

Type de jeux en ligne sur Steam

Hoang-Viêt LE

2025-04-24

Avant l'analyse...

Importation des données

Dans cette partie, on doit importer les données dans les différents fichiers et réaliser tout le nécessaire afin d'obtenir un dataframe avec toutes les informations que l'on veut afin de l'exploiter pour notre analyse et nos visualisations.

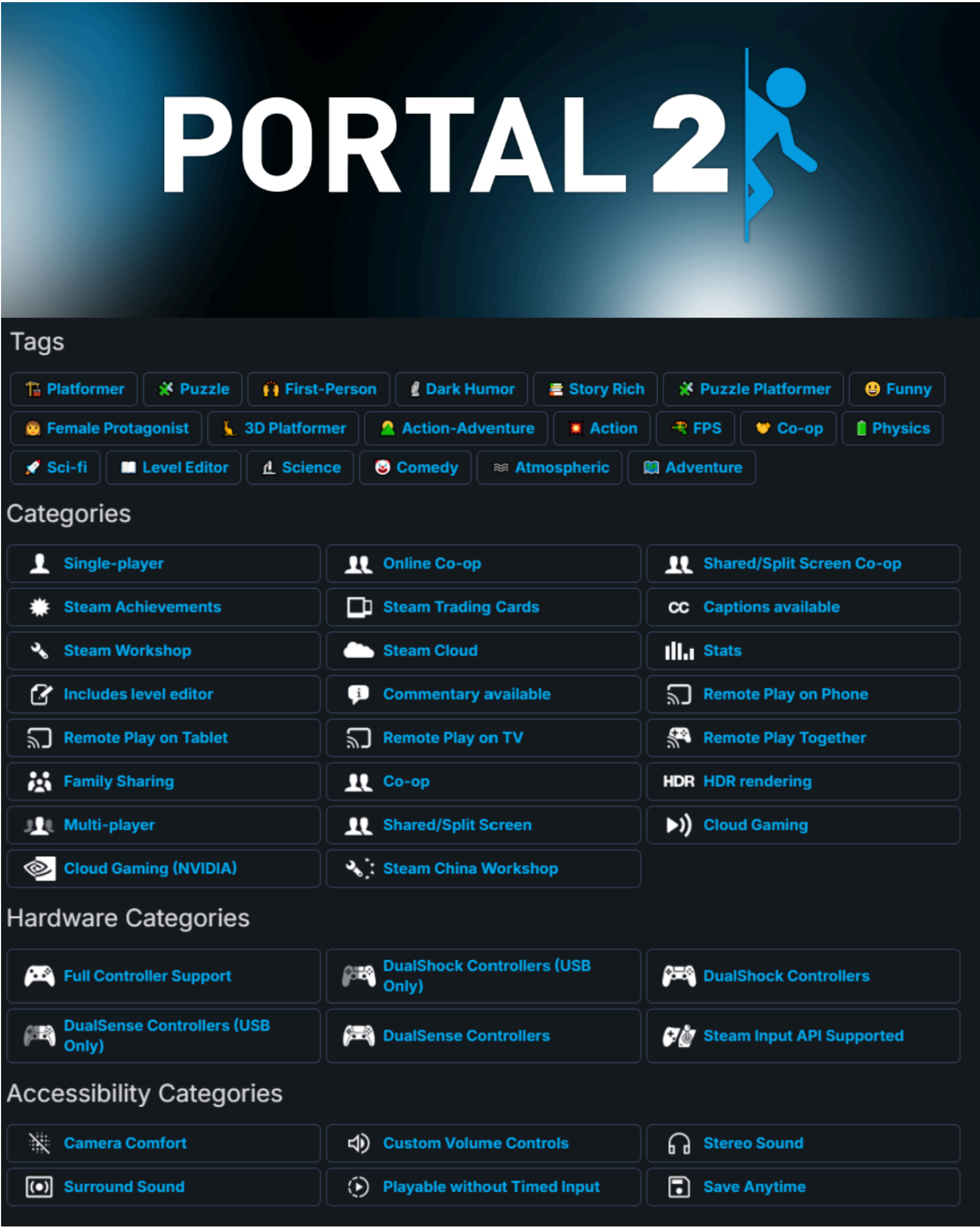
Ce que l'on a à faire :

- Tous les tableaux partagent une colonne commune : `app_id`, qui sera utilisée pour les jointures. On va alors créé un grand dataframe avec l'ensemble des variables que l'on veut exploiter.
- Certaines colonnes nécessitent un nettoyage, notamment la colonne `initial_price` contenant parfois la valeur "N"
- Regrouper les tags et les catégories sur une même ligne (même `app_id`) des fichiers `tags.csv` et `categories.csv` : les colonnes `tags` et `categories` seront des chaînes de caractères contenant l'ensemble des tags/catégories séparés chacun par une virgule
- Convertir `initial_price` en une donnée exploitable dans `steamspy.csv` (ex : 999 veut dire 9.99 USD)

Différences entre “Catégorie” et “Tag”

Sur Steam, il existe deux types de variables qui permettent de classer les jeux présents sur la plateforme :

- Catégories : classifications officielles attribuées par Steam. Elles décrivent les fonctionnalités principales d'un jeu (par exemple : Multijoueur, Succès Steam, Support manette, Mode coopératif, etc.). Ce sont des critères souvent liés à l'expérience utilisateur et aux fonctionnalités techniques du jeu.
- Tags : classifications collaboratives créées par les utilisateurs de Steam. Ils permettent de décrire le contenu, l'ambiance ou le style d'un jeu (par exemple : Puzzle, Narratif, Indépendant, FPS, Monde ouvert, etc.). Les tags sont plus subjectifs et peuvent varier en popularité selon la communauté.



Tags et Catégories de Portal 2 sur Steam

Analyse

Contexte et problématique

C'est sur la plateforme Steam que l'on a vu naître les premiers jeux en ligne et le développement de celui-ci avec des jeux mythiques comme Counter Strike, Team Fortress, etc... D'autres types de jeux sont aussi devenus populaires au cours du temps comme les MMORPG, les Battle-Royale, etc... Nous voulons voir les tendances actuelles et quels types de jeux attirent principalement les joueurs de Steam.

Notre problématique est alors la suivante :

Quels sont les types de jeu en ligne les plus populaires sur Steam ? (FPS, MMO, etc...)

Définition de jeu en ligne

Une définition de jeu en ligne sera un jeu vidéo dont l'expérience se repose sur l'interaction entre plusieurs joueurs en temps réel. On distingue différents types de jeu en ligne :

- MMO (Massive Multiplayer Online)
- Jeux compétitifs (FPS, MOBA, Battle Royale)
- Jeux collaboratifs (Eurotruck 2)

Critères “jeu en ligne” :

Dans nos données, nous identifierons un jeu comme "jeu en ligne" s'il possède les catégories ou tags suivants :

- "Multi-player", "Massively Multiplayer", "Online Co-op", ou d'autres tags associés.

