Netflix analysis

Tamar Saad & Or Arbel & Rachel Weinberger

4/3/2022

In this code we received an excel file of the Netflix data set.

We created a set of plots in order to analyze the given data and to learn about it.

Libraries uploading:

```
library(plyr)
library(dplyr)
library(stringr)
library(gplots)
library(ggplot2)
library(forcats)
library(data.table)
library(reshape2)
```

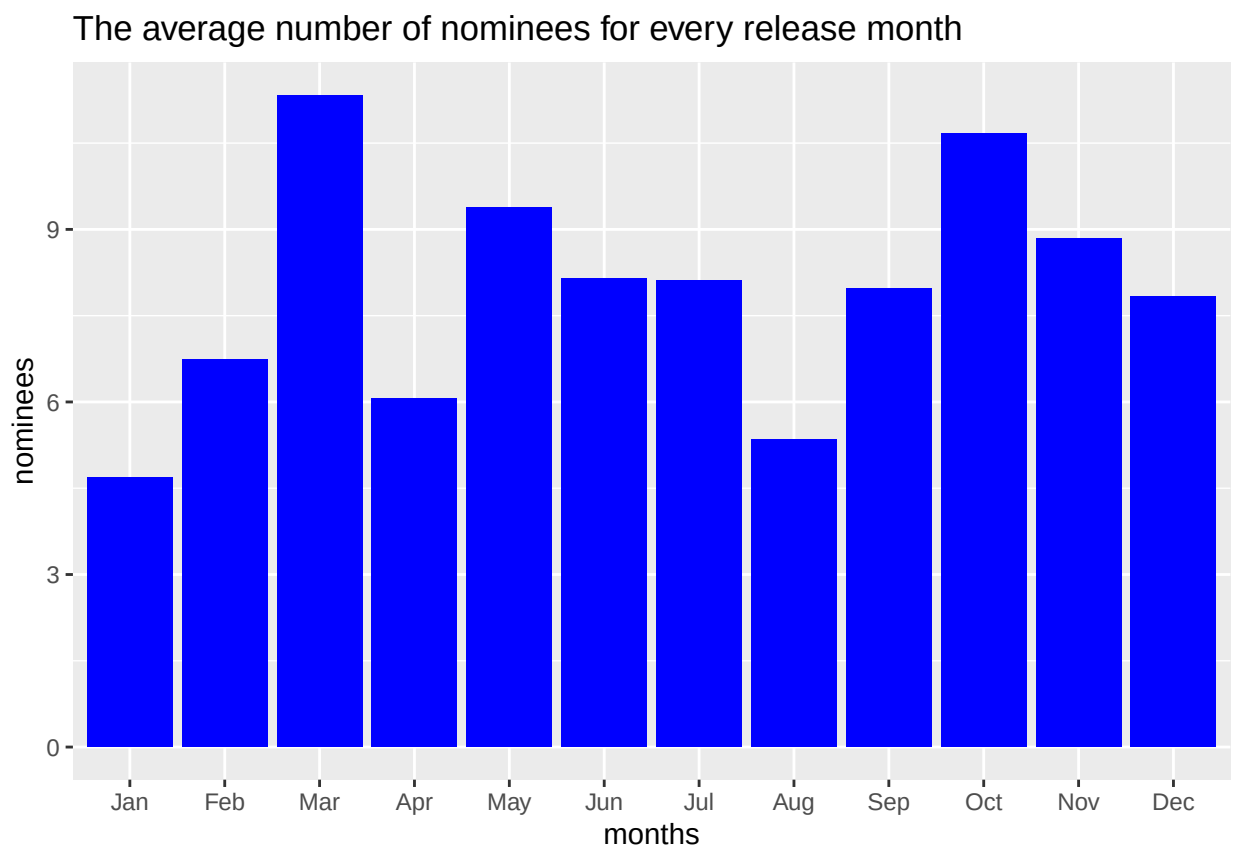
upload and clean the data from unneeded columns

```
netflix<-read.csv("netflix-rotten-tomatoes-metacritic-imdb.csv")
netflix_clean<-netflix[,-c(8,12,21:24,26:29)]
```

