

11. Les bases de ggplot2

Le b-a-ba de ggplot2

La grammaire des graphiques est impléée dans ggplot2, un ensemble d'outils qui met en correspondance les données aux éléments d'affichage visuel et qui permettent aux utilisateurs de contrôler les détails de l'affichage du graphique.

L'aspect le plus important est que ggplot2 peut être utilisé pour réfléchir à la **structure logique** des graphiques.

Un graphique ggplot2 possède deux composantes principales :

- **mappings esthétiques** (aes – connexions entre les données et les éléments du graphique)
- **géométrie** (geom – spécifie le type de graphique)
- *facettes, *coordonnées, *échelles, *étiquettes, *guides, etc. (facultatifs)

La grammaire de ggplot2

1. Tidy Data

```
p <- ggplot(data = gapminder, ...
```

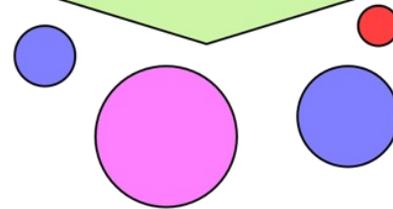
gdp	lifexp	pop	continent
340	65	31	Euro
227	51	200	Amer
909	81	80	Euro
126	40	20	Asia

2. Mapping

```
p <- ggplot(data = gapminder, mapping =
  aes(x = gdp, y = lifexp, size = pop,
      color = continent))
```

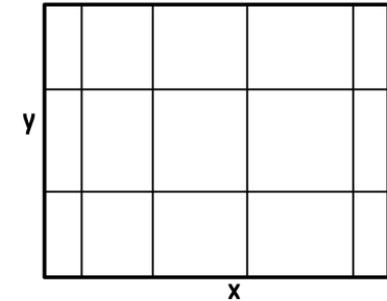
3. Geom

```
p + geom_point()
```



4. Co-Ordinates & Scales

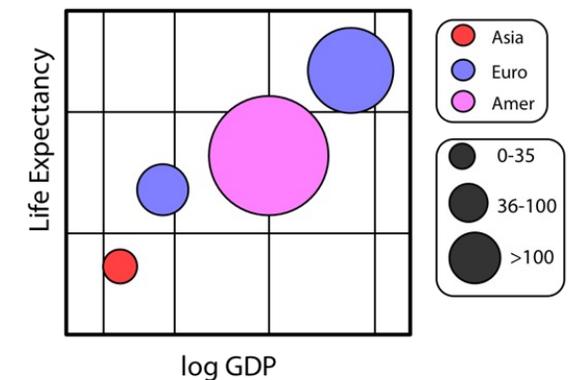
```
p + coord_cartesian() + scale_x_log10()
```



5. Labels & Guides

```
p + labs(x = "log GDP", y = "Life Expectancy",
  title = "A Gapminder Plot")
```

A Gapminder Plot



[Healey, K., *Data Visualization : A Practical Introduction*].

Les géométries de ggplot2

La source de données et les variables à tracer sont spécifiées via `ggplot()`.

Les différentes fonctions `geom()` précisent **comment** ces variables doivent être représentées visuellement :

- en utilisant des points, des barres, des lignes, des régions ombrées, etc.

Il y a actuellement plus de 37 géométries disponibles.

Function	Adds	Options
<code>geom_bar()</code>	bar chart	color, fill, alpha
<code>geom_boxplot()</code>	boxplot	color, fill, alpha, notch, width
<code>geom_density()</code>	density plot	color, fill, alpha, linetype
<code>geom_histogram()</code>	histogram	color, fill, alpha, linetype, binwidth
<code>geom_hline()</code>	horizontal lines	color, alpha, linetype, size
<code>geom_line()</code>	jittered points	color, size, alpha, shape
<code>geom_jitter()</code>	line graph	color, alpha, linetype, size
<code>geom_point()</code>	scatterplot	color, alpha, shape, size
<code>geom_rug()</code>	rug plot	color, side
<code>geom_smooth()</code>	fitted line	method, formula, color, fill, linetype, size
<code>geom_text()</code>	text annotations	many; see the help for this function
<code>geom_violin()</code>	violin plot	color, fill, alpha, linetype
<code>geom_vline()</code>	vertical lines	color, alpha, linetype, size

