TP3

Sarah THEOULLE

2024-04-10

```
library(readr)
library(here)
```

```
## here() starts at /home/sarah/Documents/DO3/S6/DataViz/do3-dataviz
```

```
library(tidyr)
library(dplyr)
```

```
##
## Attachement du package : 'dplyr'

## Les objets suivants sont masqués depuis 'package:stats':
##
##     filter, lag

## Les objets suivants sont masqués depuis 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

On cherche à répondre à la question : "Les couples pacsés ont-ils plus d'enfants de moins de 25 ans que les couples mariés en France en 2017 ?"

- Les individus dans ce contexte sont les couples résidant en France en 2017.
- La population étudiée est l'ensemble des couples résidant en France en 2017. Cela comprend à la fois les couples pacsés et les couples mariés.
- La variable mesurée est le nombre d'enfants de moins de 25 ans dans chaque couple. Cette variable est quantitative, continue et positive, car elle mesure un nombre entier non négatif. Les modalités de cette variable sont les différentes valeurs possibles qu'elle peut prendre, c'est-à-dire le nombre d'enfants de moins de 25 ans dans un couple donné. Les modalités peuvent varier de 0 (pas d'enfant de moins de 25 ans) à un nombre maximal n.

```
famille <- read_delim("../data/rp2017_td_fam2.csv",
    delim = ";", escape_double = FALSE, col_types = cols(...8 = col_skip()),
    trim_ws = TRUE, skip = 6)
```

```
## New names:
## * `` -> `...1`
## * `` -> `...8`
```

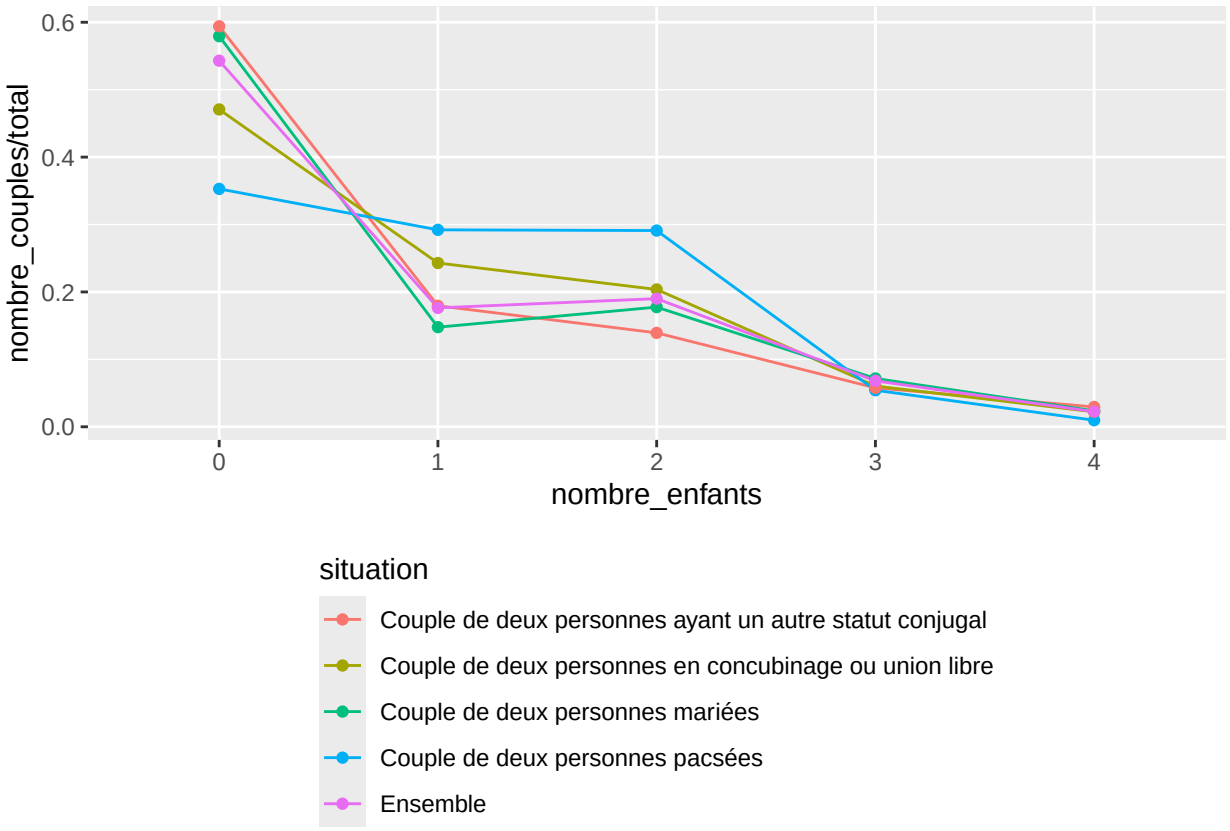
```r
colnames(famille) <- c("situation", "0","1", "2", "3", "4", "total")
famille
```

```
## # A tibble: 5 x 7
##   situation                                ‘0‘    ‘1‘    ‘2‘    ‘3‘    ‘4‘  total
##   <chr>                                  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 Couple de deux personnes mariées     6.45e6 1.64e6 1.98e6 7.98e5 263408 1.11e7
## 2 Couple de deux personnes pacsées     4.07e5 3.37e5 3.36e5 6.26e4  11225 1.15e6
## 3 Couple de deux personnes en concubi~ 1.30e6 6.73e5 5.64e5 1.68e5  61358 2.77e6
## 4 Couple de deux personnes ayant un a~ 1.77e5 5.36e4 4.16e4 1.72e4   8778 2.99e5
## 5 Ensemble                             8.34e6 2.71e6 2.92e6 1.05e6 344769 1.54e7
```

