

Livrable 1 - Pokemon

Musset Aurélien
Arulraj Sinthujan
Schummer Lucas
Khuu Sophie

2025-04-20

1. Visualisation des différents combos de types de Pokémon

Objectif du graphique :

Ici, nous allons répondre à la question suivante : **Quel est le nombre de Pokémon par combinaison de types ?** En effet, les pokémons possèdent un type principal (type1) et un type secondaire (type2). Le but est de voir ici, si ces types sont équitablement répartis, où si on peut observer des tendances. Nous allons donc créer une heatmap permettant de visualiser les combinaisons de types de Pokémon.

Présentation :

Pour réaliser ce graphique, nous avons utilisé les données du dataset "Pokemon_data". Nous avons filtré les Pokémon qui n'ont pas de type secondaire (type2 = "None"). Nous avons ensuite compté le nombre de Pokémon par combinaison de types, puis nous avons ajouté des totaux pour chaque type principal et secondaire.

Code et Visualisation

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.4.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.4.3
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
##      filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      intersect, setdiff, setequal, union
```

```
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 4.4.3
```

```
#Ouverture du dataset
```

```
df <- read.csv("data/Pokemon_data.csv", sep = ",")
```

```
#On ne garde que les colonnes nécessaires : name, type1 et type2
```

```
df <- df %>%
```

```

select(name, type1, type2) %>%
mutate(type1 = as.character(type1), type2 = as.character(type2))

# Comptage
df_filtered <- df %>%
  filter(type2 != "None") %>%
  count(type1, type2)

# Totaux
row_totals <- df_filtered %>%
  group_by(type1) %>%
  summarise(n = sum(n)) %>%
  mutate(type2 = "Total")

col_totals <- df_filtered %>%
  group_by(type2) %>%
  summarise(n = sum(n)) %>%
  mutate(type1 = "Total")

# Fusion
df_augmented <- bind_rows(df_filtered, row_totals, col_totals)

# Marquage des cellules de total
df_augmented <- df_augmented %>%
  mutate(
    is_total = type1 == "Total" | type2 == "Total",
    fill_value = ifelse(is_total, NA, n),
    text_color = ifelse(is_total, "black", "gray10"),
    border_color = ifelse(is_total, "black", "white")
  )

# Forcer l'ordre pour mettre "Total" à droite / en haut
type1_levels <- c(sort(unique(df_augmented$type1[df_augmented$type1 != "Total"])), "Total")
type2_levels <- c(sort(unique(df_augmented$type2[df_augmented$type2 != "Total"])), "Total")

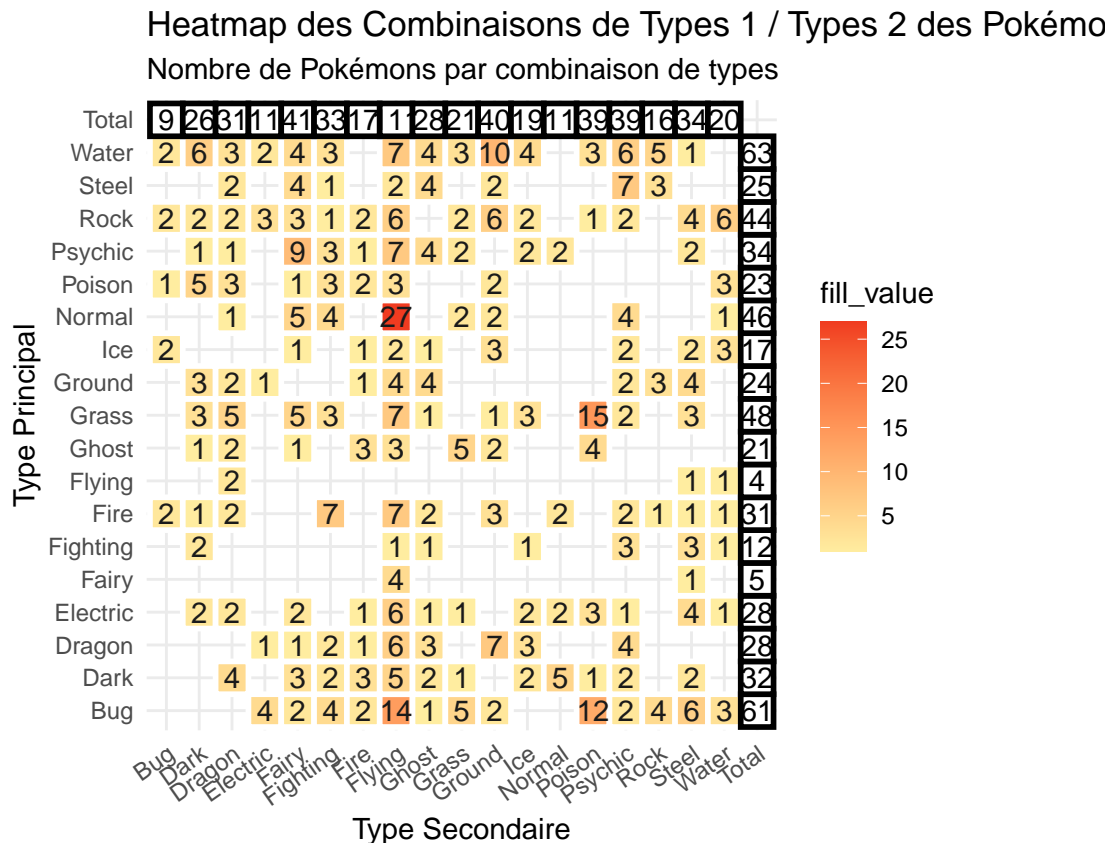
df_augmented <- df_augmented %>%
  mutate(
    type1 = factor(type1, levels = type1_levels),
    type2 = factor(type2, levels = type2_levels)
  )

