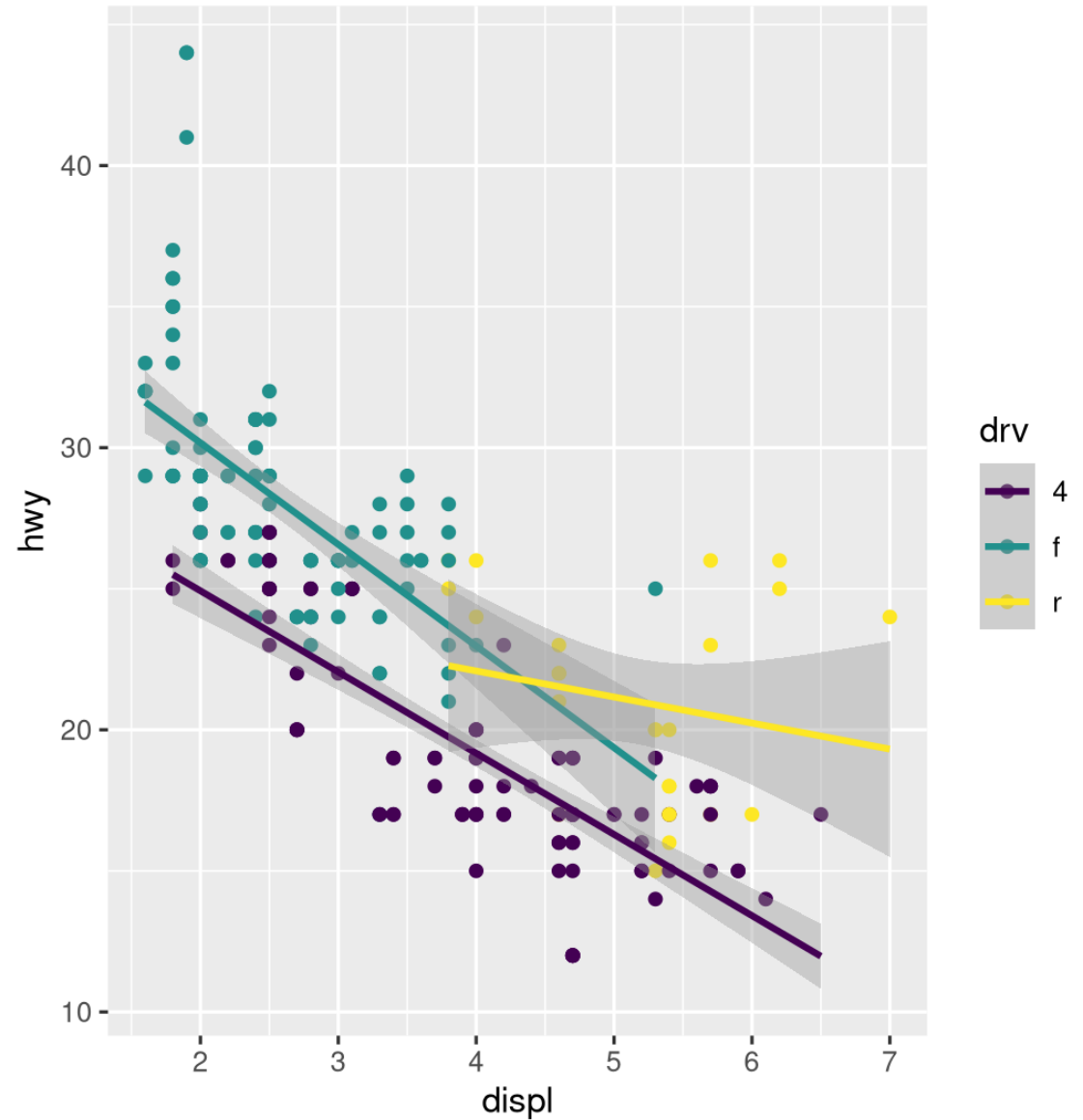
# Changer la droite de lissage par une régression linéaire