```r
famille_long <- pivot_longer(famille,
                             cols = c("0", "1", "2", "3", "4"),
                             names_to = "nombre_enfants",
                             values_to = "nombre_couples")

famille_long
```

```
## # A tibble: 25 x 4
##    situation                           total nombre_enfants nombre_couples
##    <chr>                               <dbl> <chr>                   <dbl>
##  1 Couple de deux personnes mariées 11129960 0                     6448133
##  2 Couple de deux personnes mariées 11129960 1                     1644613
##  3 Couple de deux personnes mariées 11129960 2                     1975639
##  4 Couple de deux personnes mariées 11129960 3                      798166
##  5 Couple de deux personnes mariées 11129960 4                      263408
##  6 Couple de deux personnes pacsées  1153862 0                      407144
##  7 Couple de deux personnes pacsées  1153862 1                      337083
##  8 Couple de deux personnes pacsées  1153862 2                      335833
##  9 Couple de deux personnes pacsées  1153862 3                       62577
## 10 Couple de deux personnes pacsées  1153862 4                       11225
## # i 15 more rows
```

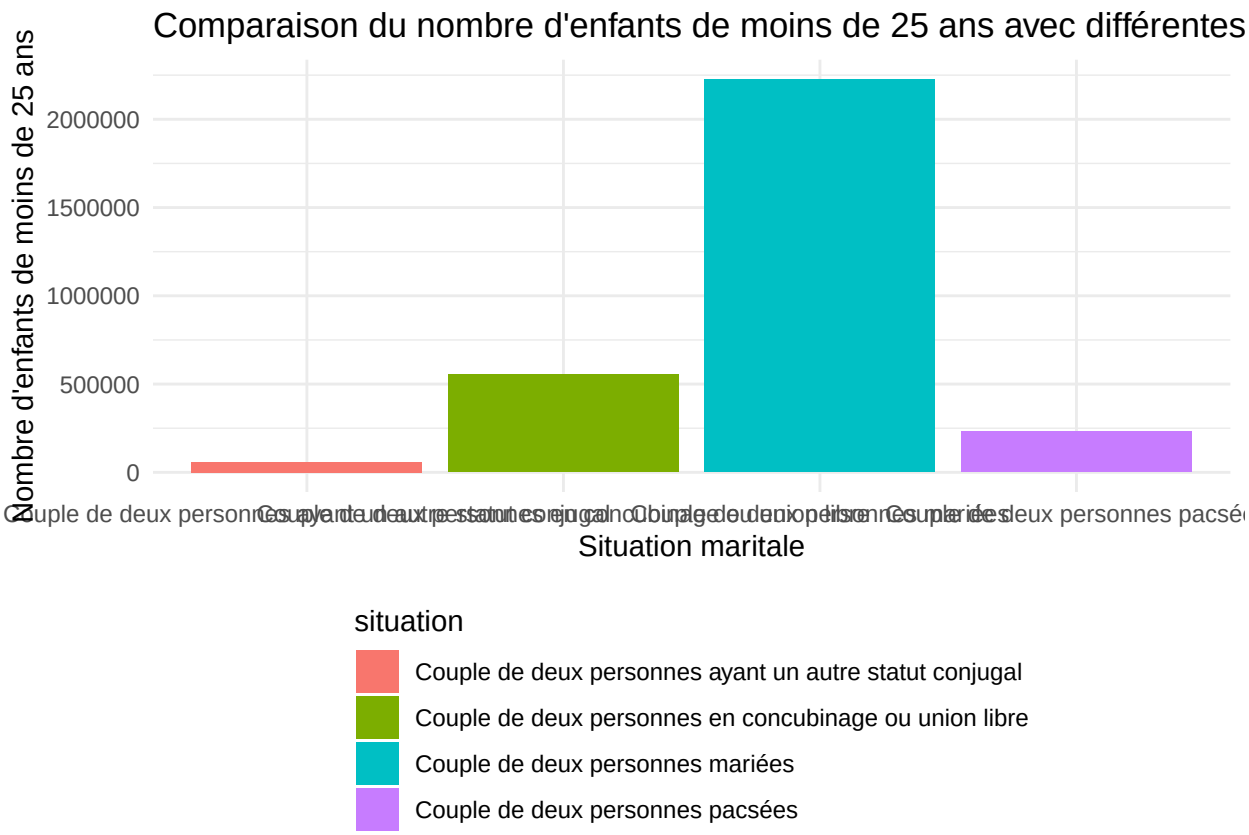
```r
ggplot(famille_long, aes(x = nombre_enfants, y = nombre_couples/total, color=situation, group=situation
  geom_line() +
  geom_point() +
  theme(legend.position = "bottom",  # Placer la légende en dessous du graphique
        legend.direction = "vertical")
```




```
moyenne_enfants <- famille_long %>%
  group_by(situation) %>%
  summarise(moyenne_enfants = mean(nombre_couples))

moyenne_enfants <- moyenne_enfants %>%
  filter(situation != "Ensemble")

print(moyenne_enfants)
```

```
## # A tibble: 4 x 2
##   situation                                                moyenne_enfants
##   <chr>                                                              <dbl>
## 1 Couple de deux personnes ayant un autre statut conjugal          59702.
## 2 Couple de deux personnes en concubinage ou union libre          554210
## 3 Couple de deux personnes mariées                               2225992.
## 4 Couple de deux personnes pacsées                                230772.
```