Option	Specifies
color	colour of points, lines, and borders around filled regions
fill	colour of filled areas such as bars and density regions
alpha	transparency of colors, ranging from 0 (fully transparent) to 1 (opaque)
linetype	pattern for lines (1 = solid, 2 = dashed, 3 = dotted, 4 = dotdash, 5 = longdash, 6 = twodash)
size	point size and line width
shape	point shapes (same as pch, with 0 = open square, 1 = open circle, 2 = open triangle, and so on)
position	position of plotted objects such as bars and points. For bars, <code>dodge''</code> places grouped bar charts side by side, <code>stacked''</code> vertically stacks grouped bar charts, and <code>fill''</code> vertically stacks grouped bar charts and standardizes their heights to be equal; for points, <code>jitter''</code> reduces point overlap
binwidth	bin width for histograms
notch	indicates whether box plots should be notched (TRUE/FALSE)
sides	placement of rug plots on the graph (<code>b''</code> = bottom, <code>l''</code> = left, <code>t''</code> = top, <code>r''</code> = right, <code>bl''</code> = both bottom and left, and so on)
width	width of box plots

Les geom() de ggplot2

```
library("ggplot2")
data(singer, package="lattice")
# Using data from the 1979 ed. of the
# New York Choral Society

# Histogram of heights
ggplot(singer, aes(x=height)) +
  geom_histogram()

# Boxplot of heights by voice part
ggplot(singer, aes(x=voice.part, y=height)) +
  geom_boxplot()
```

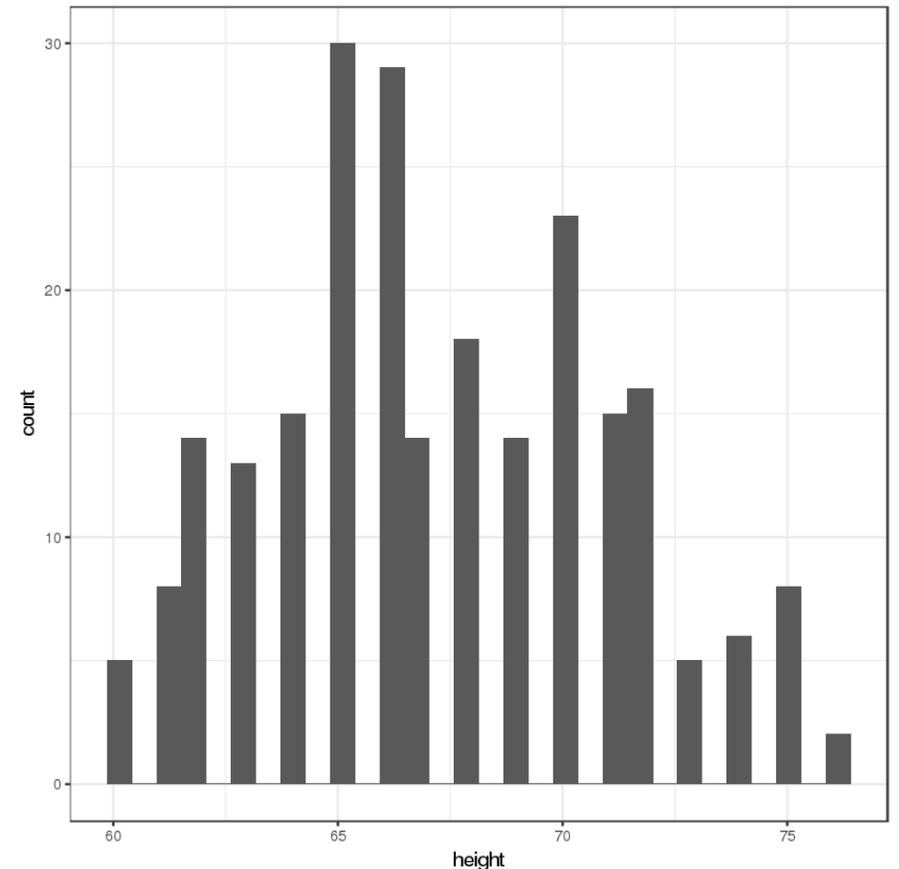
Quel est le résultat attendu ?