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



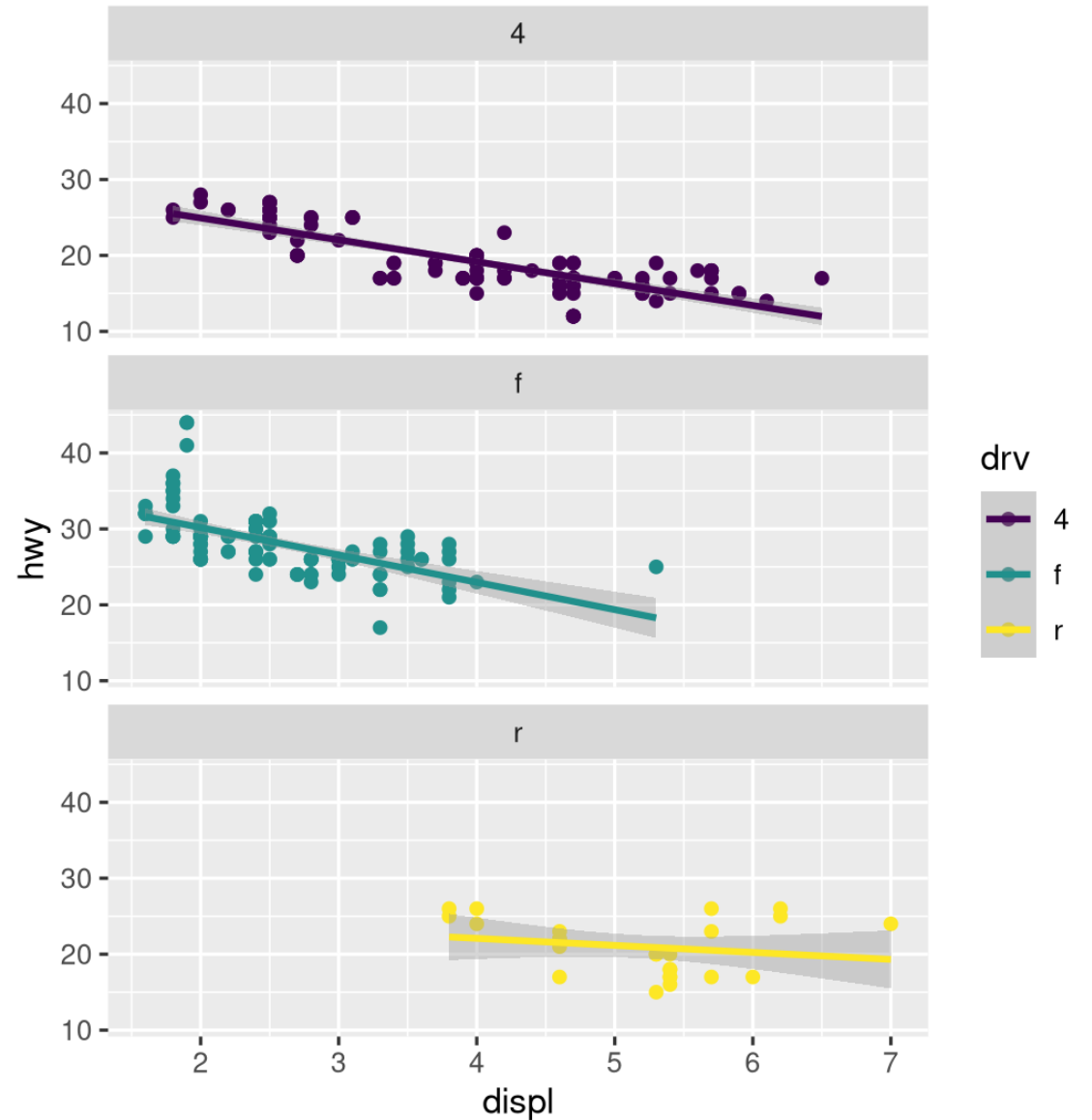
# Choisir la palette de couleur

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



# Séparer en panneaux

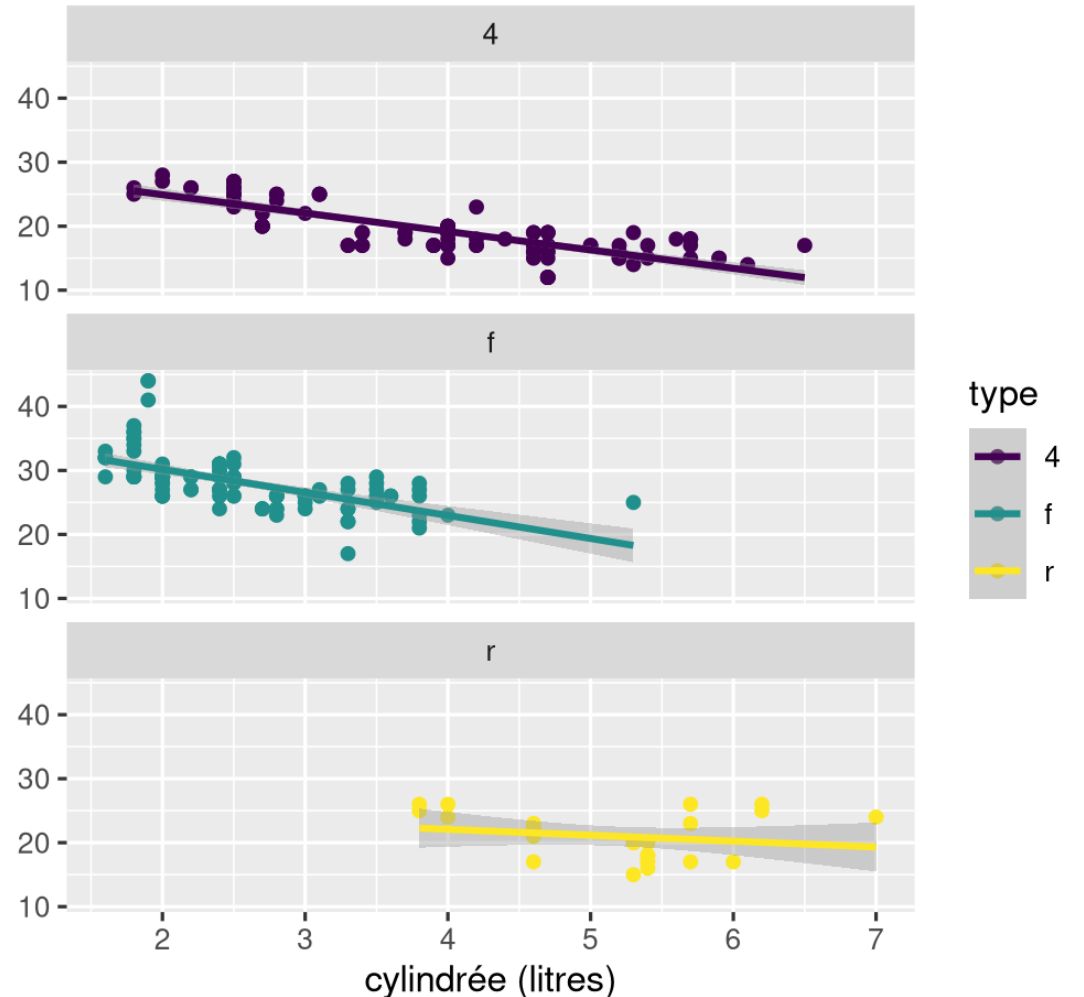
```
ggplot(data = mpg,  
       mapping = aes(x = displ,  
                     y = hwy,  
                     color = drv)) +  
  
  geom_point() +  
  geom_smooth(method = "lm") +  
  scale_color_viridis_d() +  
  facet_wrap(vars(drv), ncol = 1)
```



# Ajouter des étiquettes

```
ggplot(data = mpg,
       mapping = aes(x = displ,
                     y = hwy,
                     color = drv)) +
  geom_point() +
  geom_smooth(method = "lm") +
  scale_color_viridis_d() +
  facet_wrap(vars(drv), ncol = 1) +
  labs(x = "cylindrée (litres)",
       y = "",
       subtitle = "autonomie sur l'autoroute (mil",
       color = "type",
       title = "Cylindrée imposante rime avec moi",
       caption = "Je connais rien aux voitures")
```

Cylindrée imposante rime avec moins d'autonomie.  
autonomie sur l'autoroute (mile au gallon)