```
ggplot(moyenne_enfants, aes(x = situation, y = moyenne_enfants, fill = situation)) +
  geom_bar(stat = "identity") +
  labs(title = "Comparaison du nombre d'enfants de moins de 25 ans avec différentes relations de couple
       x = "Situation maritale",
       y = "Nombre d'enfants de moins de 25 ans") +
  theme_minimal() +
  theme(legend.position = "bottom",  # Placer la légende en dessous du graphique
        legend.direction = "vertical")
```



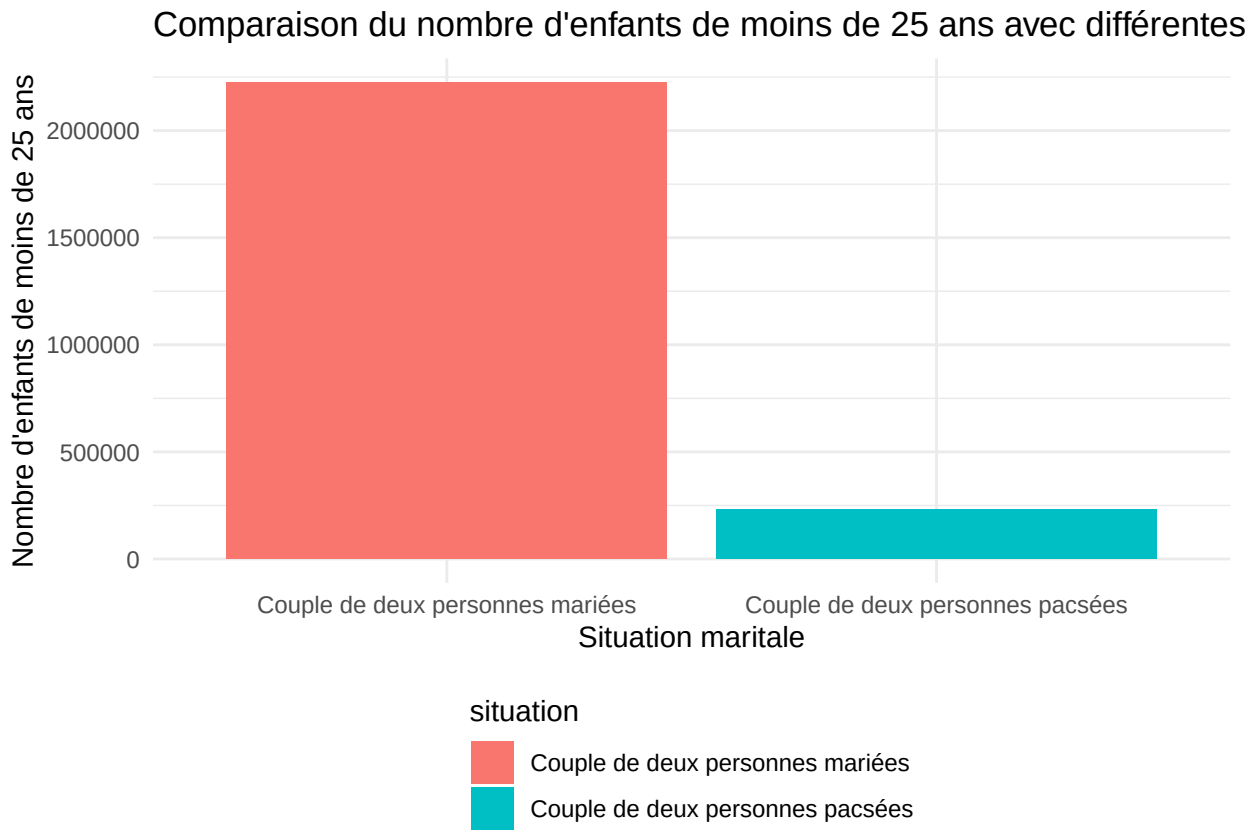
On peut filtrer les données pour ne garder que les deux colonnes qui ont la réponse à notre question

```
moyenne_enfants <- famille_long %>%
  group_by(situation) %>%
  summarise(moyenne_enfants = mean(nombre_couples))

moyenne_enfants <- moyenne_enfants %>%
  filter(situation != "Ensemble" &
           situation != "Couple de deux personnes ayant un autre statut conjugal" &
           situation != "Couple de deux personnes en concubinage ou union libre")

print(moyenne_enfants)
```

```
## # A tibble: 2 x 2
##   situation                          moyenne_enfants
##   <chr>                                        <dbl>
## 1 Couple de deux personnes mariées           2225992.
## 2 Couple de deux personnes pacsées            230772.
```

```
ggplot(moyenne_enfants, aes(x = situation, y = moyenne_enfants, fill = situation)) +
  geom_bar(stat = "identity") +
  labs(title = "Comparaison du nombre d'enfants de moins de 25 ans avec différentes relations de couple
       x = "Situation maritale",
       y = "Nombre d'enfants de moins de 25 ans") +
  theme_minimal() +
```

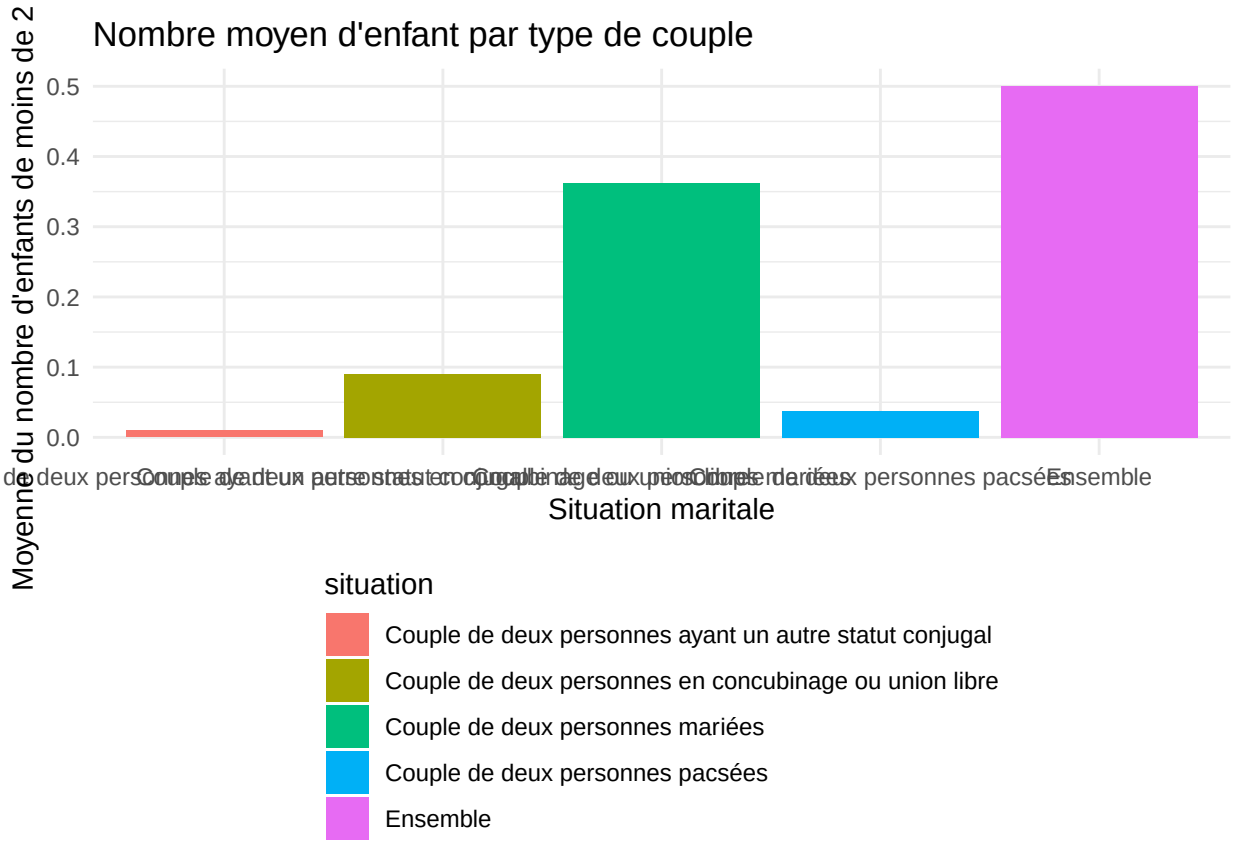
```
  theme(legend.position = "bottom",  # Placer la légende en dessous du graphique
        legend.direction = "vertical")
```



Poser une autre question sur le dataset

On peut par exemple afficher le nombre moyen d'enfants par type de couple. Pour cela on reprend nos données de départ et on effectue la moyenne du nombre d'enfant pour chaque type de couple.