Les geom() de ggplot2

```
library("ggplot2")
data(singer, package="lattice")
# Using data from the 1979 ed. of the
# New York Choral Society

# Histogram of heights
ggplot(singer, aes(x=height)) +
  geom_histogram()

# Boxplot of heights by voice part
ggplot(singer, aes(x=voice.part, y=height)) +
  geom_boxplot()
```

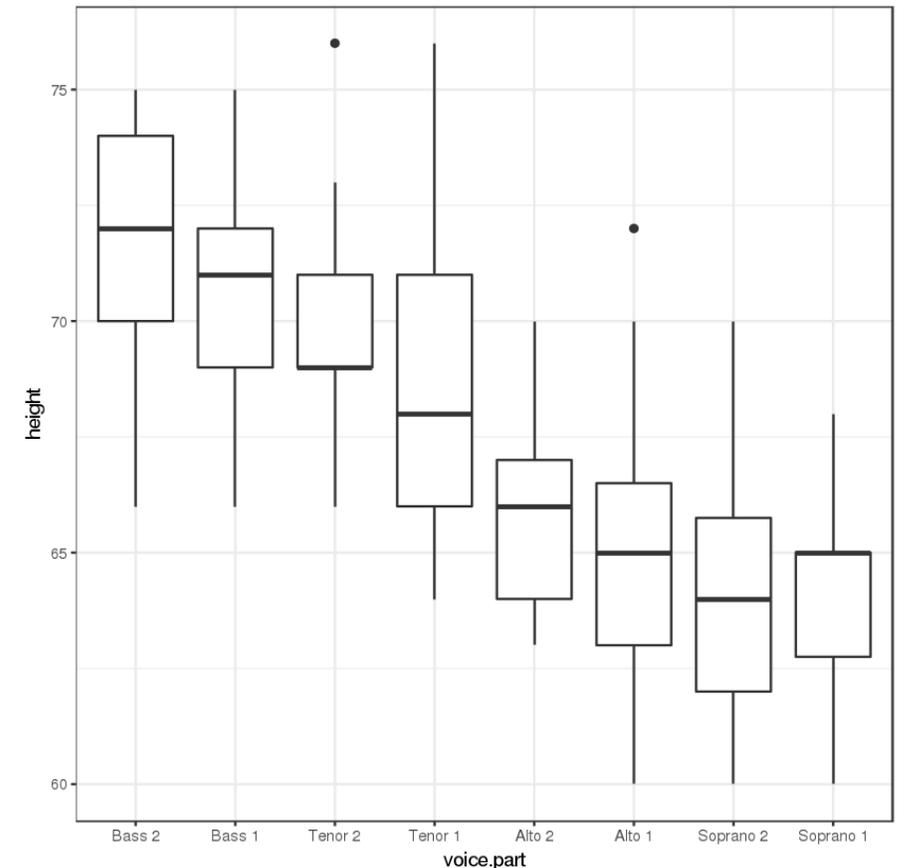


Les geom() de ggplot2

```
library("ggplot2")
data(singer, package="lattice")
# Using data from the 1979 ed. of the
# New York Choral Society

# Histogram of heights
ggplot(singer, aes(x=height)) +
  geom_histogram()

# Boxplot of heights by voice part
ggplot(singer, aes(x=voice.part, y=height)) +
  geom_boxplot()
```