Critère “popularité” :

Le critère "popularité" sera basée sur le nombre de joueurs actifs en Octobre 2024.

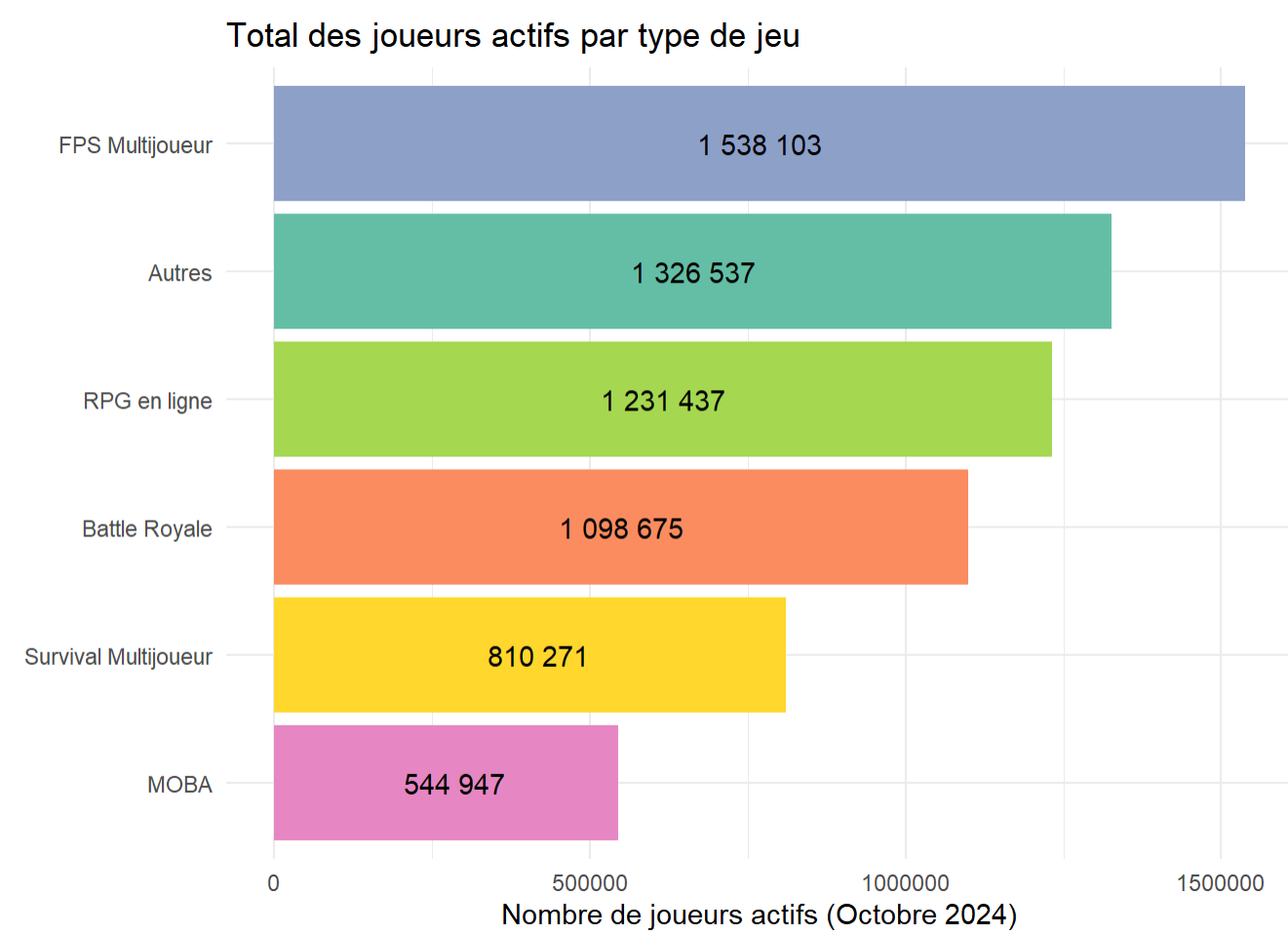
Types de jeux en ligne

Pour analyser les type de jeux les plus populaires pour les jeux en ligne et regarder sa distribution dans le magasin Steam, on va considérer les tags suivants uniquement :

- FPS Multijoueur
- Battle Royale
- RPG en ligne
- Coopération en ligne
- Sandbox
- Survival Multijoueur
- Autres (correspondant aux autres tags non classés)

Analyse des résultats

Voici la répartition des joueurs sur la plateforme Steam sur les jeux en lignes :



Observations

On voit que le genre de jeu en ligne le plus populaire sur Steam est le FPS Multijoueur rassemblant plus d'un tiers des joueurs des jeux en ligne sur Steam représentant 1.5 millions de joueurs avec CS 2 qui représente plus de 1,1 millions de joueurs.