```
moyenne_enfants_par_type <- famille %>%
  select(situation, total)

moyenne_enfants_par_type <- moyenne_enfants_par_type %>%
  mutate(moyenne_enfants = total / sum(total))

ggplot(moyenne_enfants_par_type, aes(x = situation, y = moyenne_enfants, fill = situation)) +
  geom_bar(stat = "identity") +
  labs(title = "Nombre moyen d'enfant par type de couple",
       x = "Situation maritale",
       y = "Moyenne du nombre d'enfants de moins de 25 ans") +
  theme_minimal() +
  theme(legend.position = "bottom",  # Placer la légende en dessous du graphique
        legend.direction = "vertical")
```



Expéditions sur l'Everest

```
members <- readr::read_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2
```

```
## 'curl' package not installed, falling back to using 'url()'
## Rows: 76519 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (10): expedition_id, member_id, peak_id, peak_name, season, sex, citizen...
## dbl  (5): year, age, highpoint_metres, death_height_metres, injury_height_me...
## lgl  (6): hired, success, solo, oxygen_used, died, injured
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
members
```

```
## # A tibble: 76,519 x 21
##    expedition_id member_id    peak_id peak_name   year season sex     age
##    <chr>         <chr>        <chr>   <chr>      <dbl> <chr>  <chr> <dbl>
##  1 AMAD78301     AMAD78301-01 AMAD    Ama Dablam  1978 Autumn M        40
##  2 AMAD78301     AMAD78301-02 AMAD    Ama Dablam  1978 Autumn M        41
```

```
##  3 AMAD78301     AMAD78301-03 AMAD    Ama Dablam  1978 Autumn M        27
##  4 AMAD78301     AMAD78301-04 AMAD    Ama Dablam  1978 Autumn M        40
##  5 AMAD78301     AMAD78301-05 AMAD    Ama Dablam  1978 Autumn M        34
##  6 AMAD78301     AMAD78301-06 AMAD    Ama Dablam  1978 Autumn M        25
##  7 AMAD78301     AMAD78301-07 AMAD    Ama Dablam  1978 Autumn M        41
##  8 AMAD78301     AMAD78301-08 AMAD    Ama Dablam  1978 Autumn M        29
##  9 AMAD79101     AMAD79101-03 AMAD    Ama Dablam  1979 Spring M        35
## 10 AMAD79101     AMAD79101-04 AMAD    Ama Dablam  1979 Spring M        37
## # i 76,509 more rows
## # i 13 more variables: citizenship <chr>, expedition_role <chr>, hired <lgl>,
## #   highpoint_metres <dbl>, success <lgl>, solo <lgl>, oxygen_used <lgl>,
## #   died <lgl>, death_cause <chr>, death_height_metres <dbl>, injured <lgl>,
## #   injury_type <chr>, injury_height_metres <dbl>
```

Données

Le jeu de données "members" contient des informations sur les membres participant à des expéditions en montagne. Il comprend des détails tels que l'identifiant de l'expédition, l'identifiant du membre, le nom du sommet, l'année et la saison de l'expédition, le sexe et l'âge des membres, la citoyenneté, le rôle dans l'expédition, les informations sur l'embauche, la hauteur du point culminant atteint, le succès de l'expédition, les détails sur les ascensions en solitaire, l'utilisation d'oxygène, les décès éventuels et les causes de décès.

Age des membres d'une expédition réussie

Expérience statistique :

```
Individu : Les membres d'une expédition vers le Mont Everest.
Population : Tous les membres ayant participé à des expéditions vers le Mont Everest.
Échantillon : Les membres d'une expédition réussie vers le Mont Everest.
Variable mesurée : L'âge des membres.
```

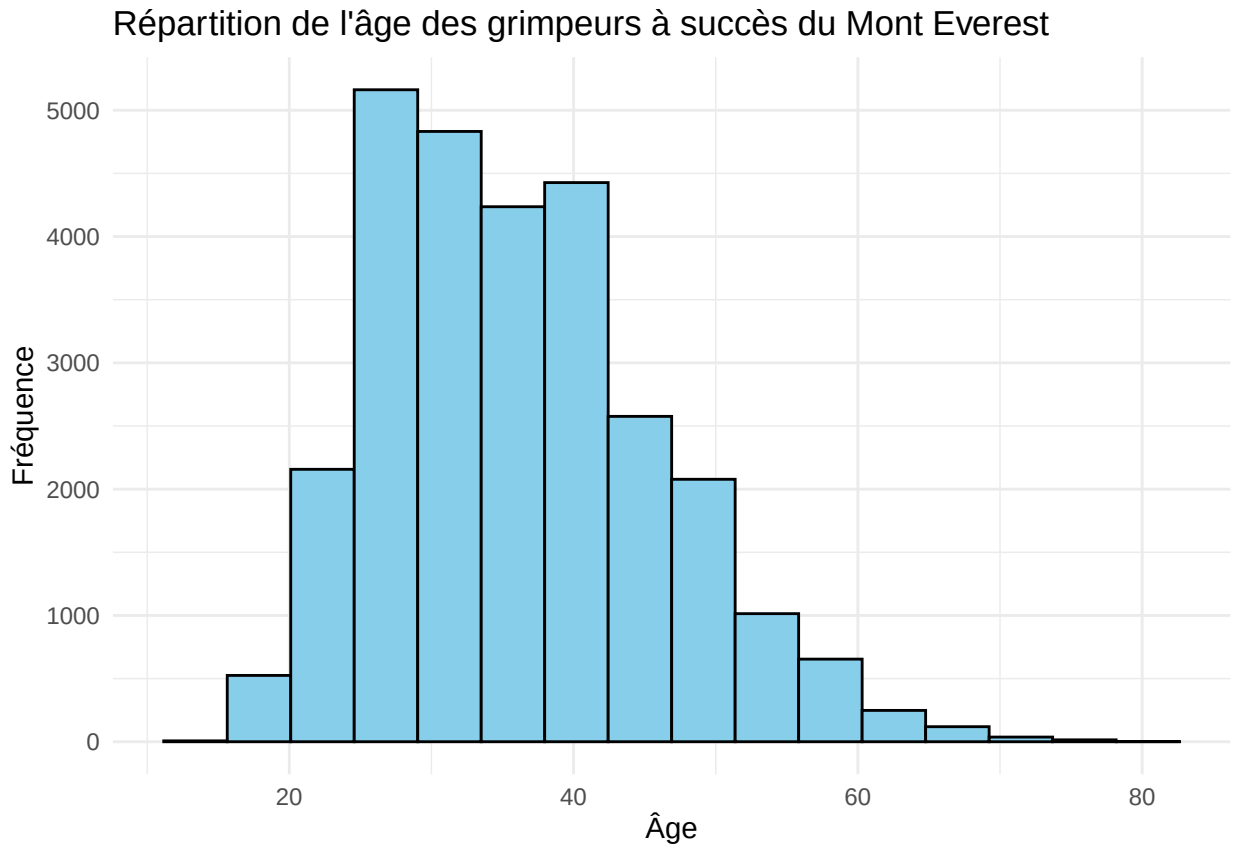
Pour répondre à la question, nous devons sélectionner les lignes du tableau correspondant aux expéditions réussies vers le Mont Everest et dont l'âge des membres n'est pas manquant.

