Unfolding-Cinemas

SJS

November 24, 2019

Let's load the packegess we need to use.

```
library(tidyverse) # Multiple packages
library(ggthemes) # Visualization themes
library(gridExtra) # Grids for visualizations
library(lubridate) # Working with dates
```

Importing the dataset.

```
setwd('C:/Users/Symon/Desktop/BoxOffice/BoxOffice')
train_data <- read_csv('train.csv')
test_data <- read_csv('test.csv')
```

Let's investigate missing values and condition the data set little bit to carry out our further analysis.

```
train_data <- train_data %>% mutate(budget = replace(budget, budget == '0', NA))

train_data$homepage[!is.na(train_data$homepage)] <- "YES"
train_data$homepage[is.na(train_data$homepage)] <- "NO"


imdb <- str_replace(train_data$imdb_id, "tt", '')
train_data["imdb_id"] <- imdb
train_data$release_date <- parse_date_time2(train_data$release_date, "mdy",
                                            cutoff_2000 = 20)

train_data <- train_data %>% separate(release_date, c("Year", "Month", "Day"))

sum(is.na(train_data$runtime))
```

```
## [1] 2
```

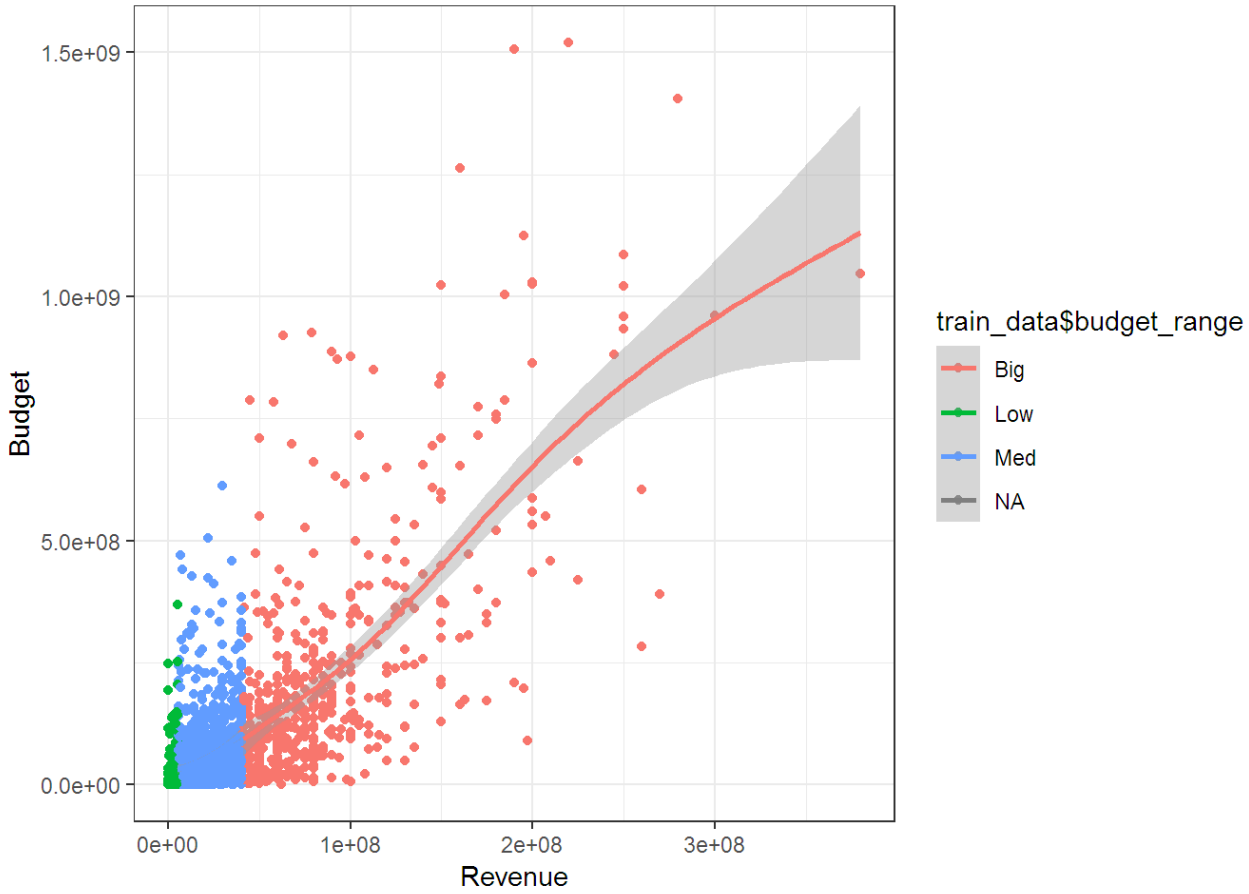
```
which(is.na(train_data$runtime))
```

```
## [1] 1336 2303
```

```
train_data <- train_data %>% drop_na(runtime)

train_data$tagline[!is.na(train_data$tagline)] <- "Yes"
train_data$tagline[is.na(train_data$tagline)] <- "NO"


train_data$collection_name <- str_extract(train_data$belongs_to_collection,
                                          pattern = "(?<=name\\'\\:\\s{1}\\').+(?=\\'\\,\\s{1}\\'poster)")


train_data$Franchise[!is.na(train_data$collection_name)] <- "YES"
train_data$Franchise[is.na(train_data$collection_name)] <- "No"

train_data$prod_country <- str_extract(string = train_data$production_countries, pattern = "[:upper:]+")

genres_matching_point <- "Comedy|Horror|Action|Drama|Documentary|Science Fiction|
Crime|Fantasy|Thriller|Animation|Adventure|Mystery|War|Romance|Music|
Family|Western|History|TV Movie|Foreign"

train_data$main_genre <- str_extract(train_data$genres, genres_matching_point)


train_data$budget_range[train_data$budget <= 5.10e+06] <- "Low"
train_data$budget_range[train_data$budget > 5.10e+06 & train_data$budget <= 4.00e+07 ] <- "Med"
train_data$budget_range[train_data$budget > 4.00e+07] <- "Big"
```

