IMDB EDA

IMDB (EDA)

00. Load Libraries

```r
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.2     ✓ tibble    3.3.0
## ✓ lubridate 1.9.4     ✓ tidyr     1.3.1
## ✓ purrr     1.1.0
## ── Conflicts ────────────────────────────────────────────────────────────── tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

01. Import Data

```r
imdb <- read_csv("IMDB.csv")
```

```
## Rows: 10178 Columns: 11
## ── Column specification ──────────────────────────────────────────────────────────────────────────
## Delimiter: ","
## chr (7): country, date_x, genre, names, orig_lang, orig_title, status
## dbl (4): budget_x, Sum of Profit/Loss, revenue, score
##
## 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.
```

02. EDA

```
glimpse(imdb)
```

```
## Rows: 10,178
## Columns: 11
## $ budget_x             <dbl> 1e+00, 1e+00, 1e+00, 1e+02, 1e+02, 1e+02, 1e+02, 1e+04, 1e+04, 1e+04, 1e+05, 1e+05, 1e+05…
## $ country              <chr> "AU", "JP", "US", "GB", "NL", "US", "US", "US", "US", "US", "AU", "AU", "AU", "AU", "ES",…
## $ date_x               <chr> "04-06-2023", "01-08-1972", "02-01-2019", "03/14/2017", "04-06-2023", "04-06-2023", "04-0…
## $ genre                <chr> "Drama, Thriller", NA, "Horror, Drama, Thriller", "Drama, Romance", NA, "Adventure, Fanta…
## $ names                <chr> "DADDY OWL!!!", "Onsen porno chitai", "Down", "Picture of Beauty", "De man uit Rome", "Lu…
## $ orig_lang            <chr> "English", "Japanese", "English", "English", "Dutch, Flemish", "English", "English", "Eng…
## $ orig_title           <chr> "Beneath Us", "温泉ポルノ痴帯", "Down", "Picture of Beauty", "De man uit Rome", "Luigi’s Mans
ion…
## $ `Sum of Profit/Loss` <dbl> 0.0, 23580102.6, 257720412.2, 2264758.8, 1240161.6, 1240161.6, 1240161.6, 17877093.8, 794…
## $ revenue              <dbl> 1, 23580104, 257720413, 2264859, 1240262, 1240262, 1240262, 17887094, 79423925, 10801446,…
## $ score                <dbl> 0, 60, 69, 45, 0, 0, 0, 53, 61, 56, 74, 58, 71, 73, 60, 72, 63, 50, 75, 69, 67, 77, 62, 7…
## $ status               <chr> "Released", "Released", "Released", "Released", "Released", "Released", "Released", "Rele…
```

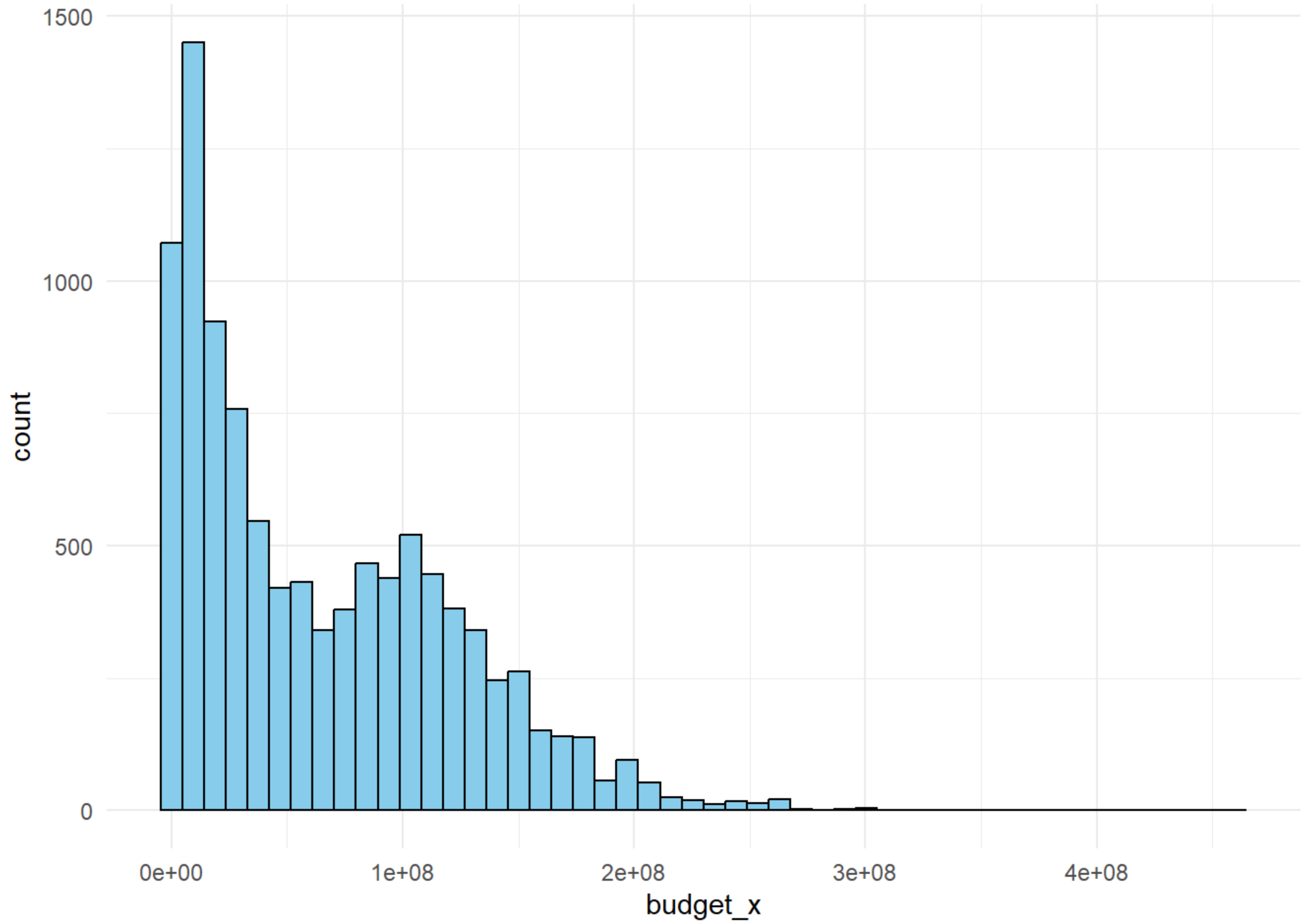
```
summary(imdb)
```