Les geom() de ggplot2

```
library(ggplot2)
data(Salaries, package="car")
# Using data on salaries of a sample of
# US university professors (2018-2019)
# var: rank, sex, yrs.since.phd, yrs.service, salary

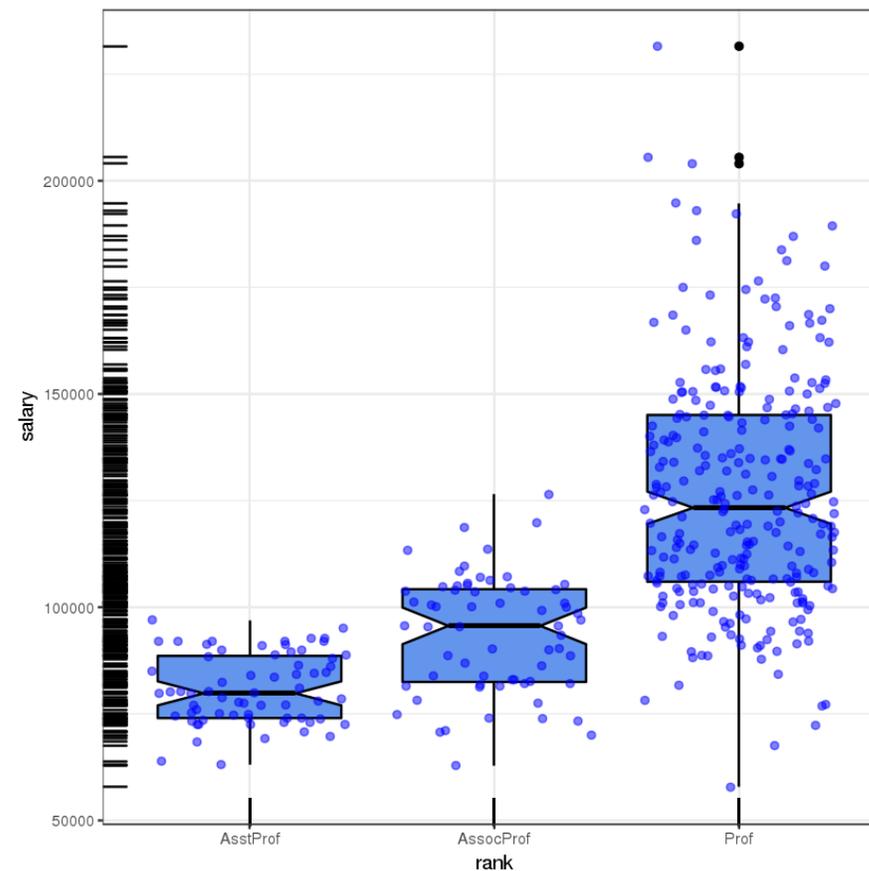
ggplot(Salaries, aes(x=rank, y=salary)) +
  geom_boxplot(fill="cornflowerblue", color="black", notch=TRUE) +
  geom_point(position="jitter", color="blue", alpha=.5) +
  geom_rug(side="l", color="black")
```

Quel est le résultat attendu ?

Les geom() de ggplot2

```
library(ggplot2)
data(Salaries, package="car")
# Using data on salaries of a sample of
# US university professors (2018-2019)
# var: rank, sex, yrs.since.phd, yrs.service, salary

ggplot(Salaries, aes(x=rank, y=salary)) +
  geom_boxplot(fill="cornflowerblue",color="black", notch=TRUE) +
  geom_point(position="jitter", color="blue", alpha=.5) +
  geom_rug(side="l", color="black")
```



L'esthétique de ggplot2

L'**esthétique** fait référence aux attributs affichés des données.

La **correspondance** associe les données à des attributs (tels que la taille ou la forme d'un marqueur) et génèrent une légende appropriée.

Les aspects esthétiques sont spécifiées avec la fonction `aes()`, soit dans l'appel de données, soit dans un appel à `geom()`. Si elles sont spécifiées dans `ggplot()`, elles s'appliquent à toutes les géométries spécifiées.

Les aes() de ggplot2

À titre d'exemple, les esthétiques disponibles pour `geom_point()` sont les suivantes :

- `x` , `y` , `alpha` , `colour` , `fill` , `shape`, `size`

Il y a une **différence importante** entre la spécification des caractéristiques (comme la couleur, la forme) à l'intérieur et à l'extérieur de l'appel à `aes()`

- à l'**intérieur** : couleur ou forme attribuée automatiquement en fonction des données
- à l'**extérieur** : les attributs ne sont pas mis en correspondance avec les données

Les aes() de ggplot2

```
library(ggplot2)
# Using the mpg dataset

# specifying characteristics inside aes()
ggplot(mpg, aes(cty, hwy)) +
  geom_point(aes(colour = class))

# specifying characteristics inside aes()
ggplot(mpg, aes(cty, hwy)) +
  geom_point(colour = "red")
```