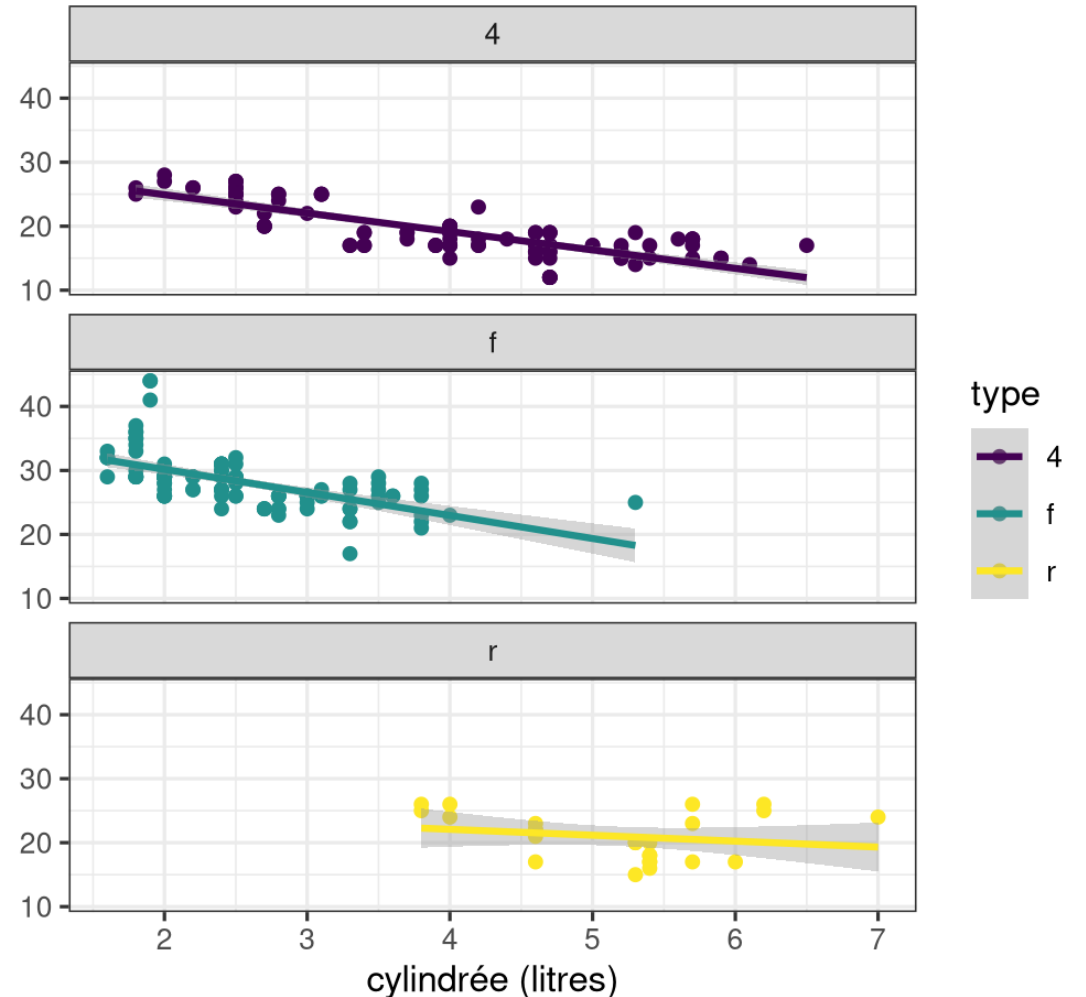
Je connais rien aux voitures



# Ajouter un thème

```
ggplot(data = mpg,
       mapping = aes(x = displ,
                     y = hwy,
                     color = drv)) +
  geom_point() +
  geom_smooth(method = "lm") +
  scale_color_viridis_d() +
  facet_wrap(vars(drv), ncol = 1) +
  labs(x = "cylindrée (litres)",
       y = "",
       subtitle = "autonomie sur l'autoroute (mil",
       color = "type",
       title = "Cylindrée imposante rime avec moi",
       caption = "Je connais rien aux voitures")
  theme_bw()
```

Cylindrée imposante rime avec moins d'autonomie.  
autonomie sur l'autoroute (mille au gallon)

