Netflixdata

Patara P.

2022-09-29

Netflix Movies and TV shows Data

This markdown files contains analysis about Movies and TV shows added to Netflix streaming platform.

Data Source : [kaggle]https://www.kaggle.com/datasets/shivamb/netflix-shows Credit to : SHIVAM BANSAL

This analysis is use for training purpose.

First, Install and import Tidyverse library

```
install.packages('tidyverse', repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/c6/r9mjgh5x7w7cdf7_7b1nbkbw0000gn/T//RtmpmcLqXb/downloaded_packages
```

```
install.packages('lubridate', repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/c6/r9mjgh5x7w7cdf7_7b1nbkbw0000gn/T//RtmpmcLqXb/downloaded_packages
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

Import Data Using readr package

```
netflix <- read_csv("netflix_titles.csv")
```

```
## Rows: 8807 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (11): show_id, type, title, director, cast, country, date_added, rating,...
## dbl  (1): release_year
##
## 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.
```

Review Data

```
head(netflix)
```

```
## # A tibble: 6 x 12
##   show_id type    title    director cast  country date_added release_year rating
##   <chr>   <chr>   <chr>    <chr>    <chr> <chr>   <chr>             <dbl> <chr>
## 1 s1      Movie   Dick Jo~ Kirsten~ <NA>  United~ September~         2020 PG-13
## 2 s2      TV Show Blood &~ <NA>     Ama ~ South ~ September~         2021 TV-MA
## 3 s3      TV Show Ganglan~ Julien ~ Sami~ <NA>    September~         2021 TV-MA
## 4 s4      TV Show Jailbir~ <NA>     <NA>  <NA>    September~         2021 TV-MA
## 5 s5      TV Show Kota Fa~ <NA>     Mayu~ India   September~         2021 TV-MA
## 6 s6      TV Show Midnigh~ Mike Fl~ Kate~ <NA>    September~         2021 TV-MA
## # ... with 3 more variables: duration <chr>, listed_in <chr>, description <chr>
```

```
colnames(netflix)
```

```
##  [1] "show_id"      "type"         "title"        "director"     "cast"
##  [6] "country"      "date_added"   "release_year" "rating"       "duration"
## [11] "listed_in"    "description"
```

Check if there are any duplicated data

```
sum(duplicated(netflix))
```

```
## [1] 0
```

Check Missing Values in Column we going to use

Since after we reviewed the data we found that there are some missing value in several column, so we will check if there are any missing value in column that we going to use.

```
sum(is.na(netflix$country))
```

```
## [1] 831
```

```
sum(is.na(netflix$duration))
```

```
## [1] 3
```

```
sum(is.na(netflix$date_added))
```

```
## [1] 10
```

```
sum(is.na(netflix$release_year))
```

```
## [1] 0
```

Drop rows of Data that are missing

Since we will use 4 column mainly in this analysis so we will drop only 4 column so it will not affect most of data.