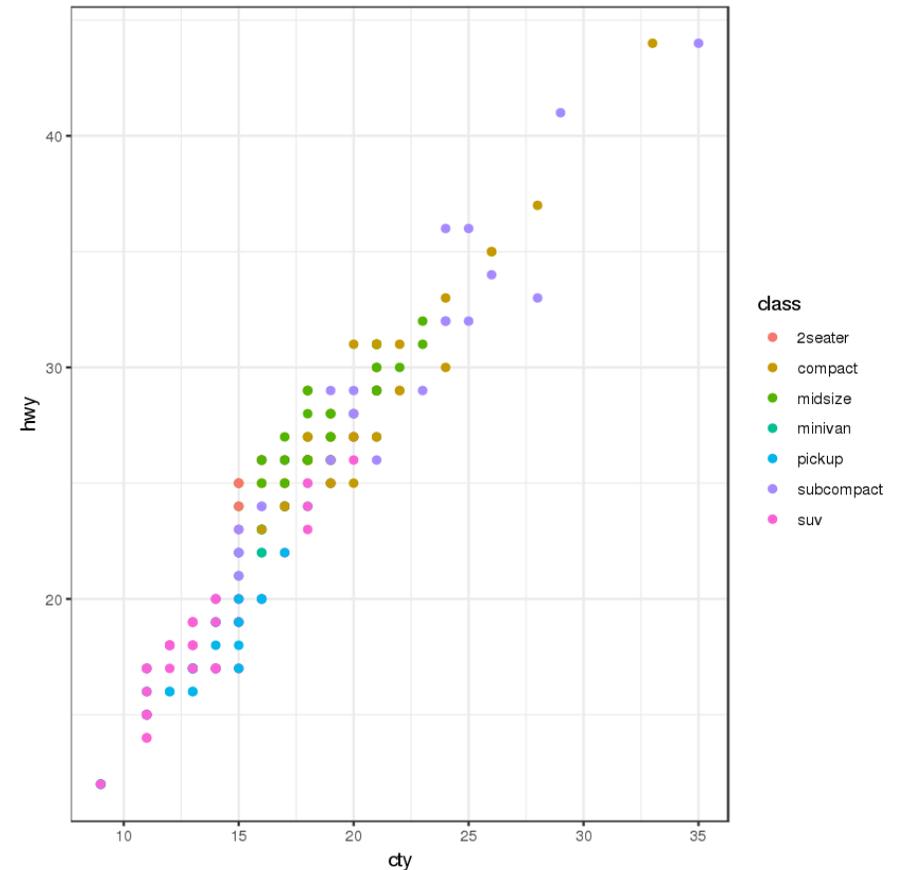
Quel est le résultat attendu ?

Les aes() de ggplot2

```
library(ggplot2)
# Using the mpg dataset

# specifying characteristics inside aes()
ggplot(mpg, aes(cty, hwy)) +
  geom_point(aes(colour = class))

# specifying characteristics inside aes()
ggplot(mpg, aes(cty, hwy)) +
  geom_point(colour = "red")
```

