

# Labtime:Psy.code.logist club

< Session 2 >

ggplot2 sur



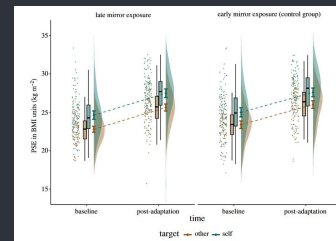
# Table des matières

01 The Grammar of Graphics  
< La logique derrière ggplot2 >



02 A Royal Society Plot

< Mise en pratique - Chazelle et al. 2023 >



03 Beyond The Plot

< Le "combo" .svg + Inscap >



04 Temps d'échange

< Des problèmes sur R ? >

# 01 The Grammar of Graphics

- Elaborée par Wilkinson (2005) pour décrire les éléments fondamentaux d'un graphique
- Implémentée sur R par Wickham (2009) avec le paquet ggplot puis ggplot2

<https://ggplot2-book.org>



TOUT est là

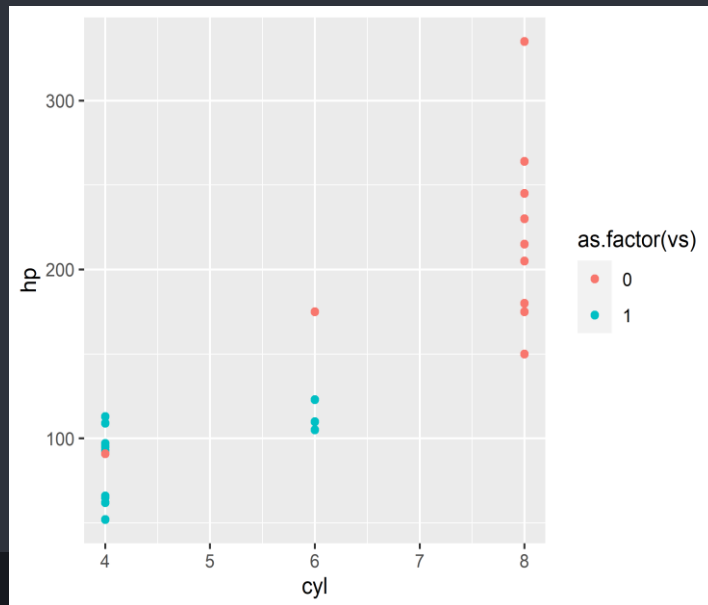


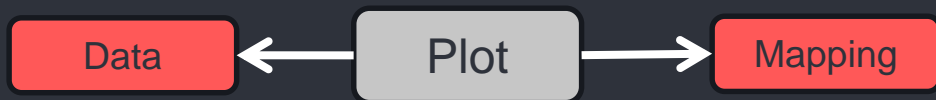
établit correspondance  
entre **variables** et  
**attributs esthétiques**  
("aes")

6  
7  
8  
9  
10  
11  
12  
13  
14

**aes typiques:**  
**x y color fill shape size ...**

```
data("mtcars")  
ggplot(data = mtcars, aes(x = cyl, y = hp,  
color = as.factor(vs)))+  
  geom_point()
```





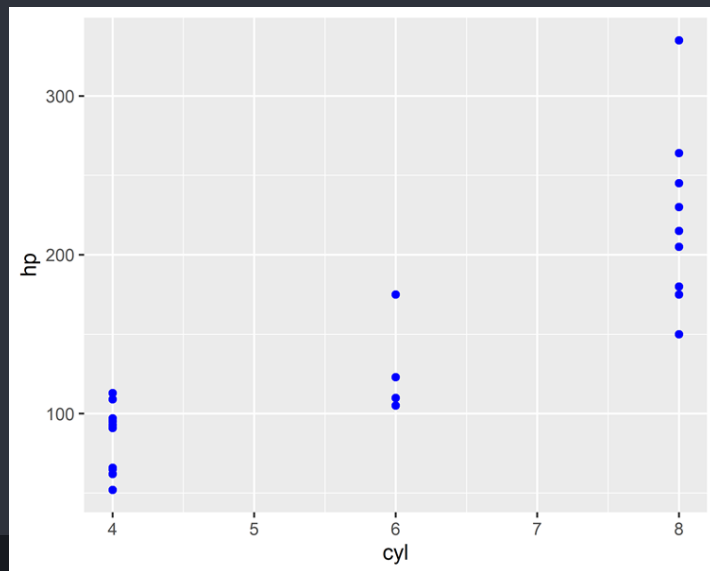
### mapping vs setting

avec `aes()`: attribut variable

sans `aes()`: attribut fixe

```
data("mtcars")
ggplot(data = mtcars, aes(x = cyl, y = hp,
  color = as.factor(vs))) +
  geom_point(color = "blue")
```

