Movie Collections

Tariq HASDI

28 MARS 2019

BIG DATA AND STATISTICS

Le sommaire

Chargement des bibliothèques de base

```
Sys.setenv(PATH=paste(Sys.getenv("PATH"),"C:/Users/Tariq/AppData/Local/Programs/MiKTeX 2.9/miktex/bin/x64/",sep=";"))
knitr::opts_chunk$set(echo = TRUE)

library(dplyr)
library(ggplot2)
library(magrittr)
library(scales)
library(tidyr)
library(rjson)

options(scipen=999)  # turn-off scientific notation like 1e+48
theme_set(theme_bw())  # pre-set the bw theme.

# Read CSV into R
data <- read.csv("movies_metadata.csv", header=TRUE, sep = "," )
##View(data)
```

Introduction

Les films Ces fichiers contiennent des métadonnées pour les 45 000 films répertoriés dans le jeu de données Full MovieLens. L'ensemble de données comprend les films sortis au plus tard en juillet 2017. Les points de données comprennent la distribution, l'équipe de tournage, les mots clés de l'intrigue, le budget, les recettes, les affiches, les dates de sortie, les langues, les sociétés de production, les pays, le nombre de votes TMDB et la moyenne des votes. Cet ensemble de données contient également des fichiers contenant 26 millions d'évaluations de 270 000 utilisateurs pour les 45 000 films. Les évaluations sont sur une échelle de 1 à 5 et ont été obtenues sur le site Web officiel de GroupLens.

Le lien vers le jeu de données https://www.kaggle.com/rounakbanik/the-movies-dataset#movies_metadata.csv

QUESTION:

L'interrogatoire sera présenté dans une perspective d'analyse de données. Nous allons omettre le genre de film afin de faciliter l'étude. Dans quel langue, on a les meilleurs films? Pour donner une réponse, nous devons trouver : Pour quelle langue, nous avons le meilleur vote d'évaluation ? Y a-t-il une corrélation entre le vote_moy et les revenues des films regroupeé par la langue d'origine ?

———————————— Préparation des données ————————————

On supprime les colonnes, qu'on n'a pas besoin pour notre analyse

```
data %>% select(-adult) -> data;
data %>% select(-belongs_to_collection) -> data;
data %>% select(-budget) -> data;
data %>% select(-original_title) -> data;
data %>% select(-production_countries) -> data;
data %>% select(-release_date) -> data;
data %>% select(-homepage) -> data;
data %>% select(-id) -> data;
data %>% select(-imdb_id) -> data;
data %>% select(-overview) -> data;
data %>% select(-poster_path) -> data;
data %>% select(-production_companies) -> data;
data %>% select(-runtime) -> data;
data %>% select(-status) -> data;
data %>% select(-tagline) -> data;
data %>% select(-title) -> data;
data %>% select(-video) -> data;
#data %>% select(-revenue) -> data;
```

Nombre d'objets dans data

```
nbObjet <- nrow(data);
nbObjet;
```

```
## [1] 45466
```

Le pourcentage des objets qui n'ont pas de valeur pour attribut "vote_count"

```
p_vote_count_NA_Values <- sum(is.na(data$vote_count)) / nbObjet;
percent(p_vote_count_NA_Values);

## [1] "0.0132%"

nb_vote_count_NA_Values <- sum(is.na(data$vote_count));
nb_vote_count_NA_Values;

## [1] 6
```

Le pourcentage des objets qui n'ont pas de valeur pour attribut "vote_average"

```
p_vote_average_NA_Values <- sum(is.na(data$vote_average)) / nbObjet;
percent(p_vote_average_NA_Values);

## [1] "0.0132%"

nb_vote_average_NA_Values <- sum(is.na(data$vote_average));
nb_vote_average_NA_Values;

## [1] 6
```

