SEANCE DATA VISUALISATION

Brun BAHOUN

2025-09-15

```
library(readr)
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(readxl)
library(ggplot2)
library(forcats)
library(lubridate)
```

```
##
## Attachement du package : 'lubridate'
```

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

#_____ PACKAGES _____

readr pour le chargement des fichiers plats et dyplr pour l'Utilisation de Tidyverse pour le traitement de données

```
data_allocine = read_csv2("DATA FICHIER/data_allocine.csv")
```

```
## i Using "','" as decimal and "'.'" as grouping mark. Use `read_delim()` for more control.
```

```
## Rows: 8238 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ";"
## chr  (7): url, titre, genre, nationalite, type_film, couleur, recompenses
```

```
## dbl  (4): id_film, duree, note_presse, note_spectateurs
## lgl  (1): reprise
## date (1): date_sortie
##
## 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.
```

```
correspondances = read_excel("DATA FICHIER/correspondances_allocine.xlsx") %>%
  rename(nationalite = nationalité )
```

Nous avions fait une IMPORTATION de nos DONNEES en la stocket dans un objet data_allocine

```
summary(data_allocine)
```

```
##     id_film           url                titre             date_sortie
##  Min.   :     3   Length:8238        Length:8238        Min.   :1990-01-10
##  1st Qu.: 29962   Class :character   Class :character   1st Qu.:2002-07-18
##  Median :110661   Mode  :character   Mode  :character   Median :2008-08-09
##  Mean   :111553                                         Mean   :2007-06-05
##  3rd Qu.:193525                                         3rd Qu.:2013-04-03
##  Max.   :259212                                         Max.   :2016-12-28
##
##      duree           genre            nationalite         type_film
##  Min.   :  1.0   Length:8238        Length:8238        Length:8238
##  1st Qu.: 88.0   Class :character   Class :character   Class :character
##  Median : 96.0   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 97.3
##  3rd Qu.:106.0
##  Max.   :750.0
##  NA's   :198
##   reprise          couleur           recompenses         note_presse
##  Mode :logical   Length:8238        Length:8238        Min.   :1.000
##  FALSE:7663      Class :character   Class :character   1st Qu.:2.800
##  TRUE :575       Mode  :character   Mode  :character   Median :3.200
##                                                        Mean   :3.195
##                                                        3rd Qu.:3.600
##                                                        Max.   :5.000
##                                                        NA's   :2349
##  note_spectateurs
##  Min.   :0.900
##  1st Qu.:2.700
##  Median :3.100
##  Mean   :3.052
##  3rd Qu.:3.400
##  Max.   :4.700
##  NA's   :990
```

Nous avons donc un petit appercu des variables numériques de notre objet

```
glimpse(data_allocine)
```

```
## Rows: 8,238
```

```
## Columns: 13
## $ id_film          <dbl> 145369, 243983, 111093, 243762, 242046, 251659, 24094~
## $ url              <chr> "http://www.allocine.fr/film/fichefilm_gen_cfilm=1453~
## $ titre            <chr> "A la recherche de Beethoven", "A la recherche de Cho~
## $ date_sortie      <date> 2016-12-01, 2016-12-01, 2016-12-01, 2016-12-01, 2016~
## $ duree            <dbl> 139, 115, 128, 102, 115, 52, 118, 90, 102, 128, 90, 8~
## $ genre            <chr> "Documentaire", "Documentaire", "Documentaire", "Docu~
## $ nationalite      <chr> "britannique", "britannique", "britannique", "britann~
## $ type_film        <chr> "Long-métrage", "Long-métrage", "Long-métrage", "Long~
## $ reprise          <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALS~
## $ couleur          <chr> "Couleur", "Couleur", "Couleur", "Couleur", "Couleur"~
## $ recompenses      <chr> NA, NA, NA, NA, NA, NA, "2 prix", NA, NA, "1 prix et ~
## $ note_presse      <dbl> NA, NA, NA, NA, NA, NA, 3.2, 3.3, 2.5, 3.9, 3.1, 2.8,~
## $ note_spectateurs <dbl> NA, NA, NA, NA, 2.9, NA, 4.3, 3.7, 2.2, 3.4, 1.8, 3.1~
```

