

ggplot2 implémente la grammaire de la visualisation

Les essentiels

Data: Données source.

Geoms: Marques de la visualisation (points, lignes, ...)

Scales: Echelles de la visualisation (position, taille, couleur,...)

ggplot2 implémente la grammaire de la visualisation

Les bonus

Stats: Opérations de modification des données sources (moyenne, ...)

Faceting: Découpe le dataset pour créer des sous-graphes

Coordinates: système de coordonnées de la visualisation

Architecture d'un graph avec ggplot2

```
ggplot(data = <DATA.FRAME>, # chargement des données
        aes(x=<VARIABLE1>,   # définition des aesthetics (aes)
            y=<VARIABLE2>,   # = échelles
            ...), # grand nombre d'aesthetics existent
        ) +
geom_<*>()
```

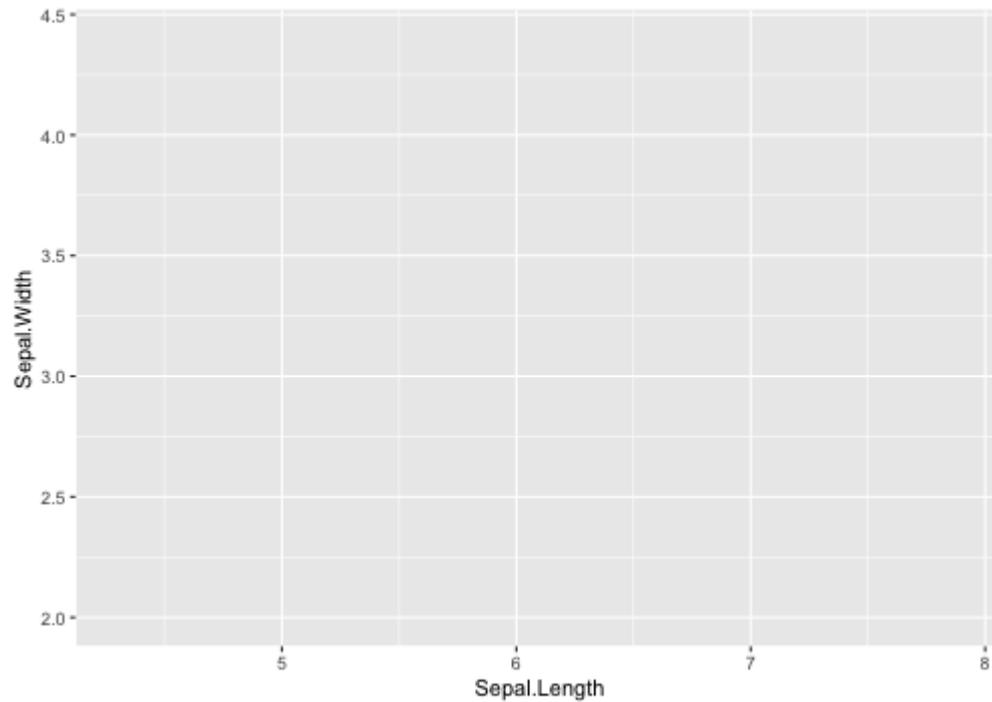
Ressources

Reference: <https://ggplot2.tidyverse.org/reference/>

aesthetics: <https://frama.link/tidyverse-aesthetics>

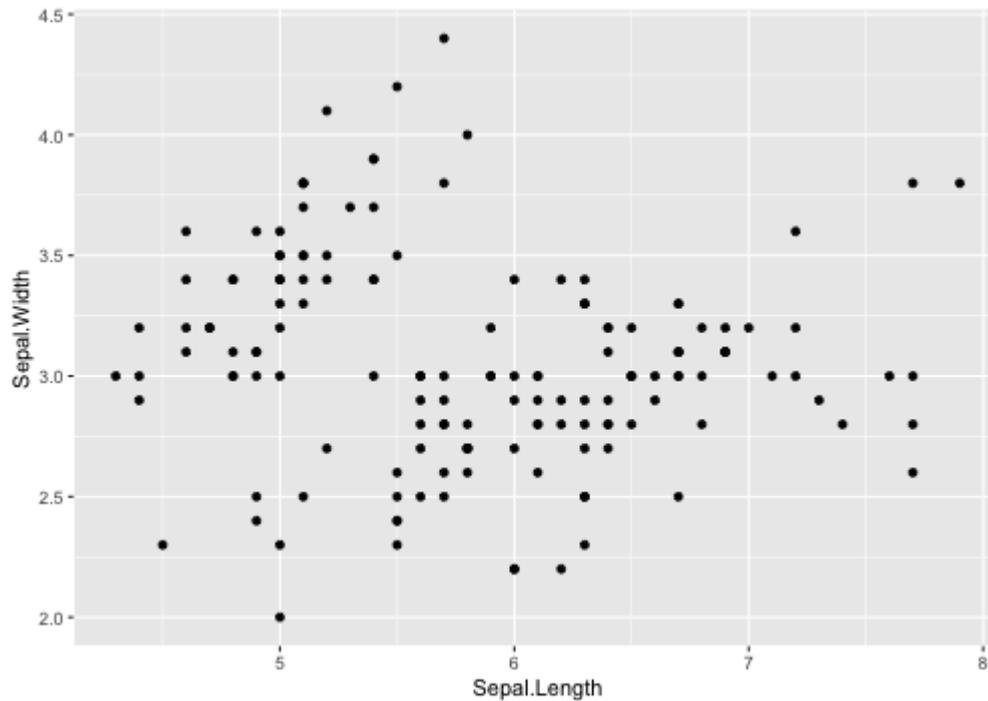
La base

```
data("iris")  
  
ggplot(iris,  
       aes(x=Sepal.Length,  
           y=Sepal.Width))
```



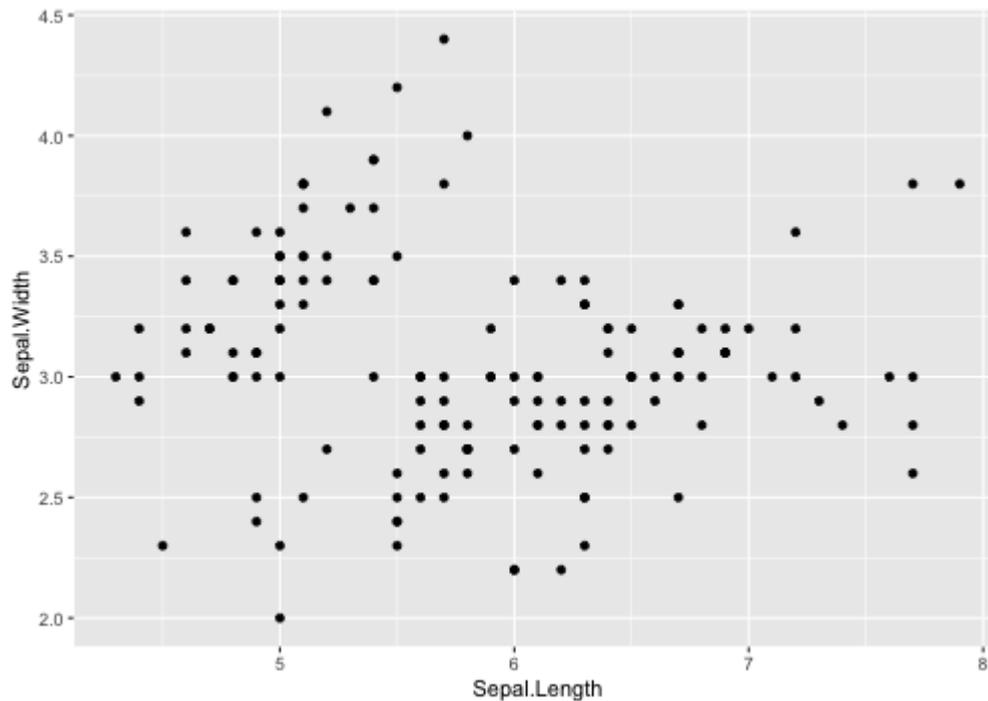
Ajouter une geometrie

```
ggplot(iris,  
  aes(x=Sepal.Length,  
      y=Sepal.Width)) + # le plus ajoute un layer  
  geom_point()
```



Ajouter une géométrie [2]

```
ggplot(iris) +  
  geom_point(aes(x=Sepal.Length,  
                 y=Sepal.Width))
```