Let’s start to explore the data and relations between our variables.

```
train_data$budget_range <- as.factor(train_data$budget_range)

ggplot(train_data, aes(train_data$budget, train_data$revenue, color = train_data$budget_range)) + geom_point() + geom_smooth() + theme_bw()+xlab("Revenue")+ylab
("Budget")
```



There are very few Low budget movies, Most of them are medium or high budget movies.

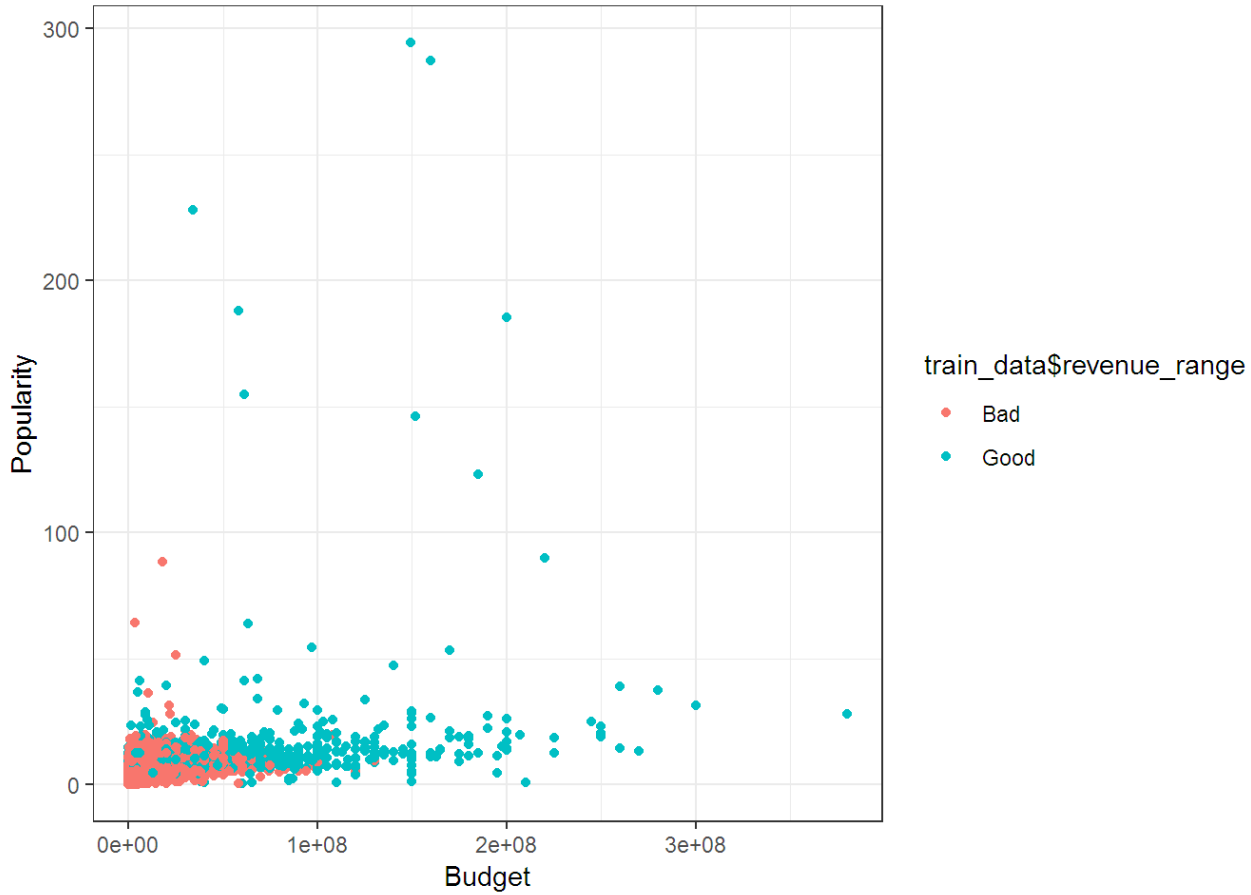
Budget seems very related with revenue. More budget seems to earn more.

```
train_data$revenue_range[train_data$revenue <= 6.677e+07] <- "Bad"
train_data$revenue_range[train_data$revenue > 6.677e+07] <- "Good"

table(train_data$budget_range, train_data$revenue_range)
```

```
##
##      Bad Good
##  Big 106  396
##  Low 523   24
##  Med 818  320
```

Observations: 1. Out of every 5 Big Budget, 4 will do good. 2. Only 4% Low buget can make the cut-off 3. 30% medium budget movie earning good 4. Money Brings Money