Visualisation de la Structure des données. Nous pouvions également utilisé str pour cette visualisation

```
head(data_allocine, 10)
```

```
## # A tibble: 10 x 13
##    id_film url       titre date_sortie duree genre nationalite type_film reprise
##      <dbl> <chr>     <chr> <date>      <dbl> <chr> <chr>       <chr>     <lgl>
##  1  145369 http://w~ A la~ 2016-12-01    139 Docu~ britannique Long-mét~ FALSE
##  2  243983 http://w~ A la~ 2016-12-01    115 Docu~ britannique Long-mét~ FALSE
##  3  111093 http://w~ A la~ 2016-12-01    128 Docu~ britannique Long-mét~ FALSE
##  4  243762 http://w~ A la~ 2016-12-01    102 Docu~ britannique Long-mét~ FALSE
##  5  242046 http://w~ L'As~ 2016-12-03    115 Poli~ britannique Long-mét~ FALSE
##  6  251659 http://w~ Jésu~ 2016-12-06     52 Docu~ français    Long-mét~ FALSE
##  7  240944 http://w~ Dema~ 2016-12-07    118 Comé~ français    Long-mét~ FALSE
##  8  239764 http://w~ Papa~ 2016-12-07     90 Comé~ français    Long-mét~ FALSE
##  9  239043 http://w~ Sex ~ 2016-12-07    102 Drame français    Long-mét~ FALSE
## 10  241700 http://w~ Bacc~ 2016-12-07    128 Drame roumain     Long-mét~ FALSE
## # i 4 more variables: couleur <chr>, recompenses <chr>, note_presse <dbl>,
## #   note_spectateurs <dbl>
```

Affichage des x premiers entetes de notre objet

```
class(data_allocine)
```

```
## [1] "spec_tbl_df" "tbl_df"      "tbl"         "data.frame"
```

La connaissance de la classe de l'objet a donc été donné par cette console

```
count(data_allocine, nationalite)
```

```
## # A tibble: 34 x 2
##    nationalite     n
##    <chr>       <int>
##  1 allemand      277
##  2 autrichien     57
##  3 belge         154
```

```
##  4 bosniaque       8
##  5 britannique   740
##  6 bulgare        11
##  7 croate          5
##  8 danois         77
##  9 espagnol      301
## 10 estonien        6
## # i 24 more rows
```

Remarquons donc que nous visualisons le nombre de films par nationalité

```
data_sans_recompenses = select(data_allocine, -recompenses)
```

Nous avions procédé à une suppresion de la colonne recompense de notre objet tout en stockant le nouveau objet obtenu dans une variable. SELECT a été utilisé pour la sélection en colonnes

```
data_sans_recompenses = filter(data_sans_recompenses, type_film == "Long-métrage")
count(data_sans_recompenses, type_film)
```

```
## # A tibble: 1 x 2
##   type_film        n
##   <chr>        <int>
## 1 Long-métrage  7978
```

Utilisation de FILTRER : pour sélectionner des lignes via une condition et nous avons procéder à la condition suivante qui est de Ne garder que les longs métrages dans data_sans_recompenses puis après avons posséder à une vérification pour vérifier la commande exécutée

```
data_sans_recompenses = rename(data_sans_recompenses, titre_film = titre)
glimpse(data_sans_recompenses)
```

```
## Rows: 7,978
## Columns: 12
## $ id_film          <dbl> 145369, 243983, 111093, 243762, 242046, 251659, 24094~
## $ url              <chr> "http://www.allocine.fr/film/fichefilm_gen_cfilm=1453~
## $ titre_film       <chr> "A la recherche de Beethoven", "A la recherche de Cho~
## $ date_sortie      <date> 2016-12-01, 2016-12-01, 2016-12-01, 2016-12-01, 2016~
## $ duree            <dbl> 139, 115, 128, 102, 115, 52, 118, 90, 102, 128, 90, 8~
## $ genre            <chr> "Documentaire", "Documentaire", "Documentaire", "Docu~
## $ nationalite      <chr> "britannique", "britannique", "britannique", "britann~
## $ type_film        <chr> "Long-métrage", "Long-métrage", "Long-métrage", "Long~
## $ reprise          <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALS~
## $ couleur          <chr> "Couleur", "Couleur", "Couleur", "Couleur", "Couleur"~
## $ note_presse      <dbl> NA, NA, NA, NA, NA, NA, 3.2, 3.3, 2.5, 3.9, 3.1, 2.8,~
## $ note_spectateurs <dbl> NA, NA, NA, NA, 2.9, NA, 4.3, 3.7, 2.2, 3.4, 1.8, 3.1~
```