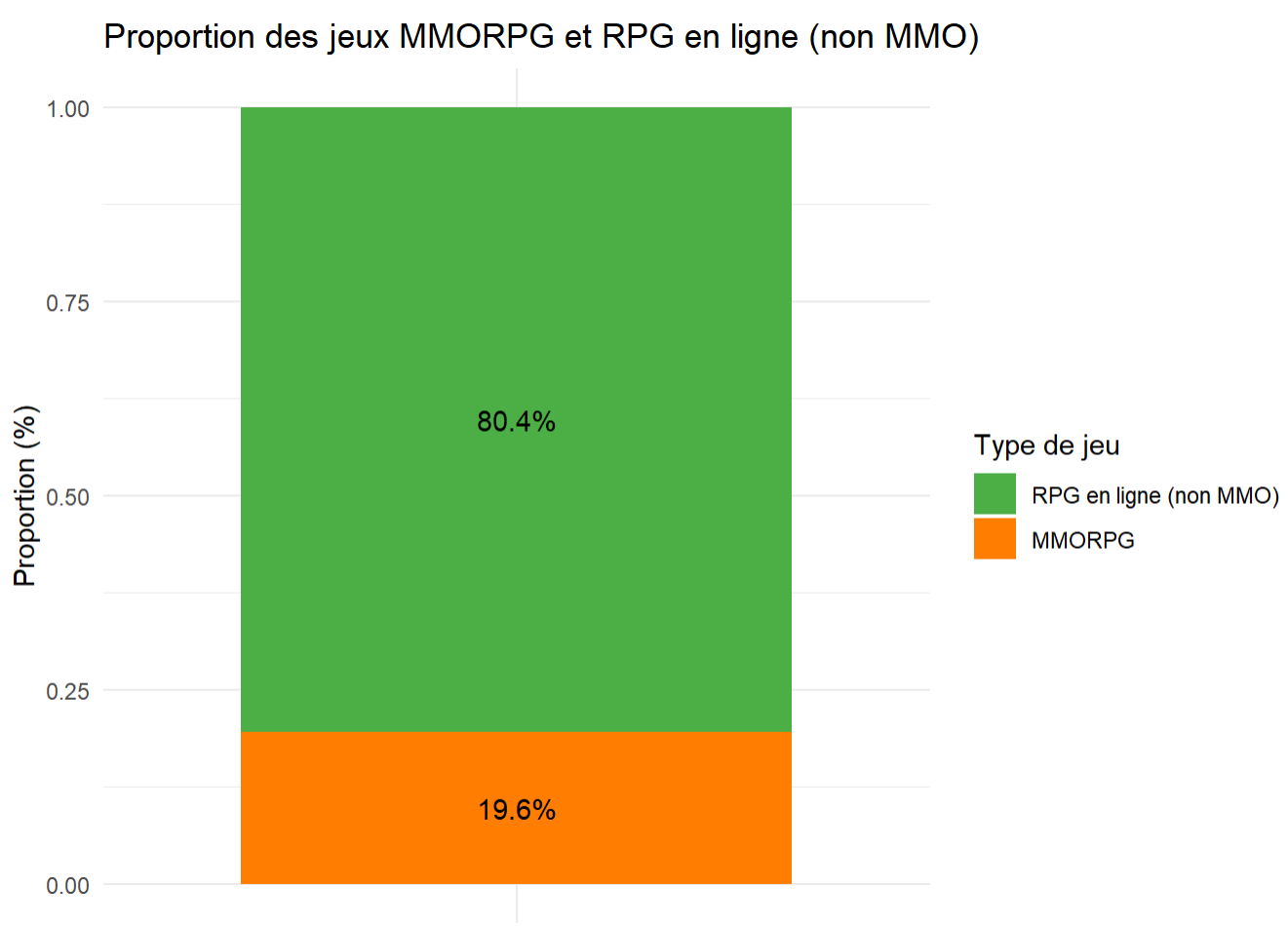
Ensuite, dans la catégorie "Autres", On y compte 1.3 millions joueurs sur des jeux comme Liar's Bar, War Thunder, Civilization VI, Forza Horizon et Eurotruck Simulator 2. C'est une catégorie regroupant pleins de genres différents.

Ensuite, les Battle Royale rassemble 1 million de joueurs (environ 16.5% des joueurs). Le genre semble être resté populaire mais elle n'a plus la même popularité qu'entre 2018-2020.

Les RPG en ligne représentent 1.2 million de joueurs mais il faut distinguer les différents genre de RPG en ligne.

En effet, dans les RPG en ligne, ils se trouvent les MMO comme Monster Hunter: World, New World: Aeternum ou Destiny 2 mais également des RPG qui peut se jouer en coopération comme Stardew Valley, Baldur's Gate 3, etc...

Voici la répartition des joueurs jouant à des MMORPG et des joueurs de RPG classique :



Note : Stardew Valley a le tag RPG et est un jeu qui peut se jouer en multijoueur mais peut-on considérer comme un jeu en ligne ? C'est le cas d'autres jeux principalement singleplayer mais qui peut se jouer en multijoueur.

Les Survival Multijoueur représentent une base de 800 000 joueurs sur des jeux phares comme Rust, Factorio, Satisfactory, etc...

Enfin, les MOBA représentent uniquement moins de 550 000 joueurs dont la majorité des joueurs viennent de Dota 2 (500 000 joueurs).

Limites de résultats obtenus

- Les tags Steam et les catégories créées ne sont pas assez précises pour trouver les jeux en ligne car on se retrouve souvent à la frontière d'un jeu avec un système multijoueur et un jeu en ligne.
- D'autres visualisations au sein des catégories nous permettraient de voir qu'uniquement quelques jeux portent la base de jeu de certains types de jeux comme CS 2 pour les FPS et Dota 2 pour les MOBA.
- Le fait que la catégorie "Autres" possèdent de nombreux jeux en ligne que l'on a pas pu classer montre que le domaine des jeux en ligne est très variée et non cantonnée aux genres que l'on pense habituellement. Globalement, réaliser une classification sur l'ensemble des jeux Steam semble difficile tant certains jeux sont assez particulier pour les catégoriser.
- Enfin, il faut être conscient que ces données ne concernent que les utilisateurs Steam et non la globalité des joueurs. Par exemple, pour les MMORPG, ceux-ci peuvent être lancés hors Steam. De plus avec l'avènement de l'Epic Game Store, une partie des jeux sont passés sur l'autre plateforme comme Rocket League.

rapport_analyse_prix

Florian Lopitiaux

2025-04-25

Les facteurs qui influencent le prix des jeux

Dans cette section, nous allons nous intéresser aux facteurs qui peuvent avoir une influence sur le prix des jeux. Nous avons à partir des données de notre jeu de données relevé différents aspects qui seraient intéressants d'analyser afin de mieux comprendre comment est défini le prix d'un jeu.

Chaque aspect sera approfondi dans sa partie : contexte (pourquoi on se pose la question), traitement des données, analyse (graphique) et conclusion.

Voici les aspects que nous analyserons :

- le nombre de langues disponibles sur un jeu : est-ce que le coût de traduction impacte le prix du jeu ?
- les genres/tags des jeux : est-ce qu'on peut observer une différence en moyenne significative entre certains genres/tag de jeu ? Par exemple : Un jeu RPG open-world en 3D serait plus complexe et cher à développer qu'un plateforme 2D et donc plus cher ?
- l'évolution des prix au fil du temps ? Est-ce qu'on peut observer une hausse des prix des jeux en fonction de leur date de sortie ?
- Y a t-il des différences de prix notables entre les studios de développement ? Ce dernier aspect sera à mettre en relation avec l'aspect étudié précédant sur les genres/tags de jeu que développent les studios.

Préambule : Comme nous sommes actuellement pour le premier rendu, seul le premier aspect (les langues) de cette section sur les prix sera traité dans ce rendu. D'autres parts, de nouveaux aspects pourraient être ajoutés d'ici le rendu final si nous en trouvons durant notre analyse.

Commençons par importer notre jeu de données et les librairies nécessaires à l'analyse :

```
library(tidyverse)

games <- read_delim("../data/clean_games.csv", delim=";", show_col_types = FALSE)
head(games)
```

app_id	type	name	release_date	is_free	price	currency	
<dbl>	<chr>	<chr>	<date>	<dbl>	<dbl>	<chr>	►
10	game	Counter-Strike	2000-11-01	0	8.19	EUR	
20	game	Team Fortress Classic	1999-04-01	0	4.99	EUR	
30	game	Day of Defeat	2003-05-01	0	4.99	EUR	
40	game	Deathmatch Classic	2001-06-01	0	4.99	EUR	
50	game	Half-Life: Opposing Force	1999-11-01	0	4.99	EUR	
60	game	Ricochet	2000-11-01	0	4.99	EUR	

6 rows | 1-7 of 8 columns

Filtrage des données

Avant de passer à l'analyse, nous devons d'abord filtrer nos données, notamment celles où nous n'avons pas d'informations sur le prix. Ce manque d'information s'explique soit car le jeu est gratuit (free-to-play <=> F2P), soit par une erreur dans les données.