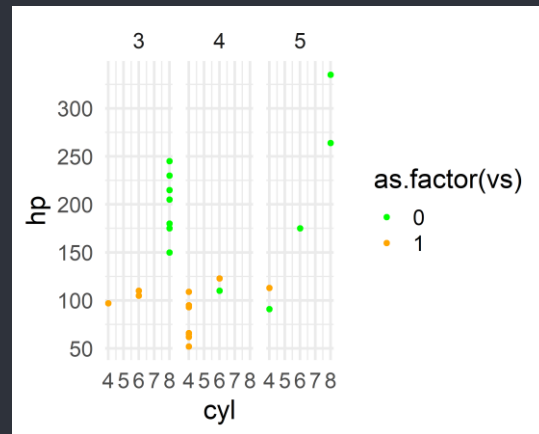
établit correspondance  
entre **variables** et  
**attributs esthétiques**  
("aes")





```
data("mtcars")
ggplot(data = mtcars, aes(x = cyl, y = hp, color = as.factor(vs)))
+
  geom_point() +
  scale_color_manual(values = c("green", "orange")) +
  scale_y_continuous(breaks = seq(50, 350, 50)) +
  facet_wrap(~gear) +
  theme_minimal(base_size = 18)
```

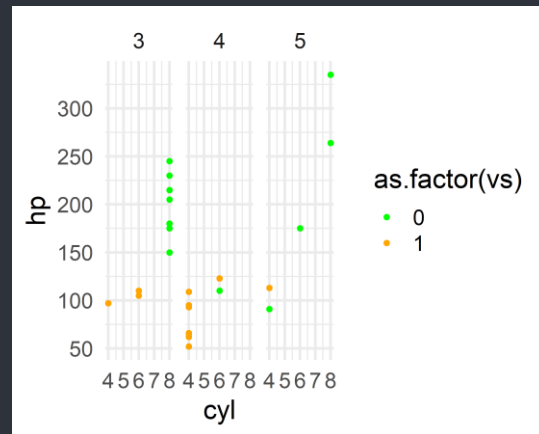
- mapping initial (moche) modifiable  
via un système de **couches (layer)**  
qui s'additionnent (+)





```
data("mtcars")
ggplot(data = mtcars, aes(x = cyl, y = hp, color = as.factor(vs)))
+
  geom_point() +
  scale_color_manual(values = c("green", "orange")) +
  scale_y_continuous(breaks = seq(50, 350, 50)) +
  facet_wrap(~gear) +
  theme_minimal(base_size = 18)
```

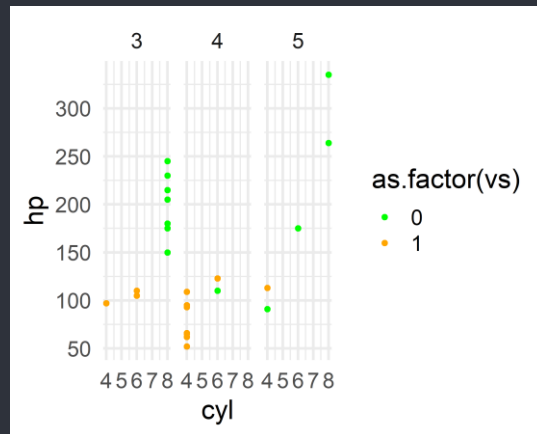
- geom : éléments géométriques
- geom\_line
- geom\_boxplot
- geom\_violin





```
data("mtcars")
ggplot(data = mtcars, aes(x = cyl, y = hp, color = as.factor(vs)))
+
  geom_point() +
  scale_color_manual(values = c("green", "orange")) +
  scale_y_continuous(breaks = seq(50, 350, 50)) +
  facet_wrap(~gear) +
  theme_minimal(base_size = 18)
```

- scale : 1 aes → 1 scale
- ggplot2 convert **data** into **aesthetics** (e.g., 'orange', 'green') with a **scale**