RENAME : utiliser pour renommer une variable et donc nous voulons renommer la variable titre en titre_film puis nous avons procéder à la vérification de notre commande

```
data_sans_recompenses = arrange(data_sans_recompenses, id_film)
head(data_sans_recompenses)
```

```
## # A tibble: 6 x 12
##   id_film url   titre_film date_sortie duree genre nationalite type_film reprise
##     <dbl> <chr> <chr>      <date>      <dbl> <chr> <chr>       <chr>     <lgl>
## 1       3 http~ Fanfan la~ 2003-05-21    102 Aven~ français    Long-mét~ TRUE
## 2      14 http~ Il Bidone  2009-10-18    112 Comé~ italien     Long-mét~ TRUE
## 3      29 http~ À bout de~ 2010-06-16     89 Poli~ français    Long-mét~ TRUE
## 4      34 http~ Accident   2007-04-22    105 Drame britannique Long-mét~ TRUE
## 5      39 http~ Adieu Phi~ 2005-09-25    108 Comé~ français    Long-mét~ TRUE
## 6      45 http~ Affreux, ~ 2009-07-15    115 Comé~ italien     Long-mét~ TRUE
## # i 3 more variables: couleur <chr>, note_presse <dbl>, note_spectateurs <dbl>
```

ARRANGE :utiliser pour trier un dataframe / tibble sur une ou des colonne(s) ainsi nous avons trier data_sans_recompenses selon l'id_film puis avons procéder à une vérification

```
data_allocine_italien = filter(data_sans_recompenses, nationalite == "italien")
data_allocine_italien = arrange(data_allocine_italien, desc(duree))
head(data_allocine_italien, 5)
```

```
## # A tibble: 5 x 12
##   id_film url   titre_film date_sortie duree genre nationalite type_film reprise
##     <dbl> <chr> <chr>      <date>      <dbl> <chr> <chr>       <chr>     <lgl>
## 1   52597 http~ Nos meill~ 2003-07-09    366 Drame italien     Long-mét~ FALSE
## 2    2990 http~ La Chine   2009-04-13    210 Docu~ italien     Long-mét~ TRUE
## 3  142817 http~ Sonetàula  2009-09-09    163 Drame italien     Long-mét~ FALSE
## 4   38714 http~ Carmen     2011-06-09    152 Drame italien     Long-mét~ TRUE
## 5   61263 http~ Romanzo c~ 2006-03-22    152 Poli~ italien     Long-mét~ FALSE
## # i 3 more variables: couleur <chr>, note_presse <dbl>, note_spectateurs <dbl>
```

Nous avions faire un ENCHAINEMENT : Lister les 5 films italiens les plus longs sans utilisation du pipe

```
data_sans_recompenses %>%
      filter(nationalite == "italien") %>%
      arrange(desc(duree)) %>%
      head(5)
```

```
## # A tibble: 5 x 12
##   id_film url   titre_film date_sortie duree genre nationalite type_film reprise
##     <dbl> <chr> <chr>      <date>      <dbl> <chr> <chr>       <chr>     <lgl>
## 1   52597 http~ Nos meill~ 2003-07-09    366 Drame italien     Long-mét~ FALSE
## 2    2990 http~ La Chine   2009-04-13    210 Docu~ italien     Long-mét~ TRUE
## 3  142817 http~ Sonetàula  2009-09-09    163 Drame italien     Long-mét~ FALSE
## 4   38714 http~ Carmen     2011-06-09    152 Drame italien     Long-mét~ TRUE
## 5   61263 http~ Romanzo c~ 2006-03-22    152 Poli~ italien     Long-mét~ FALSE
## # i 3 more variables: couleur <chr>, note_presse <dbl>, note_spectateurs <dbl>
```

Ici nous avions utilisé le pipe pour lister les 5 films italiens les plus longs

```
data_sans_recompenses %>%
  filter(nationalite == "français" & note_presse >= 4 ) %>%
  arrange(note_presse) %>%
  select(titre_film,note_presse) %>%
  write_excel_csv2("Liste_bons_films_fr.csv")
```