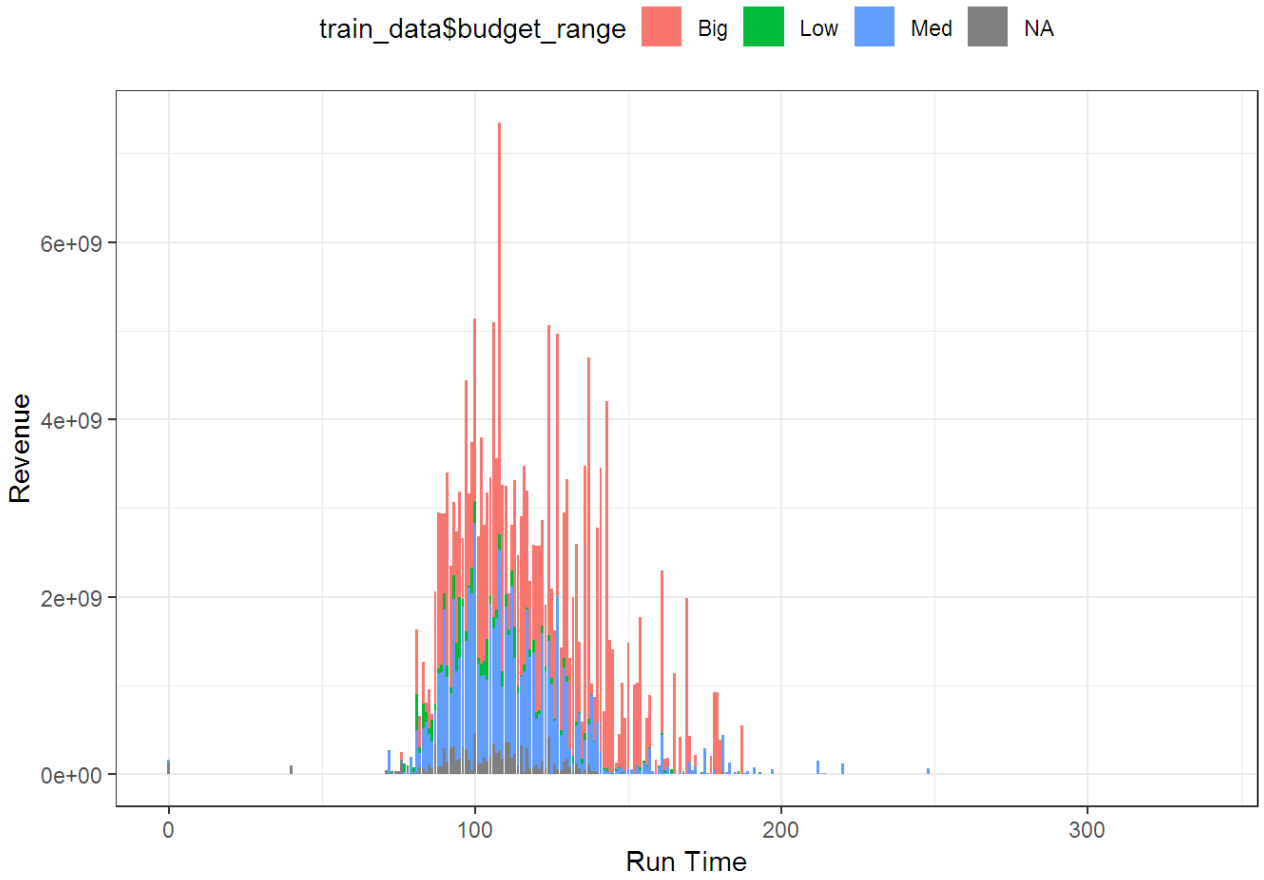
```
ggplot(train_data, aes(train_data$budget, train_data$popularity, color = train_data$revenue_range)) + geom_point()+theme_bw()+xlab("Budget")+ylab("Popularity")
```



Popularity also has a positive correlation with budget as expected but not as much as revenue.

There is a sweetspot where low budget movie seems to have good revenue and good popularity: These movies will be interesting to study. Maybe some other time.

```
ggplot(train_data, aes(train_data$runtime, train_data$revenue, fill = train_data$budget_range)) + geom_col()+xlab("Run Time")+ylab("Revenue")+theme_bw()+theme(legend.position = "top")
```



1. Runtime is important, people tend to spend money on movies ranging from 90 min - 145 min.
2. Big budgets and medium budgets movie are clearly aware of this fact.
3. This says something about our attention span, isn't it?

```
Median_budget <- train_data %>% group_by(train_data$budget_range) %>% summarise(median_revenue=median(revenue))


Median_revenue <- train_data %>% group_by(train_data$budget_range) %>% summarise(median_budget=median(budget))

Median <- merge(Median_revenue, Median_budget, by = "train_data$budget_range")
```

Top 10 Big Budget titles based on Revenue

```
train_data %>% filter(budget_range == "Big") %>% arrange(desc(revenue)) %>% select(title, revenue) %>% head(10)
```

```
## # A tibble: 10 x 2
##    title                                revenue
##    <chr>                                  <dbl>
##  1 The Avengers                        1519557910
##  2 Furious 7                           1506249360
##  3 Avengers: Age of Ultron             1405403694
##  4 Beauty and the Beast                1262886337
##  5 Transformers: Dark of the Moon      1123746996
##  6 The Dark Knight Rises               1084939099
##  7 Pirates of the Caribbean: On Stranger Tides 1045713802
##  8 Finding Dory                        1028570889
##  9 Alice in Wonderland                 1025491110
## 10 Zootopia                            1023784195
```