```
netflix_new <- netflix %>%  drop_na(country,duration,date_added,release_year)
```

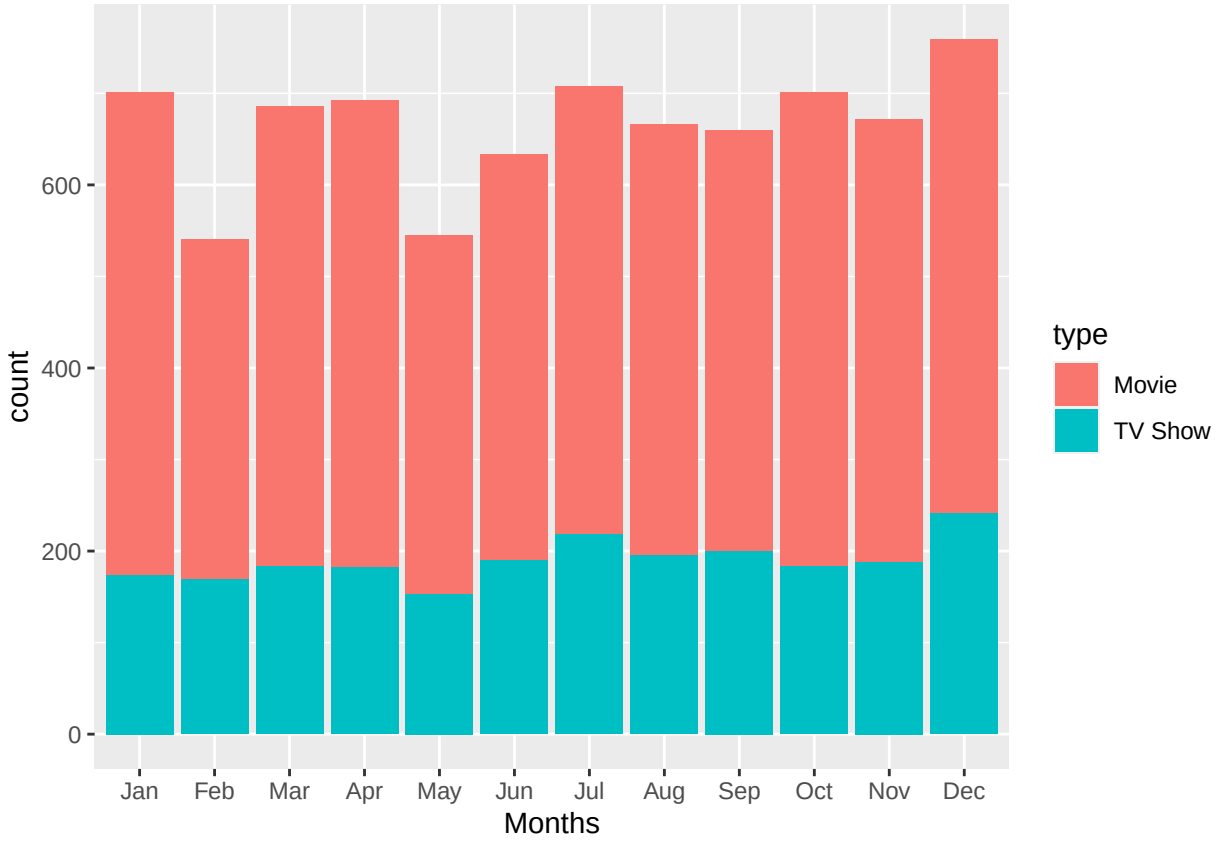
Format and Mutate Months column.

Next, After format and mutate new column i put those data in temporary data frame so i can use this further.

