Analyse exploratoire des données

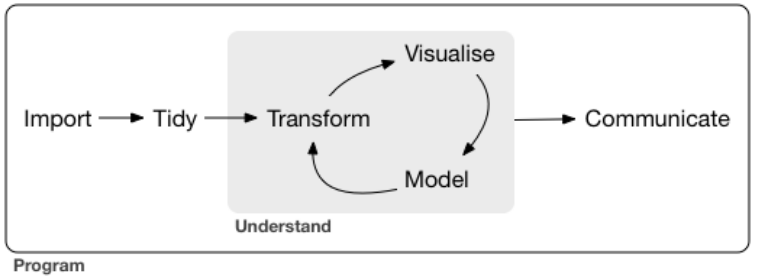
Faire connaissance avec ses données

Benjamin Louis

16/10/2019 (MàJ: 18/10/2019)

Analyse exploratoire

- ⚠️ **Part essentielle de toute analyse !!!**
- ⚠️ **À effectuer en premier !!!**



- **Objectifs** :
  - Vérifier la qualité des données (données aberrantes, manquantes, ...)
  - Vérifier le besoin de transformer les données
  - Observer la variation/covariation des données
  - Générer des questions/hypothèses sur les données

Image : https://r4ds.had.co.nz/introduction.html

Analyse exploratoire

- Pas de règles formelles
- Toute question est bonne
- Trois phases importantes :
  - Généralités sur les jeu de données
  - Variation des variables
  - Covariation des variables

Généralités sur le jeu de données

Objectifs

- Premier contact
- Dimension du jeu de données
- Type des variables
- *tidy data* ?
- Données manquantes
- Données avec beaucoup de 0
- Données avec valeurs infinies

Dans la console

```
msleep
```

```
## # A tibble: 83 x 11
##    name  genus vore  order conservation sleep_total sleep_rem sleep_cycle
##    <chr> <chr> <chr> <chr> <chr>               <dbl>     <dbl>       <dbl>
##  1 Chee~ Acin~ carni Carn~ lc                   12.1      NA        NA
##  2 Owl ~ Aotus omni  Prim~ <NA>                 17         1.8      NA
##  3 Moun~ Aplo~ herbi Rode~ nt                   14.4       2.4      NA
##  4 Grea~ Blar~ omni  Sori~ lc                   14.9       2.3       0.133
##  5 Cow   Bos   herbi Arti~ domesticated          4         0.7       0.667
##  6 Thre~ Brad~ herbi Pilo~ <NA>                 14.4       2.2       0.767
##  7 Nort~ Call~ carni Carn~ vu                    8.7       1.4       0.383
##  8 Vesp~ Calo~ <NA>  Rode~ <NA>                  7        NA        NA
##  9 Dog   Canis carni Carn~ domesticated         10.1       2.9       0.333
## 10 Roe ~ Capr~ herbi Arti~ lc                    3        NA        NA
## # ... with 73 more rows, and 3 more variables: awake <dbl>, brainwt <dbl>,
## #   bodywt <dbl>
```

summary()

```
summary(msleep)
```

```
##      name               genus               vore          
##  Length:83          Length:83          Length:83         
##  Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character  
##                                                          
##                                                          
##                                                          
##                                                          
##     order           conservation        sleep_total      sleep_rem     
##  Length:83          Length:83          Min.   : 1.90   Min.   :0.100  
##  Class :character   Class :character   1st Qu.: 7.85   1st Qu.:0.900  
##  Mode  :character   Mode  :character   Median :10.10   Median :1.500  
##                                        Mean   :10.43   Mean   :1.875  
##                                        3rd Qu.:13.75   3rd Qu.:2.400  
##                                        Max.   :19.90   Max.   :6.600  
##                                                        NA's   :22     
##   sleep_cycle         awake           brainwt             bodywt        
##  Min.   :0.1167   Min.   : 4.10   Min.   :0.00014   Min.   :   0.005  
##  1st Qu.:0.1833   1st Qu.:10.25   1st Qu.:0.00290   1st Qu.:   0.174  
##  Median :0.3333   Median :13.90   Median :0.01240   Median :   1.670  
##  Mean   :0.4396   Mean   :13.57   Mean   :0.28158   Mean   : 166.136  
##  3rd Qu.:0.5792   3rd Qu.:16.15   3rd Qu.:0.12550   3rd Qu.:  41.750  
##  Max.   :1.5000   Max.   :22.10   Max.   :5.71200   Max.   :6654.000  
##  NA's   :51                       NA's   :27
```

