

Exercice 3

Gabriel Aubin-Moreau

2022-09-14

Tâche 1 : Réflexion

Pourquoi est-il important de visualiser les variables et les données ? Il est important de visualiser les variables et les données pour: voir des corrélations, des tendances, faire des comparaisons, etc.

Que signifie mapper des données pour représenter graphiquement l'esthétique ? Cela signifie qu'une fois l'ensemble de données chargé, il faut indiquer ou mapper à R, quelle variable doit aller sur quel axe et d'autres spécifications qui vont créer le rendu visuel ou l'esthétique de notre graphique. Par exemple, on indique la majorité du temps quelle variable utiliser pour l'axe des x et des y et parfois on sépare les données par couleur selon une autre variable.

Quelles données ont été mappées à quelle esthétique dans la vidéo de Rosling ? L'espérance de vie a été mappée à l'axe des y, le revenu par habitant a été mappé à l'axe des x, le continent a été mappé à la couleur et la population a été mappé à la taille.

Tâche 2 : Le Seigneur des Anneaux

Charger et nettoyer les données

Nous chargeons, restructurons et nettoyons d'abord les données.

```
# Vous n'aurez besoin que de la bibliothèque tidyverse pour cet exercice
library(tidyverse)

# Charger les jeux de données séparés
communauté <- read_csv("data/The_Fellowship_Of_The_Ring.csv")
tt <- read_csv("data/The_Two_Towers.csv")
roth <- read_csv("data/The_Return_Of_The_King.csv")

# bind_rows() empile les blocs de données les uns sur les autres
lotr_wide <- bind_rows(communauté, tt, roth) %>%
  # Faites de la colonne Film une variable catégorielle (facteur) et placez-la dans
  # l'ordre d'apparition des catégories (pour que les films soient dans le bon ordre)
  mutate(Film = fct_inorder(Film))

# Rendez ces données larges bien rangées
lotr <- lotr_wide %>%
  # C'est la nouvelle façon de rendre les données longues
  pivot_longer(cols = c(Female, Male),
    names_to = "Sexe", values_to = "Mots")
```

Course

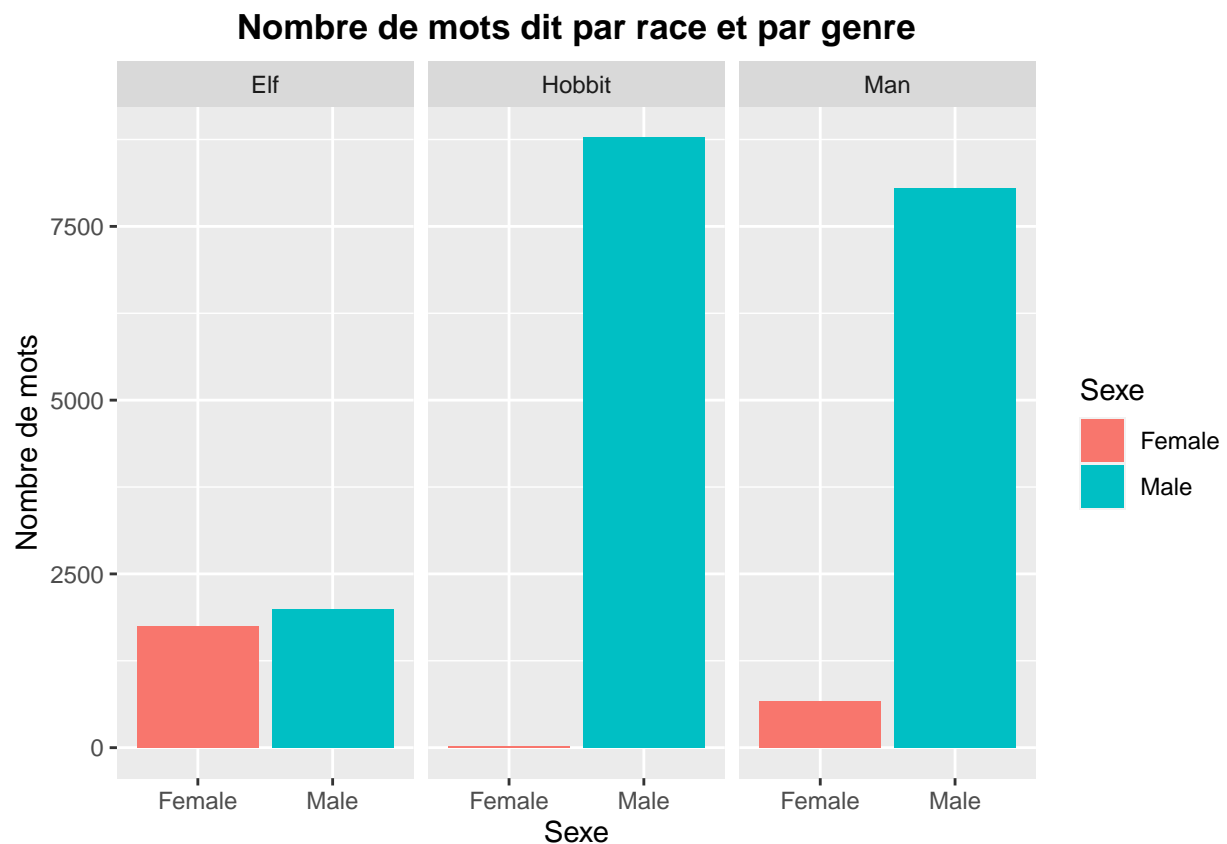
1. Les hobbit masculins ont dit 8780 mots.