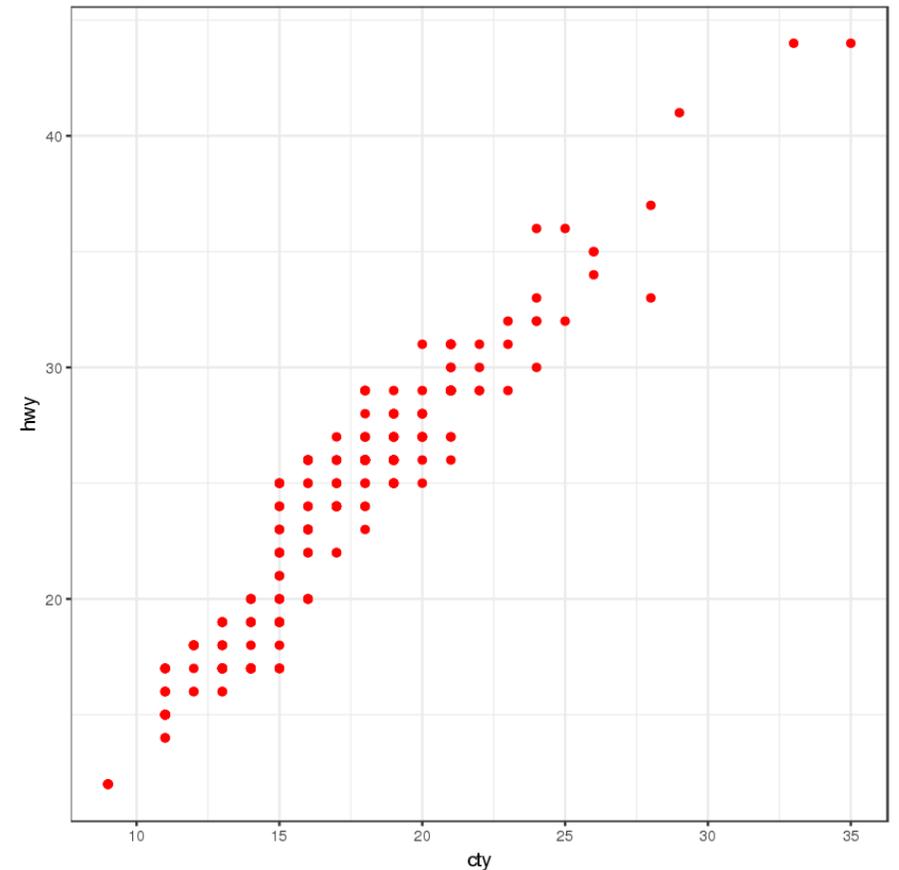


Les aes() de ggplot2

```
library(ggplot2)
# Using the mpg dataset

# specifying characteristics inside aes()
ggplot(mpg, aes(cty, hwy)) +
  geom_point(aes(colour = class))

# specifying characteristics outside aes()
ggplot(mpg, aes(cty, hwy)) +
  geom_point(colour = "red")
```



Les facettes de ggplot2

Dans le langage de ggplot2, les petits multiples sont appelés des facettes

- `facet_wrap()`, `facet_grid()`

Par défaut, tous les panneaux (un pour chaque facteur) partagent les mêmes axes (à l'échelle).

En séparant le graphique en une série de petits tracés côte-à-côte, il est plus facile de procéder à des comparaisons.

Les “wraps” n'affichent que les petits multiples pour lesquels il existe des données, les “grids” affichent tous les multiples, même ceux qui sont vides.

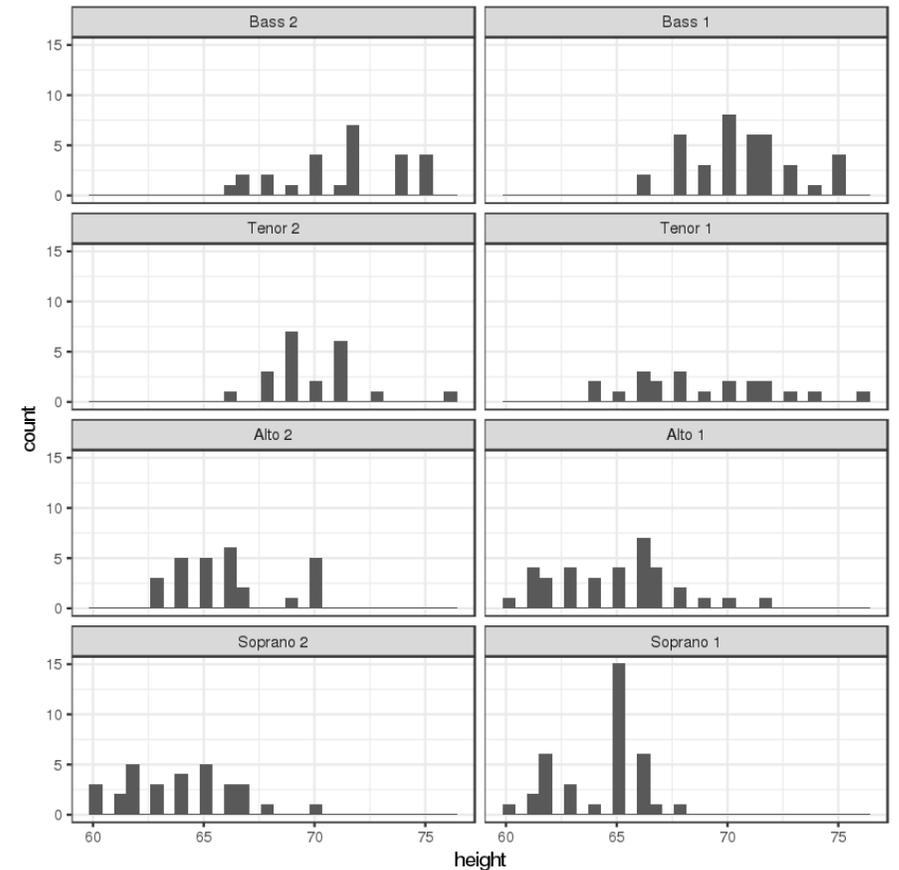
Les facet() de ggplot2

```
data(singer, package="lattice")  
library(ggplot2)  
ggplot(data=singer, aes(x=height)) +  
  geom_histogram() +  
  facet_wrap(~voice.part, nrow=4)
```

Quel est le résultat attendu ?

