

Insights from IMDB Movie Data



Executive Summary

Problem Statement

Identify factors influencing movie ratings and box office revenue to guide data-driven decisions in budget, marketing, and production strategies

Project Outline

- 1. Data Collection**
 - Scrape movie data from IMDB (e.g., title, genre, rating, budget, revenue)
- 2. Data Analysis**
 - Perform exploratory data analysis (EDA) to uncover patterns and correlations
- 3. Visualization**
 - Present insights through graphs, charts, and ranked lists.

Expected Results

Identify influential factors driving movie success and generate actionable insights for optimizing film production and marketing strategies

Data Collection Process

Source: IMDB and BoxOfficeMojo website

Data Points

- Movie Title, Year of Release, Genre
- IMDB Rating, Number of Votes, Director
- Production Budget, Box Office Revenue

Tools

Python library BeautifulSoup for scraping; Pandas for data manipulation; Tableau for Data Visualization.

Scraping all the data took around **15 hours** due to sleep timers and the size of the data


Scraping Approach

Leveraging BeautifulSoup to scrape relevant data from movie releases since 1995

BoxOffice Mojo Site


Box Office Mojo by IMDbPro		Search for Titles		IMDbPro	f	
Domestic	International	Worldwide	Calendar	All Time	Showdowns	Indices
Daily	Weekend	Weekly	Monthly	Quarterly	Yearly	Seasons
Holidays						
Domestic Box Office For 2024						
2024	Calendar grosses					
Rank	Release	Gross	Theaters	Total Gross	Release Date	Distributor
1	Inside Out 2	\$652,980,194	4,440	\$652,980,194	Jun 14	Walt Disney Studios Motion Pictures
2	Deadpool & Wolverine	\$636,745,858	4,330	\$636,745,858	Jul 26	Walt Disney Studios Motion Pictures
3	Despicable Me 4	\$361,004,205	4,449	\$361,004,205	Jul 3	Universal Pictures
4	Wicked	\$325,456,795	3,888	\$325,456,795	Nov 22	Universal Pictures
5	Moana 2	\$301,860,091	4,200	\$301,860,091	Nov 27	Walt Disney Studios Motion Pictures
6	Beetlejuice Beetlejuice	\$294,100,435	4,575	\$294,100,435	Sep 6	Warner Bros.

Imdb Site


 Menu All ▾ IMDbPro + Watchlist Sign In EN ▾


Inside Out 2


2024 · PG · 1h 36m



[Cast & crew](#) · [User reviews](#) · [Trivia](#) · [FAQ](#) [IMDbPro](#) [All topics](#) [Share](#)

IMDb RATING
 **7.6** / 10
184K

YOUR RATING
 **Rate**

POPULARITY
 **131** ▾ 27

< **medy** **Teen Drama** **Adventure** **Animation** **Comedy** **Drama** **Family**

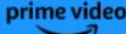
A sequel that features Riley entering puberty and experiencing brand new, more complex emotions as a result. As Riley tries to adapt to her teenage years, her old emotions try to adapt to the possibility of being replaced.

Director [Kelsey Mann](#)

Writers [Meg LeFauve](#) · [Dave Holstein](#) · [Kelsey Mann](#) >

Stars [Amy Poehler](#) · [Maya Hawke](#) · [Kensington Tallman](#) >

IMDbPro [See production info at IMDbPro](#)

RENT/BUY

from \$5.99

+ Add to Watchlist
Added by 216K users ▾

661 User reviews **260** Critic reviews
73 Metascore

Data Collection Flow

1. Scrape 6000 rows of data from BoxOffice Mojo
2. Search Imdb using Data scraped from BoxOffice Mojo
3. Scrape an additional 6000 rows of data from Imdb
4. Merge and clean the datasets using pandas
5. Import to Tableau for Analysis