Top 10 Medium Budget titles based on Revenue

```
train_data %>% filter(budget_range == "Med") %>% arrange(desc(revenue)) %>% select(title, revenue) %>% head(10)
```

```
## # A tibble: 10 x 2
##    title                    revenue
##    <chr>                      <dbl>
##  1 The Passion of the Christ 611899420
##  2 Ghost                     505000000
##  3 Jaws                      470654000
##  4 The Hangover              459270619
##  5 The Exorcist              441306145
##  6 The Intouchables          426480871
##  7 Dances with Wolves        424208848
##  8 The Bodyguard             411006740
##  9 Monster Hunt              385284817
## 10 Toy Story                 373554033
```

Top 10 Big Budgets based on Revenue

```
train_data %>% filter(budget_range == "Low") %>% arrange(desc(revenue)) %>% select(title, revenue) %>% head(10)
```

```
## # A tibble: 10 x 2
##    title                    revenue
##    <chr>                      <dbl>
##  1 My Big Fat Greek Wedding 368744044
##  2 Get Out                  252434250
##  3 The Blair Witch Project  248000000
##  4 Paranormal Activity 3    205703818
##  5 Paranormal Activity      193355800
##  6 Lights Out               148868835
##  7 Paranormal Activity 4    142817992
##  8 Animal House             141000000
##  9 Love Story               136400000
## 10 Porky's                  125728258
```

Observations:

The top 10 movies by earning for low budget criteria seems very interesting: 6 of them horror, 4 of them commedy

If you get less money in the movie business, either bring good wit or crazy vison to scare people off

Franchise movies dominate big budget genre

Seems like medium budget movie can hold more creativity and experiments

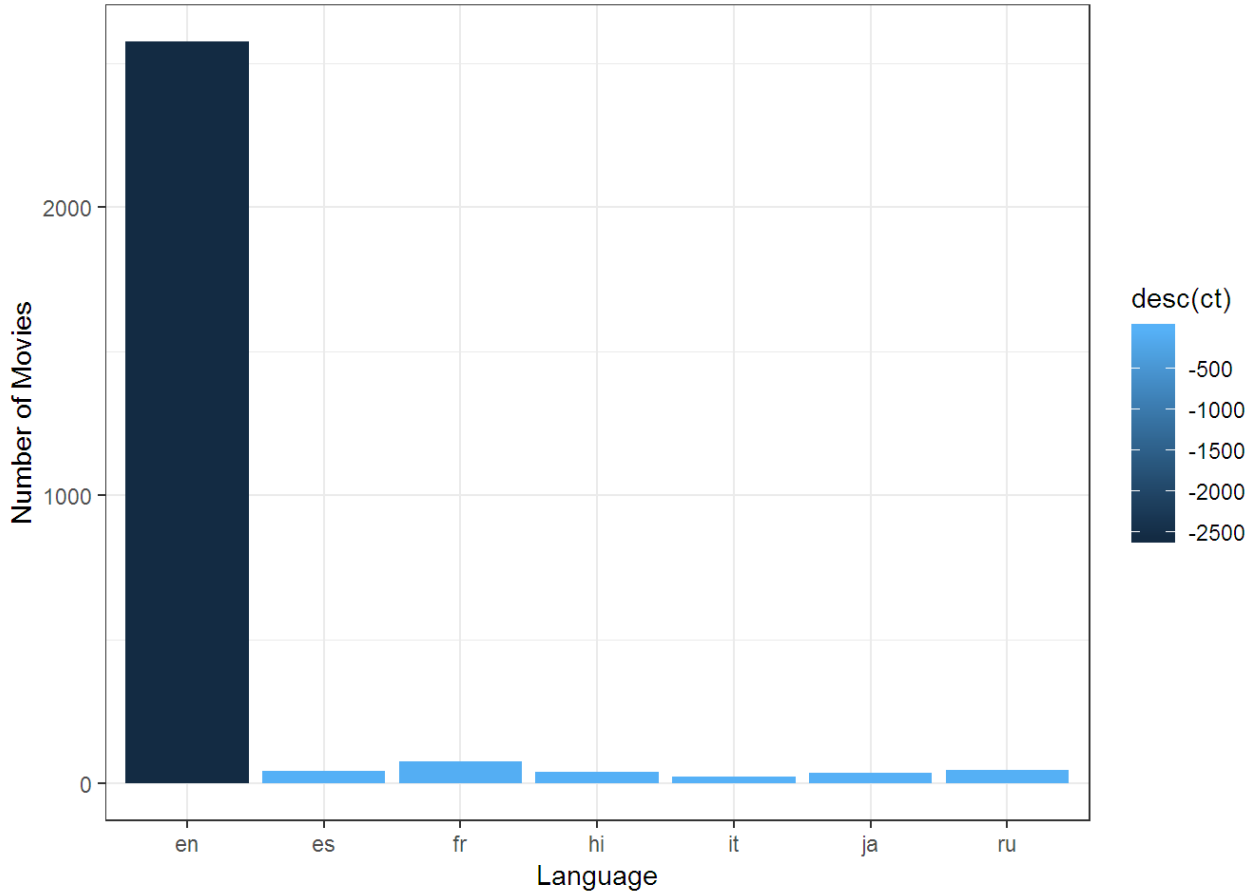
Top 10 Big Budget yet Fails:

```
train_data %>% filter(budget_range == "Big") %>% arrange(desc(revenue)) %>% select(title, revenue) %>% tail(10)
```

```
## # A tibble: 10 x 2
##    title                      revenue
##    <chr>                        <dbl>
##  1 Stay                       8342132
##  2 Gigli                      7266209
##  3 1492: Conquest of Paradise 7191399
##  4 The Adventures of Pluto Nash 7103973
##  5 The Big Bounce             6808550
##  6 A Sound of Thunder         5989640
##  7 Heaven's Gate              3484331
##  8 Child 44                   3324330
##  9 Shadow Conspiracy          2154540
## 10 Lolita                     1060056
```