```
  members_not_null <- members %>%
    filter(success == TRUE & !is.null(age))
members_not_null
```

```
## # A tibble: 29,199 x 21
##    expedition_id member_id    peak_id peak_name   year season sex     age
##    <chr>         <chr>        <chr>   <chr>      <dbl> <chr>  <chr> <dbl>
##  1 AMAD79101     AMAD79101-04 AMAD    Ama Dablam  1979 Spring M        37
##  2 AMAD79101     AMAD79101-05 AMAD    Ama Dablam  1979 Spring M        23
##  3 AMAD79101     AMAD79101-02 AMAD    Ama Dablam  1979 Spring M        42
##  4 AMAD79101     AMAD79101-10 AMAD    Ama Dablam  1979 Spring M        30
##  5 AMAD79101     AMAD79101-11 AMAD    Ama Dablam  1979 Spring M        28
##  6 AMAD79101     AMAD79101-12 AMAD    Ama Dablam  1979 Spring M        35
##  7 AMAD79101     AMAD79101-13 AMAD    Ama Dablam  1979 Spring M        33
##  8 AMAD79101     AMAD79101-15 AMAD    Ama Dablam  1979 Spring M        29
##  9 AMAD79101     AMAD79101-16 AMAD    Ama Dablam  1979 Spring M        26
## 10 AMAD79101     AMAD79101-18 AMAD    Ama Dablam  1979 Spring M        23
```

```
## # i 29,189 more rows
## # i 13 more variables: citizenship <chr>, expedition_role <chr>, hired <lgl>,
## #   highpoint_metres <dbl>, success <lgl>, solo <lgl>, oxygen_used <lgl>,
## #   died <lgl>, death_cause <chr>, death_height_metres <dbl>, injured <lgl>,
## #   injury_type <chr>, injury_height_metres <dbl>
```

```r
members_filtered <- members_not_null[!is.na(members_not_null$age), ]

nb_classes <- 15
plage_donnees <- range(members_filtered$age, na.rm = TRUE)  # Ignorer les valeurs non finies lors du ca
largeur_classe <- diff(plage_donnees) / nb_classes

histogram <- ggplot(members_filtered, aes(x = age)) +
  geom_histogram(binwidth = largeur_classe, fill = "skyblue", color = "black") +
  labs(title = "Répartition de l'âge des grimpeurs à succès du Mont Everest",
       x = "Âge",
       y = "Fréquence") +
  theme_minimal()