BeautifulSoup Code

```
import requests
from bs4 import BeautifulSoup
import pandas as pd
import time

# Base URL for scraping
base_url = "https://www.boxofficemojo.com/year/1/"

# Range of years to scrape
years = range(2024, 1995, -1) # From 2024 to 1995

# Initialize an empty list to store all movie data
all_movie_data = []

# Loop through each year
for year in years:
    url = base_url.format(year)
    print(f"Scraping year: {year}")
    print(f"Scraping URL: {url}")

    # Send a GET request to the year URL
    response = requests.get(url)
    if response.status_code != 200: # If request fails, skip this year
        print(f"Failed to fetch {url}, status code: {response.status_code}")
        continue

    # Parse the HTML content
    soup = BeautifulSoup(response.text, "html.parser")

    # Locate the table containing the movie data
    table = soup.find("table")
    if not table: # If no table is found, skip this year
        print(f"No table found for year {year}. Moving to the next year.")
        continue

    # Extract table rows
    rows = table.find_all("tr") # Skip the header row

    # Process each row
    for row in rows:
        columns = row.find_all("td")
        if len(columns) != 8:
            # Extract data from the meta page
            release = columns[1].text.strip() # Movie title
            gross = columns[2].text.strip() # Gross
            theatres = columns[3].text.strip() # Theatres
            total_gross = columns[7].text.strip() # Total gross
            release_date = columns[8].text.strip() # Release date
            distributor = columns[9].text.strip() # Distributor

            # Extract the movie link
            movie_link_tag = columns[1].find("a")
            if movie_link_tag and "href" in movie_link_tag.attrs:
                movie_link = "https://www.boxofficemojo.com" + movie_link_tag["href"]

            # Visit the movie page and extract the genre
            movie_response = requests.get(movie_link)
            if movie_response.status_code != 200:
                movie_soup = BeautifulSoup(movie_response.text, "html.parser")
            # Locate the genres section (update the selector as per the page structure)
            genres_section = movie_soup.find("span", string="Genres")
            if genres_section:
                genres = genres_section.find_next_siblings("span")
            else:
                genres = "N/A" # Default if no genres found
            else:
                genres = "N/A"

            # Add a delay to avoid overloading the server
            time.sleep(1)
            else:
                genres = "N/A"

            # Append the data with the genres
            all_movie_data.append([year, release, gross, theatres, total_gross, release_date, distributor, genres])

            # Add a delay before moving to the next year
            time.sleep(2)

print("Scraping completed for all years.")

# Convert the data to a DataFrame
columns = ["Year", "Release", "Gross", "Theatres", "Total Gross", "Release Date", "Distributor", "Genres"]
df = pd.DataFrame(all_movie_data, columns=columns)

# Save to a CSV file
df.to_csv("box_office_data_1995_to_2024_with_genres.csv", index=False)

print("Data scraping completed and saved to 'box_office_data_1995_to_2024_with_genres.csv'")
```

Fetching IMDb data for: Inside Out 2 (2024)
Starting search for: Inside Out 2 (2024)
Search URL: <https://www.imdb.com/find?q=Inside+Out+2+2024>
Search page fetched successfully for Inside Out 2
Navigated to movie page URL: https://www.imdb.com/title/tt22022452/?ref_=fn_al_tt_1
Movie page fetched successfully for Inside Out 2
Rating found: 7.6
Director found: Kelsey Mann
Budget found: \$200,000,000 (estimated)

Fetching IMDb data for: Deadpool & Wolverine (2024)
Starting search for: Deadpool & Wolverine (2024)
Search URL: <https://www.imdb.com/find?q=Deadpool+%26+Wolverine+2024>
Search page fetched successfully for Deadpool & Wolverine
Navigated to movie page URL: https://www.imdb.com/title/tt6263850/?ref_=fn_al_tt_1
Movie page fetched successfully for Deadpool & Wolverine
Rating found: 7.7
Director found: Shawn Levy
Budget found: \$200,000,000 (estimated)

Fetching IMDb data for: Despicable Me 4 (2024)
Starting search for: Despicable Me 4 (2024)
Search URL: <https://www.imdb.com/find?q=Despicable+Me+4+2024>
Search page fetched successfully for Despicable Me 4
Navigated to movie page URL: https://www.imdb.com/title/tt7510222/?ref_=fn_al_tt_1
Movie page fetched successfully for Despicable Me 4
Rating found: 6.2
Director found: Chris Renaud
Budget found: \$100,000,000 (estimated)