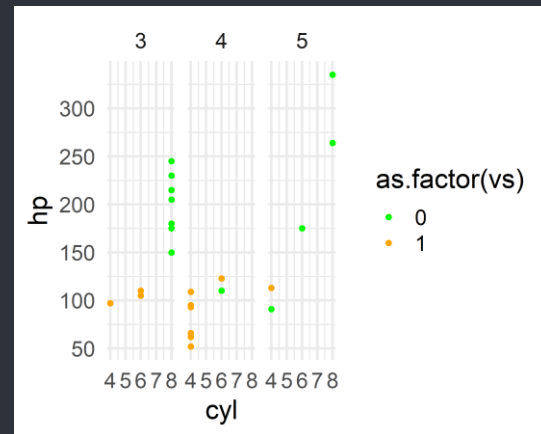






```

data("mtcars")
ggplot(data = mtcars, aes(x = cyl, y = hp, color = as.factor(vs)))
+
  geom_point() +
  scale_color_manual(values = c("green", "orange"))+
  scale_y_continuous(breaks = seq(50, 350, 50)) +
  facet_wrap(~gear)+
  theme_minimal(base_size = 18)
  
```

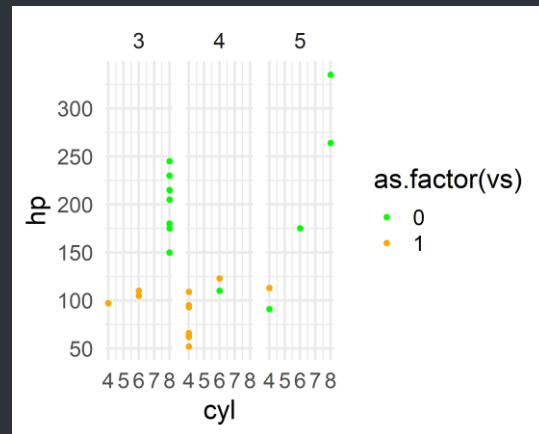


- système de coordonnées
- limites axes
- labels axes
- nombres de tirets...



```
data("mtcars")
ggplot(data = mtcars, aes(x = cyl, y = hp, color = as.factor(vs)))
+
  geom_point() +
  scale_color_manual(values = c("green", "orange")) +
  scale_y_continuous(breaks = seq(50, 350, 50)) +
  facet_wrap(~gear) +
  theme_minimal(base_size = 18)
```

- facet : rajout d'une 4<sup>ème</sup> variable
- theme : customisation du plot





- **Avantages de ggplot2 par rapport à graphics (base R)**
- Code plus **épuré** et **intuitif**
- chaque couche (layer) **hérite des aes de la 1<sup>ère</sup> couche**
- une couche n+5 peut **modifier les couches précédentes !** (e.g. avec theme)

Pour une comparaison ggplot2 vs graphics voir :

<https://flowingdata.com/2016/03/22/comparing-ggplot2-and-r-base-graphics/>

```
par(mfrow=c(2,4))
days <- c("Thur", "Fri", "Sat", "Sun")
sexes <- unique(tips$sex)
for (i in 1:length(sexes)) {
  for (j in 1:length(days)) {
    currdata <- tips[tips$day == days[j] & tips$sex == sexes[i],]
    plot(currdata$total_bill, currdata$tip/currdata$total_bill,
         main=paste(days[j], sexes[i], sep=", "), ylim=c(0,0.7), las=1)
  }
}
```