histogram
```



Dans ce cas particulier, nous avons choisi 15 classes pour plusieurs raisons :

1. **Taille de l'échantillon :** Avec un échantillon suffisamment grand, diviser les données en 15 classes permet de mieux représenter la distribution de l'âge des grimpeurs tout en évitant d'avoir des classes trop larges ou trop étroites.

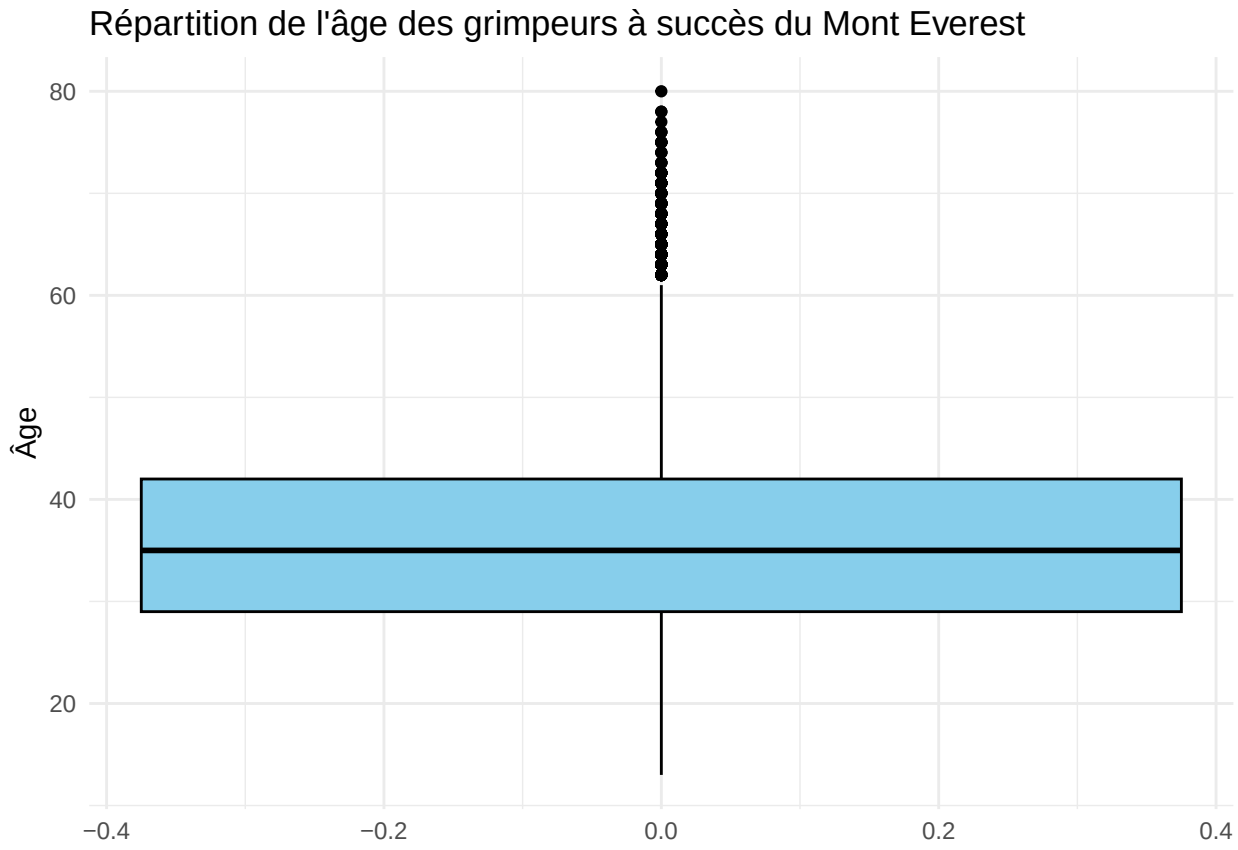
2. **Précision :** En ayant un nombre modéré de classes, nous pouvons encore observer les tendances générales de la distribution de l'âge sans trop de détail, ce qui rend l'interprétation de l'histogramme plus facile.

3. **Visibilité :** Un nombre trop élevé de classes peut entraîner un histogramme surchargé, rendant difficile l'interprétation visuelle. Avec 15 classes, nous obtenons un bon équilibre entre détails et lisibilité.

4. **Facilité d'interprétation :** Avec un nombre raisonnable de classes, il est plus facile d'interpréter l'histogramme et de tirer des conclusions sur la distribution de l'âge des grimpeurs.

On peut afficher les mêmes données dans une boîte à moustaches

```
boxplot <- ggplot(members_filtered, aes(y = age)) +
  geom_boxplot(fill = "skyblue", color = "black") +
  labs(title = "Répartition de l'âge des grimpeurs à succès du Mont Everest",
       y = "Âge") +
  theme_minimal()

print(boxplot)
```



La représentation en histogramme est davantage informative que celle à moustache. En effet la boîte à moustache ne donne pas assez de détails sur la répartition de l'âge des grimpeurs. On ne voit que l'age moyen et médian avec les extremas, sans avoir des détails sur les catégories les plus représentées ou les anomalies sur certaines années.

```r
age_min <- min(members_filtered$age, na.rm = TRUE)

age_max <- max(members_filtered$age, na.rm = TRUE)

print(paste("Âge minimum :", age_min))
```

```
## [1] "Âge minimum : 13"
```

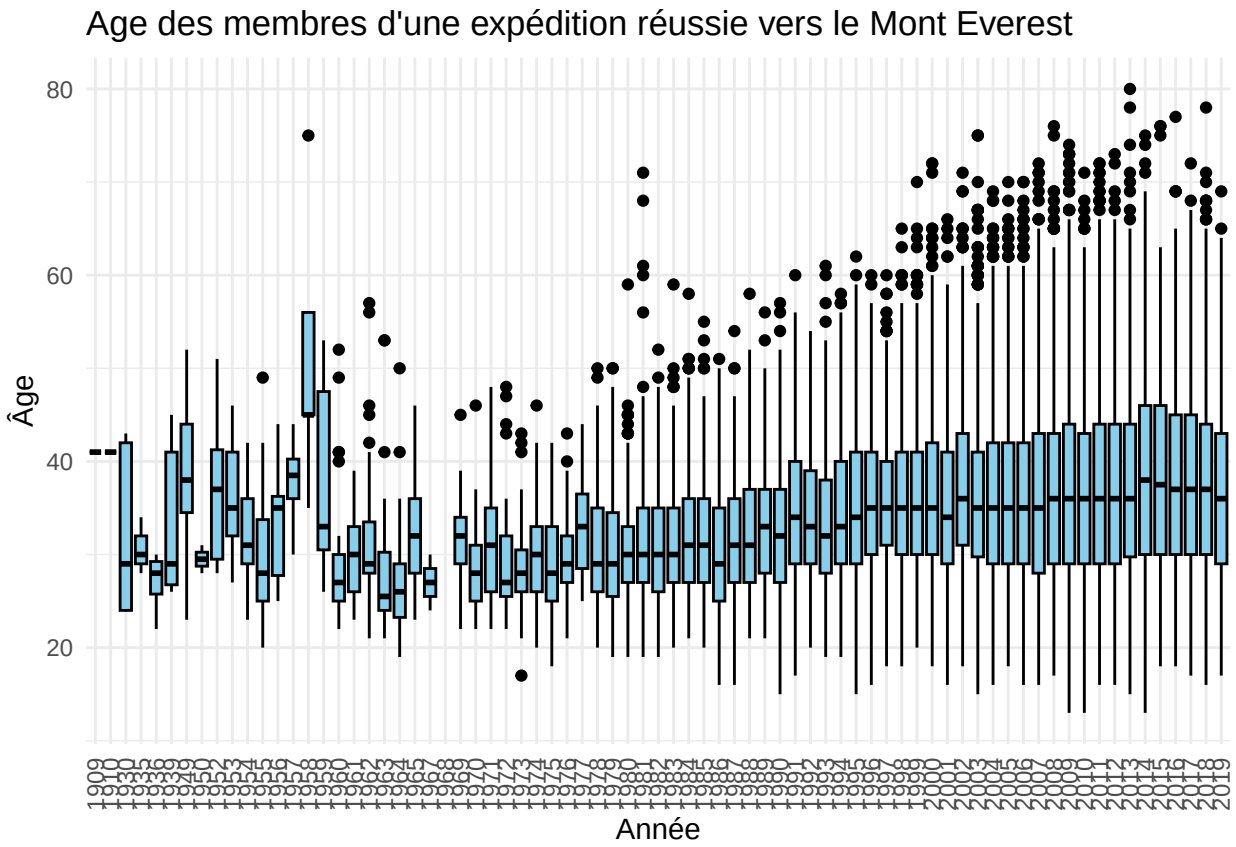
```r
print(paste("Âge maximum :", age_max))
```