```
##     budget_x           country             date_x             genre              names             orig_lang
##  Min.   :        1   Length:10178       Length:10178       Length:10178       Length:10178       Length:10178
##  1st Qu.: 15000000   Class :character   Class :character   Class :character   Class :character   Class :character
##  Median : 50000000   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 64882379
##  3rd Qu.:105000000
##  Max.   :460000000
##   orig_title        Sum of Profit/Loss      revenue              score          status
##  Length:10178       Min.   :-340000000   Min.   :0.000e+00   Min.   :  0.0   Length:10178
##  Class :character   1st Qu.:   5901626   1st Qu.:2.859e+07   1st Qu.: 59.0   Class :character
##  Mode  :character   Median :  84515466   Median :1.529e+08   Median : 65.0   Mode  :character
##                     Mean   : 188257715   Mean   :2.531e+08   Mean   : 63.5
##                     3rd Qu.: 317666522   3rd Qu.:4.178e+08   3rd Qu.: 71.0
##                     Max.   :2686706026   Max.   :2.924e+09   Max.   :100.0
```

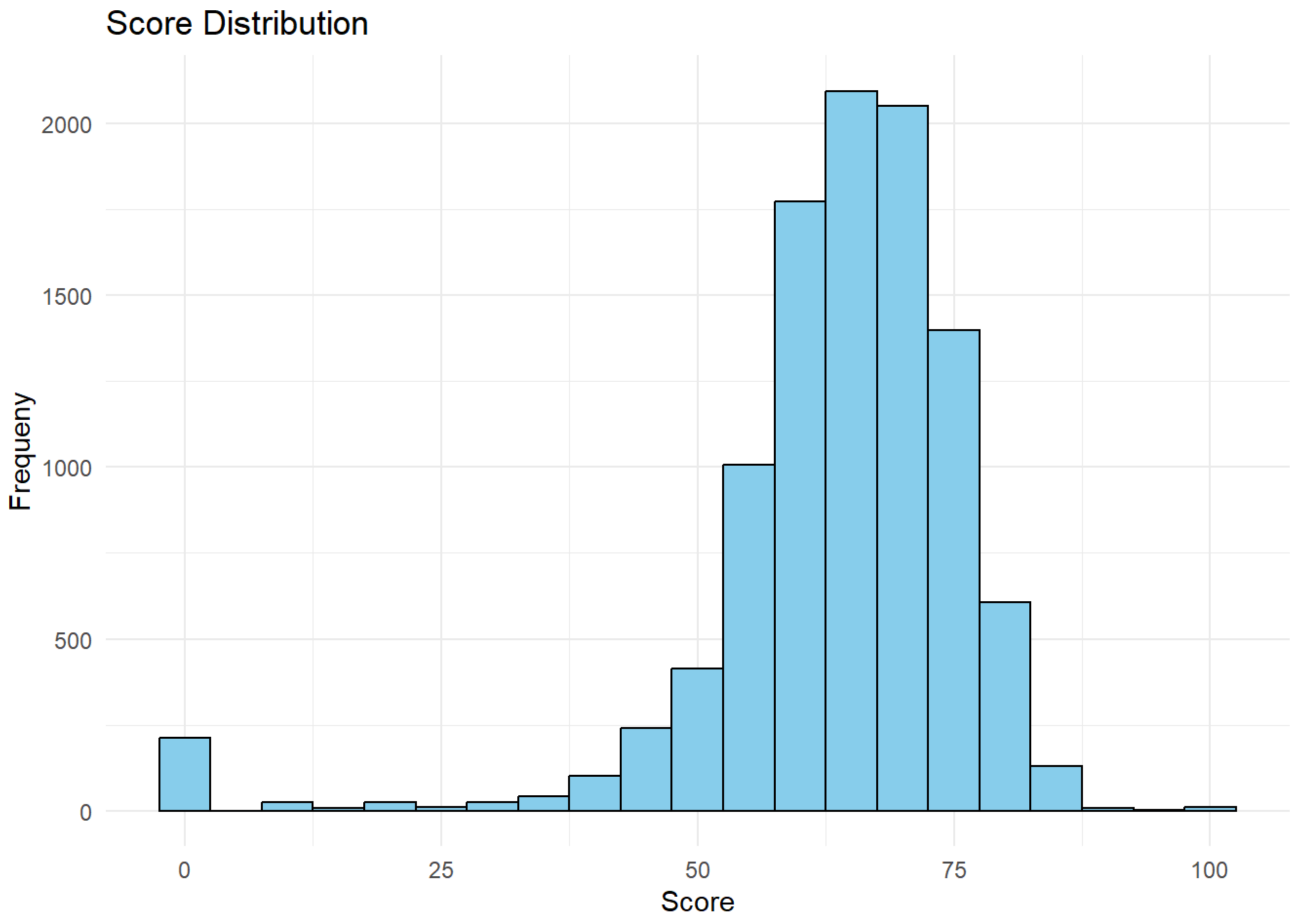
```
view(imdb)
```

03. Plotting