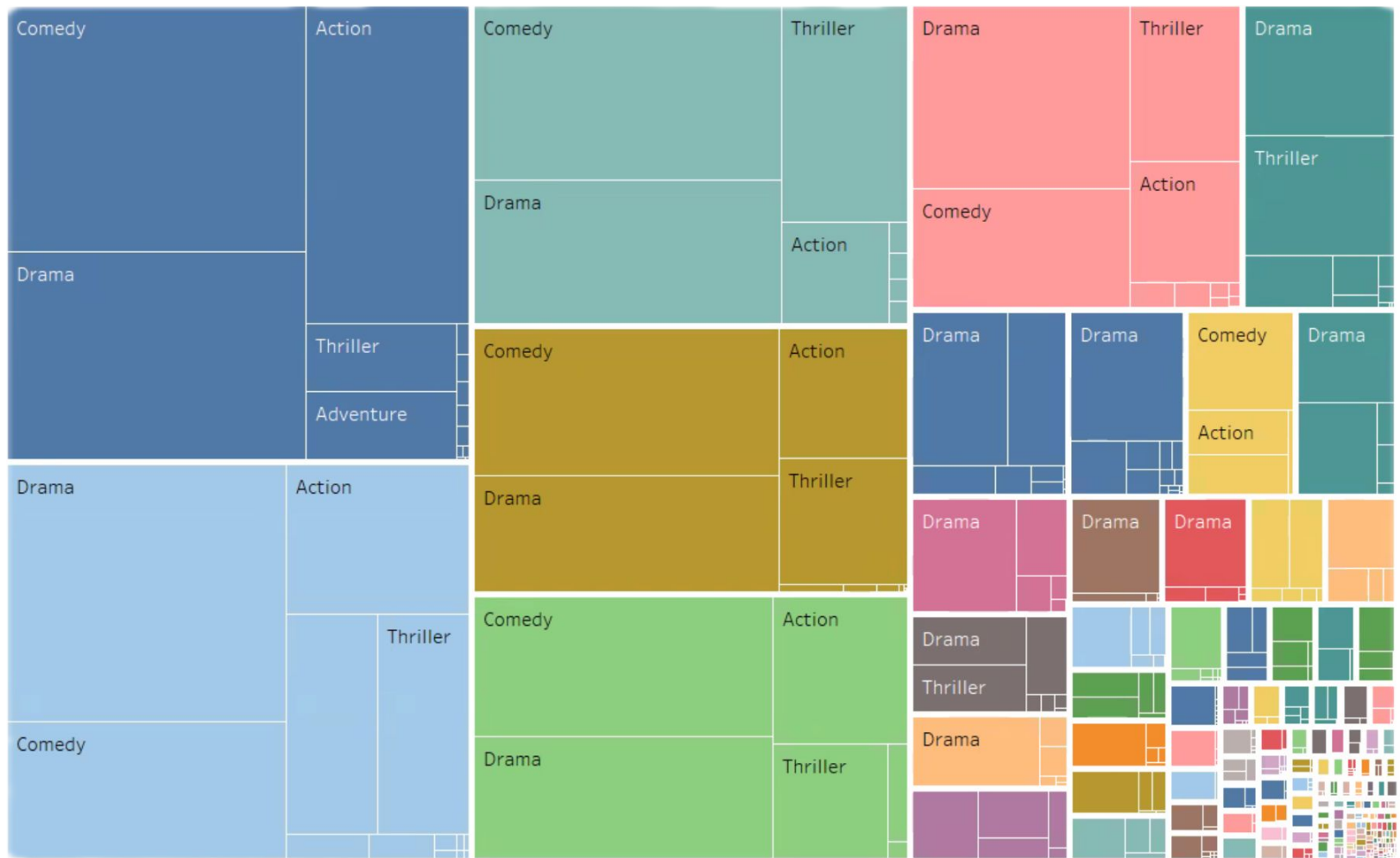
Fetching IMDb data for: Beetlejuice Beetlejuice (2024)
Starting search for: Beetlejuice Beetlejuice (2024)
Search URL: <https://www.imdb.com/find?q=Beetlejuice+Beetlejuice+2024>
Search page fetched successfully for Beetlejuice Beetlejuice
Navigated to movie page URL: https://www.imdb.com/title/tt2049403/?ref_=fn_al_tt_1
Movie page fetched successfully for Beetlejuice Beetlejuice
Rating found: 6.8
Director found: Tim Burton
Budget found: \$100,000,000 (estimated)