Nous décidons de ne pas prendre en compte les jeux gratuits dans nos analyses du prix et de les analyser distinctement dans une autre section.

```
print(paste0("Nombre total de lignes du dataframe : ", nrow(games)))

## [1] "Nombre total de lignes du dataframe : 140082"

games_price <- games %>% filter(is_free == 0)
print(paste0("Nombre de lignes sans les F2P : ", nrow(games_price)))

## [1] "Nombre de lignes sans les F2P : 106421"

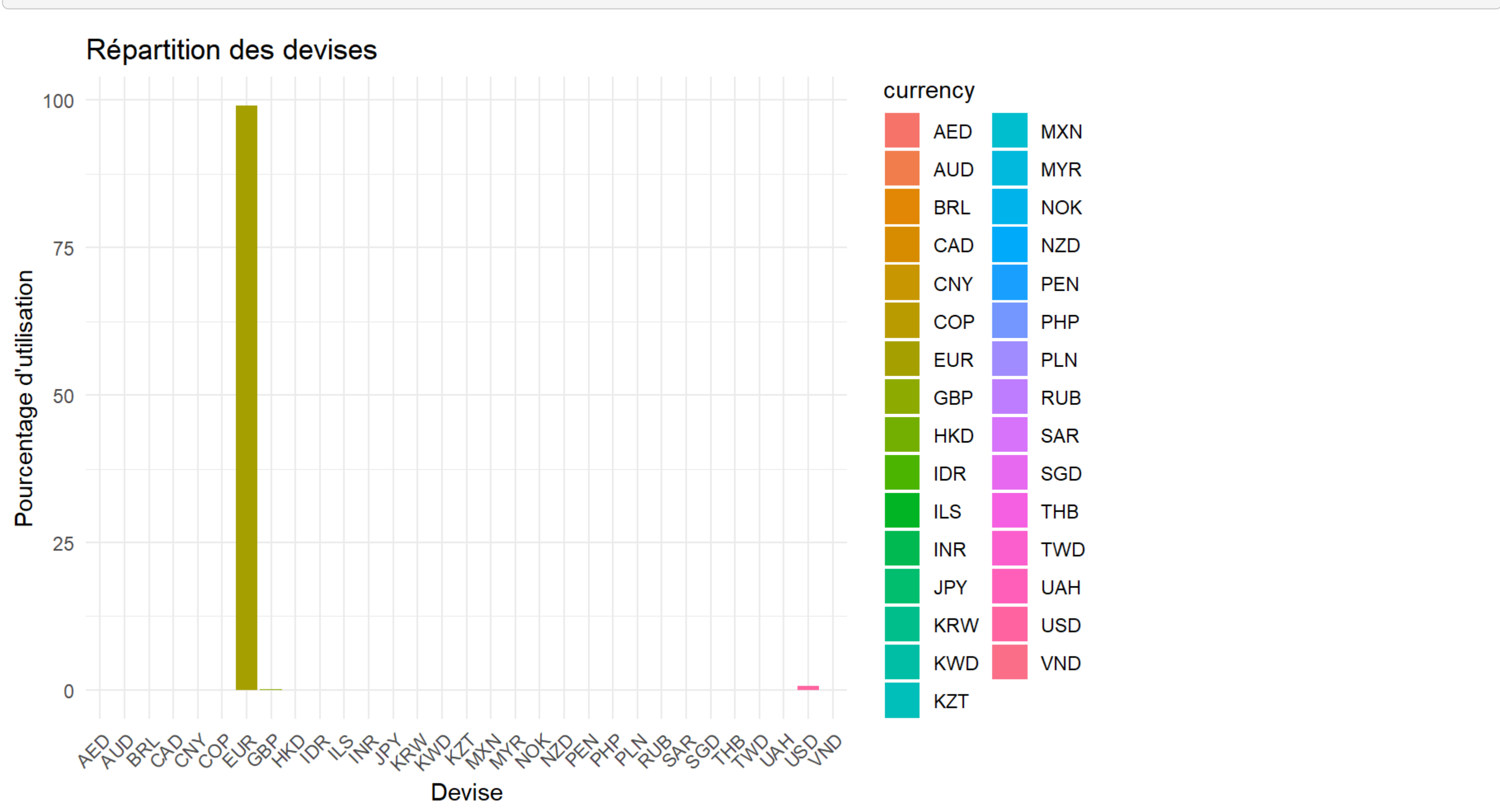
games_price <- games_price %>% filter(!is.na(price))
print(paste0("Nombre de lignes sans les F2P + erreurs de données : ", nrow(games_price)))

## [1] "Nombre de lignes sans les F2P + erreurs de données : 76332"
```

De plus, notre jeu de données contient une colonne "currency" contenant la devise de la monnaie utilisée pour exprimé le prix du jeu. Voici un bar plot montrant le pourcentage de représentation de chaque devise présente dans le dataframe.

```
currency_stats <- games_price %>%
  count(currency) %>%
  mutate(percentage = n / sum(n) * 100)

ggplot(currency_stats, aes(x = currency, y = percentage, fill = currency)) +
  geom_bar(stat = "identity") +
  labs(title = "Répartition des devises", x = "Devise", y = "Pourcentage d'utilisation") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



On observe que l'écrasante majorité des prix sont formulés sous la devise de l'EURO. La solution optimale serait de convertir les prix avec une autre devise en EURO afin de pouvoir les comparer. Toutefois, dû à la complexité de la tâche et au manque de temps nous décidons de simplifier et d'utiliser uniquement les jeux exprimés en EURO.

```
print(paste0("Nombre de lignes avec toutes les devises : ", nrow(games_price)))

## [1] "Nombre de lignes avec toutes les devises : 76332"

games_price <- games_price %>% filter(currency == "EUR")
print(paste0("Nombre de lignes avec uniquement les jeux en EURO : ", nrow(games_price)))

## [1] "Nombre de lignes avec uniquement les jeux en EURO : 75621"
```

Enfin, nous devons également enlever les lignes qui ne contiennent pas d'informations sur les langues disponibles des jeux.