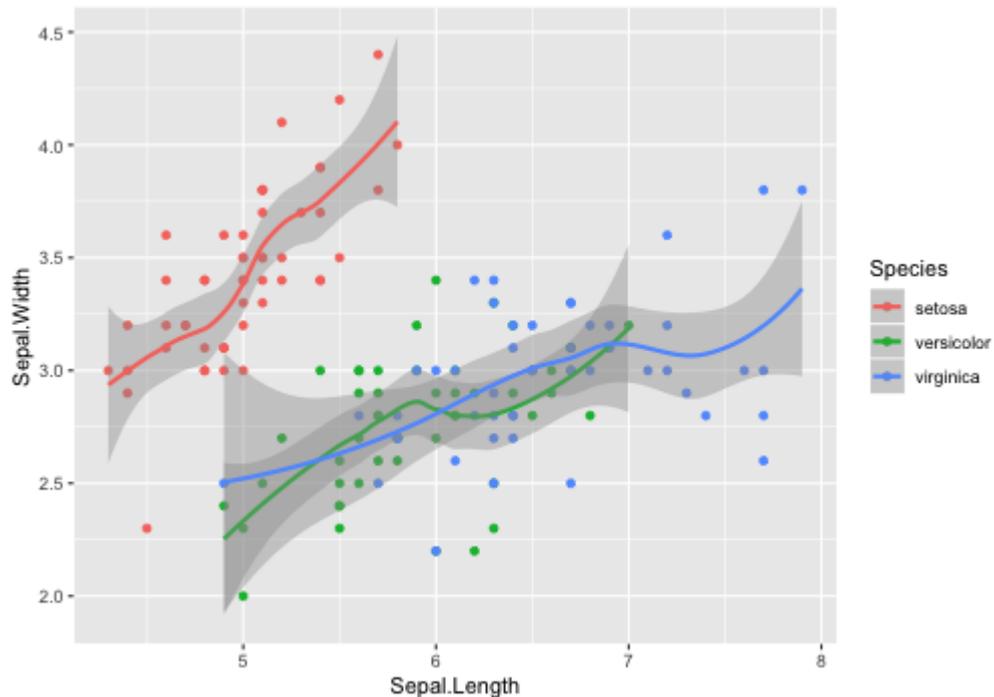
Ajouter un encodage (aesthetics)

```
ggplot(iris,  
       aes(x=Sepal.Length,  
           y=Sepal.Width,  
           color = Species)) +  
geom_point()
```