2.

```
lotr_group_race_gender <- lotr %>% group_by(Race, Sexe) %>% summarise(NbMots = sum(Mots))
```

'summarise()' has grouped output by 'Race'. You can override using the '.groups' argument.

```
ggplot(lotr_group_race_gender, mapping=aes(Sexe, NbMots, fill=Sexe)) +  
  geom_col() +  
  facet_grid(cols=vars(Race)) +  
  labs(title="Nombre de mots dit par race et par genre", y="Nombre de mots") +  
  theme(plot.title = element_text(face = "bold", hjust = 0.5))
```

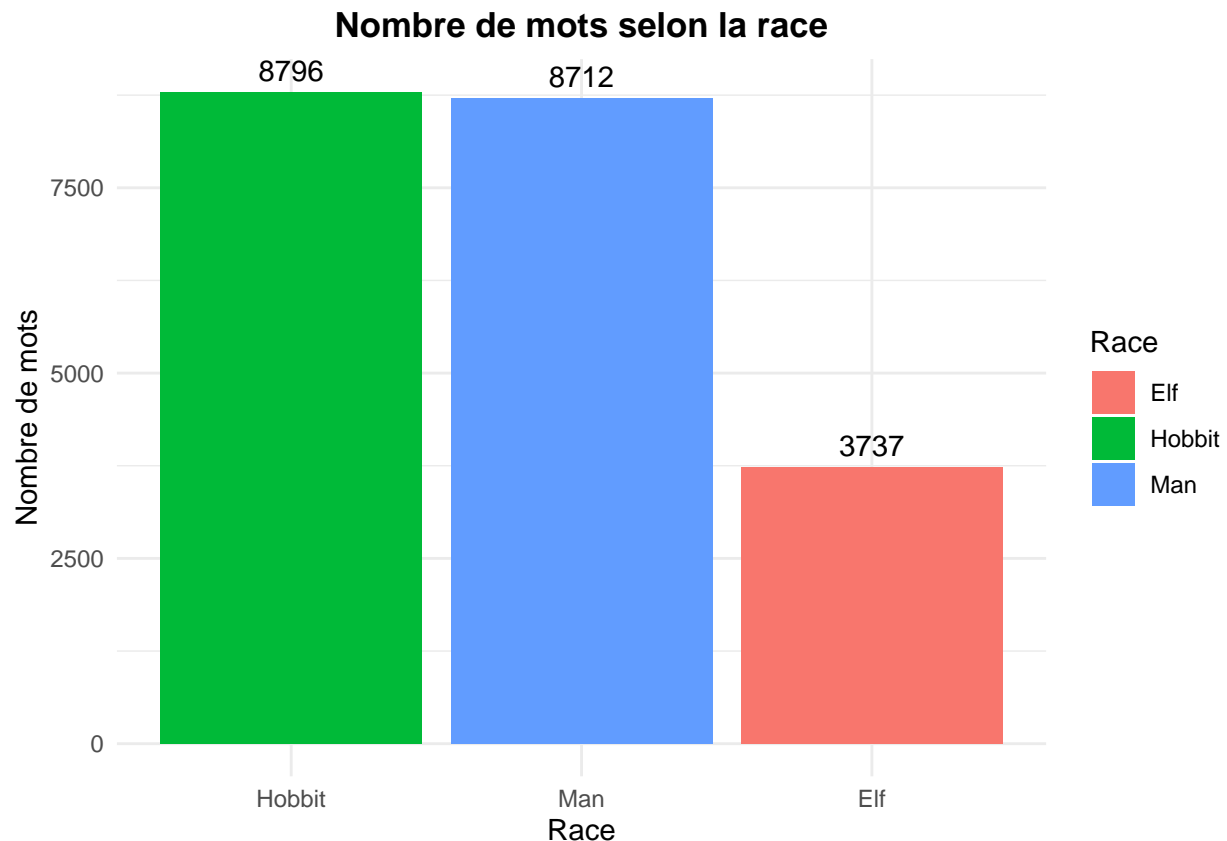


3. Une certaine race domine-t-elle toute la trilogie ? (indice : regrouper par Race)

Réponse: Hobbit

```
lotr_group <- lotr %>% group_by(Race) %>% summarise(NbMots = sum(Mots))  