```
## [1] "Âge maximum : 80"
```

Age des membres d'une expédition réussie ou non

```r
members_filtered <- members_not_null %>%
  filter(success == TRUE)

members_filtered$year <- as.factor(members_filtered$year)

boxplot_age_year <- ggplot(members_filtered, aes(x = year, y = age)) +
  geom_boxplot(fill = "skyblue", color = "black") +
  labs(title = "Age des membres d'une expédition réussie vers le Mont Everest",
       x = "Année",
       y = "Âge") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))

print(boxplot_age_year)
```

```
## Warning: Removed 1111 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```



Les médianes des boîtes sont assez stables d'une année à l'autre, on peut en conclure que l'âge médian des membres n'a pas beaucoup changé au fil du temps, sauf dans les premières anneés des relevés, où moins de grimpeurs réussissaient à gravir l'Everest. Les moustaches des boxplots indiquent la variabilité de l'âge des membres pour chaque année. On remarque que la taille des moustaches tend à augmenter montrant que la dispersion des âges a augmenté au fil du temps. Cela suggère une diversification de l'âge des membres des expéditions réussies vers le Mont Everest, avec une plus grande variété d'âges représentés au fur et à mesure que les années passent.

Age des membres d'une expédition réussie ou non

On se pose la question suivante : "Y-a-t-il une différence d'âge entre les membres d'une expédition réussie, et ceux d'une expédition qui a échouée, avec ou sans oxygène ?"

Pour répondre à cette question, nous devons comparer l'âge des membres des expéditions réussies avec ceux des expéditions échouées, en tenant compte de l'utilisation de l'oxygène. Voici comment nous pourrions décrire l'expérience statistique :

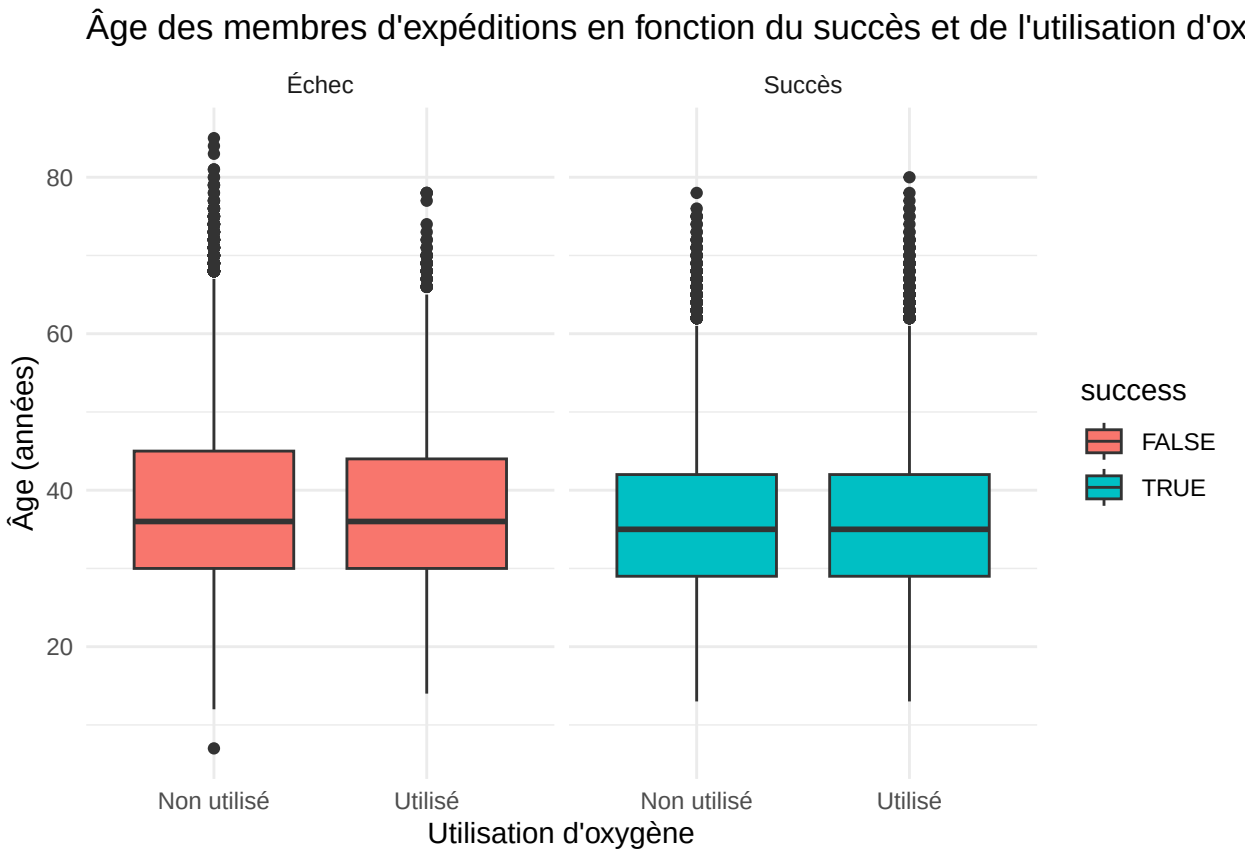
```
Individus : Les membres de différentes expéditions vers le Mont Everest.
Population : Toutes les expéditions vers le Mont Everest.
Échantillon : Les membres des expéditions réussies et des expéditions échouées, en distinguant ceux qui
Variable mesurée : L'âge des membres de l'expédition.
Critères de sélection : Nous sélectionnerons les lignes où l'expédition est réussie ou échouée, et où l
```

```
expeditions_filtered <- members %>%
  filter(success %in% c(TRUE, FALSE), !is.na(age))
```

```r
head(expeditions_filtered)
```