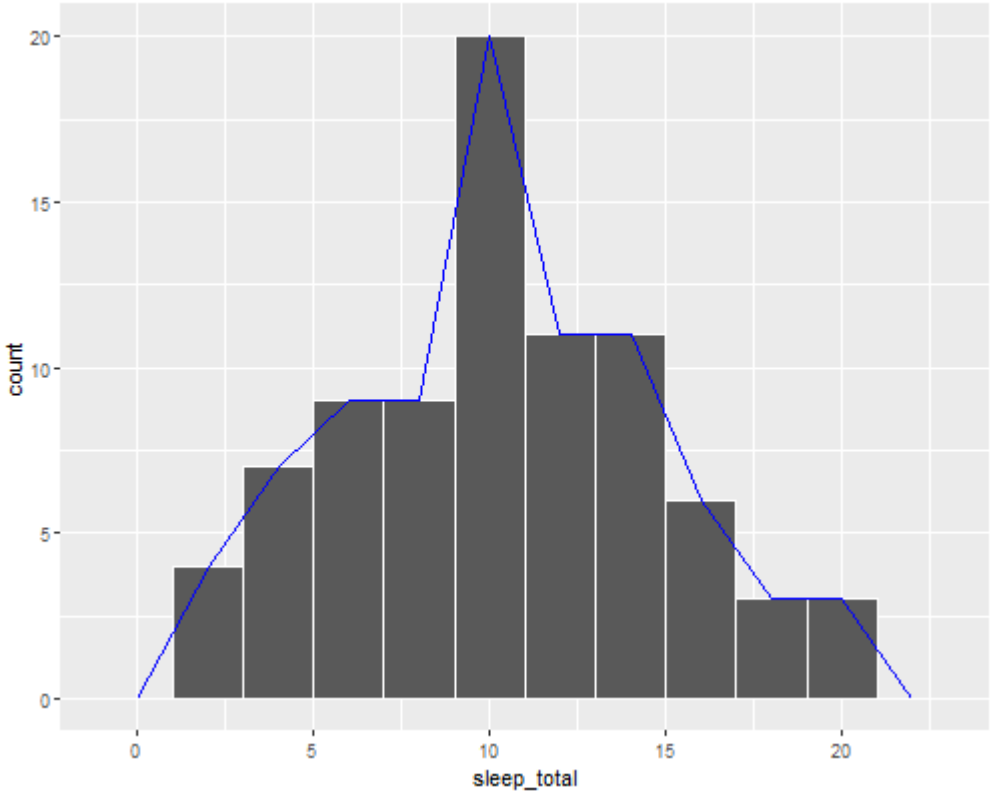
Variations

Objectifs

- Distribution des données
- Quelles sont les valeurs les plus typiques ?
- Quelles sont les données rares ?
- Données aberrantes ou juste rares ?
- Tester la qualité des données

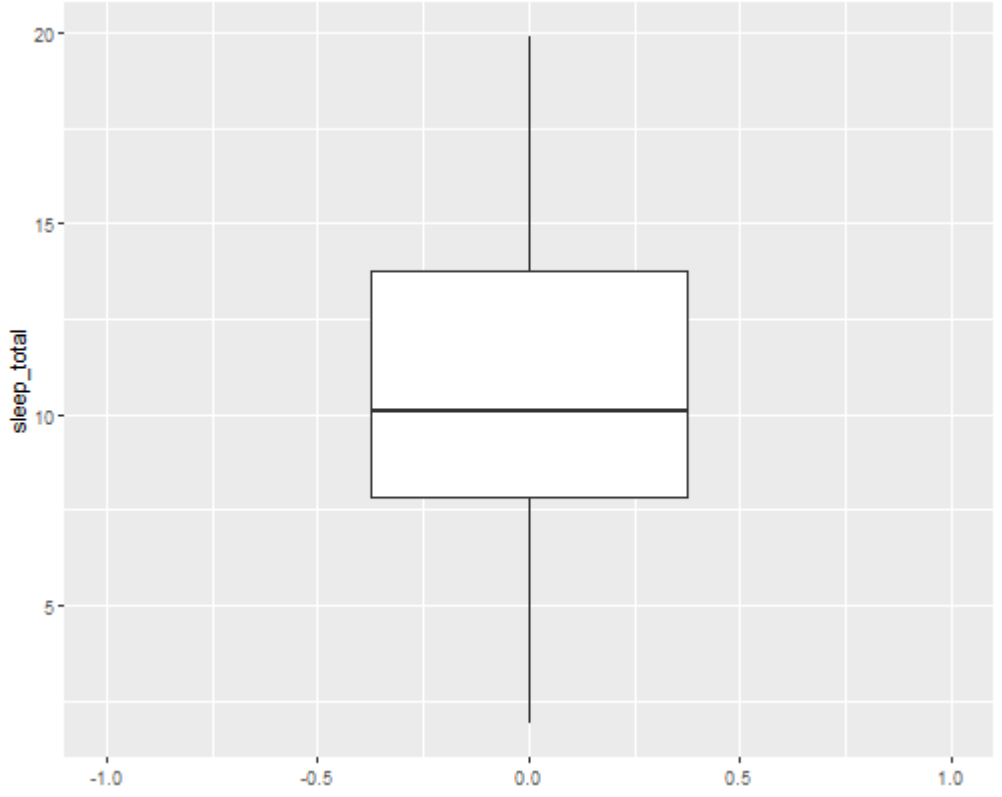
Variables quantitatives : histogramme 👍

```
ggplot(msleep) +
  aes(x = sleep_total) +
  geom_histogram(binwidth = 2, color = "white") +
  geom_freqpoly(binwidth = 2, color = "blue")
```