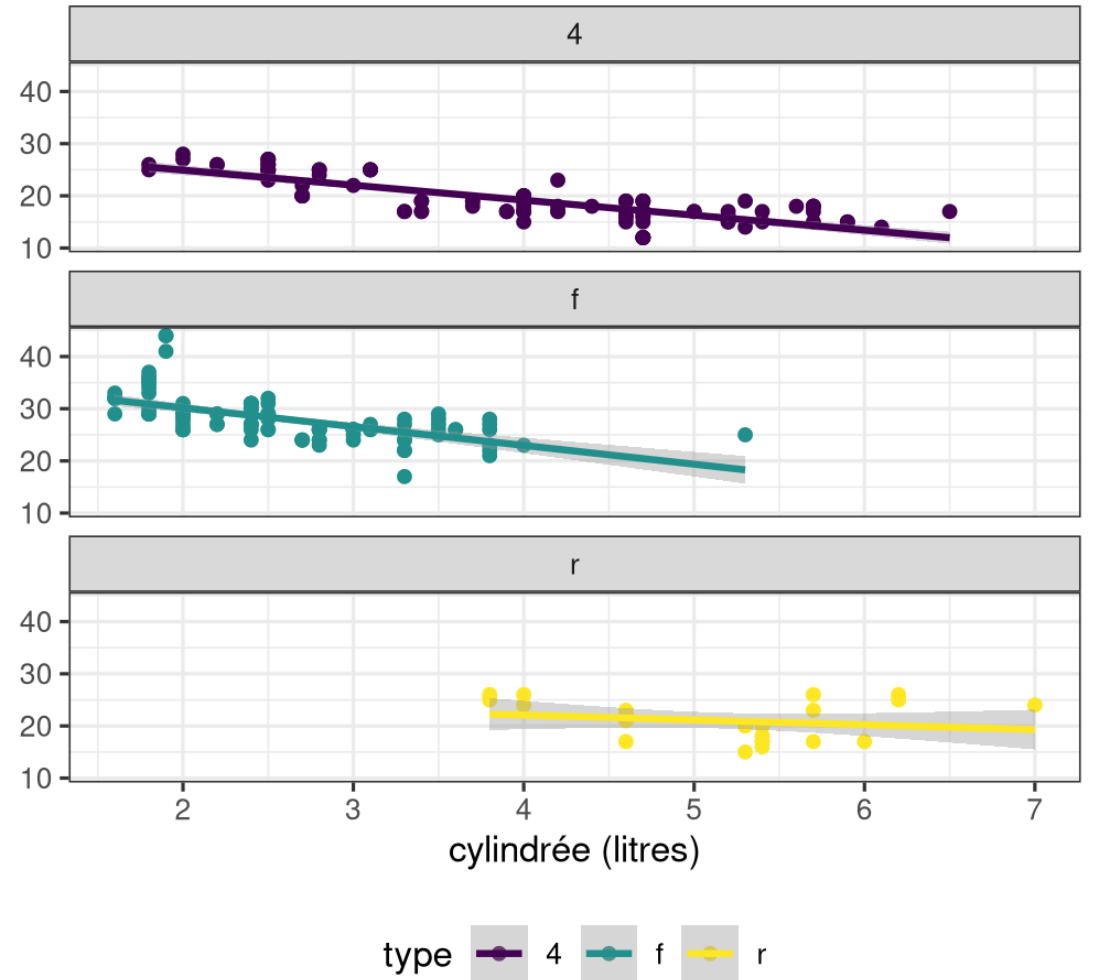


Je connais rien aux voitures

# Modifier le thème

```
ggplot(data = mpg,
       mapping = aes(x = displ,
                     y = hwy,
                     color = drv)) +
  geom_point() +
  geom_smooth(method = "lm") +
  scale_color_viridis_d() +
  facet_wrap(vars(drv), ncol = 1) +
  labs(x = "cylindrée (litres)",
       y = "",
       subtitle = "autonomie sur l'autoroute (mil",
       color = "type",
       title = "Cylindrée imposante rime avec moi",
       caption = "Je connais rien aux voitures")
  theme_bw() +
  theme(legend.position = "bottom")
```

Cylindrée imposante rime avec moins d'autonomie.  
autonomie sur l'autoroute (mile au gallon)