# Heatmap finale
ggplot(df_augmented, aes(x = type2, y = type1)) +
  geom_tile(aes(fill = fill_value), color = df_augmented$border_color, size = 1) +
  geom_text(aes(label = n, color = text_color), show.legend = FALSE) +
  scale_fill_gradient(low = "#ffeda0", high = "#f03b20", na.value = "white") +
  scale_color_identity() +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 35, hjust = 1)) +
  labs(
    title = "Heatmap des Combinaisons de Types 1 / Types 2 des Pokémon avec un Type Secondaire",
    subtitle = "Nombre de Pokémon par combinaison de types",
    x = "Type Secondaire",

```

```
y = "Type Principal"
) +
coord_fixed()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Analyse :

Nous pouvons observer que la majorité des Pokémon possèdent un type principal de type “Eau” (Water) et un type secondaire de type “Vol” (Flying). Nous pouvons également observer que les Pokémon de type “Normal” (Normal) possèdent le plus fréquemment un type secondaire de type “Vol” (Flying) et que les Pokémon de type “Plante” (Grass) possèdent souvent un type secondaire de type “Poison” (Poison). **Ce sont les combos les plus fréquents.** On peut également remarquer que de nombreux combos n’existent pas, comme par exemple “Poison” (Poison) et “Acier” (Steel), ou encore “Fée” (Fairy) et “Combat” (Fighting). Cela peut sembler logique pour certaines combinaisons (par exemple “Feu” (Fire) et “Eau” (Water)), mais il est intéressant de le noter. Malgré tout, on peut remarquer que la plupart des combinaisons sont représentées par la même échelle de couleur, ce qui signifie qu’il n’y a pas de combo de type qui se démarque vraiment des autres (hormis ceux notés au début).

2. Visualisation de l’évolution du nombre de ventes

Objectif du graphique :

Ici, nous allons répondre à la question suivante : **Y a t-il une corrélation entre le nombre de ventes et**

l'année de sortie du jeu ?

Présentation :

Pour réaliser ce graphique, j'ai utilisé les données du dataset "vgsales". Ce graphique présente l'évolution des ventes des jeux Pokémon (> 6 Millions de ventes) au cours du temps. Il présente l'évolution des ventes des jeux Pokémon global et également pour chaque région.

L'axe des abscisses représente le temps en année de 1996 à 2014.

L'axe des ordonnées représente le nombre de ventes en millions de 0 à plus de 30 millions.

Code et Visualisation

```
# Charger les packages
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.4.3
```

```
## Warning: package 'tibble' was built under R version 4.4.3
```

```
## Warning: package 'readr' was built under R version 4.4.3
```

```
## Warning: package 'forcats' was built under R version 4.4.3
```

```
## Warning: package 'lubridate' was built under R version 4.4.3
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
```

```
## v forcats 1.0.0      v readr 2.1.5
```

```
## v lubridate 1.9.4    v stringr 1.5.1
```

```
## v purrr 1.0.4       v tibble 3.2.1
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
```

```
## x dplyr::lag()     masks stats::lag()
```

```
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
# Charger les données
```

```
df <- read.csv("data/vgsales.csv")
```

```
# Filtrer les jeux Pokémon avec >6M ventes
```

```
pokemon_sales <- df %>%
```

```
  filter(grepl("Pokemon", Name, ignore.case = TRUE), Global_Sales > 6) %>%
```

```
  filter(!is.na(Year)) %>%
```

```
  mutate(Year = as.integer(as.character(Year)))
```

```
# Agréger ventes par année
```

```
pokemon_region_year <- pokemon_sales %>%
```

```
  group_by(Year) %>%
```

```
  summarise(
```

```
    North_A = sum(NA_Sales, na.rm = TRUE),
```

```
    EU = sum(EU_Sales, na.rm = TRUE),
```

```
    JP = sum(JP_Sales, na.rm = TRUE),
```

```
    Other = sum(Other_Sales, na.rm = TRUE),
```

```
    Total = sum(Global_Sales, na.rm = TRUE)
```

```
  ) %>%
```

```
  pivot_longer(cols = c(North_A, EU, JP, Other, Total), names_to = "Region", values_to = "Sales")
```

```
# Graphique esthétique
```

```
ggplot(pokemon_region_year, aes(x = Year, y = Sales, color = Region, group = Region)) +
```

```
  geom_line(aes(size = Region == "Total", linetype = Region == "Total")) +
```

```
  geom_point(size = 2.5) +
```