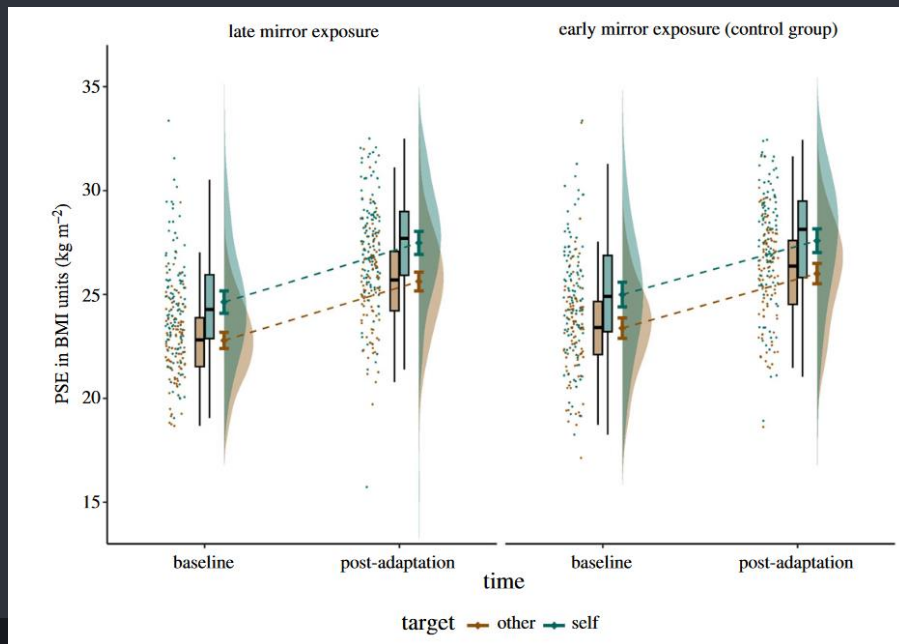
```
sp <- ggplot(tips, aes(x=total_bill, y=tip/total_bill))
  + geom_point(shape=1)
sp + facet_grid(sex ~ day)
```

nb : pour un très  
haut degrés de  
**souplesse**,  
graphics est  
cependant plus  
pratique que  
ggplot2

# 02 A Royal Society Plot

THE ROYAL SOCIETY  
PUBLISHING

- Elaboré par **Chazelle** et al. (2023)  
<https://doi.org/10.1098/r sos.221589>
- Haut degré de complexité et de finesse
- Challenge du labtime : reproduire (à peu près) ce plot



- Données au format long "[chazelle2007labtime.csv](#)"
- Code dans le fichier .Rmd
- Accessible sur <https://github.com/lafitter/labtime>

**VI target**

- tâche : juger si corps plus large que moi OU Emma Watson

- mesure **pre** vs **post** intervention **VI time**

**VI Exposure**

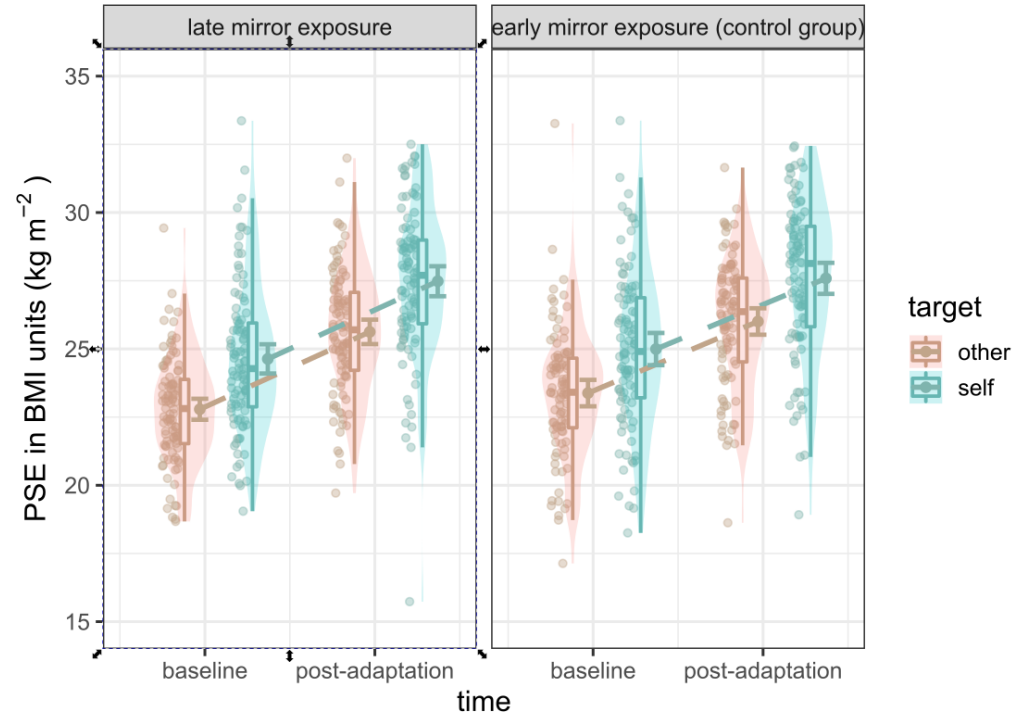
- intervention 1 : se regarder puis regarder un corps large (**cond c**)
- intervention 2 : regarder corps large puis se regarder (**cond m**)