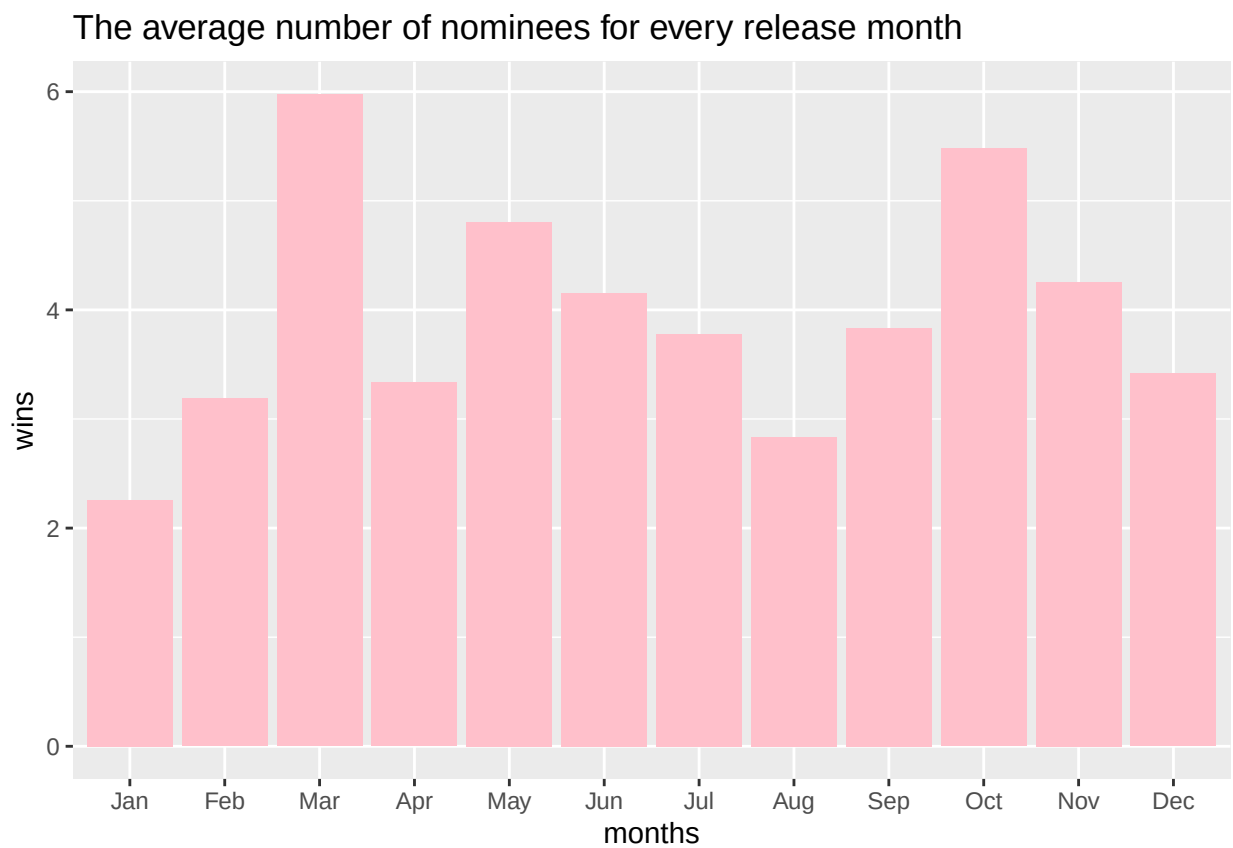
check correlation between release month of movies + series and number of awards. we wanted to see whether the month of release affects the number of award nominees or winnings, thinking that perhaps movies or series that were released shortly before the award committees will be nominated to more awards than movies that were released longer before.

```
#extract the month names from the list
date<-netflix_clean$Release.Date
date_split<-strsplit(date, " ")
months<-sapply(date_split, function(x) {x[2]})
months<-as.factor(months)
#order month names chronologically
levels(months)<-levels(months)[order(match(levels(months), month.abb))]
#vector of number of nominees, instead of NA there's 0
nominees<-netflix_clean$Awards.Nominated.For
nominees[is.na(nominees)]<-0
#and now the same for the winnings
wins<-netflix_clean$Awards.Received
wins[is.na(wins)]<-0
#remove data with NA values in 'months' vector, from month, wins and nominees
nominees<-nominees[!is.na(months)]
wins<-wins[!is.na(months)]
months<-months[!is.na(months)]
```

```r
#create a dataframe with the different vectors
df<-data.frame(months, nominees, wins)
#bar plot of the number of award nominees per month
ggplot(df, aes(x=months, y=nominees)) +geom_bar(stat = "summary", fun= "mean", fill="blue") +
  ggtitle("The average number of nominees for every release month")
```




```r
#bar plot of the number of award winnings per month
ggplot(df, aes(x=months, y=wins)) +geom_bar(stat = "summary", fun= "mean", fill="pink") +
    ggtitle("The average number of nominees for every release month")
```