```
df1 <- netflix_new %>%  mutate(formated_date = mdy(date_added)) %>%
  mutate(month_added = month.abb[month(formated_date)]) %>%
  mutate(month_added = factor(month_added ,
  levels = c("Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov","Dec"))) %>%
  subset(select = -formated_date)
```

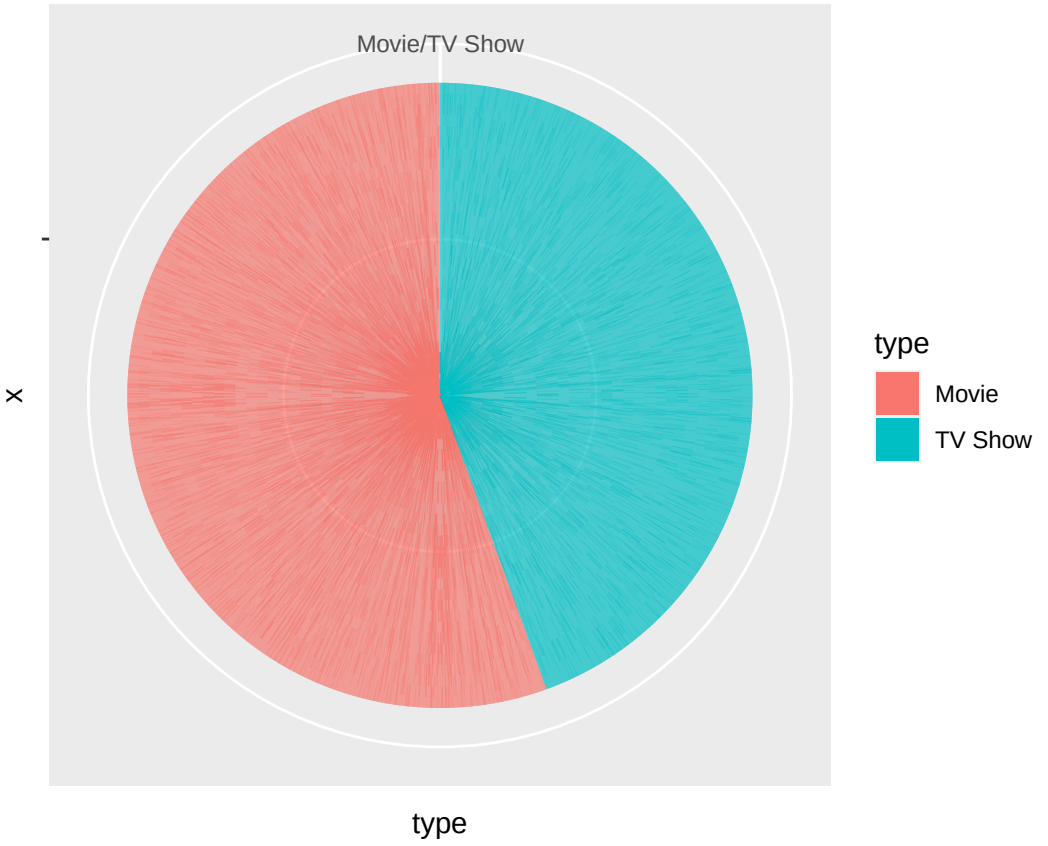
Plot Amount of Movies and TV shows added in each month

```
ggplot(df1,aes(x=month_added , fill = type )) + geom_bar() + xlab("Months")
```



Plot propotion of total Movies and TV shows

```
ggplot(df1, aes(x= "",y = type , fill = type)) + geom_col() + coord_polar(theta = "y")
```



Next we will find Average Movie duration categorized by Genres (Action,Comedy,Drama,Scifi)

I started with cleaning duration column before we categorized by removing "min" word and white spaces and put those data in new data frame call movies.

```
movies <- df1 %>%  filter(type == "Movie") %>%
  mutate(duration1 = str_remove_all(duration,"min")) %>%
  mutate(duration_min = str_trim(duration1)) %>%
  subset(select = - duration) %>%
  subset(select = - duration1)
```

##Action

```
action_movies <- movies %>% filter(str_detect(listed_in,"Action"))
act_avg <- mean(as.integer(action_movies$duration_min))
```

Comedy

```
comedy_movies <- movies %>% filter(str_detect(listed_in,"Comedy"))
com_avg <- mean(as.integer(comedy_movies$duration_min))
```

##Drama

```
drama_movies <- movies %>% filter(str_detect(listed_in,"Drama"))
dra_avg <- mean(as.integer(drama_movies$duration_min))
```

##Scifi

```
scifi_movies <- movies %>% filter(str_detect(listed_in,"Sci"))
sci_avg <- mean(as.integer(scifi_movies$duration_min))
```

Create new Data frame

```
genre <- c("Action","Comedy","Drama","Scifi")
avg_dur <- c(act_avg,com_avg,dra_avg,sci_avg)

average_duration <- data.frame(genre,avg_dur)
```

##Plot Average duration chart categorized by genre

```
ggplot(data = average_duration, aes(x=genre , y=avg_dur , fill = genre)) +
  geom_bar(stat='identity')
```