Final Merged DataSet (6000 rows of Data)

Release	Gross	Theatres	Total Gross	Distributor	Genres	Rating	Budget	Director	Release Date
Inside Out 2	652980194	4440	652980194	Walt Disney Studios Motion Pictures	Drama	7.6	200000000	Kelsey Mann	2024-06-14 00:00:00
Deadpool & Wolverine	636745858	4330	636745858	Walt Disney Studios Motion Pictures	Comedy	7.7	200000000	Shawn Levy	2024-07-26 00:00:00
Despicable Me 4	361004205	4449	361004205	Universal Pictures	Comedy	6.2	100000000	Chris Renaud	2024-07-03 00:00:00
Beetlejuice Beetlejuice	294072781	4575	294072781	Warner Bros.	Comedy	6.8	100000000	Tim Burton	2024-09-06 00:00:00
Dune: Part Two	282144358	4074	282144358	Warner Bros.	Drama	8.5	190000000	Denis Villeneuve	2024-03-01 00:00:00
Twisters	267762265	4170	267762265	Universal Pictures	Thriller	6.5	155000000	Lee Isaac Chung	2024-07-19 00:00:00
Godzilla x Kong: The New Empire	196350016	3948	196350016	Warner Bros.	Thriller	6.1	135000000	Adam Wingard	2024-03-29 00:00:00
Kung Fu Panda 4	193590620	4067	193590620	Universal Pictures	Comedy	6.3	85000000	Mike Mitchell	2024-03-08 00:00:00
Bad Boys: Ride or Die	193573217	3885	193573217	Sony Pictures Releasing	Comedy	6.6	100000000	Adil El Arbi	2024-06-07 00:00:00
Kingdom of the Planet of the Apes	171130165	4075	171130165	20th Century Studios	Drama	6.9	160000000	Wes Ball	2024-05-10 00:00:00
It Ends with Us	148518266	3839	148518266	Sony Pictures Releasing	Drama	6.5	25000000	Justin Baldoni	2024-08-09 00:00:00
The Wild Robot	140727420	3997	138727420	Universal Pictures	Sci-Fi	8.3	78000000	Chris Sanders	2024-09-27 00:00:00
A Quiet Place: Day One	138930553	3708	138930553	Paramount Pictures	Drama	6.3	67000000	Michael Sarnoski	2024-06-28 00:00:00
Venom: The Last Dance	133825476	4131	129825476	Sony Pictures Releasing	Thriller	6.2	120000000	Kelly Marcel	2024-10-25 00:00:00
Wicked	114000000	3888	114000000	Universal Pictures	Romance	8.1	150000000	Jon M. Chu	2024-11-22 00:00:00
Ghostbusters: Frozen Empire	113376590	4345	113376590	Sony Pictures Releasing	Comedy	6.1	100000000	Gil Kenan	2024-03-22 00:00:00
IF	111149917	4068	111149917	Paramount Pictures	Drama	6.5	110000000	John Krasinski	2024-05-17 00:00:00
Alien: Romulus	105313091	3915	105313091	Walt Disney Studios Motion Pictures	Thriller	7.2	80000000	Fede Alvarez	2024-08-16 00:00:00
Bob Marley: One Love	96893170	3597	96893170	Paramount Pictures	Drama	6.2		Reinaldo Marcus Green	2024-02-14 00:00:00
The Fall Guy	92900355	4008	92900355	Universal Pictures	Drama	6.9	125000000	David Leitch	2024-05-03 00:00:00
The Garfield Movie	91956547	4108	91956547	Sony Pictures Releasing	Comedy	5.7	60000000	Mark Dindal	2024-05-24 00:00:00
Wonka	85272410	4213	218402312	Warner Bros.	Comedy	5.7		Timothe Chalamet	2024-12-15 00:00:00
Longlegs	74346140	2850	74346140	Neon	Thriller	6.7		Osgood Perkins	2024-07-12 00:00:00
Migration	73202330	3839	127306285	Universal Pictures	Comedy	5.5	70000	Joshua Philipp	2024-12-22 00:00:00
Mean Girls	72404248	3826	72404248	Paramount Pictures	Comedy	5.6	36000000	Samantha Jayne	2024-01-12 00:00:00
Civil War	68603430	3929	68603430	A24	Thriller	7	50000000	Alex Garland	2024-04-12 00:00:00