```
ggplot(imdb,aes(budget_x)) +
  geom_histogram(bins = 50, fill = "skyblue", color = "black") +
  theme_minimal()
```

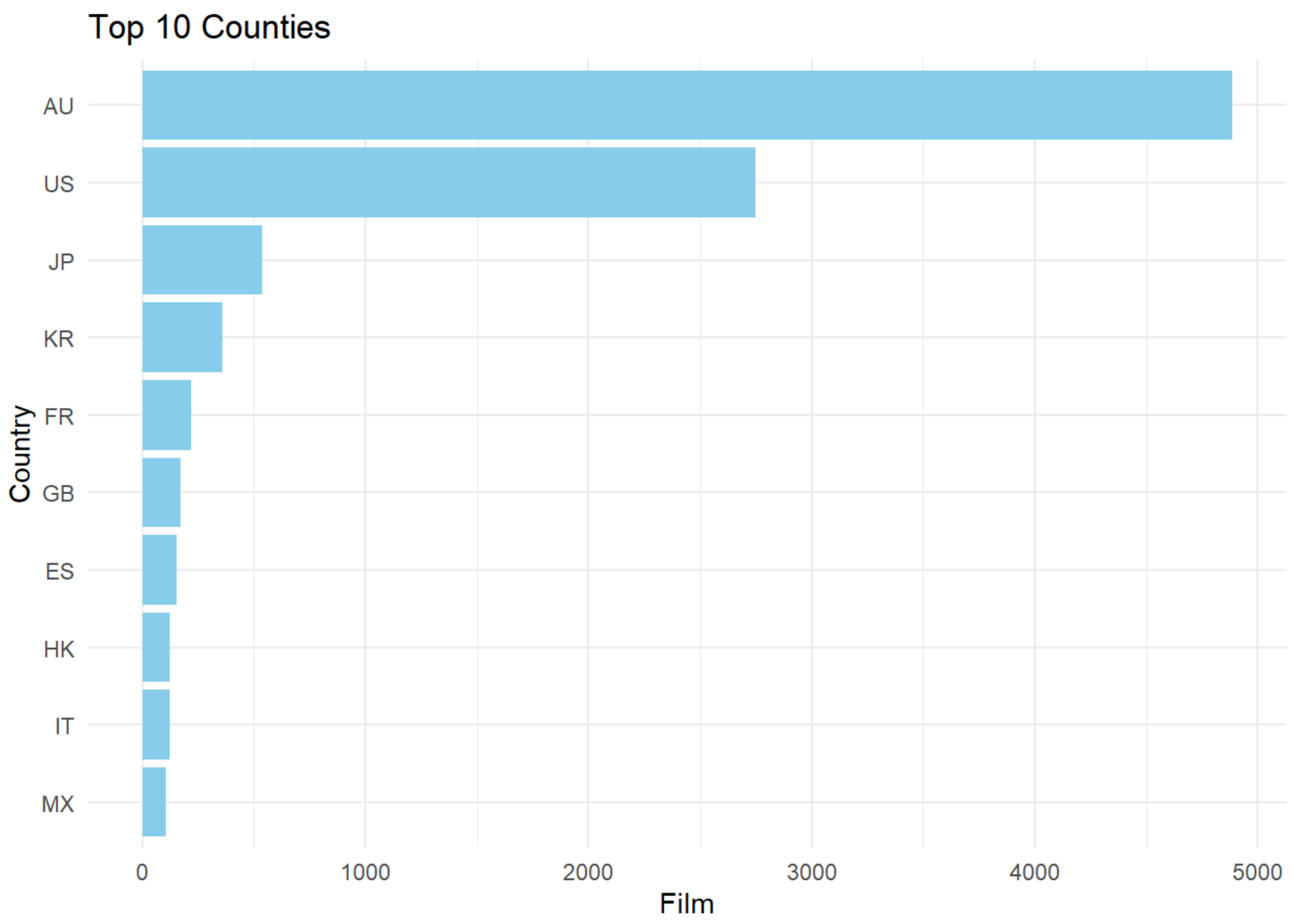