Le pourcentage des objets qui n'ont pas de valeur pour attribut "revenue"

```
p_revenue_NA_Values <- sum(is.na(data$revenue)) / nbObjet;
percent(p_revenue_NA_Values);

## [1] "0.0132%"

nb_revenue_NA_Values <- sum(is.na(data$revenue));
nb_revenue_NA_Values;

## [1] 6
```

Avec seulement 0,0132% des valeurs NA, ces lignes seront supprimées de la trame de données.

```
data %>% filter(!is.na(vote_count)) -> data;
data %>% filter(!is.na(vote_average)) -> data;
```

summary(data$vote_count)

```
summary(data$vote_count);

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     3.0    10.0   109.9    34.0 14075.0
```

Calcul de décile de data$vote_count

```
decile<-quantile(data$vote_count, probs=seq(0, 1, 0.1));
decile;
```

```
##    0%   10%   20%   30%   40%   50%   60%   70%   80%   90%  100%
##     0     1     2     4     6    10    15    25    50   160 14075
```

```
data %>% filter( data$vote_count >= 160 ) -> data;
```

summary(data$vote_count)

```
summary(data$vote_count);
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   160.0   251.0   433.0   923.3   984.0 14075.0
```

summary(data$vote_average)

```
summary(data$vote_average);
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.900   5.900   6.500   6.474   7.100   9.100
```

Calcul de décile de data$vote_count

```
decile<-quantile(data$vote_count, probs=seq(0, 1, 0.1));
decile;
```

```
##      0%     10%     20%     30%     40%     50%     60%     70%     80%
##   160.0   191.4   228.0   281.0   346.0   433.0   575.4   815.8  1236.0
##     90%    100%
##  2181.0 14075.0
```

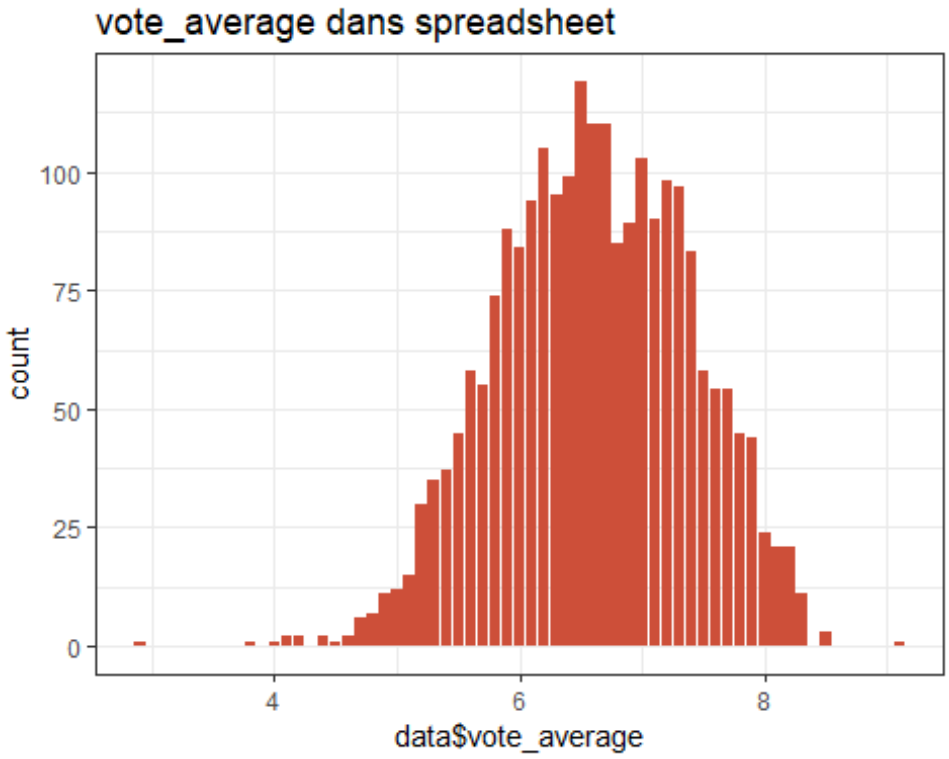
```
data %>% filter( data$vote_count >= 433 ) -> data;
```