Since the award committees occur around March, we can see that movies that were released in March did win and were nominated to more awards. Beside in March, there is no clear correlation between month of release and number of awards. Another interesting finding is that the plots of nominees and of winnings are almost the same, meaning there is a strong correlation between number of nominees and number of awards winnings.

Check correlation between number of votes in IMDb site and the IMDb score

```
#export the data from the dataframe into vectors
nimdb<-netflix_clean$IMDb.Votes
simdb<-netflix_clean$IMDb.Score
#check number of NAs
sum(is.na(nimdb))
```

```
## [1] 2101
```

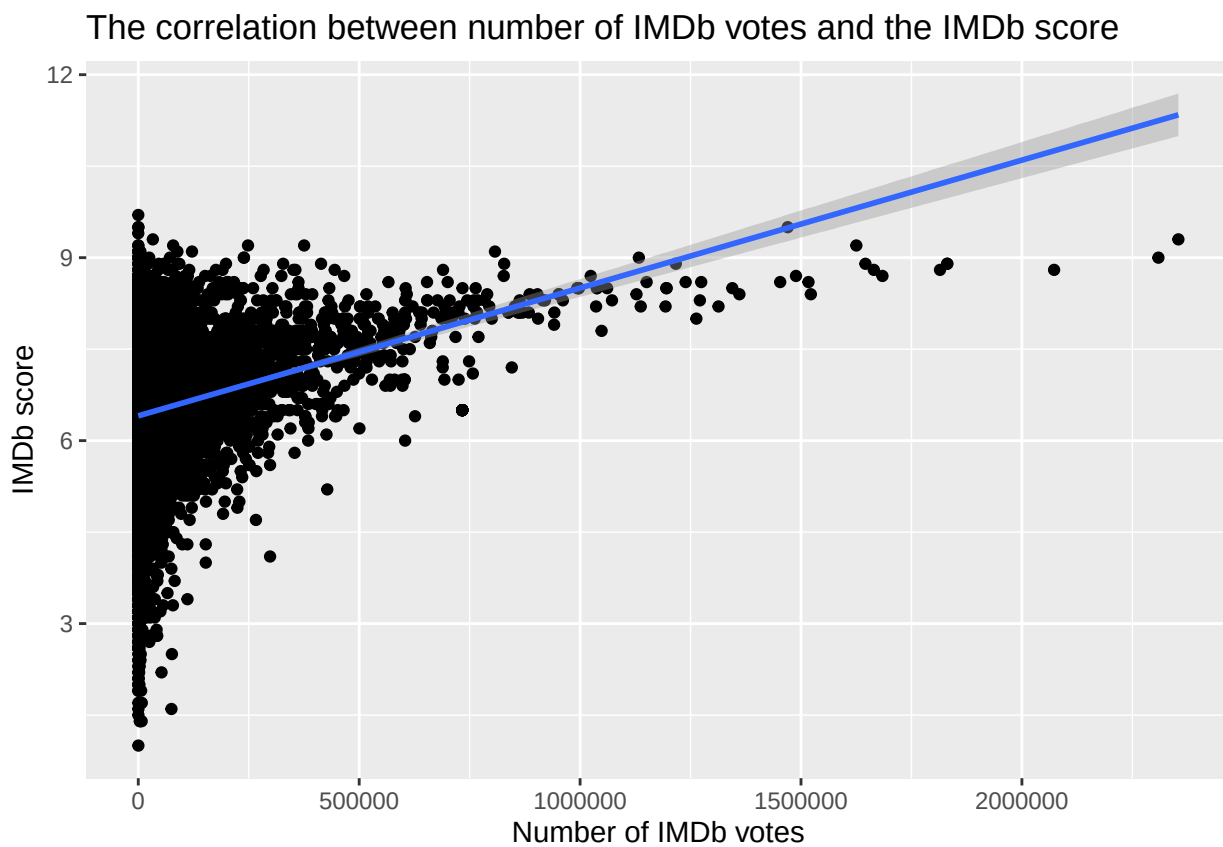
```
sum(is.na(simdb))
```

```
## [1] 2099
```

```
#there's almost the same number of NA in both vectors, and we want to see if
#they're at the same locations
sum(is.na(nimdb[is.na(simdb)]))
```

```
## [1] 2099
```