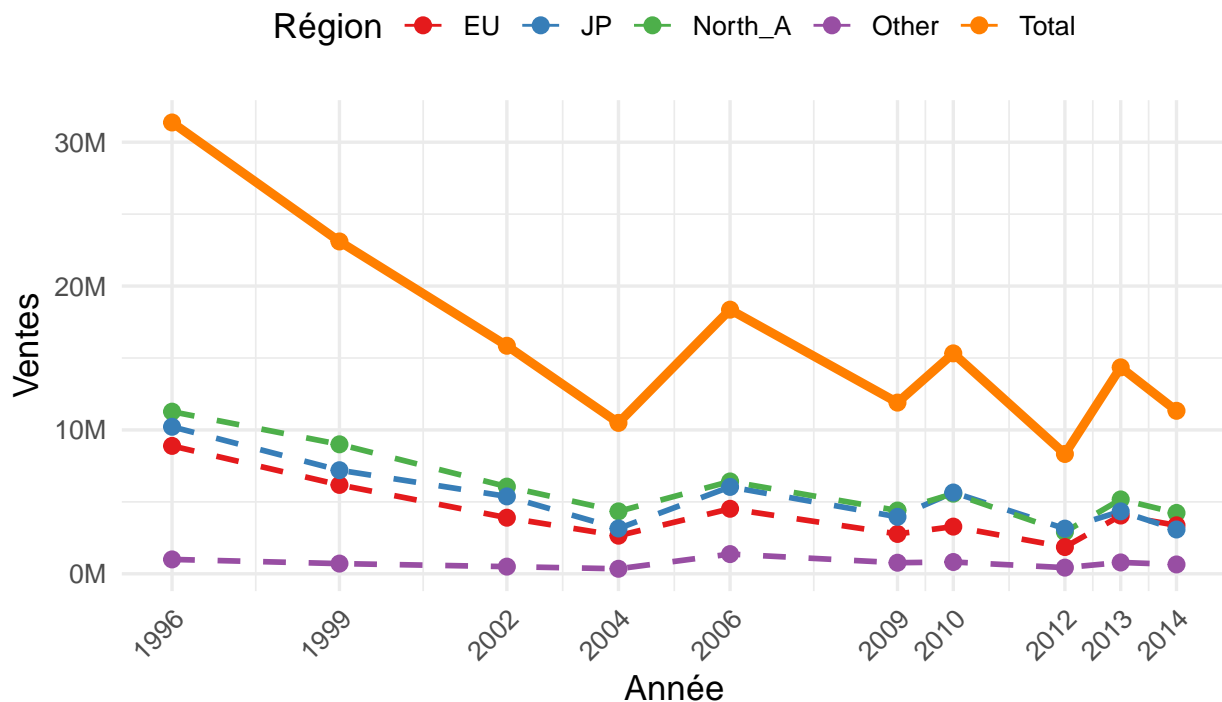
```

scale_size_manual(values = c("TRUE" = 1.6, "FALSE" = 1), guide = "none") +
scale_linetype_manual(values = c("TRUE" = "solid", "FALSE" = "dashed"), guide = "none") +
scale_color_brewer(palette = "Set1") +
scale_y_continuous(labels = scales::label_number(suffix = "M", scale = 1)) +
scale_x_continuous(breaks = unique(pokemon_region_year$Year)) +
labs(
  title = "Évolution des ventes des jeux Pokémon (>6M de ventes)",
  subtitle = "Répartition par région et total global (en millions d'unités)",
  x = "Année",
  y = "Ventes",
  color = "Région"
) +
theme_minimal(base_size = 13) +
theme(
  plot.title = element_text(face = "bold", hjust = 0.5),
  plot.subtitle = element_text(hjust = 0.5),
  legend.position = "top",
  axis.text.x = element_text(angle = 45, hjust = 1)
)

```

Évolution des ventes des jeux Pokémon (>6M de ventes)

Répartition par région et total global (en millions d'unités)



Analyse :

On peut remarquer que les jeux récents se vendent moins bien que les précédents. Le jeu Pokémon le plus vendu de l'histoire est sorti en 1996 avec plus de 30 millions de ventes, tandis que les plus récents peinent à atteindre les 20 millions de ventes.

Cela peut être surprenant, si on pense à la popularité des jeux Pokémon dans les années 2000 et 2010 et à l'essor des jeux vidéo et des consoles, qui occupaient une place centrale dans les loisirs des jeunes à cette

époque.