———————— DÉTAILS DE BASE ET DE PLOTS ————————————

Le premier Plot “Histograme” on trouve que la variable vote_average est gaussienne

```
ggplot(data=data, aes(data$vote_average)) +
  geom_histogram(stat="count", fill="tomato3") +
  labs(title="vote_average dans spreadsheet")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```



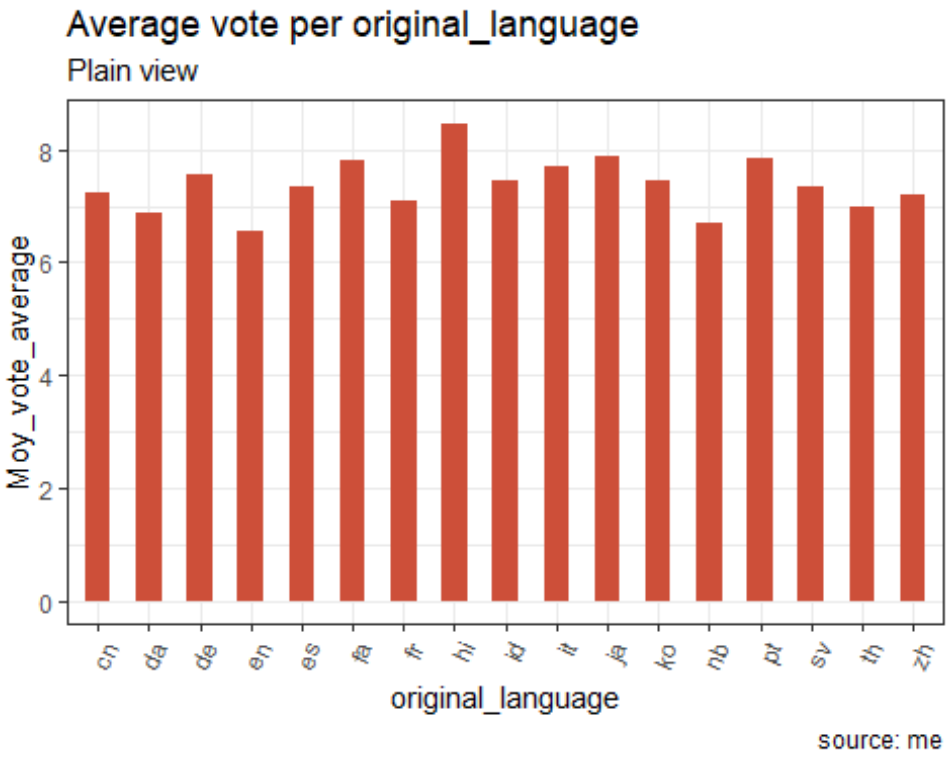
De plus, un group_by est effectué pour récapituler les données par original_language

```
data %>% group_by(original_language) %>% summarize(Moy_vote_average = mean(vo
te_average), Moy_revenue = mean(revenue)) -> origLangVote_average;
head(origLangVote_average);
```