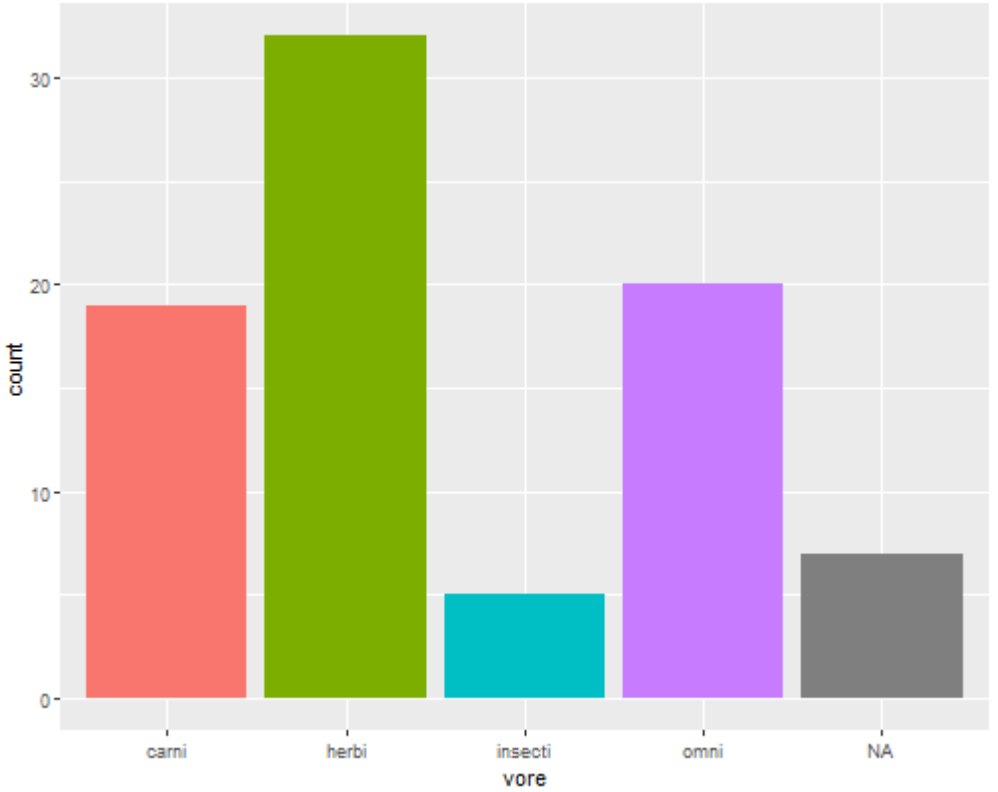
Variables quantitatives : boxplot ⚠️

```
ggplot(msleep) +
  aes(y = sleep_total) +
  geom_boxplot() +
  xlim(-1, 1)
```

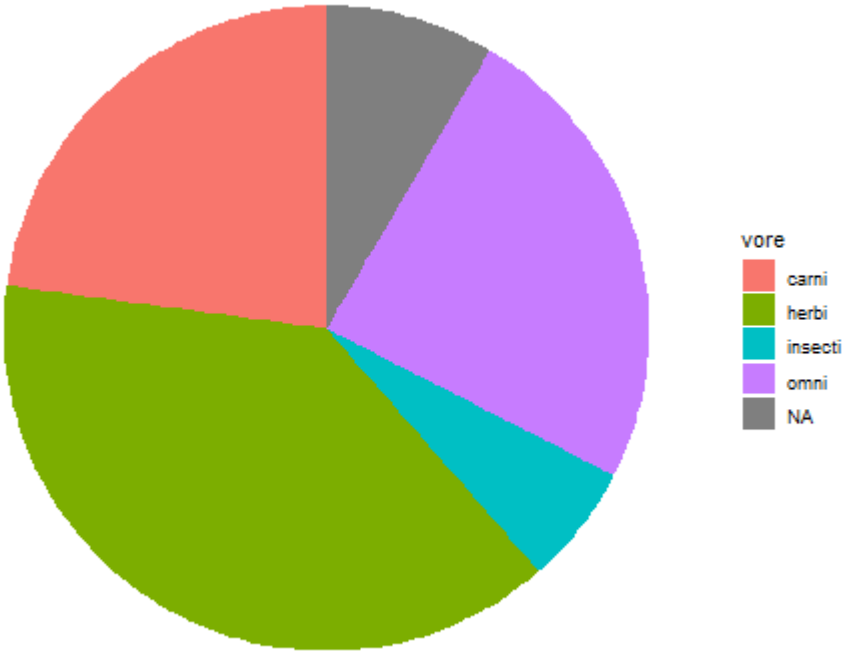


Variables qualitatives : barplot 👍

```
ggplot(msleep) +
  aes(x = vore, fill = vore) +
  geom_bar(show.legend = FALSE)
```