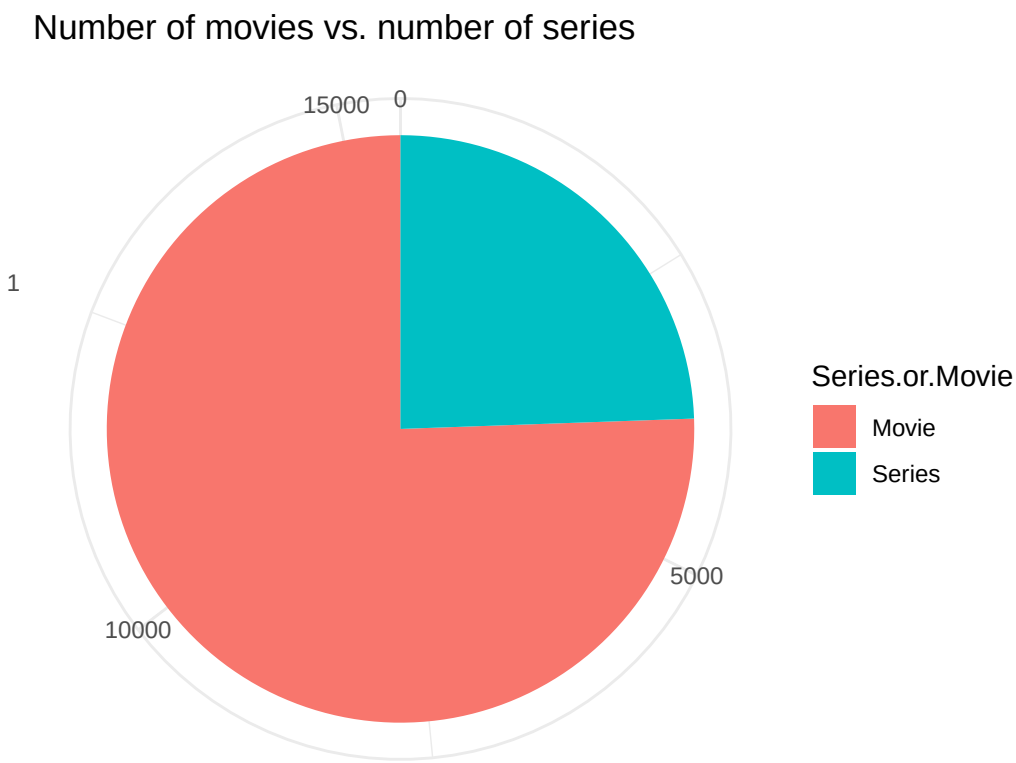
```
#all of the NAs overlap, and there are 2 more in nimdb, so we will filter all of them
simdb<-simdb[!is.na(nimdb)]
nimdb<-nimdb[!is.na(nimdb)]
#combine the vectors to dataframe
df_imdb<-data.frame(nimdb, simdb)
#create scatter plot and trend line
ggplot(df_imdb, aes(x=nimdb, y=simdb)) + geom_point() +
    geom_smooth(method=lm) + xlab("Number of IMDb votes") + ylab("IMDb score") +
  ggtitle("The correlation between number of IMDb votes and the IMDb score")
```