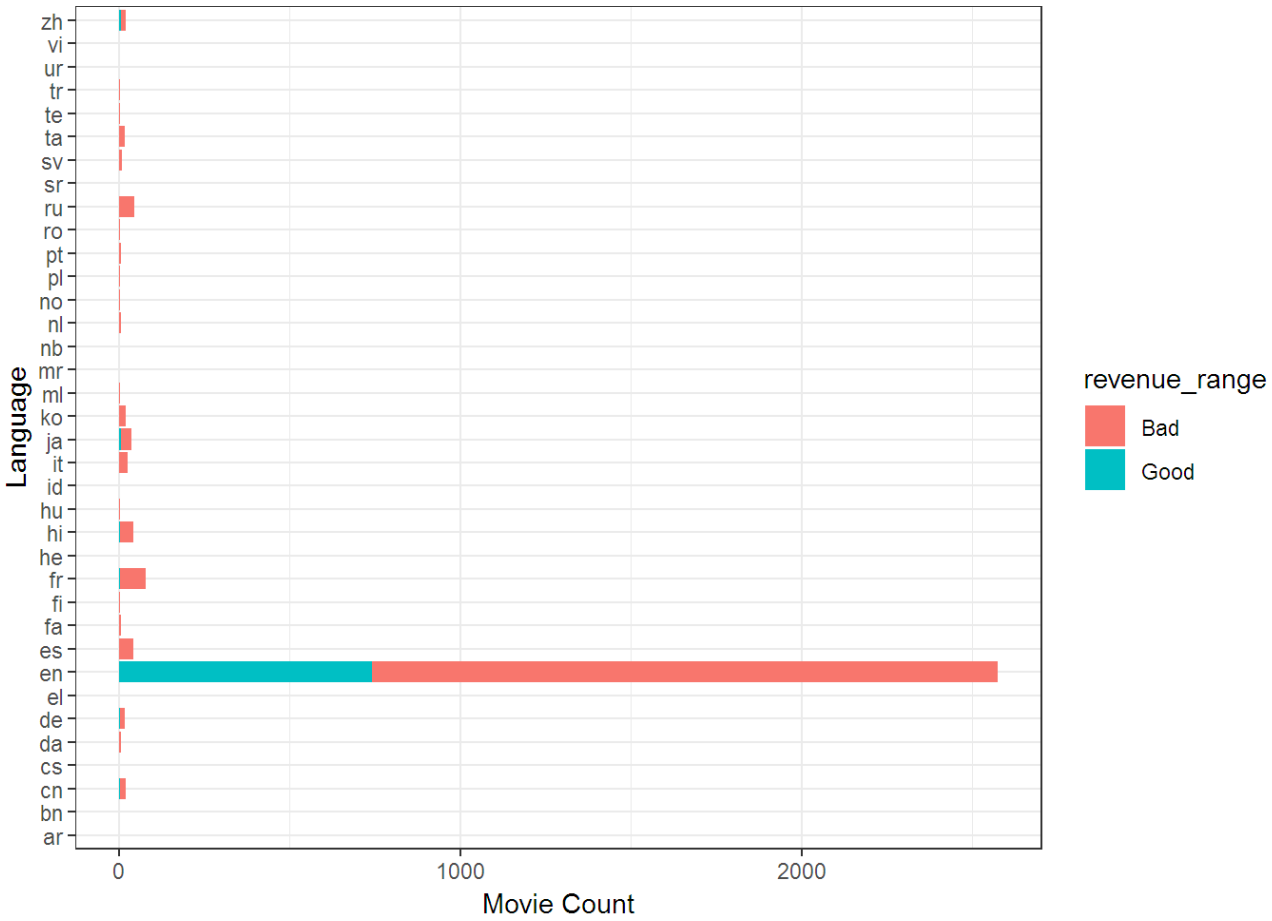
I do keep myself informed with movies at least with thee big hits: Never heard about any of these movies: seems well justified to me.

```
language_number <- train_data %>% group_by(original_language) %>% summarise(ct = n()) %>% arrange(desc(ct)) %>% head(7)
language_number$original_language <- as.factor(language_number$original_language)

ggplot(language_number, aes(language_number$original_language, language_number$ct, fill = desc(ct))) + geom_col() + xlab("Language") + ylab("Number of Movies")
+theme_bw()
```



No wonder, there will be monopoly of english language. Othere signifacnt languages are french, russian, hindi, spanish and italian