Variables qualitatives : pie plot 👎👎👎



Covariations

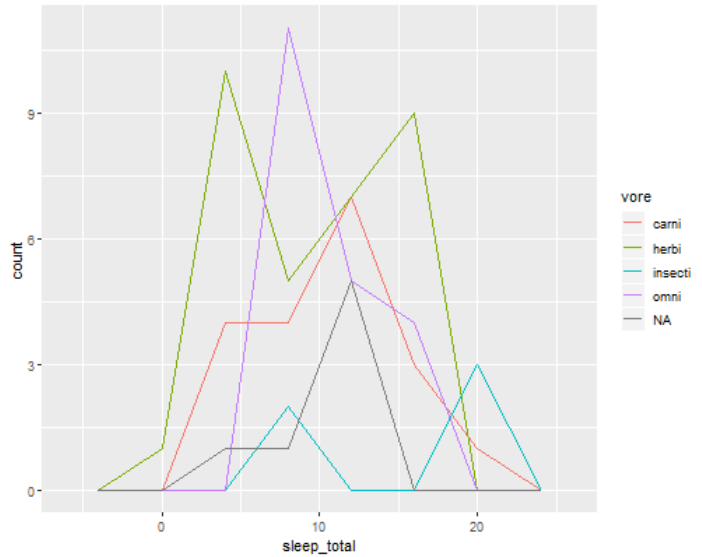
Objectifs

- Existe t-il des relatoins entre les variables ?
- Les relations observées sont-elles attendues ?
- Existe t-il de l'information redondante ?

Quantitative ~ Qualitative : histogrammes/freqpoly

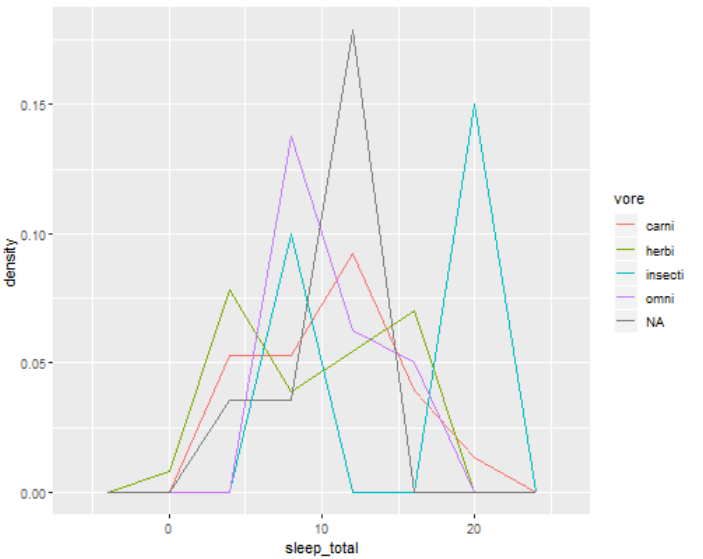
👎 Groupes de taille différente

```
ggplot(msleep) +
  aes(x = sleep_total, color = vore) +
  geom_freqpoly(binwidth = 4)
```