We can see a clear trend line that shows a straight correlation between number of votes in IMDb site and the score for every movie and series. This may mean that when a movie is highly popular, more viewers are willing to rate it.

A pie chart of the distribution of movies vs. series in the Netflix site

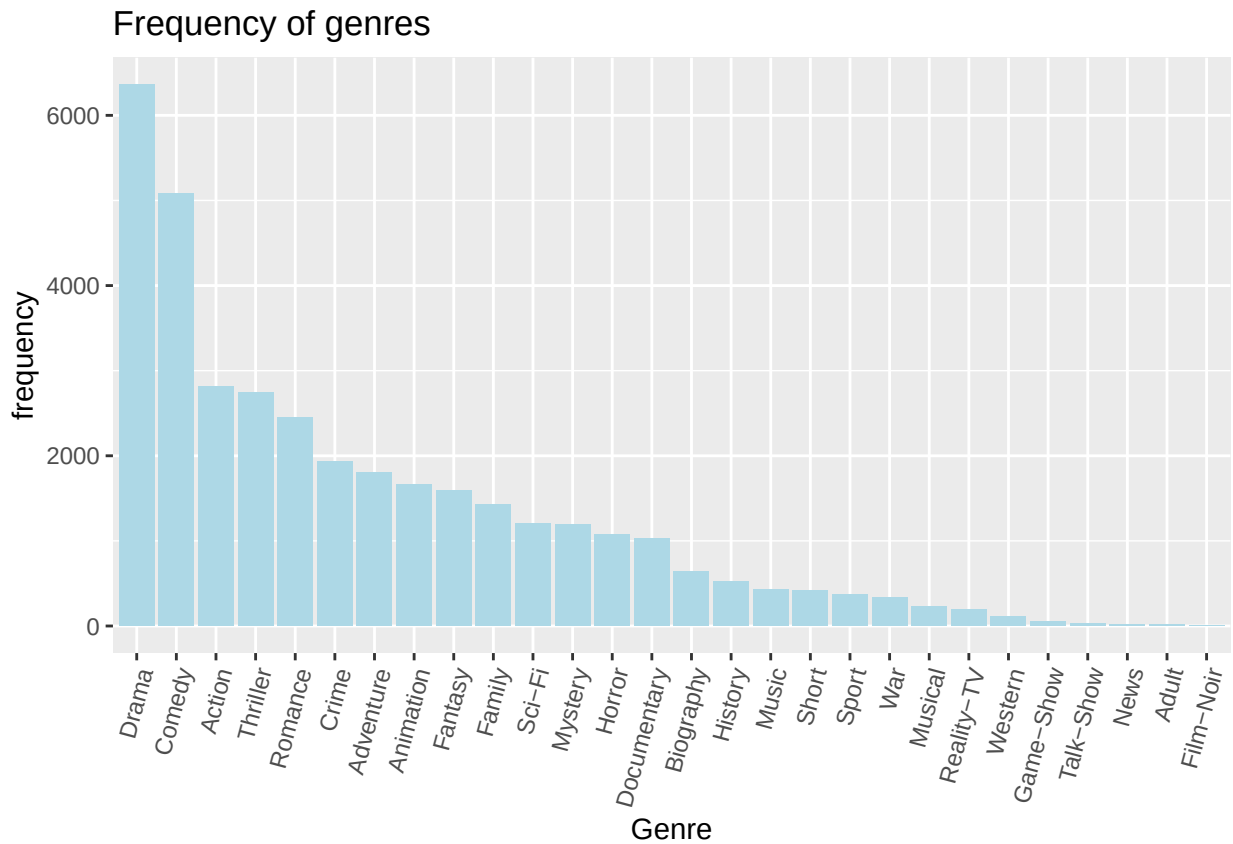
```
type <- netflix_clean$Series.or.Movie
ggplot(netflix_clean, aes(x=factor(1), fill=Series.or.Movie))+
  geom_bar(width = 1)+
  coord_polar("y") + xlab("") + ylab("") + theme_minimal() +
  ggtitle("Number of movies vs. number of series")
```