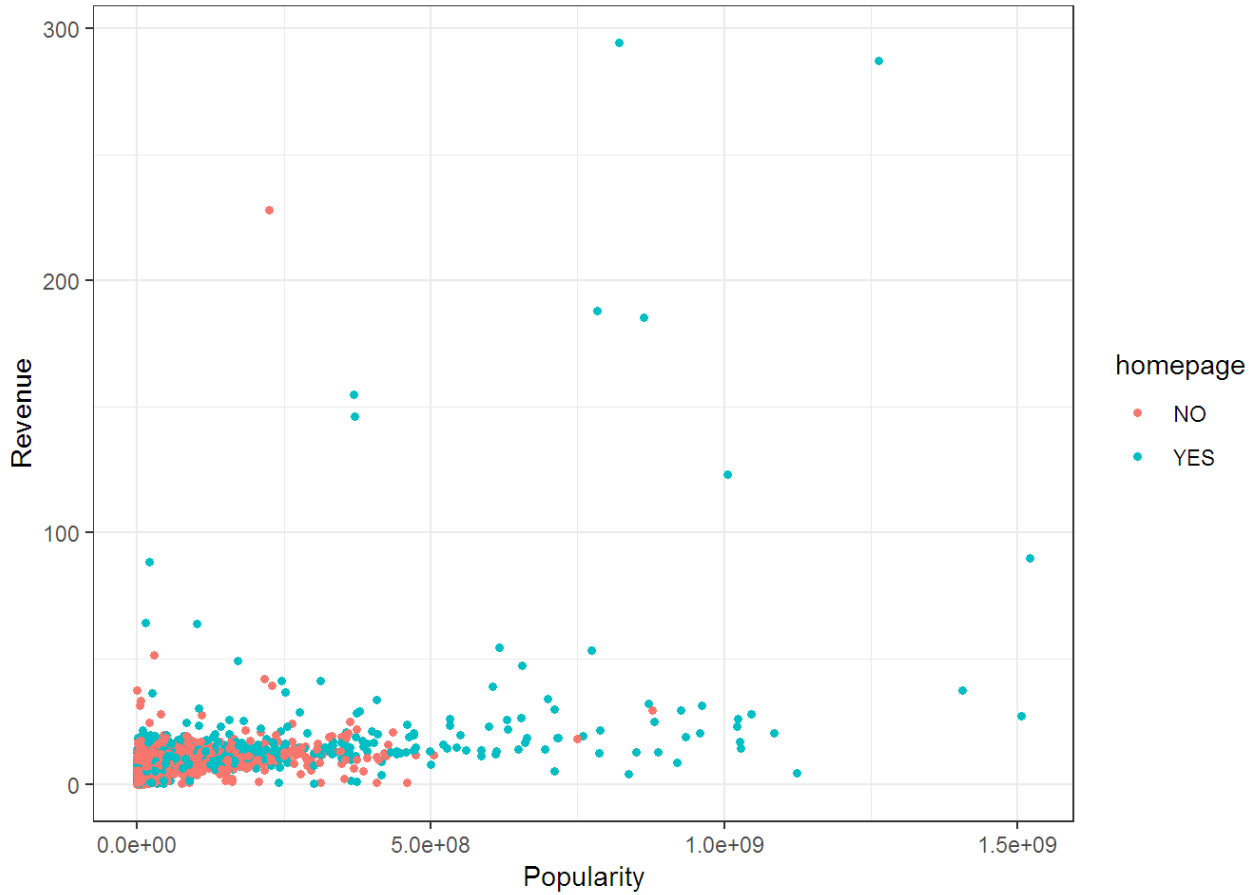
```
ggplot(train_data, aes(original_language, fill = revenue_range)) + geom_bar() + coord_flip()+xlab("Language") + ylab("Movie Count")+theme_bw()+theme(plot.margin =
 margin(.01,.01,.01,.01, "cm"))
```



By global standard, other languages other than english are not successful that much except few language like japanese, hindi, french, tr, zh, de

Russian language film seems to suffer a lot; Giant land with small population effect, I guess