```
games_langues <- games_price %>% filter(!is.na(languages))
print(paste0("Nombre de lignes sans les jeux avec aucune information sur les langues : ", nrow(games_langues)))

## [1] "Nombre de lignes sans les jeux avec aucune information sur les langues : 75529"
```

On observe que cette étape supprime peu de lignes, on en déduit donc que les lignes où il y avait une erreur dans les prix sont probablement en majorité les mêmes lignes où il y a eu une erreur avec les langues et qu'il n'y a pas d'information.

L'impact des langues disponibles

Maintenant que le filtrage de données nécessaires pour cette partie est terminé, nous pouvons commencer notre analyse !

Notre objectif est de trouver une potentielle corrélation entre le prix des jeux et le nombre de langues disponibles. En effet, hormis la langue originale du jeu (celle du studio qui l'a développé), chaque langue disponible en plus nécessite un coût supplémentaire pour le studio de développement qui doit engager une équipe de traduction. De plus, cette traduction demande parfois des adaptations visuelles d'éléments graphiques directement dans le jeu, par exemple : si les langues sont extrêmement différentes (langues asiatiques fonctionnant sur un système d'idéogrammes et langues occidentales se basant sur l'alphabet latin).

De ce fait, il nous paraît possible que ce coût puisse avoir une répercussion sur le prix du jeu et donc qu'on puisse observer une corrélation entre ces deux éléments avec nos données.

Pour ce faire, nous allons d'abord devoir calculer le nombre de langues par jeu, rajoutons une nouvelle colonne 'nb_languages' qui contiendra ce nombre :

```
games_langues <- games_langues %>%
  mutate(
    nb_languages = str_split(languages, ",") %>% map_int(~ length(.x))
  )

head(games_langues)
```

app_id	type	name	release_date	is_free	price	currency	
<dbl>	<chr>	<chr>	<date>	<dbl>	<dbl>	<chr>	►
10	game	Counter-Strike	2000-11-01	0	8.19	EUR	
20	game	Team Fortress Classic	1999-04-01	0	4.99	EUR	
30	game	Day of Defeat	2003-05-01	0	4.99	EUR	
40	game	Deathmatch Classic	2001-06-01	0	4.99	EUR	
50	game	Half-Life: Opposing Force	1999-11-01	0	4.99	EUR	
60	game	Ricochet	2000-11-01	0	4.99	EUR	

6 rows | 1-7 of 9 columns

Et voilà maintenant que ceci est fait nous pouvons maintenant créer notre graphique afin de voir, si oui ou non, une corrélation est observable. Pour ce graphique, nous décidons de partir sur un "scalar plot" avec comme axe horizontal le prix des jeux et comme axe vertical le nombre de langues disponibles :

```
ggplot(games_langues, aes(x = price, y = nb_languages)) +
  geom_point(color = "blue", alpha = 0.5) +
  labs(title = "Corrélation entre le Prix des jeux et le nombre de Langues disponibles",
    x = "Prix (€)",
    y = "Nombre de Langues") +
  theme_minimal()
```



Comme vous pouvez le constater le graphique est très peu lisible à cause de ses échelles, cela est dû aux points dits "outlier" : des points extrêmes en prix ou nombre de langues relevant "d'anomalie" et ne représentant pas la masse des données.

Nous décidons donc de les enlever pour réduire l'échelle et mieux pouvoir analyser le graphique.

```
games_langues <- games_langues %>% filter(nb_languages < 50) %>% filter(price <= 100)

ggplot(games_langues, aes(x = price, y = nb_languages)) +
  geom_point(color = "blue", alpha = 0.5) +
  labs(title = "Corrélation entre le Prix des jeux et le Nombre de Langues disponibles",
    x = "Prix (€)",
    y = "Nombre de Langues") +
  theme_minimal()
```



Comme nous pouvons l'observer, de nombreux jeux possédant le même nombre de langues se vendent à des prix très différents et inversement de nombreux jeux aux mêmes prix possèdent un nombre de langues très différents. De plus, nous n'observons pas d'augmentation de prix de façon linéaire par rapport au nombre de langues disponibles.

Il ne semble donc pas y avoir de corrélation entre le nombre de langues disponibles et le prix d'un jeu.

Notre hypothèse concernant les moyens et coûts de traduction qui pourraient impacter le prix est donc fausse. Cela nous paraît cohérent étant donné qu'aujourd'hui le support multi-langage est devenu une norme pour une grande partie des jeux. Les studios aujourd'hui visent un marché international et donc doivent rendre leurs jeux accessibles pour ces marchés.

rapport_analyse_indie_aaa

Paul Lassasseigne

2025-04-25

Introduction

L'industrie du jeu vidéo traverse une période compliquée depuis plusieurs années. Les plus grosses entreprises du milieu enchaînent des polémiques entre licencements intensifs, management toxique et jeux très moyens aux yeux de la critique dont les prix ne cessent d'augmenter.

Qu'est ce qu'un jeu AAA ?

C'est un terme utilisé pour classifier les jeux vidéo dotés de budgets de développement et de promotion élevés. Ce sont donc des jeux censés présenter de grandes qualités.

Cependant, les productions AAA font face à plusieurs critiques récurrentes telles que le manque d'innovation, une uniformisation des contenus, ou encore des pratiques commerciales controversées (microtransactions, loot boxes, DLC abusifs).

À l'opposé, les jeux indépendants sont souvent développés par des équipes réduites ou par des développeurs individuels disposant de moyens financiers et techniques limités. Ce contexte leur permet généralement une plus grande liberté créative, donnant naissance à des jeux originaux, audacieux et parfois innovants.

Problématique

Face à ce contexte qui oppose budget et innovation, une question se soulève quant à la qualité des jeux auprès du grand public :

Les jeux indépendants obtiennent-ils de meilleures évaluations que les AAA ?

Chargement et préparation des données

Afin de réaliser l'analyse, on va utiliser les fichiers [raw_reviews.csv](#) et [tags.csv](#) qui seront suffisants. Les deux fichiers sont reliés par app_id, tags.csv va nous permettre de séparer les jeux par le groupe des jeux indépendants et celui des jeux AAA, et raw_reviews.csv va nous permettre d'obtenir les notes attribuées pour chaque jeu.

```
library(dplyr)
library(readr)
library(tidyr)
library(ggplot2)

reviews <- read_csv("data/raw_reviews.csv")
tags <- read_csv("data/tags.csv")
```

Préparation des jeux

Steam possède un tag spécialement pour les jeux indépendants, qui se nomme "Indie". Ce n'est pas le cas pour les jeux AAA. Cependant, on peut facilement les obtenir en récupérant l'ensemble des jeux, puis en enlevant les jeux indépendants :