We can see that about a quarter of the content in Netflix is series, and the rest is movies.

The frequency of genres

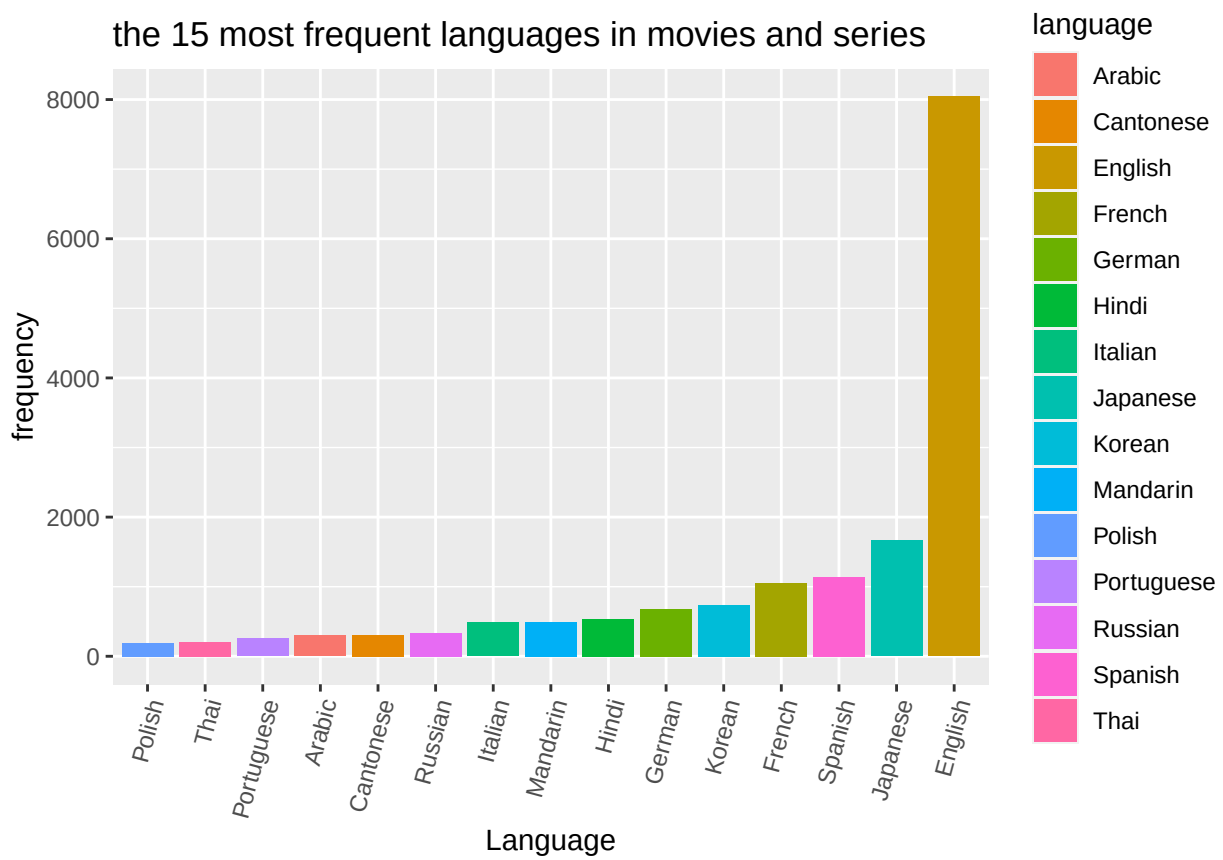
```
#get the genre column from the dataset, and split it to a 1D vector
genre = netflix_clean$Genre
genre = sapply(genre, strsplit, ', ')
genre = unlist(genre, recursive = FALSE)
genre_df = data.frame(genre)
#create a ber plot of the genre's frequency
ggplot(genre_df, aes(x = fct_infreq(genre))) + geom_bar(fill="light blue") +
  theme(axis.text.x=element_text(angle=75,hjust=1)) +
  xlab("Genre") + ylab("frequency") + ggtitle("Frequency of genres")
```