Exercice consistant à mettre dans un fichier csv la liste des films français dont la note est supérieure à 4 et dont la colonne note est en ordre croissant tout en conservant la note et le titre du film

```
data_sans_recompenses = data_sans_recompenses %>%
      left_join(correspondances, by = "nationalite")
glimpse(data_sans_recompenses)
```

```
## Rows: 7,978
## Columns: 14
## $ id_film          <dbl> 3, 14, 29, 34, 39, 45, 56, 61, 62, 82, 83, 122, 126, ~
## $ url              <chr> "http://www.allocine.fr/film/fichefilm_gen_cfilm=3.ht~
## $ titre_film       <chr> "Fanfan la Tulipe", "Il Bidone", "À bout de souffle",~
## $ date_sortie      <date> 2003-05-21, 2009-10-18, 2010-06-16, 2007-04-22, 2005~
## $ duree            <dbl> 102, 112, 89, 105, 108, 115, 71, 139, 116, 124, 94, 1~
## $ genre            <chr> "Aventure", "Comédie dramatique", "Policier", "Drame"~
## $ nationalite      <chr> "français", "italien", "français", "britannique", "fr~
## $ type_film        <chr> "Long-métrage", "Long-métrage", "Long-métrage", "Long~
## $ reprise          <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE,~
## $ couleur          <chr> "Couleur", "N&B", "N&B", "Couleur", "N&B", "Couleur",~
## $ note_presse      <dbl> NA, NA, 5.0, NA, 5.0, 5.0, 4.0, 4.0, 4.4, NA, NA, 4.6~
## $ note_spectateurs <dbl> 2.8, 3.8, 3.8, 3.4, 3.3, 3.8, 3.2, 3.5, 4.3, 3.6, 3.4~
## $ pays             <chr> "France", "Italie", "France", "Royaume-Uni", "France"~
## $ region           <chr> "Europe de l'ouest", "Europe de l'ouest", "Europe de ~
```

Nous allons Enrichir data_sans_recompenses avec les colonnes de correpsondances_allocine en faisant une jointure de gauche ainsi nous avons procédé au changement de notre objet

```
data_sans_recompenses %>%
  filter(genre == "Drame") %>%
  count(region) %>%
  arrange(desc(n)) %>%
  filter(!is.na(region))
```

```
## # A tibble: 3 x 2
##   region                n
##   <chr>             <int>
## 1 Europe de l'ouest  2395
## 2 Europe de l'est     139
## 3 Scandinavie          69
```

COMBIEN DE FILMS de drame EN EUROPE DE L'OUEST OU DE L'EST ? Telle est la question à laquelle nous voulons répondre en exécutant cette commande

```
data_sans_recompenses %>%
 summarize(
  nbre_films = n(), # comptage du nombre de lignes
  moyenne_presse = mean(note_presse, na.rm = TRUE),
  moyenne_spectateurs = mean(note_spectateurs, na.rm = TRUE)
 )
```

```
## # A tibble: 1 x 3
##   nbre_films moyenne_presse moyenne_spectateurs
##        <int>          <dbl>               <dbl>
## 1       7978           3.19                3.05
```

Utilisation de summarize en faisant leCalcul de la moyenne des notes presse et la moyenne des notes spectateurs

```
    data_sans_recompenses %>%
      group_by(region) %>%
      summarize(
        nbre_films = n(), # comptage du nombre de lignes
        moyenne_presse = mean(note_presse, na.rm = TRUE),
        moyenne_spectateurs = mean(note_spectateurs, na.rm = TRUE)
      )
```

```
## # A tibble: 4 x 4
##   region            nbre_films moyenne_presse moyenne_spectateurs
##   <chr>                  <int>          <dbl>               <dbl>
## 1 Europe de l'est          295           3.38                3.18
## 2 Europe de l'ouest       7524           3.18                3.04
## 3 Scandinavie              153           3.19                3.31
## 4 <NA>                       6           4.75                3.48
```