```
## # A tibble: 6 x 3
##   original_language Moy_vote_average Moy_revenue
##   <fct>                        <dbl>       <dbl>
## 1 cn                            7.24   42057747
## 2 da                            6.88   38052861.
## 3 de                            7.55   37899755.
## 4 en                            6.56  183703822.
## 5 es                            7.35   32979018.
## 6 fa                            7.8           0
```

Le Plot suivant affichera le moyen vote_verage par langue originale

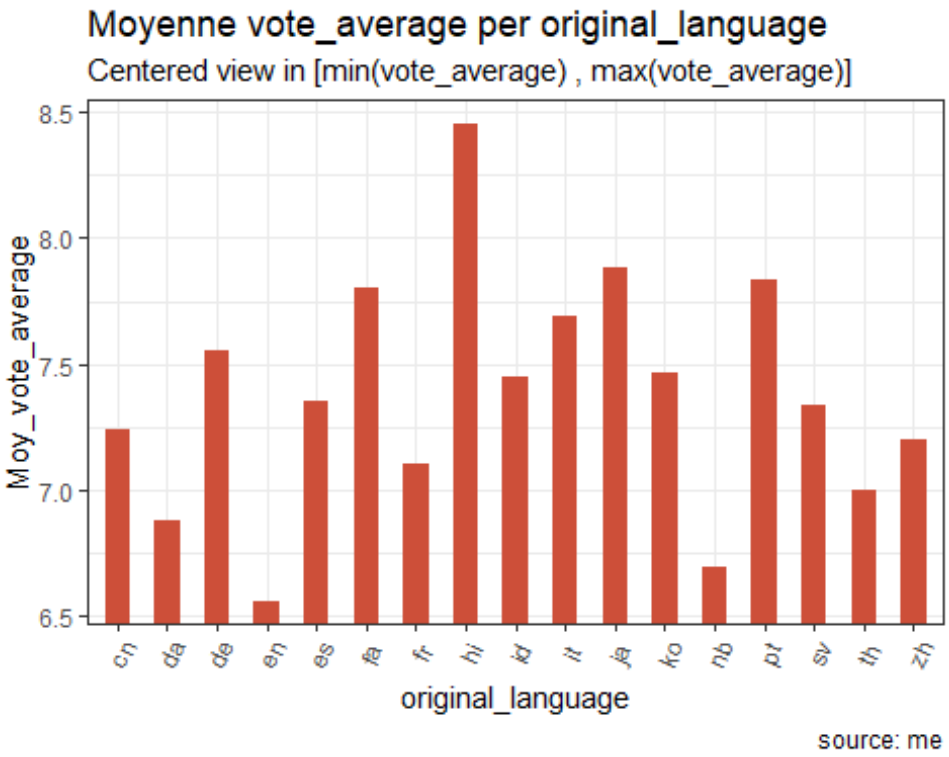
```
ggplot(origLangVote_average, aes(x=original_language, y=Moy_vote_average)) +
  geom_bar(stat="identity", width=0.5, fill="tomato3") +
  labs(title="Average vote per original_language", subtitle="Plain view", cap
tion="source: me") +
  theme(axis.text.x = element_text(angle=65, hjust=1, vjust=1))
```



Nous remarquons que toutes nos valeurs sont proches

Avec 2282 valeurs, nous pouvons entrer dans les détails concernant la moyenne vote_average par original_language

```
ggplot(origLangVote_average, aes(x=original_language, y=Moy_vote_average)) +
  geom_bar(stat="identity", width=0.5, fill="tomato3")  +
  scale_y_continuous(limits=c(min(origLangVote_average$Moy_vote_average),max(
origLangVote_average$Moy_vote_average)),oob = rescale_none) +
  labs(title="Moyenne vote_average per original_language",
       subtitle="Centered view in [min(vote_average) , max(vote_average)]",
       caption="source: me") +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))
```



Premier élément de réponse: Le language_original HINDI est le meilleur par le vote.

```
origLangVote_average %>% filter(Moy_vote_average == max(Moy_vote_average)) ->
bestOriginal_languageVote_average;
bestOriginal_languageVote_average;
```

```
## # A tibble: 1 x 3
##   original_language Moy_vote_average Moy_revenue
##   <fct>                        <dbl>       <dbl>
## 1 hi                            8.45    85000000
```