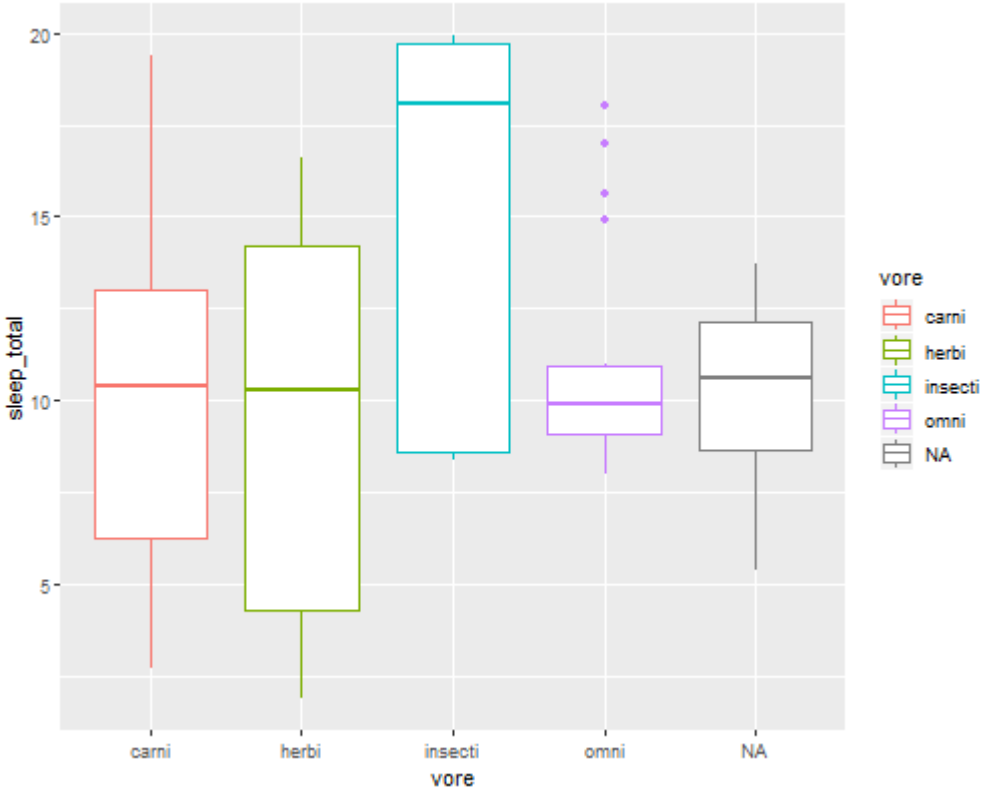
👍 Comparaison densité

```
ggplot(msleep) +
  aes(x = sleep_total, color = vore,
      y = ..density..) +
  geom_freqpoly(binwidth = 4)
```



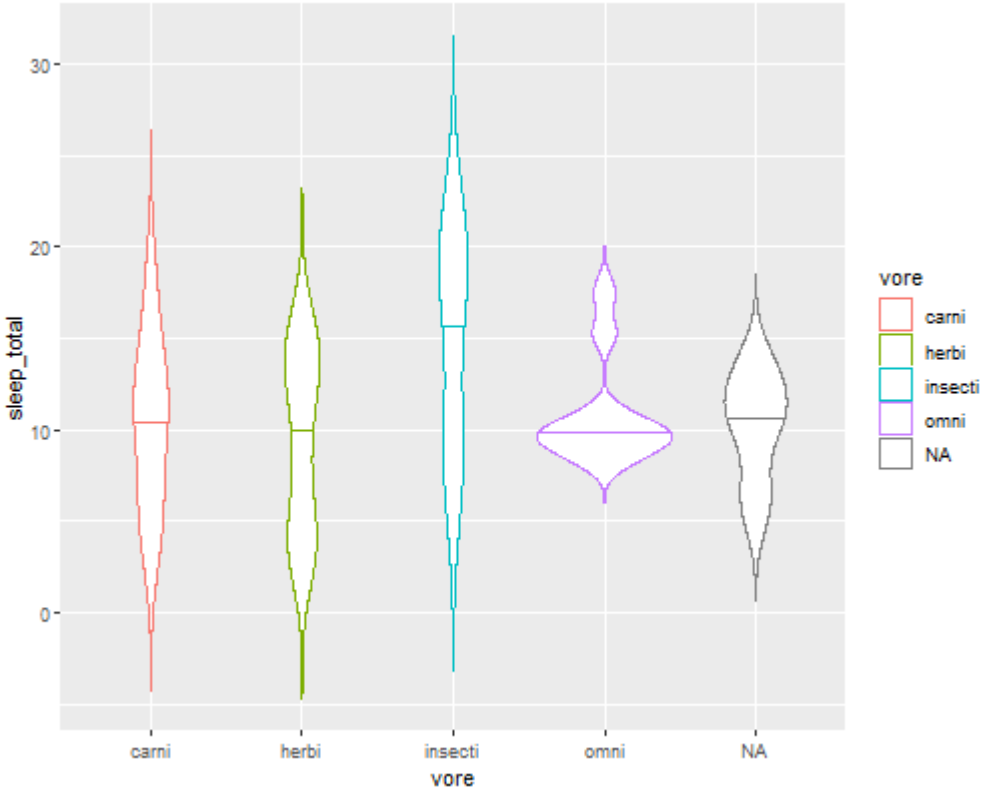
Quantitative ~ Qualitative : boxplot ⚠️

```
ggplot(msleep) +
  aes(x = vore, y = sleep_total,
      color = vore) +
  geom_boxplot()
```