```
ggplot(data = imdb, aes(x = score)) +
  geom_histogram(binwidth = 5, fill = "skyblue", color = "black") +
  labs(title = "Score Distribution",
       x = "Score",
       y = "Frequeny") +
  theme_minimal()
```



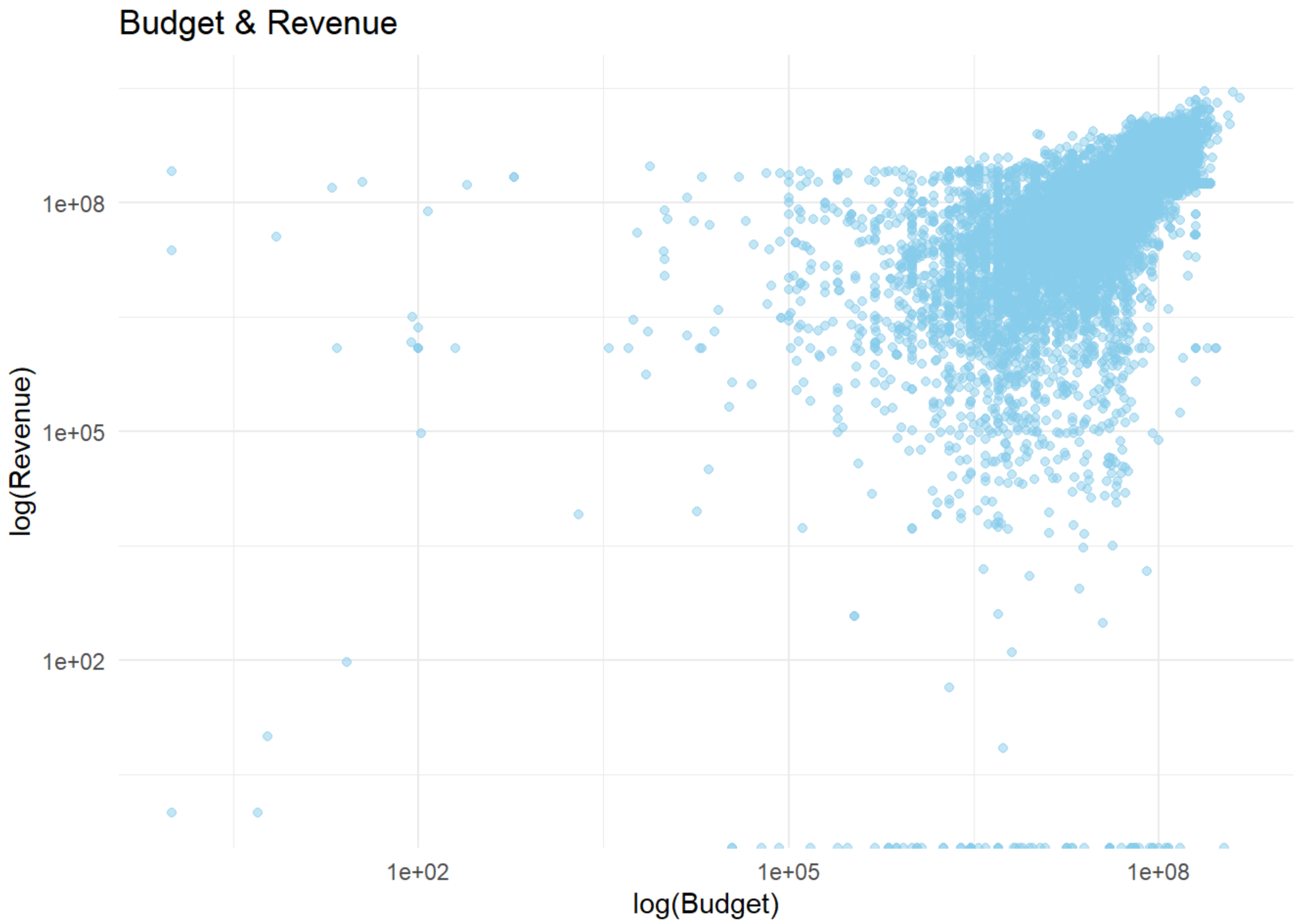
```
imdb %>%
  count(country, sort = TRUE) %>%
  top_n(10) %>%
  ggplot(aes(x = reorder(country, n), y = n)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  coord_flip() +
  labs(title = "Top 10 Counties",
       x = "Country",
       y = "Film") +
  theme_minimal()
```