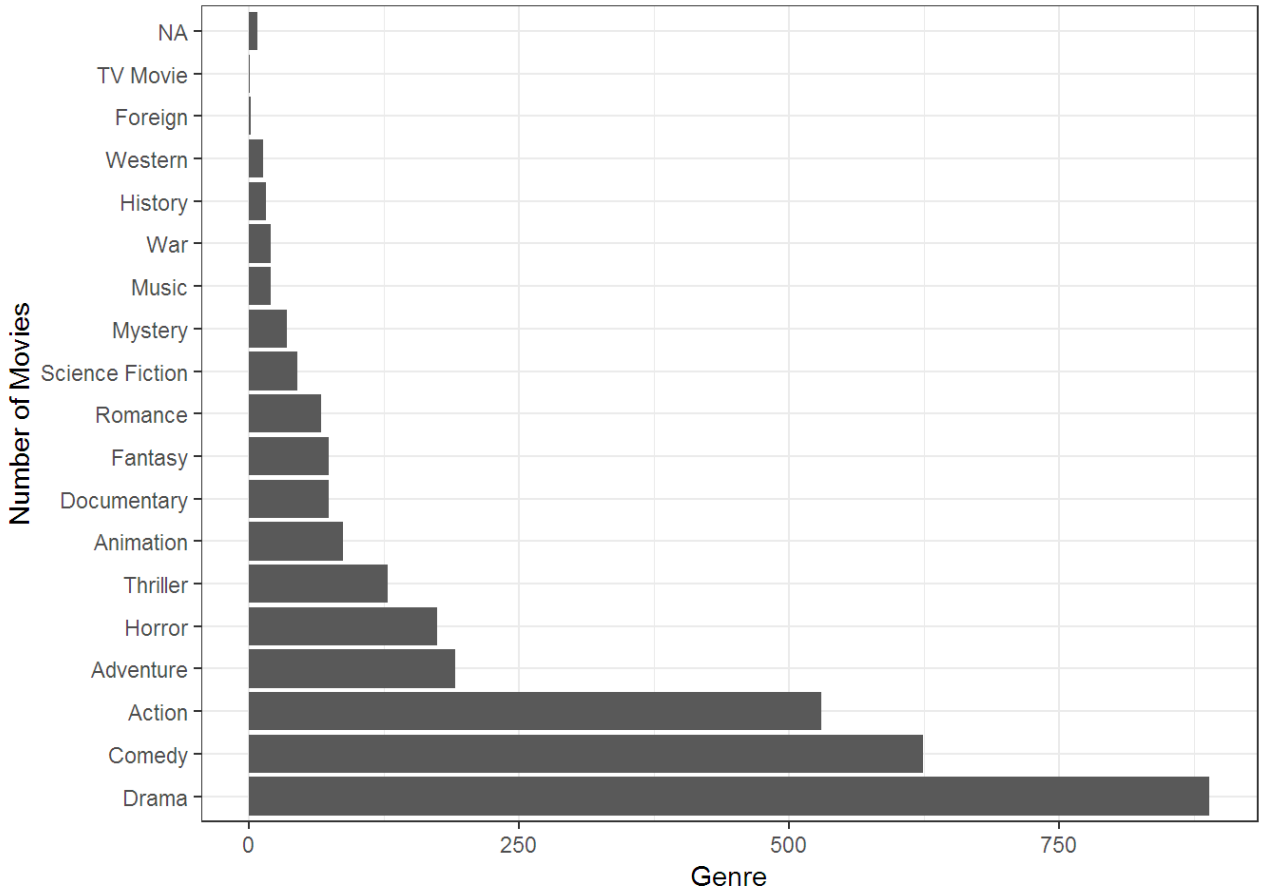
```
ggplot(train_data, aes(revenue, popularity, color = homepage)) + geom_point(size = 1.2)+theme_bw()+xlab("Popularity")+ ylab("Revenue")
```



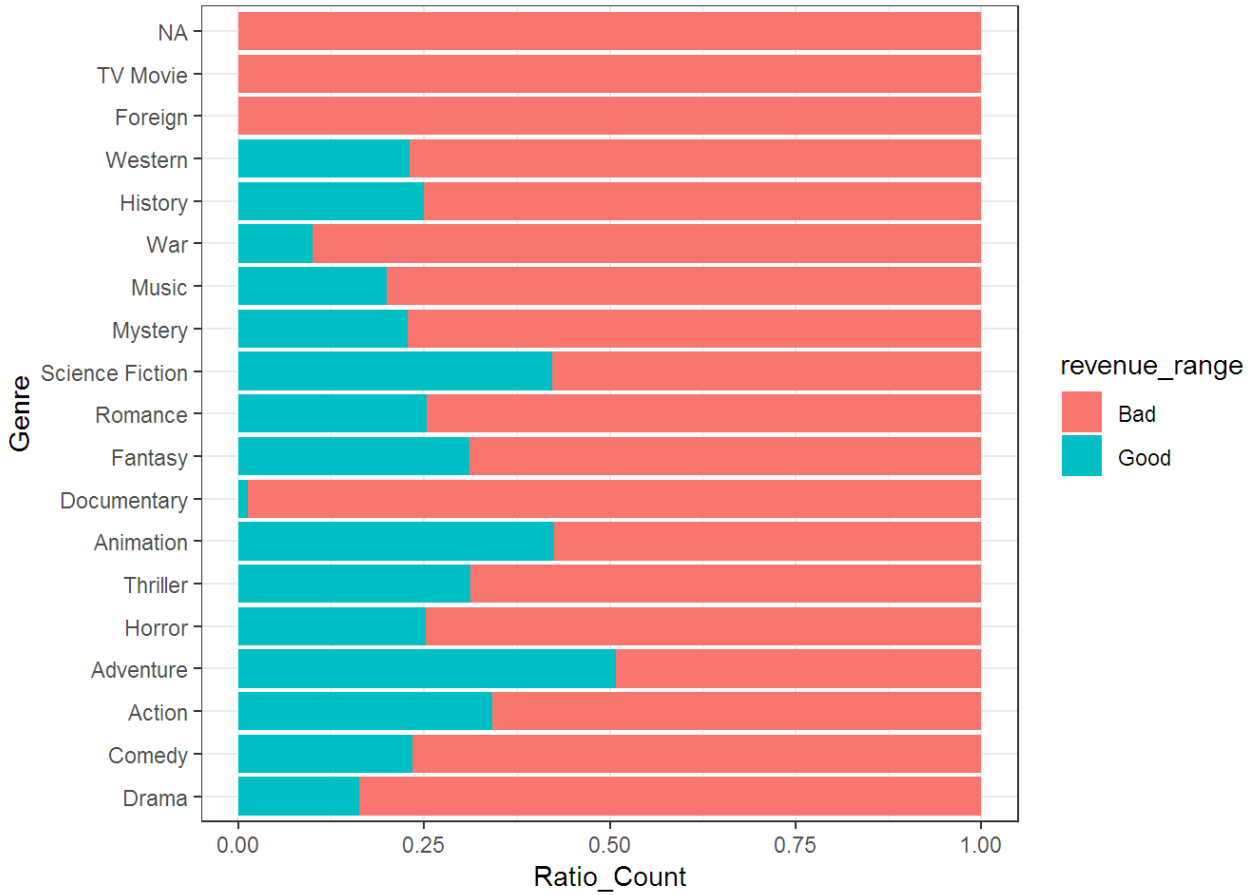
The graph might be misleading: seems like having homepage have clear effect on being the movie popular and successful