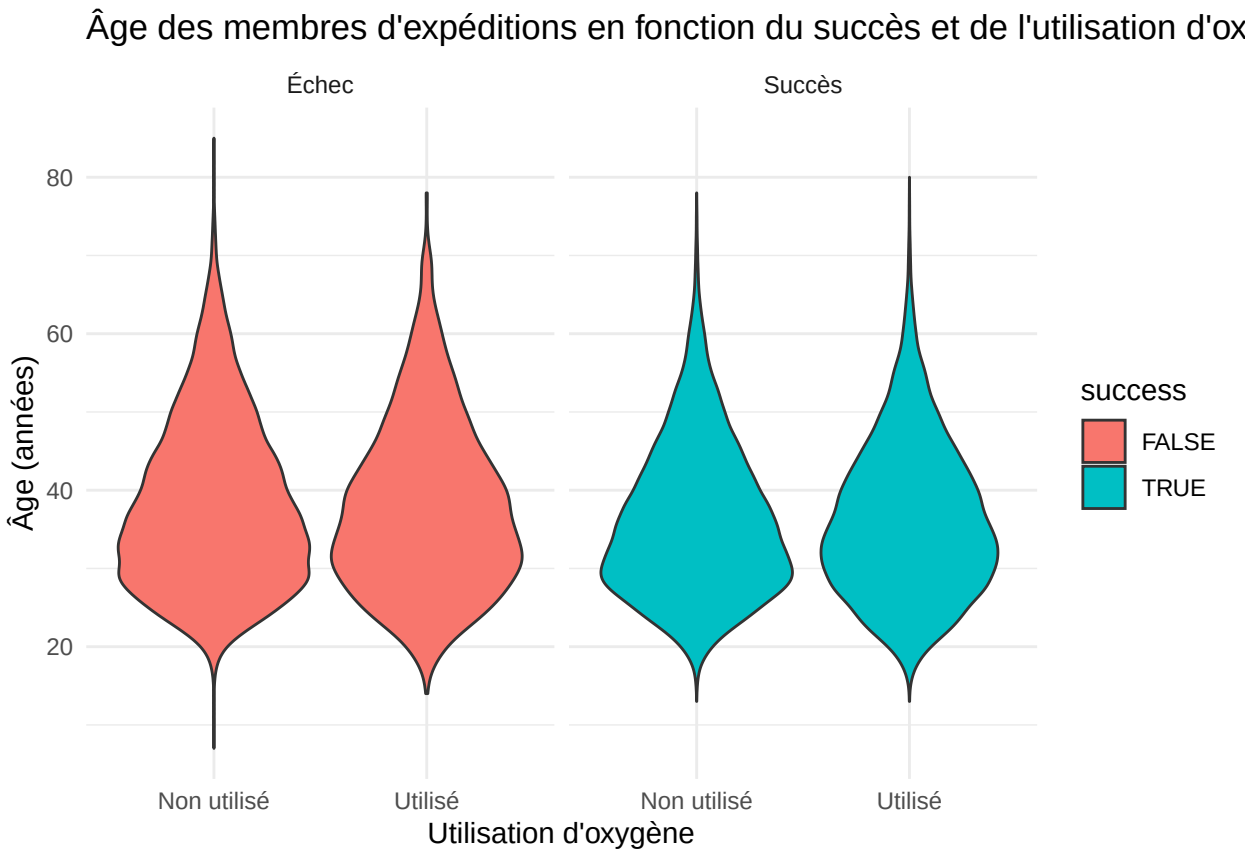
```
## # A tibble: 6 x 21
##   expedition_id member_id peak_id peak_name  year season sex     age citizenship
##   <chr>         <chr>     <chr>   <chr>     <dbl> <chr>  <chr> <dbl> <chr>
## 1 AMAD78301     AMAD7830~ AMAD    Ama Dabl~  1978 Autumn M        40 France
## 2 AMAD78301     AMAD7830~ AMAD    Ama Dabl~  1978 Autumn M        41 France
## 3 AMAD78301     AMAD7830~ AMAD    Ama Dabl~  1978 Autumn M        27 France
## 4 AMAD78301     AMAD7830~ AMAD    Ama Dabl~  1978 Autumn M        40 France
## 5 AMAD78301     AMAD7830~ AMAD    Ama Dabl~  1978 Autumn M        34 France
## 6 AMAD78301     AMAD7830~ AMAD    Ama Dabl~  1978 Autumn M        25 France
## # i 12 more variables: expedition_role <chr>, hired <lgl>,
## #   highpoint_metres <dbl>, success <lgl>, solo <lgl>, oxygen_used <lgl>,
## #   died <lgl>, death_cause <chr>, death_height_metres <dbl>, injured <lgl>,
## #   injury_type <chr>, injury_height_metres <dbl>
```

```r
boxplot <- ggplot(expeditions_filtered, aes(x = oxygen_used, y = age, fill = success)) +
  geom_boxplot() +
  labs(title = "Âge des membres d'expéditions en fonction du succès et de l'utilisation d'oxygène",
       x = "Utilisation d'oxygène",
       y = "Âge") +
  scale_x_discrete(labels = c("Non utilisé", "Utilisé")) +
  scale_y_continuous(name = "Âge (années)") +
  facet_wrap(success ~ ., labeller = as_labeller(c(`TRUE` = "Succès", `FALSE` = "Échec"))) +
  theme_minimal()

print(boxplot)
```



Il y a peu de différence entre les 4 boîtes. On remarque cependant un échec sans oxygène pour une personne mineure

```
# Représentation des données sous forme de violons
violin_plot <- ggplot(expeditions_filtered, aes(x = oxygen_used, y = age, fill = success)) +
  geom_violin() +
  labs(title = "Âge des membres d'expéditions en fonction du succès et de l'utilisation d'oxygène",
       x = "Utilisation d'oxygène",
       y = "Âge") +
  scale_x_discrete(labels = c("Non utilisé", "Utilisé")) +
  scale_y_continuous(name = "Âge (années)") +
  facet_wrap(success ~ ., labeller = as_labeller(c(`TRUE` = "Succès", `FALSE` = "Échec"))) +
  theme_minimal()

# Affichage des violons
print(violin_plot)
```



Cette représentation utilisant des violons offre une meilleure visualisation de la distribution des âges en fonction du succès de l'expédition et de l'utilisation d'oxygène. Les parties épaissies du violon représentent les régions où les valeurs sont plus fréquentes, tandis que les parties étroites représentent les valeurs moins fréquentes.