```
indie_games <- tags %>%
  mutate(app_id = as.character(app_id)) %>%
  filter(tag == "Indie") %>%
  distinct(app_id) %>%
  mutate(type_game = "Indépendant")

aaa_games <- tags %>%
  mutate(app_id = as.character(app_id)) %>%
  anti_join(indie_games, by = "app_id") %>%
  distinct(app_id) %>%
  mutate(type_game = "AAA")

nrow(indie_games)
```

```
## [1] 63253
```

```
nrow(aaa_games)
```

```
## [1] 54252
```

```
types_games <- bind_rows(indie_games, aaa_games)
```

Par la suite, on peut tout mettre dans un nouveau dataset nommé types_games qui recense l'ensemble des jeux avec leur catégorie propre.

Préparation des évaluations

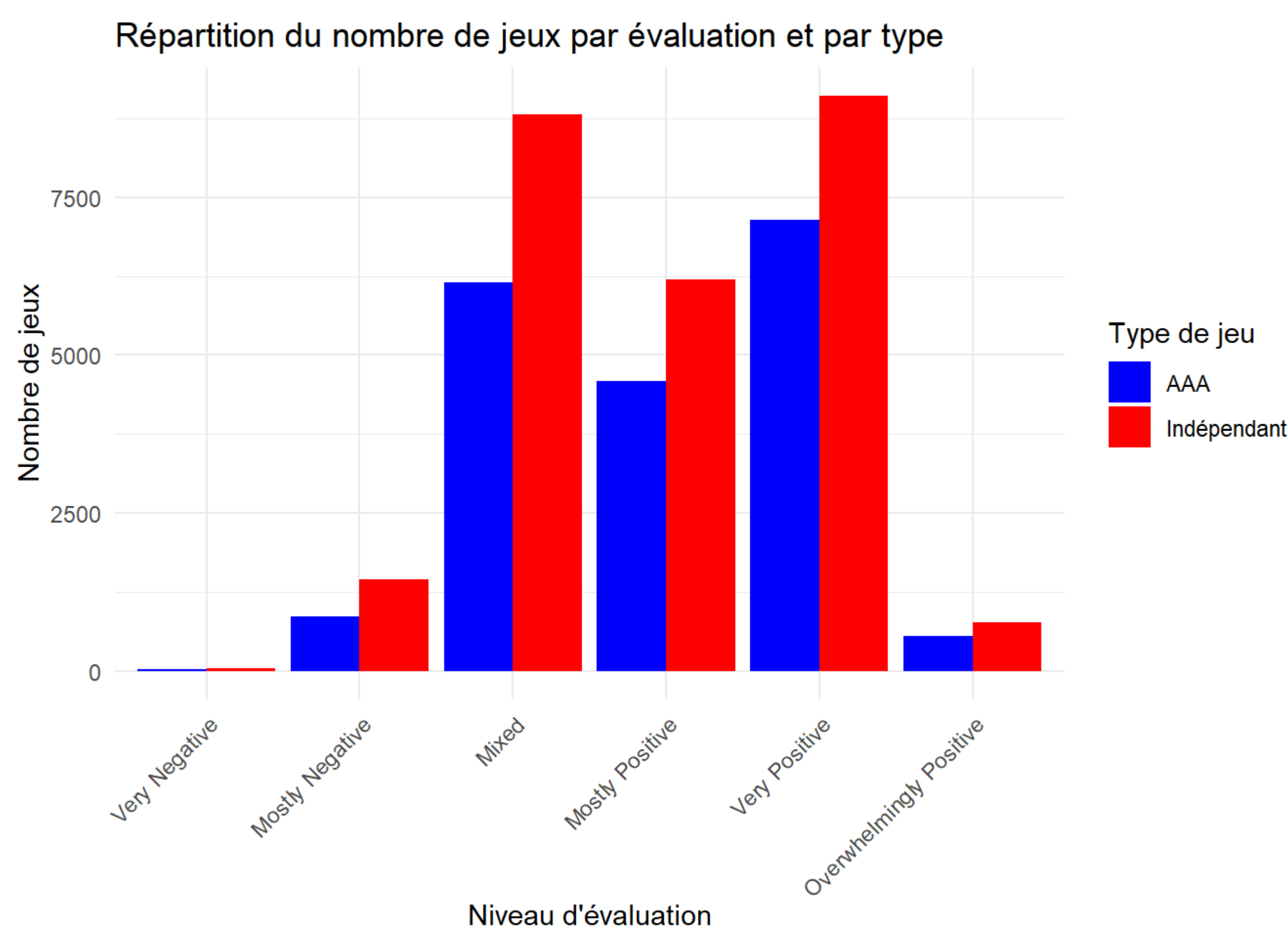
Pour les évaluations, on va créer un nouveau dataset avec seulement les informations dont nous avons besoin, c'est à dire app_id et review_score_description. On va ensuite filtrer review_score_description afin d'obtenir seulement les jeux qui ont une évaluation. La colonne peut avoir d'autres valeurs, comme "None", "1 user reviews" etc... car Steam attribue une évaluation à un jeu seulement à partir d'un certain nombre d'évaluations de la part de la communauté.