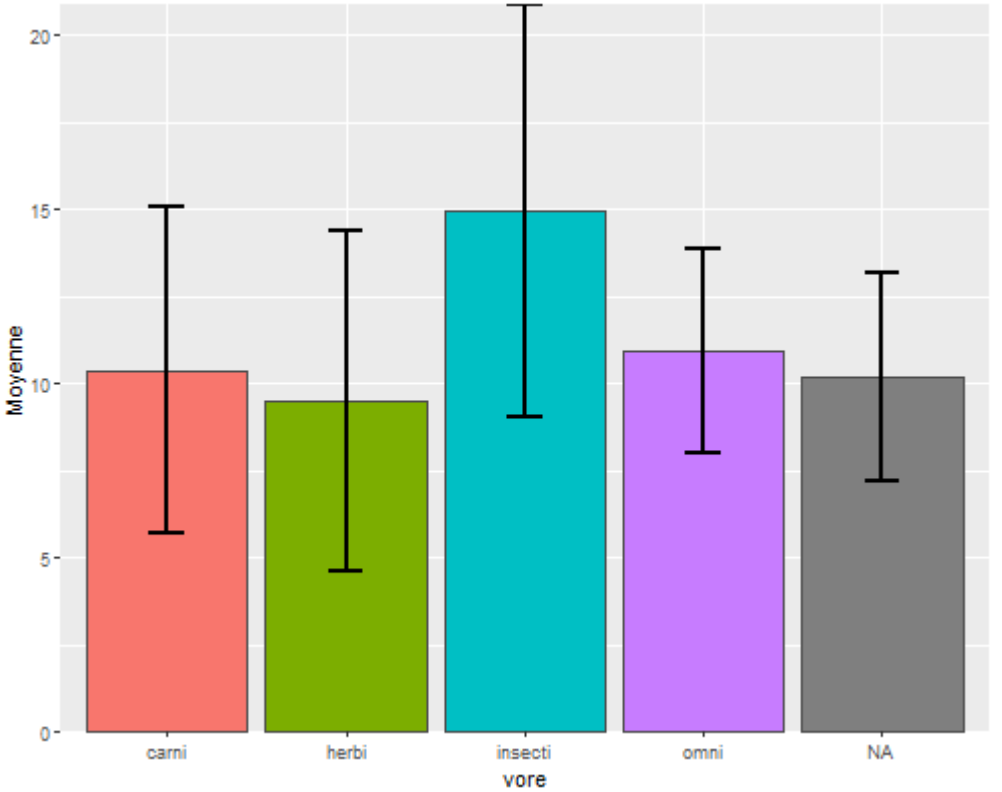
Quantitative ~ Qualitative : violin plot 👍

```
ggplot(msleep) +
  aes(x = vore, y = sleep_total,
      color = vore) +
  geom_violin(draw_quantiles = 0.5, trim = FALSE)
```



Quantitative ~ Qualitative : bar-barplot 👎

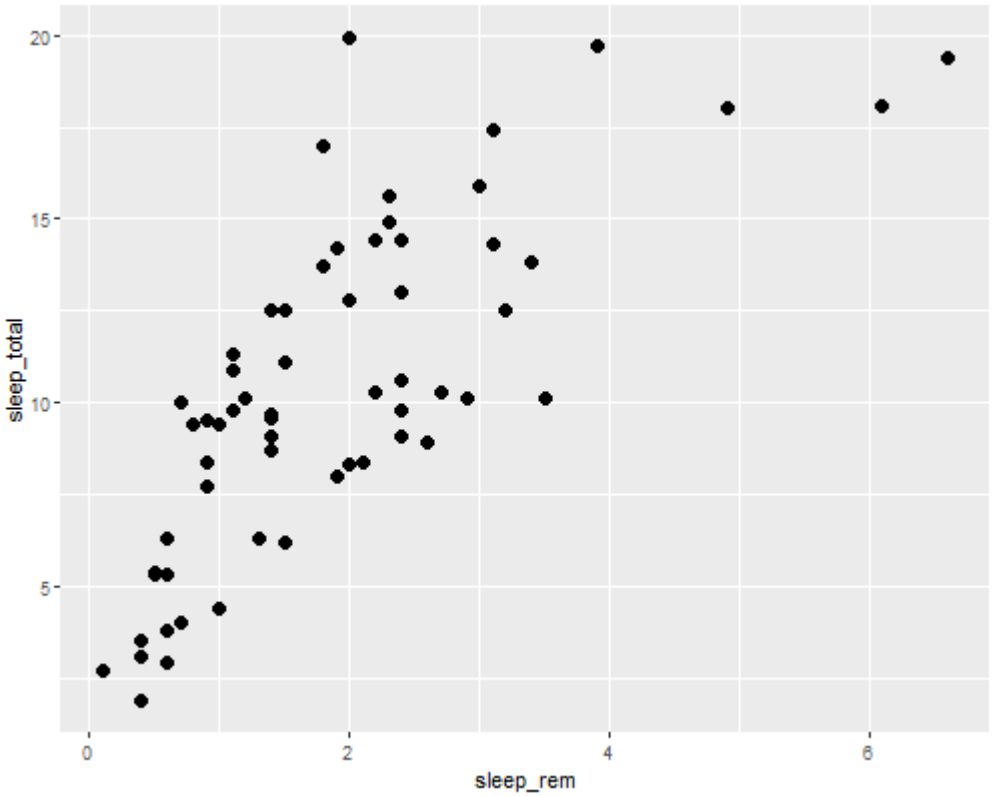
bar-barplot : barplot des valeurs moyennes et écart-type