We can see that the most frequent genres are Drama and Comedy, and the rest of the genres are significantly less popular.

The 15 most frequent languages in movies and series. The languages that are more frequent can teach us about the countries that produce the most movies that will be popular worldwide.

```
#get the languages column from the dataframe and split it to a 1D vector
languages = netflix_clean$Languages
languages = sapply(languages, strsplit, ', ')
languages = unlist(languages, recursive = FALSE)
language = data.frame(languages)
#create the counts table of the languages
languages_count = table(language)
languages_count = data.frame(languages_count)
#get the 15 most frequent languages
languages_topn = top_n(languages_count, 15)
#create bar plot
ggplot(languages_topn, aes(x=reorder(language, Freq) , y = Freq, fill=language)) + geom_col() +
  theme(axis.text.x=element_text(angle=75,hjust=1)) +xlab("Language") +ylab("frequency") +
  ggtitle("the 15 most frequent languages in movies and series")
```



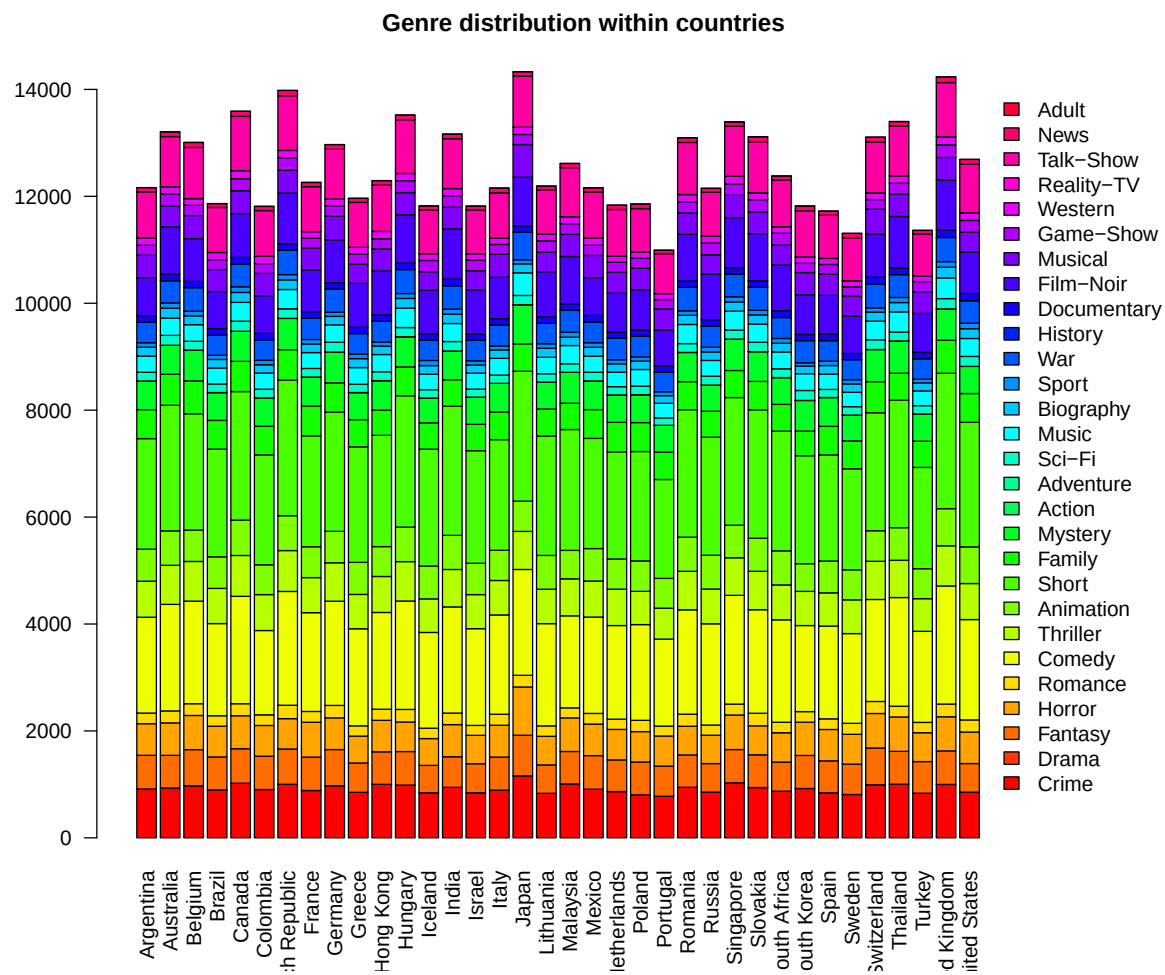
The most used language is English, by far. Meaning that there is a large number of movies created in English, and a large number of costumers that are willing to consume movies in that language. However, we should mention that Netflix is an American company so this outcome might be explained by that fact.