```
evaluations <- reviews %>%
  filter(review_score_description != c(
    "Very Negative", "Mostly Negative", "Mixed",
    "Mostly Positive", "Very Positive", "Overwhelmingly Positive"
  )) %>%
  select(app_id, review_score_description) %>%
  inner_join(types_games, by = "app_id")
```

Visualisation

Enfin, pour la visualisation de nos données, on s'est orienté sur un diagramme à barres. L'objectif est de comparer directement les types de jeu selon le niveau d'évaluation.

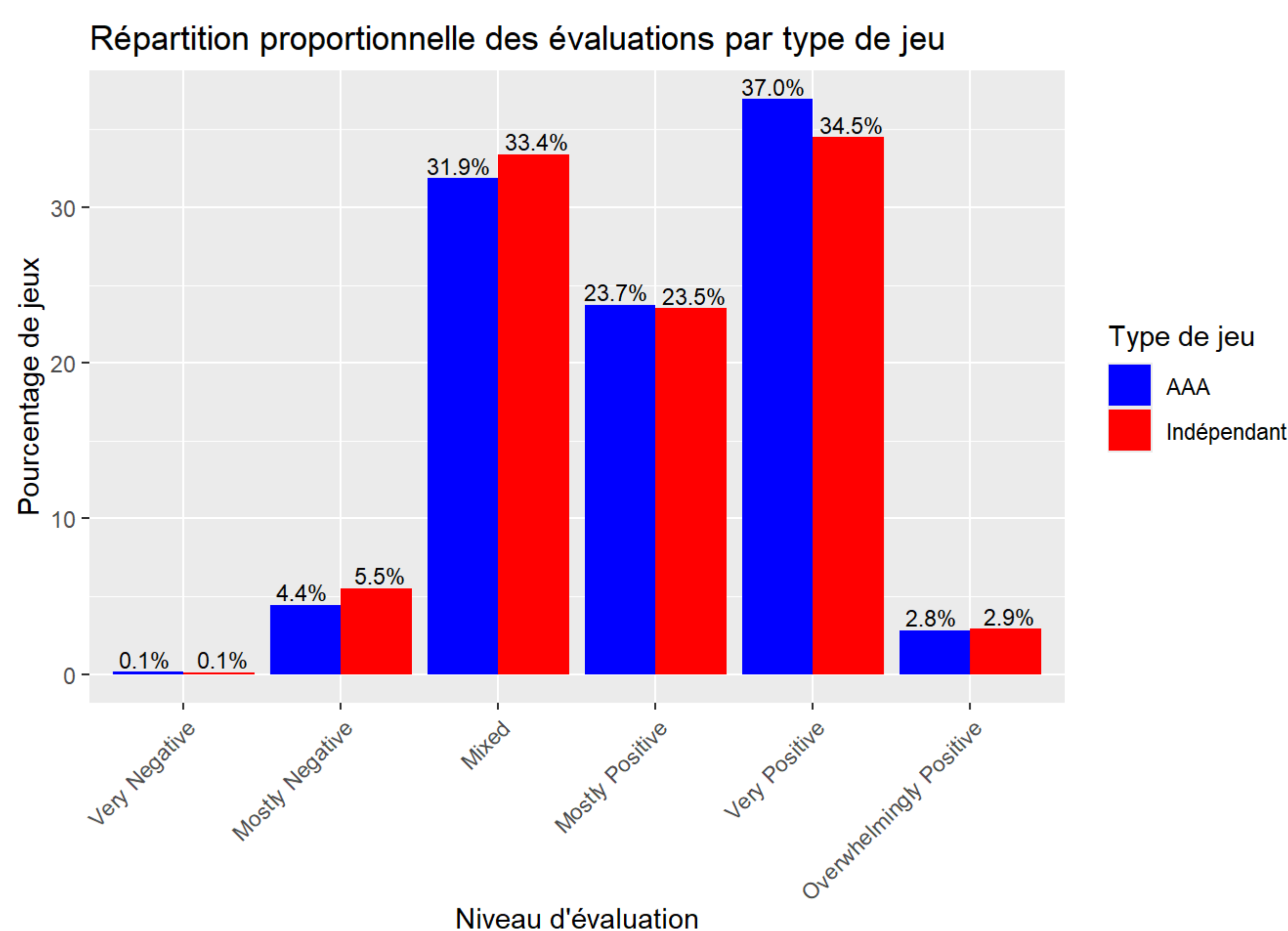
```
evaluations %>%
  mutate(review_score_description = factor(review_score_description, levels = c("Very Negative", "Mostly Negative", "Mixed",
    "Mostly Positive", "Very Positive", "Overwhelmingly Positive"))) %>%
  count(review_score_description, type_game) %>%
  ggplot(aes(x = review_score_description, y = n, fill = type_game)) +
  geom_bar(stat = "identity", position = "dodge") +
  scale_fill_manual(values = c("Indépendant" = "red", "AAA" = "blue")) +
  labs(
    title = "Répartition du nombre de jeux par évaluation et par type",
    x = "Niveau d'évaluation",
    y = "Nombre de jeux",
    fill = "Type de jeu"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Le problème est que ce n'est pas égal de faire la comparaison actuellement. Comme on peut le voir plus au dessus grâce à la fonction nrow, aaa_games possède 54 252 jeux, tandis que indie_games en possède 63 253. On va donc faire une moyenne globale pour être au même niveau.

La fonction geom_bar nous permet de définir l'option "position = dodge" afin d'avoir les barres côte à côte pour une meilleure lisibilité.

```
evaluations %>%
  mutate(review_score_description = factor(review_score_description, levels = c("Very Negative", "Mostly Negative", "Mixed",
    "Mostly Positive", "Very Positive", "Overwhelmingly Positive"))) %>%
  count(review_score_description, type_game) %>%
  group_by(type_game) %>%
  mutate(percentage = n / sum(n) * 100) %>%
  ggplot(aes(x = review_score_description, y = percentage, fill = type_game)) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = sprintf("%.1f%", percentage)),
    position = position_dodge(width = 1),
    vjust = -0.25, size = 3) +
  scale_fill_manual(values = c("Indépendant" = "red", "AAA" = "blue")) +
  labs(
    title = "Répartition proportionnelle des évaluations par type de jeu",
    x = "Niveau d'évaluation",
    y = "Pourcentage de jeux",
    fill = "Type de jeu"
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Observations

L'analyse du graphique montre que, proportionnellement, les jeux AAA obtiennent légèrement plus d'évaluations "Very Positive" que les jeux indépendants. Toutefois, les différences restent relativement faibles et la majorité des jeux indépendants se situent également dans des évaluations globalement positives. Les jeux indépendants affichent une proportion un peu plus élevée d'évaluations "Mixed", ce qui pourrait refléter une plus grande diversité de qualité au sein de cette catégorie. Cela amène à nuancer l'hypothèse initiale : si les jeux indépendants reçoivent beaucoup de bonnes évaluations, les jeux AAA conservent tout de même une légère avance en termes de perception positive moyenne sur Steam.

Cette hypothèse vient du fait que aujourd'hui, les joueurs sont focalisés sur un léger pourcentage de jeux par rapport à l'entièreté du monde du jeu vidéo. Tous les jours de nouveaux jeux sortent, tous les jours il peut y en avoir d'excellents, mais ils ne feront pas le même bruit que les jeux colossaux développés par les grandes entreprises, comme Microsoft, Ubisoft, Activision Blizzard, Rockstar etc...

rapport_analyse_genre

Jules PERRIN
2025-04-27

Contexte général

Steam est une plateforme qui proposent l'un des plus gros catalogue de jeu vidéo du marché. Chaque jeu proposé a plusieurs caractéristique (nom, prix, éditeur , etc). Parmi ces caractéristique, l'une d'entre elle nous intéresse particulièrement dans ce rapport. En effet les tags permettent aux joueur de se faire une idée rapide des jeux proposé ou encore de rechercher de nouveaux jeux selon leur préférence.

Le jeux de donnée que nous allons utiliser ne comporte pas de donnée permettant de dresser une temporalité. Ce rapport portera donc sur les données du catalogue steam e octobre 2024.

Les catégories ne doivent pas être confondu avec les tags. Ici les catégories représentent comment l'on joue au jeu (multiplayer, online pvp, remote play on phone etc). Alors que les tags représentent à quoi l'on joue (survival, tactical, FPS, old school, etc).

Problématique

Ce rapport traitera une question pour le moment:

- Quel est la tendance des jeux sortis récemment ?

Cela nous permettra d'avoir une idée global de la dispersion des joueurs dans les catégories majeurs sur steam pour les nouveaux jeux. Pourrais être utile à un studio pour orienter son futur jeux viéo par exmple.

Analyse

Import des données

Premièrement il faut importer les données nécessaire a cette analyse. Ainsi que charger les librairies requises pour travailler et visualiser les données Les données des fichiers tags, steamspy et games sont nécessaire

```
library(tidyr)
library(scales)
library(lubridate)
library(ggplot2)
library(tidyverse)

steamspy <- read_csv("data/steamspy.csv")
tags <- read_csv("data/tags.csv")
games <- read_delim("data/clean_games.csv", delim=";", show_col_types = FALSE)
head(games)