Ajouter une 2ème géométrie

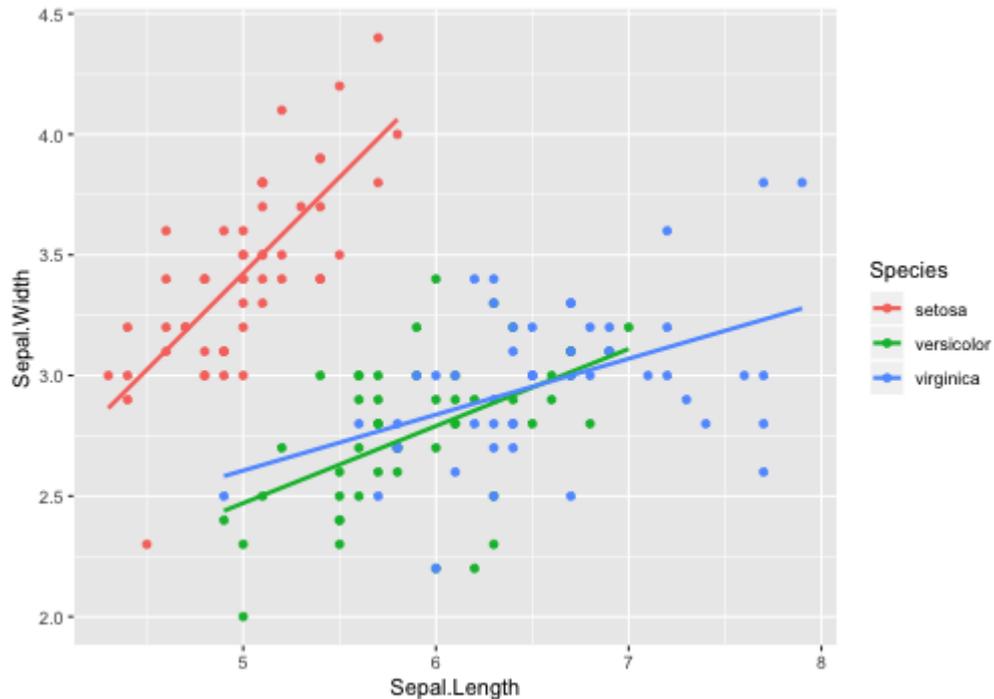
```
ggplot(iris,  
  aes(x=Sepal.Length,  
      y=Sepal.Width,  
      color = Species)) +  
  geom_point() +  
  geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



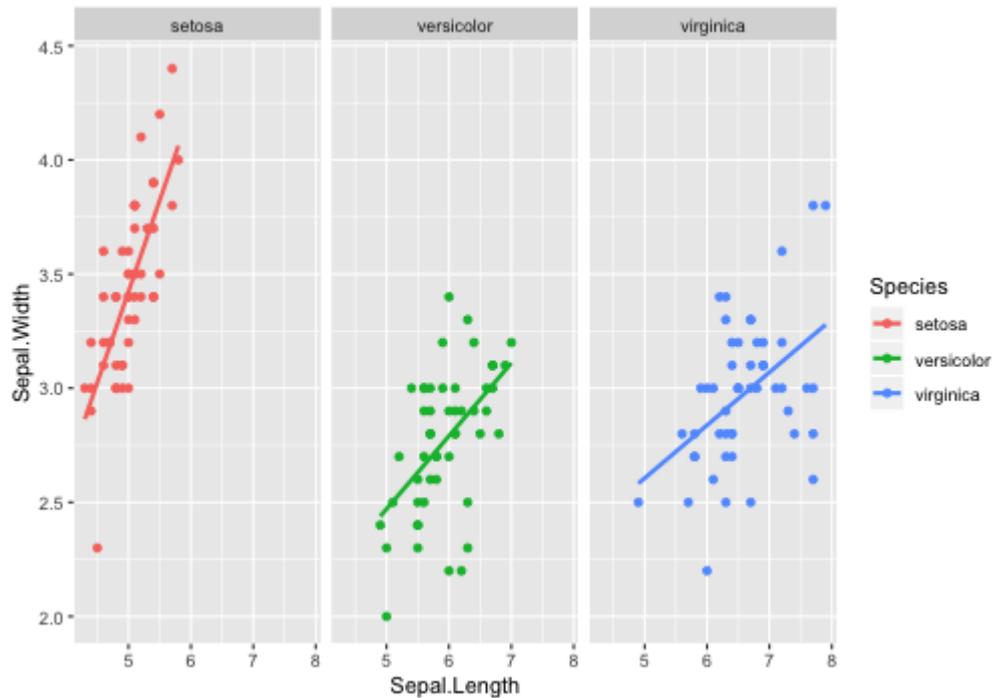
Régler les options de la géométrie

```
ggplot(iris,  
  aes(x=Sepal.Length,  
      y=Sepal.Width,  
      color = Species)) +  
  geom_point() +  
  geom_smooth(method='lm', se=FALSE)
```



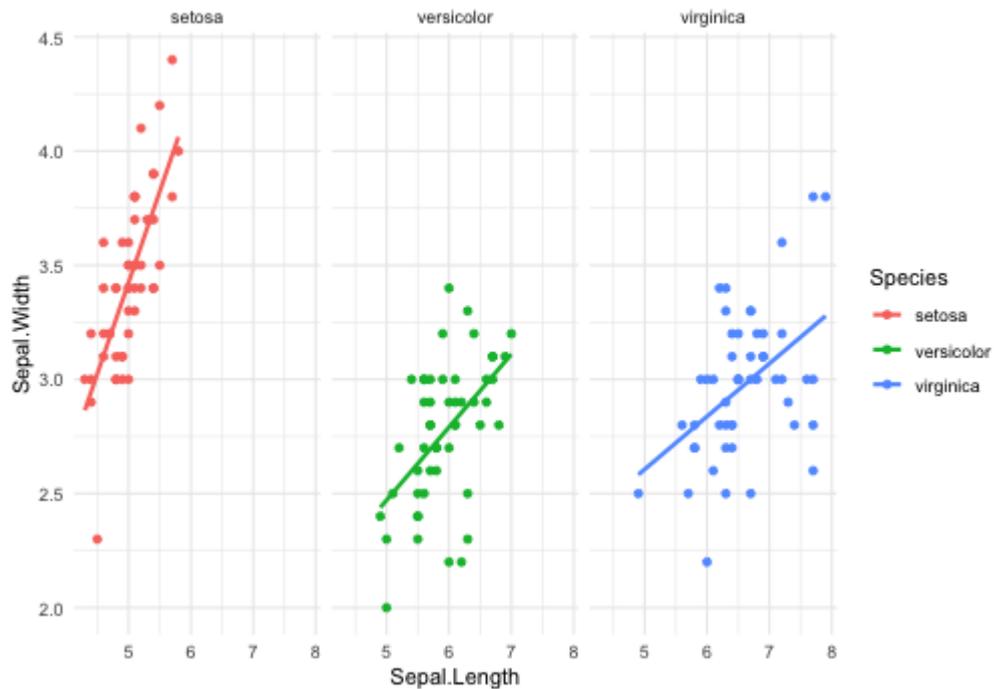
Ajouter une facette