Actually most of the successful or popular movies are high budget movies: thereby can afford to build or care about having a homepage. Domain Knowledge !!

```
ggplot(train_data, aes(fct_infreq(train_data$main_genre))) + geom_bar(na.rm= TRUE) + coord_flip() + ylab("Genre")+xlab("Number of Movies")+theme_bw()
```




```
ggplot(train_data, aes(fct_infreq(train_data$main_genre),  fill = revenue_range)) + geom_bar(position = "fill") + coord_flip()+ xlab("Genre")+ylab("Ratio_Count")+theme_bw()
```

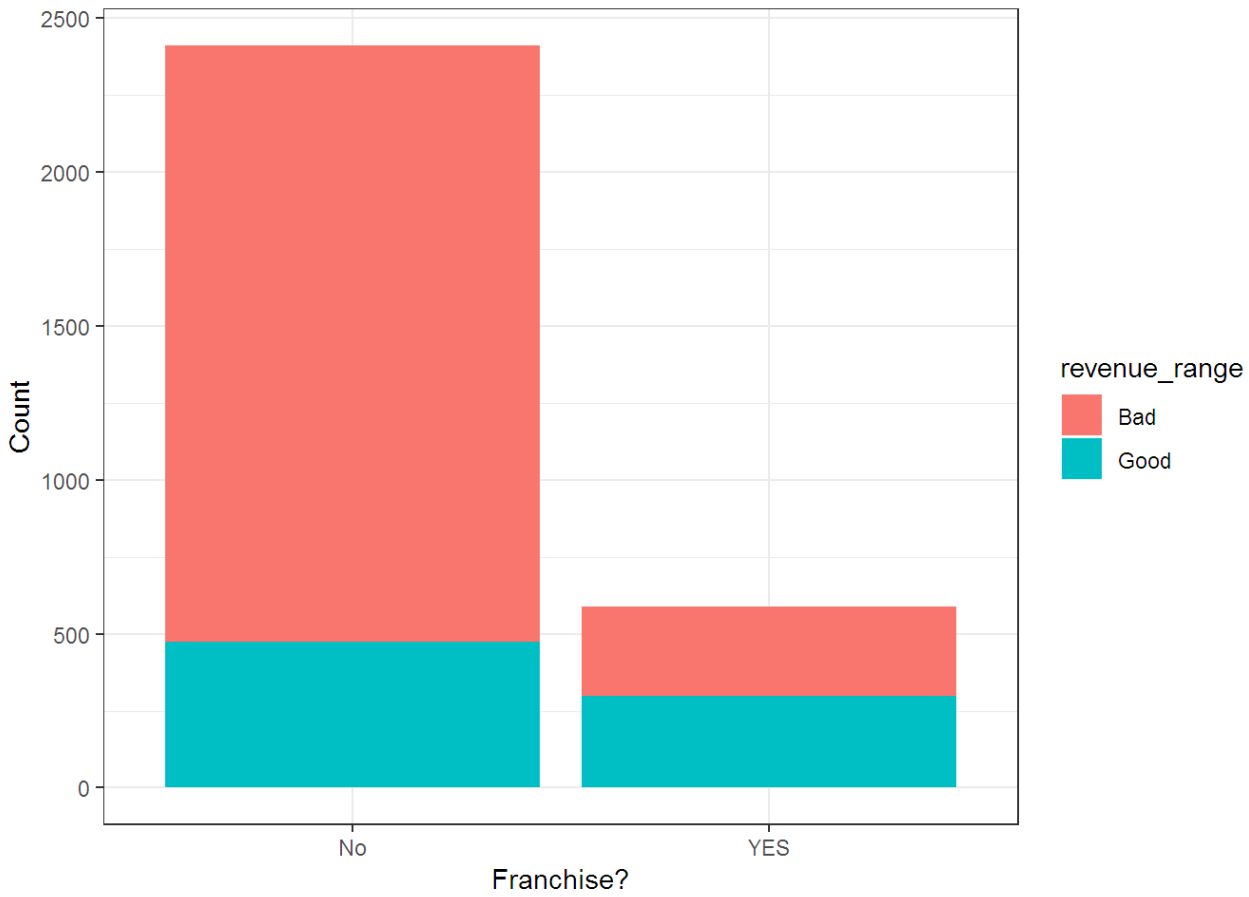


Drama, comedy, action overwhelms other genre.

But Revenue wise Adventure, Science fictions & animation beats everybody.

Hollywood stands high in every category.

```
ggplot(train_data, aes(Franchise, fill = revenue_range)) + geom_bar()+xlab("Franchise?")+ylab("Count")+theme_bw()
```



This is one interesting plot, looks like probability of being a winner is much higher for a franchise movie.