Les facet() de ggplot2

```
data(singer, package="lattice")
library(ggplot2)
ggplot(data=singer, aes(x=height)) +
  geom_histogram() +
  facet_wrap(~voice.part, nrow=4)
```



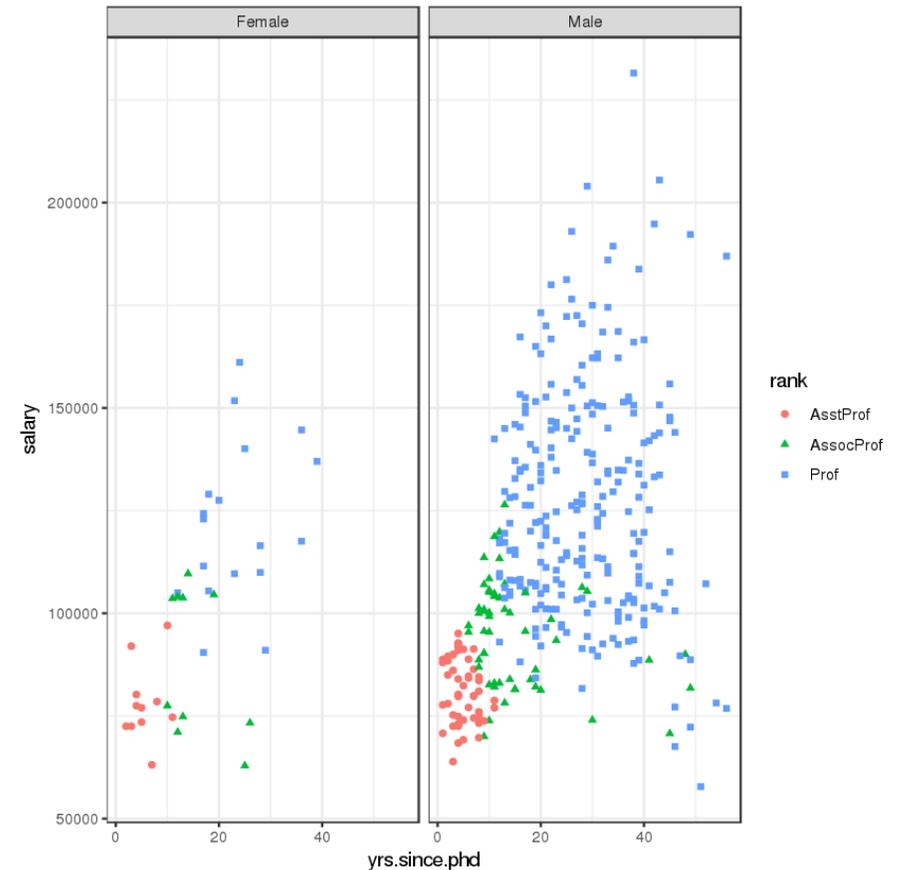
Les facet() de ggplot2

```
data(Salaries, package="car")
library(ggplot2)
ggplot(Salaries, aes(x=yrs.since.phd,
  y=salary, color=rank, shape=rank)) +
  geom_point() +
  facet_grid(~sex)
```

Quel est le résultat attendu ?

Les facet() de ggplot2

```
data(Salaries, package="car")
library(ggplot2)
ggplot(Salaries, aes(x=yrs.since.phd,
  y=salary, color=rank, shape=rank)) +
  geom_point() +
  facet_grid(~sex)
```



Lectures suggérées

Les bases de ggplot2

Data Understanding, Data Analysis, Data Science
Data Visualization and Data Exploration

ggplot2 Visualizations in R

- Basics of ggplot2's Grammar
- ggplot2 Miscellanea

Exercices

Les bases de ggplot2

Créez quelques visualisations ggplot2 simples avec les ensembles de données disponibles dans R. L'accent est mis sur la familiarisation avec diverses géométries, leur esthétique, et l'utilisation de facettes.

Vous pouvez utiliser les exemples trouvés dans ce module comme fondations de votre travail.

Les ensembles de données disponibles sont obtenus en exécutant

```
> données()
```

à l'invite de commande R.