##	Participant	Exposure	PSE	target	time
## 1	1	m	23.05948	self	pre
## 2	1	m	23.47127	other	pre
## 3	1	m	26.12832	self	post
## 4	1	m	26.75768	other	post
## 5	2	m	22.23160	self	pre
## 6	2	m	22.40942	other	pre

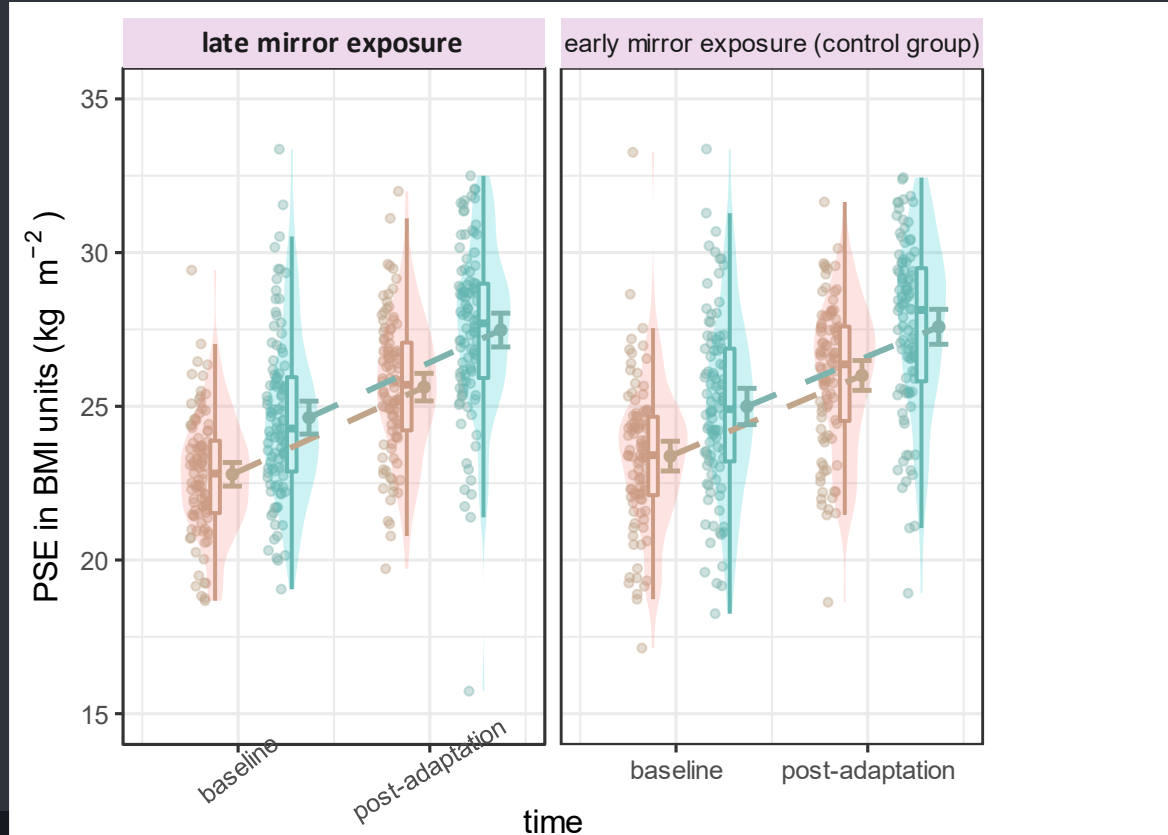
# 03 Beyond The Plot

- Parfois R ne suffit pas pour atteindre la perfection
- Stratégie intéressante : exporter le plot en .svg et le customiser avec Inkscape (ou autre logiciel)









# 04 Temps d'échange

- Des soucis sur R ?



# Merci pour votre participation,

Psy.code.logist club n°3 ?

Tout savoir (ou presque) sur GIT dans R !

**DIAPOS, SCRIPTS, BIBLIOGRAPHIE**

<https://cloud.univ-grenoble-alpes.fr/s/RPrAmF4ogA9A3Hx>