```
len<-nrow(origLangVote_average);
classement<-seq(1:len);
origLangVote_averageClass <- cbind(classement,origLangVote_average[order(-ori
gLangVote_average$Moy_vote_average),]);
origLangVote_averageClass;
```

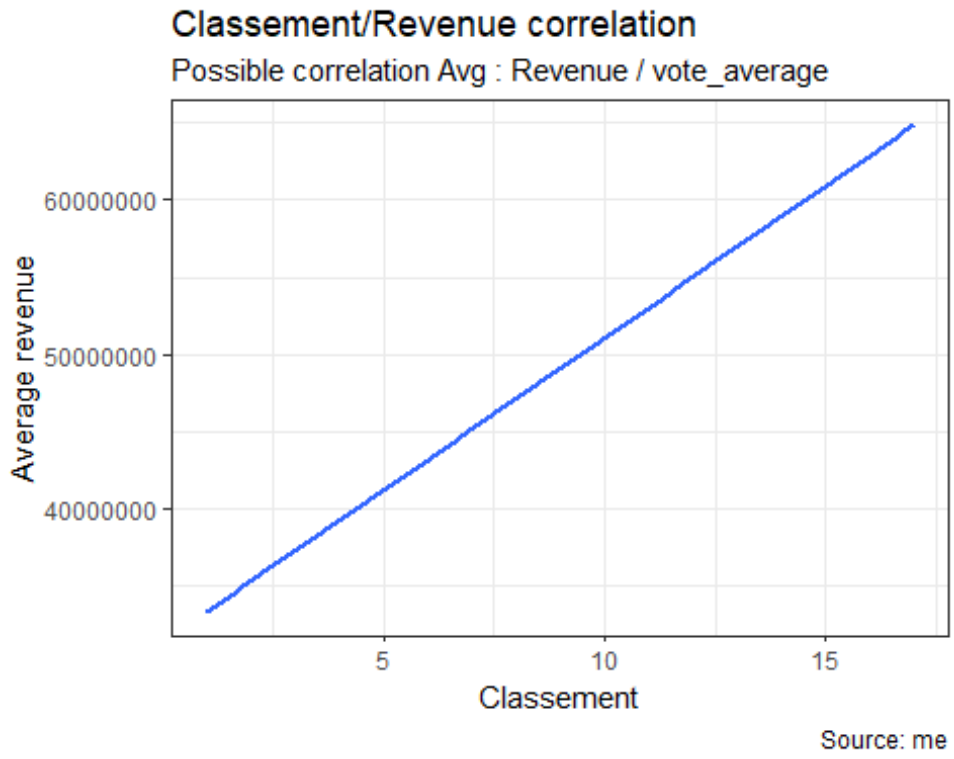
```
##    classement original_language Moy_vote_average Moy_revenue
## 1           1                hi         8.450000    85000000
## 2           2                ja         7.880000    76438773
## 3           3                pt         7.833333    31223150
## 4           4                fa         7.800000           0
## 5           5                it         7.691667    24381916
## 6           6                de         7.550000    37899755
## 7           7                ko         7.466667    32390255
## 8           8                id         7.450000     3366198
## 9           9                es         7.350000    32979018
## 10         10                sv         7.340000    45124969
## 11         11                cn         7.240000    42057747
## 12         12                zh         7.200000   161261371
## 13         13                fr         7.103448    36475215
## 14         14                th         7.000000          15
## 15         15                da         6.883333    38052861
## 16         16                nb         6.700000     4159678
## 17         17                en         6.562223   183703822
```

Corrélation langage_original/MoyRevenue

Le graphique ci-dessous nous permettra d'évaluer une corrélation possible entre langague_original et MoyRevenue.

```
ggplot(origLangVote_averageClass, aes(x=classement, y=Moy_revenue)) +
  #geom_point(aes(size= Moy_vote_count), color="violetred2" ) +
  geom_smooth(method="lm", se=F) +

  labs(subtitle="Possible correlation Avg : Revenue / vote_average",
       y="Average revenue",
       x="Classement",
       title="Classement/Revenue correlation",
       caption = "Source: me")
```



CONCLUSION

En conclusion, l'étude est principalement axée sur la relation entre la langue originale d'un film et les revenues.

Par cette étude, on a pu démontrer qu'il a une corrélation entre les revenues des films et leurs langues d'origines.