```
## Selecting by n
```



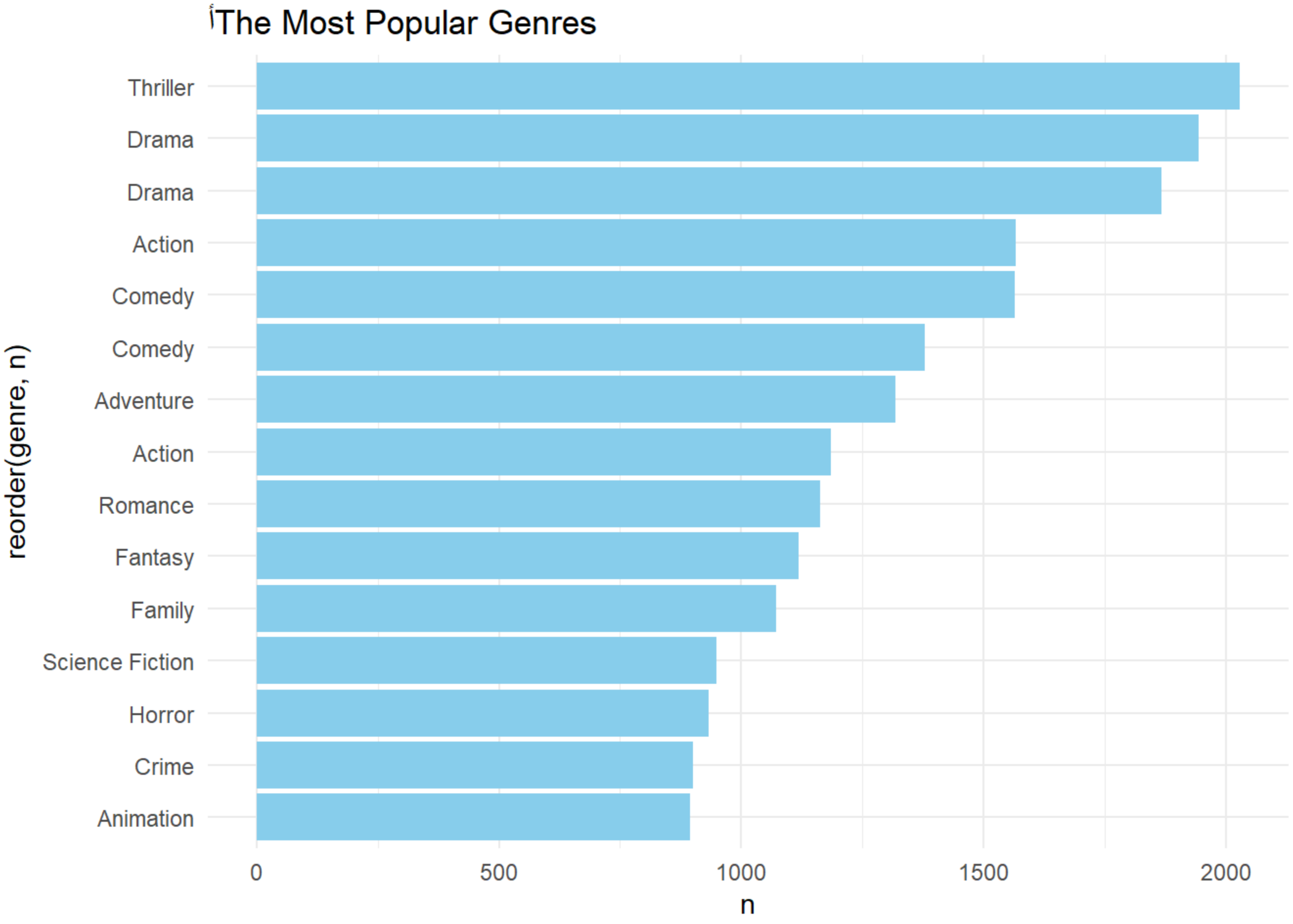
```
ggplot(data = imdb, aes(x = budget_x, y = revenue)) +
  geom_point(alpha = 0.5, color = "skyblue") +
  scale_x_log10() +
  scale_y_log10() +
  labs(title = "Budget & Revenue",
       x = "log(Budget)",
       y = "log(Revenue)") +
  theme_minimal()
```

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```



```
genres_separated <- imdb %>%
  separate_rows(genre, sep = ",\\s*") # The ",\\s*" separates by comma and any following whitespace
genres_separated %>%
  count(genre, sort = TRUE) %>%
  top_n(15) %>%
  ggplot(aes(x = reorder(genre, n), y = n)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  coord_flip() +
  labs(title = "أThe Most Popular Genres") +
  theme_minimal()
```

```
## Selecting by n
```


The Most Popular Genres
reorder(genre, n)
Thriller
Drama
Drama
Action
Comedy
Comedy
Adventure
Action
Romance
Fantasy
Family
Science Fiction
Horror
Crime
Animation
n