```
ggplot(iris,  
  aes(x=Sepal.Length,  
      y=Sepal.Width,  
      color = Species)) +  
  geom_point() +  
  geom_smooth(method='lm', se=FALSE) +  
  facet_grid(~Species)
```



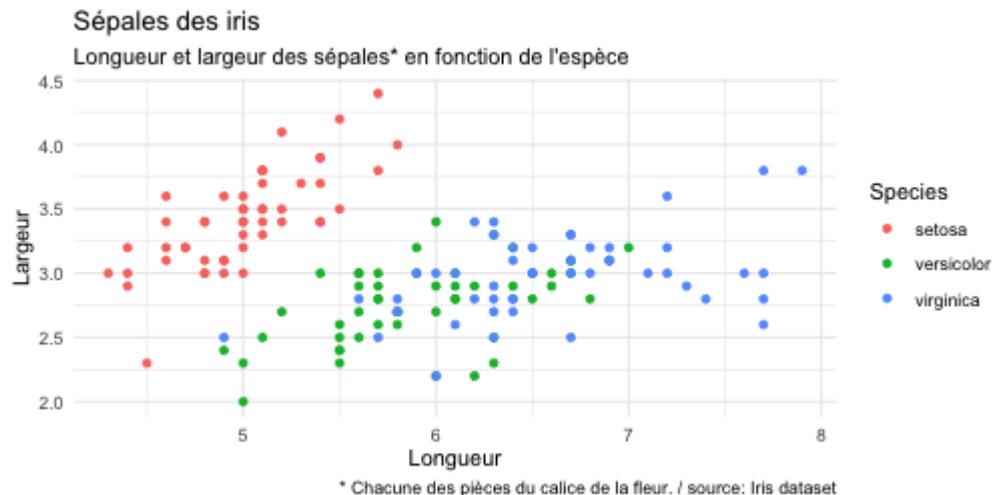
Régler le thème

```
ggplot(iris,  
       aes(x=Sepal.Length,  
           y=Sepal.Width,  
           color = Species)) +  
  geom_point() +  
  geom_smooth(method='lm', se=FALSE) +  
  facet_grid(~Species) +  
  theme_minimal()
```



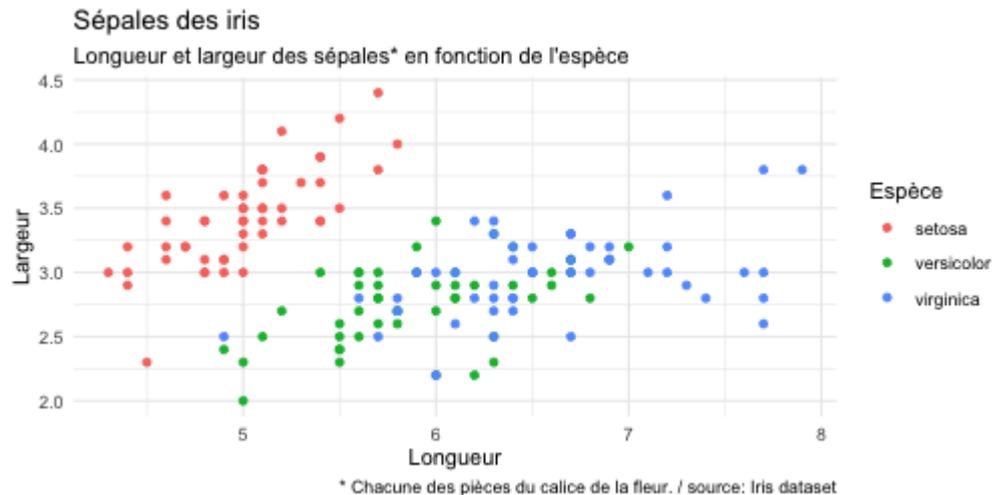
Régler les titres et labels