cf https://thinkr.fr/les-pieges-de-la-representation-de-donnees/

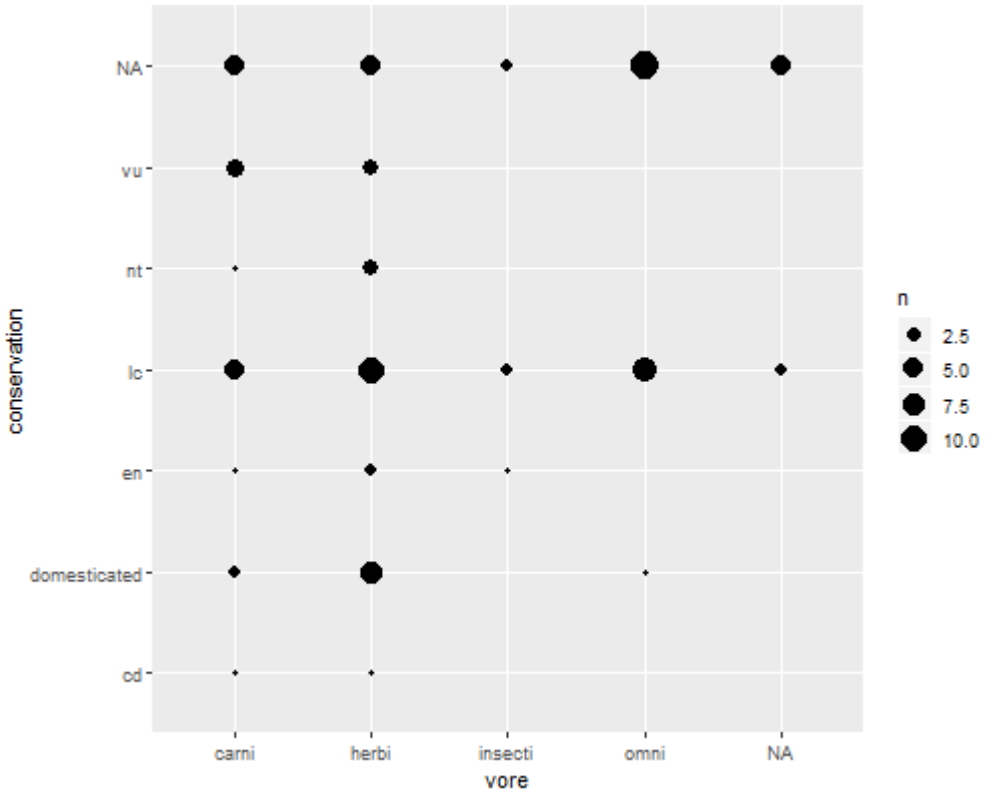
Quantitative ~ Quantitative : scatter plot 👍

```
ggplot(msleep) +
  aes(x = sleep_rem, y = sleep_total) +
  geom_point(size = 3)
```



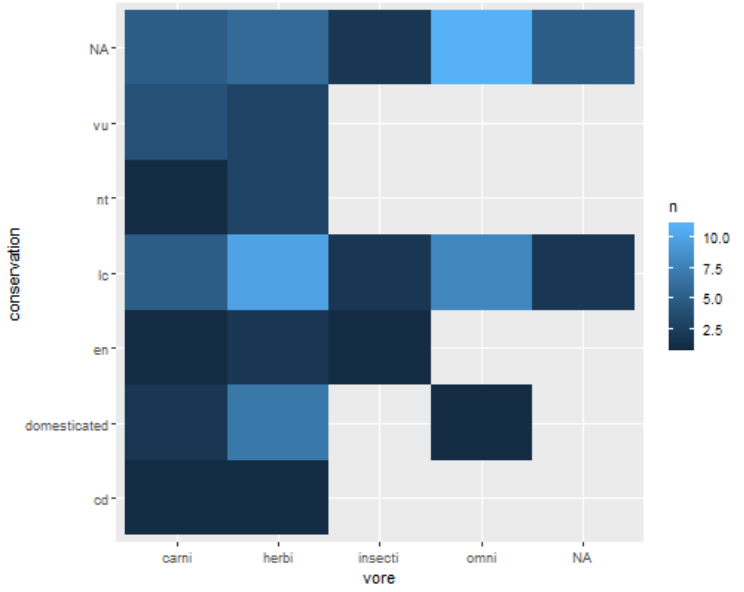
Qualitative ~ Qualitative : geom_count()

```
ggplot(msleep) +
  aes(x = vore, y = conservation) +
  geom_count()
```