## # A tibble: 6 × 8
##   app_id type  name                release_date is_free price currency languages
##   <dbl> <chr> <chr>                <date>         <dbl> <dbl> <chr>    <chr>
## 1    10 game Counter-Strike    2000-11-01         0    8.19 EUR    English,...
## 2    20 game Team Fortress Clas... 1999-04-01         0    4.99 EUR    English,...
## 3    30 game Day of Defeat        2003-05-01         0    4.99 EUR    English,...
## 4    40 game Deathmatch Classic  2001-06-01         0    4.99 EUR    English,...
## 5    50 game Half-Life: Opposin... 1999-11-01         0    4.99 EUR    English,...
## 6    60 game Ricochet            2000-11-01         0    4.99 EUR    English,...
```

précision et définition

Ici on cherche a représenter le tendance actuelle. On utilisera donc la population steam présente le jour de la construction du dataset (bien que pas objectivement representatif cf interpretation). On cherche également a se concentré sur les jeux récent. On utilisera donc tous les jeux sorite en 2024

```
games_2024 <- games %>%
  mutate(release_date = ymd(release_date)) %>%
  filter(year(release_date) == 2024) %>%
  select(app_id)

tag_players <- tags %>%
  semi_join(games_2024, by = "app_id") %>%
  left_join(steamspy %>%
    select(app_id, concurrent_users_yesterday),
    by = "app_id") %>%
  group_by(tag) %>%
  summarise(total_ccu = sum(concurrent_users_yesterday, na.rm = TRUE), .groups = "drop") %>%
  arrange(desc(total_ccu)) %>%
  slice_head(n = 10)

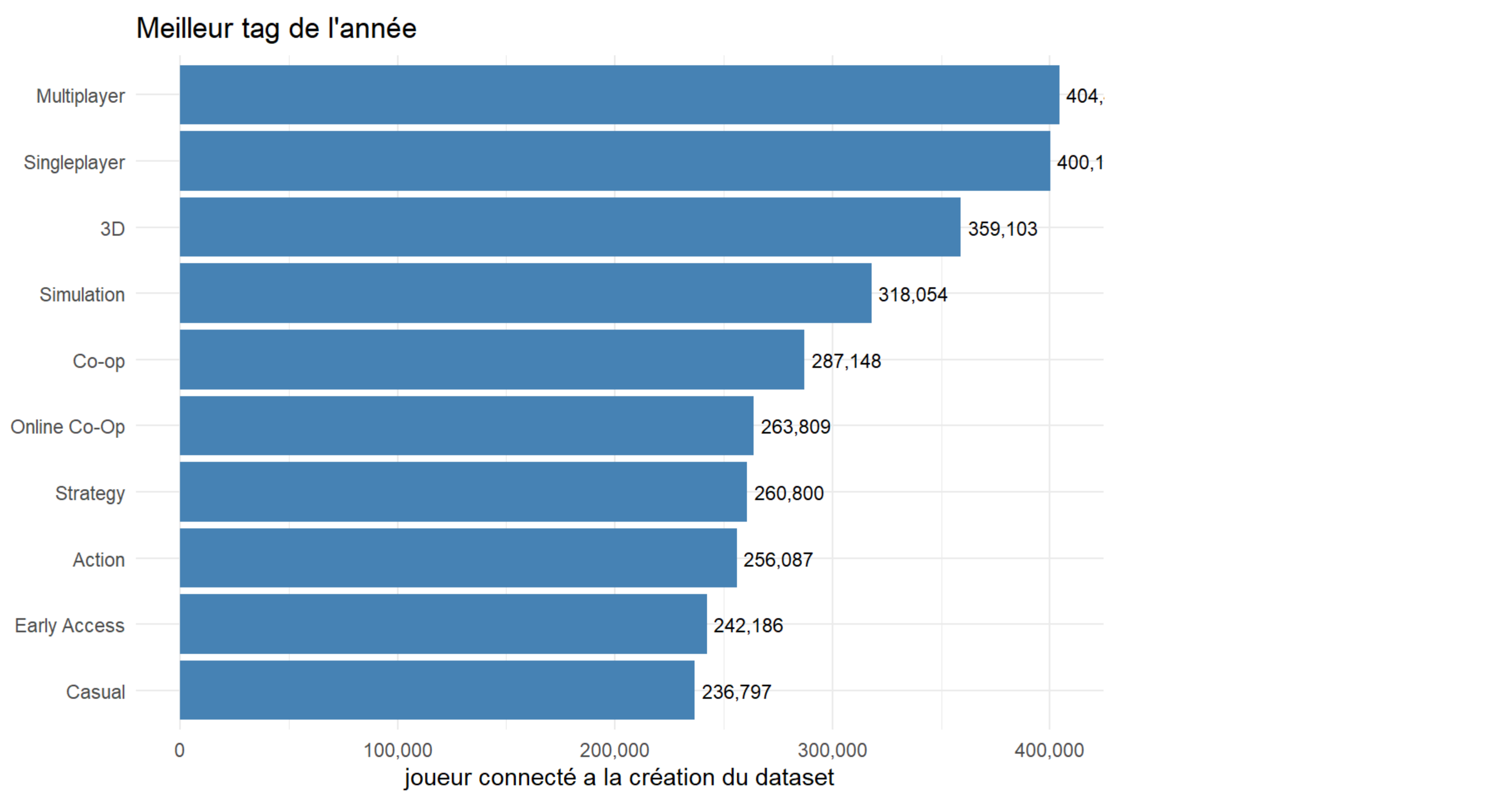
print(tag_players)

## # A tibble: 10 × 2
##   tag          total_ccu
##   <chr>          <dbl>
## 1 Multiplayer    404473
## 2 Singleplayer  400153
## 3 3D             359103
## 4 Simulation    318054
## 5 Co-op          287148
## 6 Online Co-Op  263809
## 7 Strategy      260800
## 8 Action        256087
## 9 Early Access  242186
## 10 Casual       236797
```

Voila donc un apercu des tag les plus populaire selon ce le contexte établi

visualisation

```
ggplot(tag_players,
  aes(x = reorder(tag, total_ccu),
    y = total_ccu)) +
  geom_col(fill = "steelblue") +
  coord_flip() +
  scale_y_continuous(labels = comma) +
  geom_text(aes(label = comma(total_ccu)),
    hjust = -0.1, size = 3) +
  labs(title = "Meilleur tag de l'année",
    x = NULL,
    y = "joueur connecté a la création du dataset") +
  theme_minimal()
```



interpretation

early acces voir quelle jeux c'est multi et single se valle a peu pres co op local met une vitesse a co op online = peur de tomber avec des retardes ?

limites et attentions

cross tag : CS = first person and 1 jour = peu représentatif limité par les donnees