```
ggplot(iris,  
  aes(x=Sepal.Length,  
      y=Sepal.Width,  
      color = Species)) +  
geom_point()+  
theme_minimal() +  
labs(title = "Sépales des iris",  
  subtitle = "Longueur et largeur des sépales* en fonction de l'espèce",  
  x= "Longueur",  
  y = "Largeur",  
  caption = "* Chacune des pièces du calice de la fleur. / source: Iris dataset"  
)
```



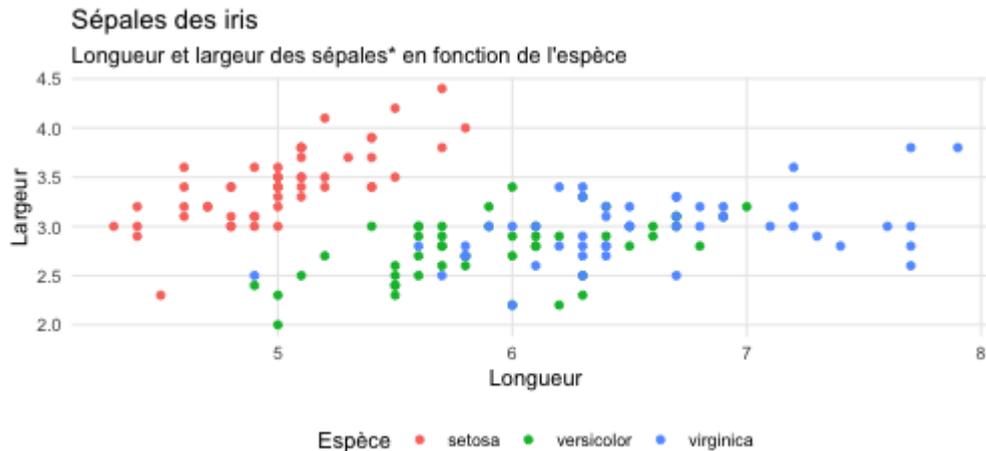
Ajuster la légende

```
ggplot(iris,  
  aes(x=Sepal.Length,  
      y=Sepal.Width,  
      color = Species)) +  
  geom_point()+  
  theme_minimal() +  
  labs(title = "Sépales des iris",  
       subtitle = "Longueur et largeur des sépales* en fonction de l'espèce",  
       x= "Longueur",  
       y = "Largeur",  
       caption = "* Chacune des pièces du calice de la fleur. / source: Iris dataset") +  
  guides(color = guide_legend(title = "Espèce"))
```



Paufiner le thème

```
ggplot(iris,  
  aes(x=Sepal.Length,  
      y=Sepal.Width,  
      color = Species)) +  
geom_point()+  
theme_minimal() +  
labs(title = "Sépales des iris",  
  subtitle = "Longueur et largeur des sépales* en fonction de l'espèce",  
  x= "Longueur",  
  y = "Largeur",  
  caption = "* Chacune des pièces du calice de la fleur. / source: Iris dataset"  
  ) +  
guides(color = guide_legend(title = "Espèce")) +  
theme(panel.grid.minor = element_blank(),  
  legend.position = "bottom")
```



* Chacune des pièces du calice de la fleur. / source: Iris dataset

A l'aide !!!

site du tidyverse: <https://ggplot2.tidyverse.org>

R for datascience: <https://r4ds.had.co.nz/>

stackoverflow: <https://stackoverflow.com>

votre moteur de recherche préféré

TODO

Ouvrir le dataset "mtcars"

représenter le "Gross horsepower" en fonction du nombre de cylindres

utiliser l'encodage multiple sur le nombre de cylindres

ajouter l'information du nombre de carburateurs

Paufiner le plot (axes, titres, thème)

TODO 2

représenter la distribution du nombre de miles per gallon en histogramme

représenter la distribution du nombre de miles per gallon en boxplot

représenter la distribution du nombre de miles per gallon en fonction du nombre de cylindres

ajouter les points par dessus la distribution

paufiner le plot (axes, titres, thème)