library(ggplot2)  
ggplot(data = lotr_group, mapping = aes(x = reorder(Race, -NbMots), y=NbMots, fill=Race)) +
```

```
geom_col()+ geom_text(aes(label= NbMots), vjust=-0.5) +
labs(title="Nombre de mots selon la race", x="Race", y="Nombre de mots") +
theme_minimal() +
theme(plot.title = element_text(face = "bold", hjust = 0.5))
```



Genre et cinéma

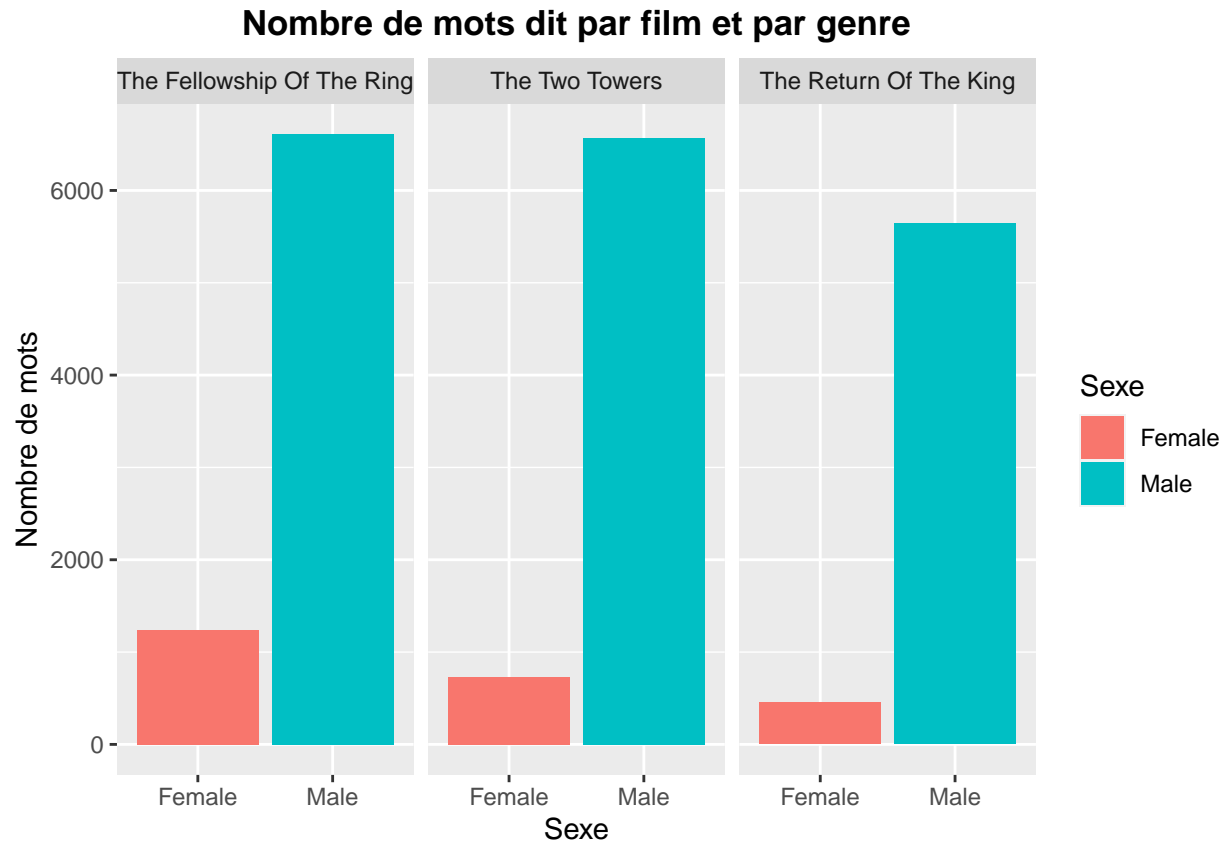
4. Est-ce qu'un certain genre domine un film ? (lol bien sûr que c'est le cas, mais quand même, représentez-le graphiquement) (Astuce : groupez à la fois par Gender et Film.) Expérimentez avec le remplissage par Gender ou Film et le facettage par Gender ou Film.