GROUP BUY : en lien avec summarize, calculs par groupe. refaire les memes calculs pendant l'utilisation de summarize comme la précédente , mais selon la région

```
data_sans_recompenses %>%
  filter(region == "Europe de l'ouest") %>%
  group_by(genre) %>%
  summarize(
    n = n(),
    duree_moyenne = mean(duree, na.rm = TRUE)
  ) %>%
  arrange(desc(n)) %>%
  head(5)
```

```
## # A tibble: 5 x 3
##   genre                   n duree_moyenne
##   <chr>               <int>         <dbl>
## 1 Drame                2395         103.
## 2 Comédie              1377          95.1
## 3 Comédie dramatique   1147         100.
## 4 Documentaire         1084          91.9
## 5 Autres                434         108.
```

Calculer sur les films de l'europe de l'ouest la durée moyenne et le nombre de films par genre puis afficher les 5 genres ayant le plus de duree moyenne

```
data_sans_recompenses = data_sans_recompenses %>%
  mutate(
    note_totale = note_presse + note_spectateurs
  )

data_sans_recompenses %>% summarize(
  note_totale_moyenne = mean(note_totale, na.rm = TRUE)
)
```

```
## # A tibble: 1 x 1
##   note_totale_moyenne
##                 <dbl>
## 1                6.22
```

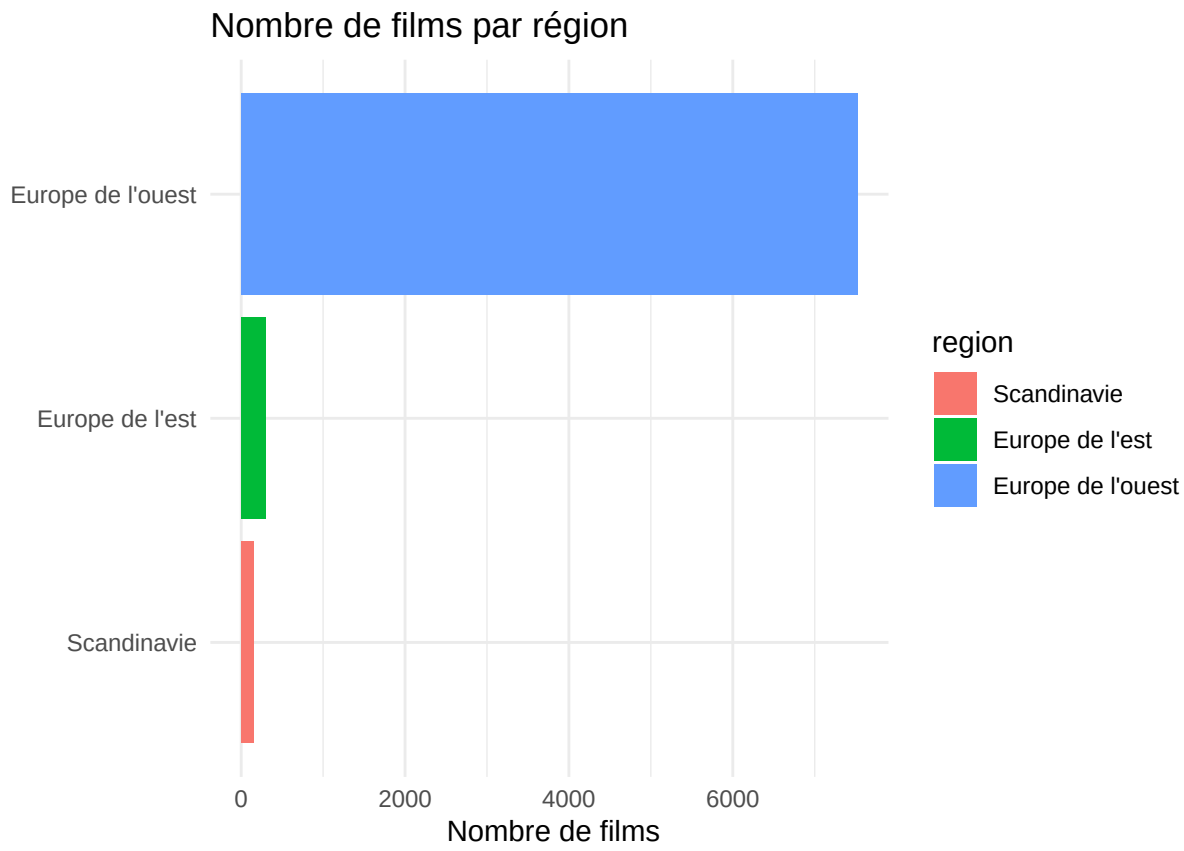
MUTATE pour créer ou mette à jou rune colonne d'un dataframe ainsi nous exécutons la commande suiavante pour créer une colonne note_totale qui est la somme de presse et spectateurs