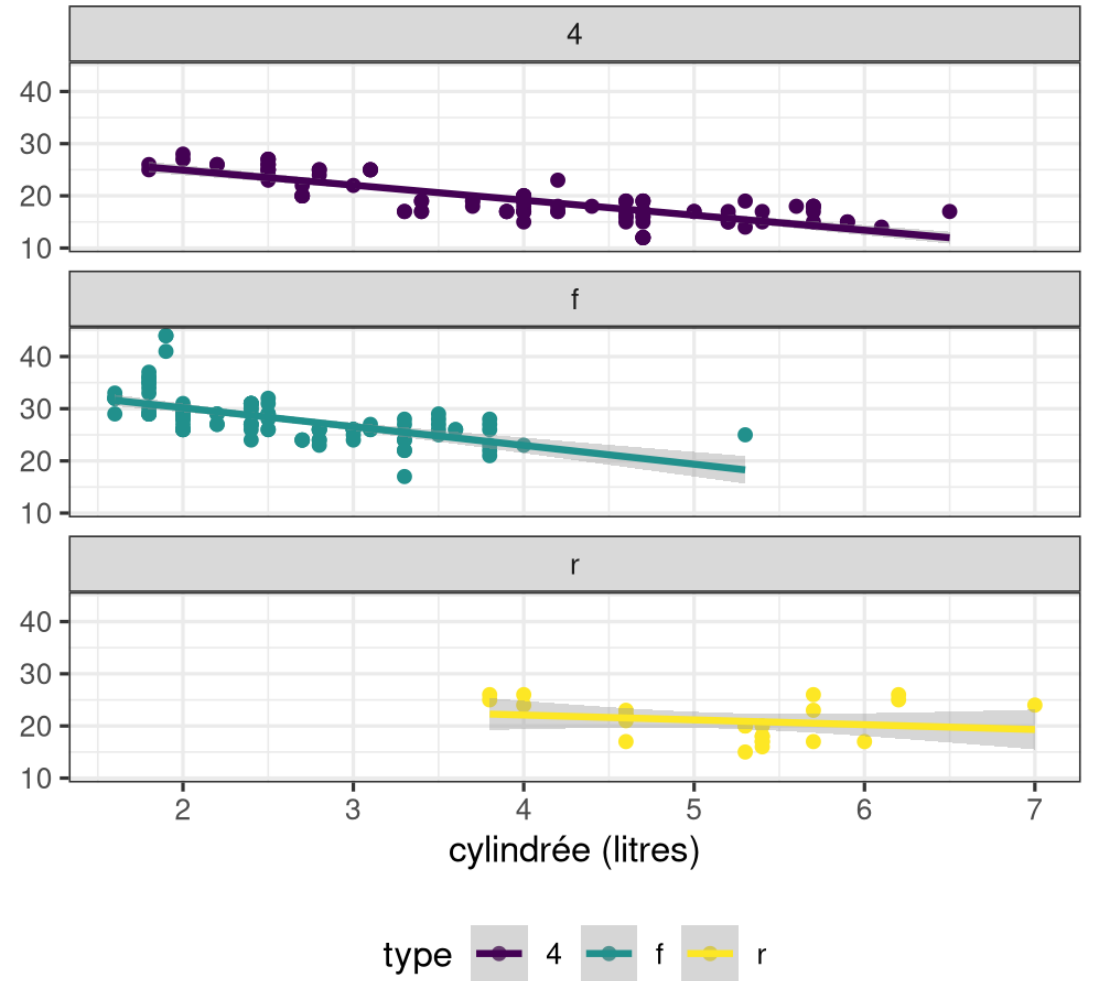


Je connais rien aux voitures

# Terminé!

```
ggplot(data = mpg,
       mapping = aes(x = displ,
                     y = hwy,
                     color = drv)) +
  geom_point() +
  geom_smooth(method = "lm") +
  scale_color_viridis_d() +
  facet_wrap(vars(drv), ncol = 1) +
  labs(x = "cylindrée (litres)",
       y = "",
       subtitle = "autonomie sur l'autoroute (mil",
       color = "type",
       title = "Cylindrée imposante rime avec moi",
       caption = "Je connais rien aux voitures")
  theme_bw() +
  theme(legend.position = "bottom")
```

Cylindrée imposante rime avec moins d'autonomie.  
autonomie sur l'autoroute (mile au gallon)



Je connais rien aux voitures