Réponse: le genre masculin domine tout les films.

```
lotr_group_gender_film <- lotr %>%
  group_by(Film, Sexe) %>%
  summarise(NbMots = sum(Mots))
```

'summarise()' has grouped output by 'Film'. You can override using the '.groups' argument.

```
ggplot(lotr_group_gender_film, mapping=aes(Sexe, NbMots, fill=Sexe)) +
  geom_col() +
  facet_grid(cols=vars(Film)) +
  labs(title="Nombre de mots dit par film et par genre", y="Nombre de mots") +
  theme(plot.title = element_text(face = "bold", hjust = 0.5))
```



Course et cinéma

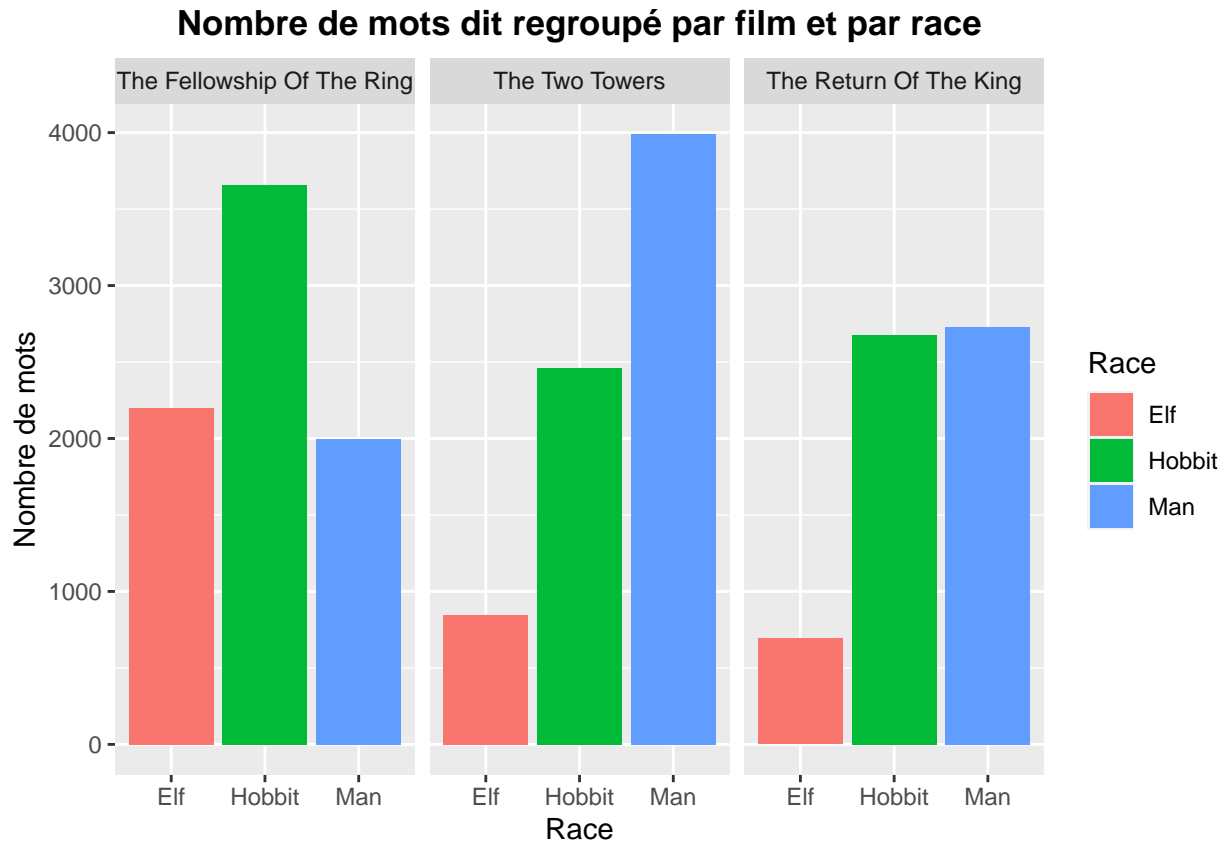
5. La race dominante diffère-t-elle d'un film à l'autre ? (Astuce : regrouper à la fois par Race et Film.) Expérimentez avec le remplissage par Race ou Film et le facettage par Race ou Film.

Réponse: Oui, dans le premier film, c'est les hobbits et dans les deux autres, c'est les humains.

```
lotr_group_film_race <- lotr %>%
  group_by(Film, Race) %>%
  summarise(NbMots = sum(Mots))
```

'summarise()' has grouped output by 'Film'. You can override using the '.groups' argument.

```
ggplot(lotr_group_film_race, mapping = aes(Race, NbMots, fill=Race)) +
  geom_col() +
  facet_grid(cols = vars(Film)) +
  labs(title = "Nombre de mots dit regroupé par film et par race", y="Nombre de mots") +
  theme(plot.title = element_text(face = "bold", hjust = 0.5))