Creating a bar plot that shows the genre distribution between countries, AKA in what genres each country watch

```
#getting the data of countries and genres
genres<- netflix_clean$Genre
countries <- netflix_clean$Country.Availability
#create a list of the countries' names
split_countries<- sapply(countries, strsplit, split=",")
unlist_countries<-unlist(split_countries, recursive = FALSE)
unique_countries<-unique(unlist_countries)
#split the genres too
split_genres<- sapply(genres, strsplit, split=",")
#create dataframe of countries and genres
df1<-data.frame(countries, genres)
x <- strsplit(as.character(df1$genres), ", ", fixed = T)
# add new rows with each genre:
df1 <- cbind(df1[rep(1:nrow(df1), lengths(x)), 1:2], genres_split = unlist(x))
data<- data.frame(countries=df1$countries, genres_split=df1$genres_split)
x <- strsplit(as.character(data$countries), ",", fixed = T)
# add new rows with each country:
data <- cbind(data[rep(1:nrow(data), lengths(x)), 1:2], countries_split = unlist(x))
data<- data.frame(countries_split=data$countries_split, genres_split=data$genres_split)
counts<- setDT(data)[,list(Count=.N),names(data)]
```

```r
#turn the data to a matrix and clean it
mat<-acast(counts, genres_split ~ countries_split)
mat[is.na(mat)] <- 0
mat[mat<50] <- 0

#create the bar plot
palette1 <- rainbow(length(unique(counts$genres_split)))
vec <- matrix()
par(mfrow=c(1,1), mar = c(5,4,4,7))
plt <-barplot(mat, main="Genre distribution within countries",
  xlab="",xlim=c(0,ncol(mat)+10 ), col= palette1,
  cex.names=1, las=2, args.legend = list(x=ncol(mat)+17, inset=c(-0.30,0), xpd=TRUE, box.col="white"),
```



This plot means about nothing, but we put a lot of thought and effort in it, so here it is. In conclusion: "beauty is in the eye of the beholder" (Our Rabbi Moses, Romeo and Juliet, <3)