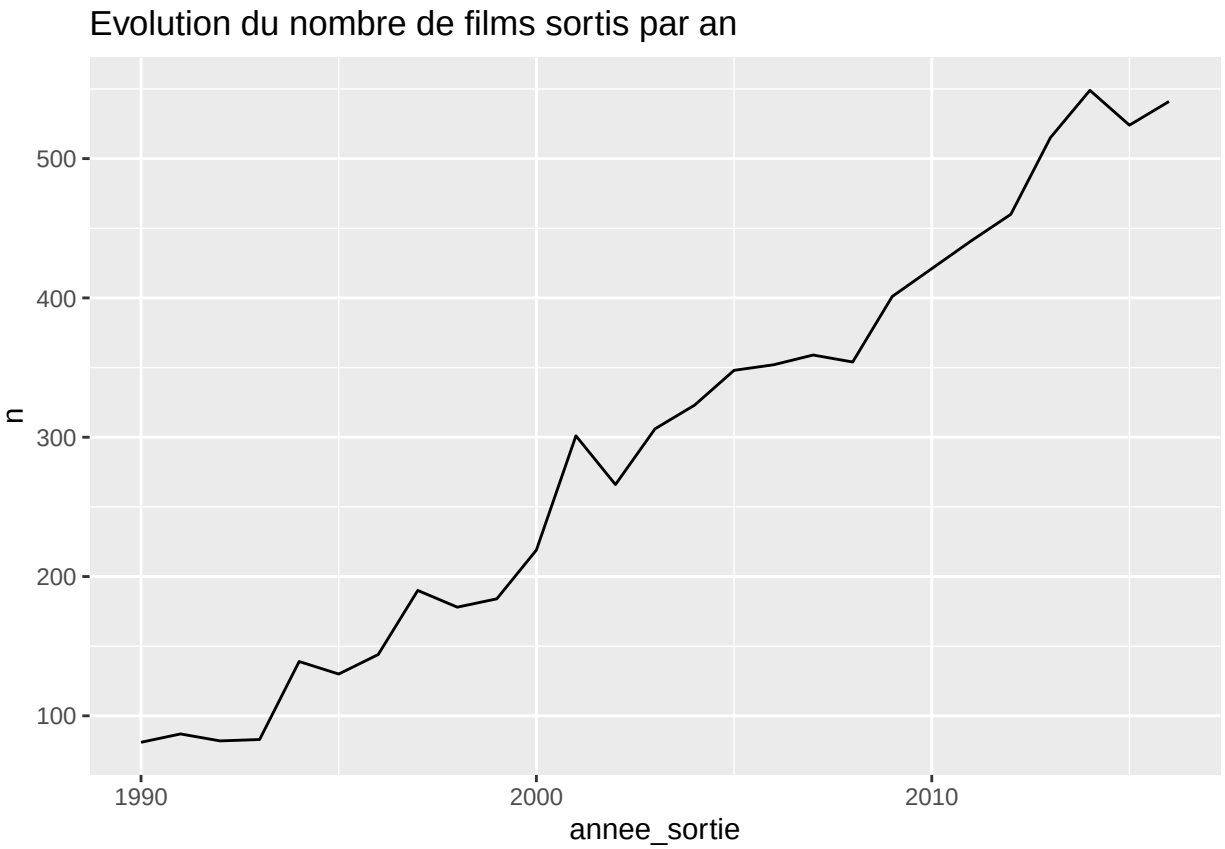
```
data_sans_recompenses = data_sans_recompenses %>%
  mutate(
    tr_note = if_else( is.na(note_totale), "Pas de note",
                       if_else(note_totale < 5, "Mauvais film",
                       if_else(note_totale < 7, "Moyen Film",
                               "Bon Film")
    ))
  )

count(data_sans_recompenses, tr_note)
```