```



Race et genre et film

6. Créez un graphique qui visualise simultanément le nombre de mots prononcés par race, sexe et film. Utilisez le bloc de données “lotr” complet. Vous n’avez pas besoin de créer un nouvel ensemble de données résumées (avec `group_by(Race, Gender, Film)`) car les données d’origine ont déjà une ligne pour chacune d’entre elles (vous pourriez créer un ensemble de données résumées, mais il serait identique à la version complète). Vous devez afficher Race, Gender et Film en même temps, mais vous n’avez que deux esthétiques possibles (x et fill), vous devrez donc également faire face à la troisième. Jouez avec différentes combinaisons (par exemple, essayez `x = Race`, puis `x = Film`) jusqu’à ce que vous en trouviez une qui raconte l’histoire la plus claire. Pour vous amuser, ajoutez un calque `labs()` pour ajouter un titre, un sous-titre et une légende.

```
ggplot(lotr, mapping = aes(Race, Mots, fill=Sexe)) +
  geom_col() + facet_grid(cols = vars(Film)) +
  labs(title="Nombre de mots dit par film et par race",
        subtitle = "Trilogie du seigneur des anneaux") +
  theme(plot.title=element_text(face = "bold", hjust=0.5),
        plot.subtitle = element_text(hjust=0.5))
```

Nombre de mots dit par film et par race

Trilogie du seigneur des anneaux