Qualitative ~ Qualitative : geom_tile() 👍

```
msleep %>%
  count(vore,conservation) %>%
  ggplot() +
  aes(x = vore, y = conservation) +
  geom_tile(aes(fill = n))
```

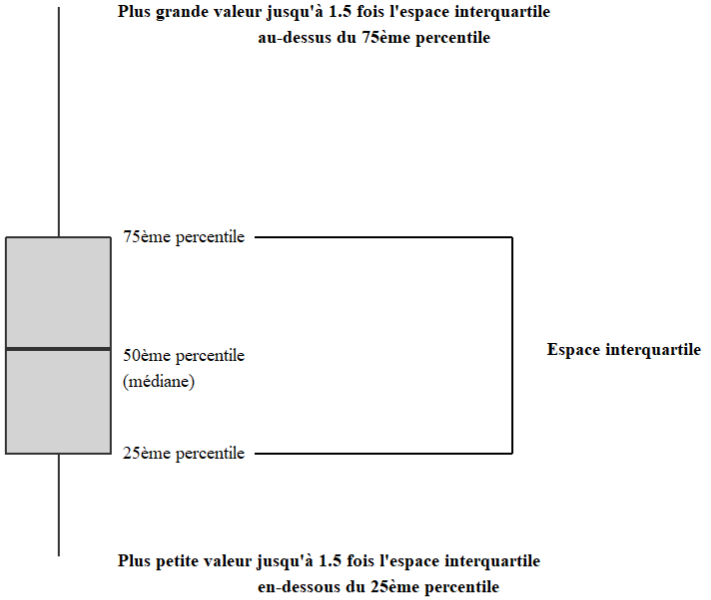


Données aberrantes

Définition

- Outliers
- Valeurs distantes des autres observations
- Contraste
- Analyse exploratoire aide à les détecter
- Des tests statistiques existent
- Leur sort dépend uniquement de l'expertise !!!

boxplot


Plus grande valeur jusqu'à 1.5 fois l'espace interquartile
au-dessus du 75ème percentile
75ème percentile
50ème percentile
(médiane)
25ème percentile
Espace interquartile
Plus petite valeur jusqu'à 1.5 fois l'espace interquartile
en-dessous du 25ème percentile
Valeur considérée comme atypique (outlier)

boxplot

➔ Récupérer les outliers

```
msleep %>%
  filter(sleep_rem %in% boxplot.stats(sleep_rem)$out)
```

```
## # A tibble: 3 x 11
##   name  genus vore  order conservation sleep_total sleep_rem sleep_cycle
##   <chr> <chr> <chr> <chr> <chr>              <dbl>     <dbl>       <dbl>
## 1 Nort~ Dide~ omni  Dide~ lc                  18         4.9       0.333
## 2 Thic~ Lutr~ carni Dide~ lc                  19.4       6.6      NA
## 3 Gian~ Prio~ inse~ Cing~ en                  18.1       6.1      NA
## # ... with 3 more variables: awake <dbl>, brainwt <dbl>, bodywt <dbl>
```

➔ Autres règles

Parfois on trouve des règles plus conservatrices --> $3 \times IQR$ (au lieu de 1,5)

```
msleep %>%
  filter(sleep_rem %in% boxplot.stats(sleep_rem, coef = 3)$out)
```

⚠ Repose sur la loi normale

Test de Grubbs

- Package **outliers**
- Test si une valeur extrême est un outlier

```
library(outliers)
# Test pour valeur max
grubbs.test(msleep$sleep_rem)
```

```
##
##     Grubbs test for one outlier
##
## data:  msleep$sleep_rem
## G = 3.6391, U = 0.7756, p-value = 0.003515
## alternative hypothesis: highest value 6.6 is an outlier
```

```
# Test pour valeur min
grubbs.test(msleep$sleep_rem, opposite = TRUE)
```

```
##
##     Grubbs test for one outlier
##
## data:  msleep$sleep_rem
## G = 1.36750, U = 0.96831, p-value = 1
## alternative hypothesis: lowest value 0.1 is an outlier
```

Packages utiles

- Généralités
  - `funModeling::df_status`
- Variations/Covariations
  - `xray::distributions`
  - `skimr` : https://ropensci.github.io/skimr/
- Transversal
  - `visdat` : http://visdat.njtierney.com/
  - `summarytools` : https://github.com/dcomtois/summarytools
  - `DataExplorer` : https://boxuancui.github.io/DataExplorer/

Exercice

Testez-vous sur des jeux de données de votre choix : `iris`, `mtcars`, `starwars`, …

References

- https://www.data-to-viz.com/
- https://statistique-et-logiciel-r.com/comment-detecter-les-outliers-avec-r/