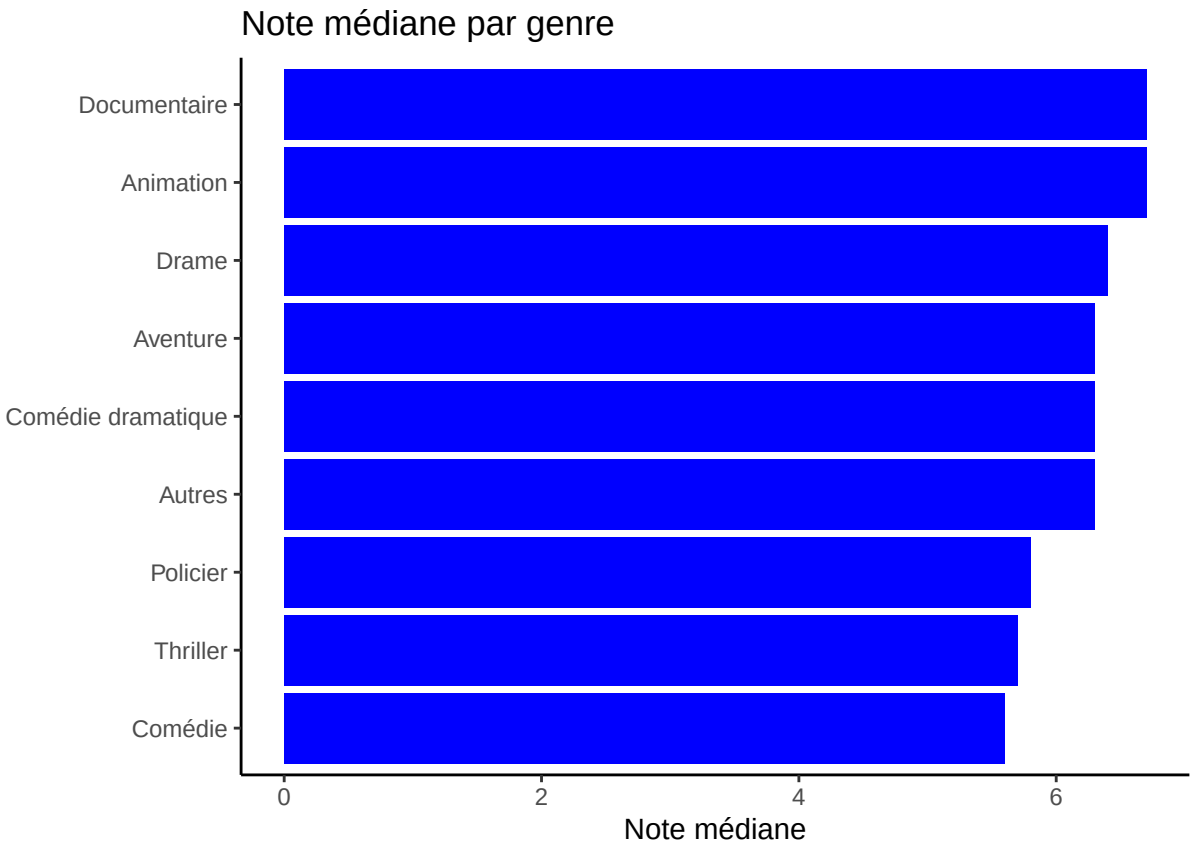
```
## # A tibble: 4 x 2
##   tr_note          n
##   <chr>        <int>
## 1 Bon Film      1410
## 2 Mauvais film   723
## 3 Moyen Film    3676
## 4 Pas de note   2169
```

Utilisation de MUTATE AVEC IFELSE qui permet de créer une variable d'un dataframe sous condition ainsi nous allons procéder à la création d'une tr_note basée sur la note totale : moins de 5, entre 5 et 7, plus de 7

#_____ VISUALISATION _____


Nombre de films par région
Europe de l'ouest
Europe de l'est
Scandinavie
0
2000
4000
6000
Nombre de films
region
Scandinavie
Europe de l'est
Europe de l'ouest


Evolution du nombre de films sortis par an
n
500
400
300
200
100
1990
2000
2010
annee_sortie


Note médiane par genre
Documentaire
Animation
Drame
Aventure
Comédie dramatique
Autres
Policier
Thriller
Comédie
0
2
4
6
Note médiane