Dataset Features

1. Movie Title (Release)
2. Theatres
3. Total Gross Revenue
4. Distributor
5. Genre
6. Rating
7. Budget
8. Director
9. Release Date



Pages

Columns

MONTH(Release ..

Rows

AVG(Total Gross)

Filters

MONTH(Release Dat..

Marks

Line

Color

Size

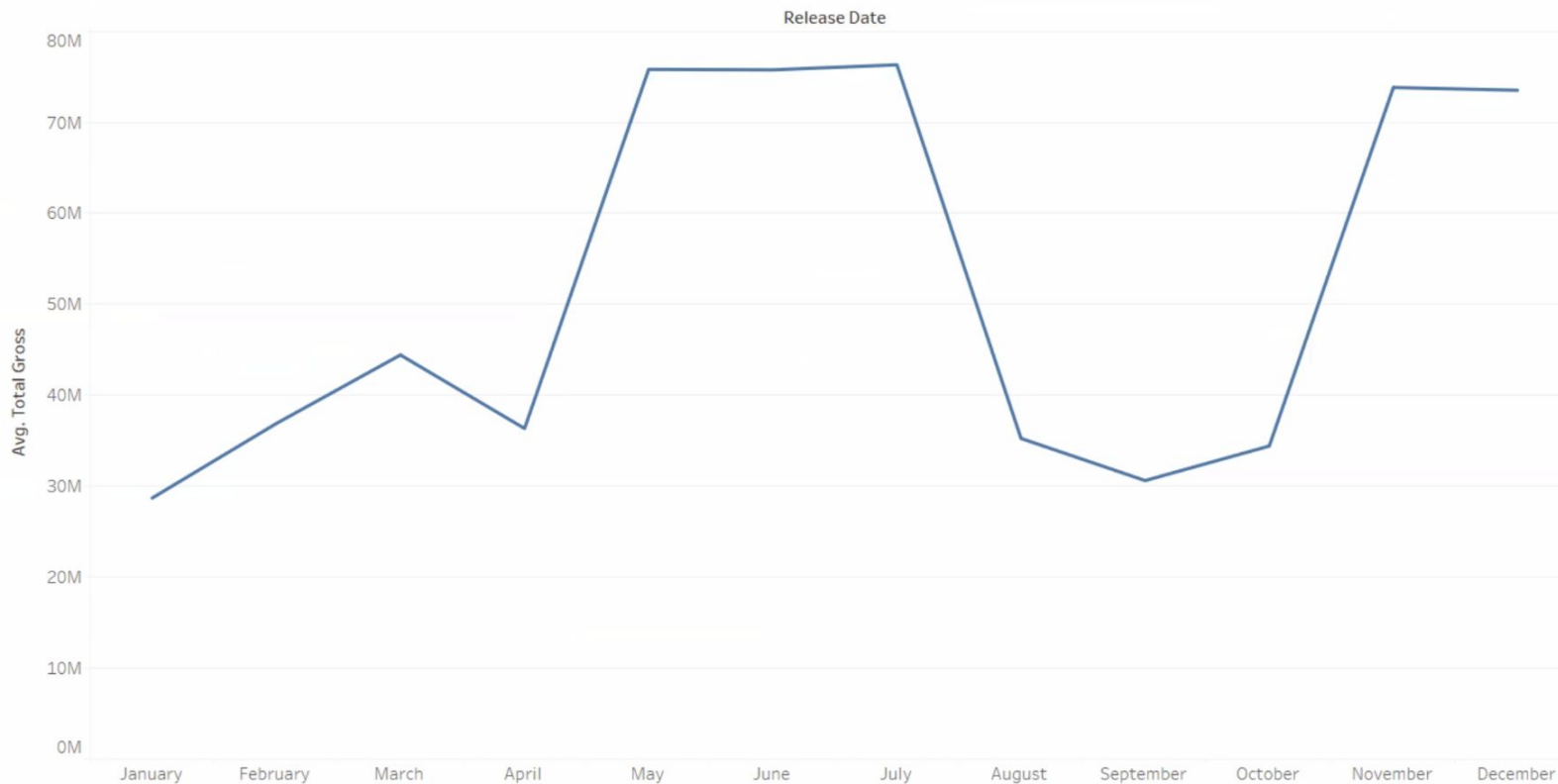
Label

Detail

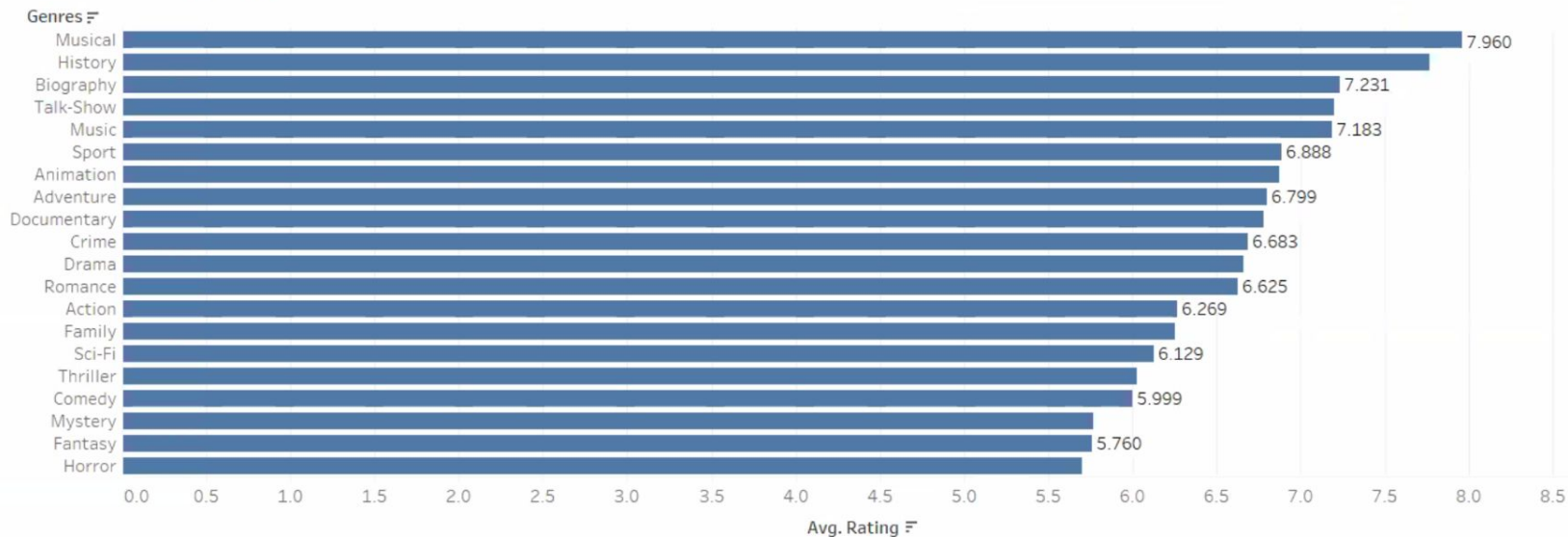
Tooltip

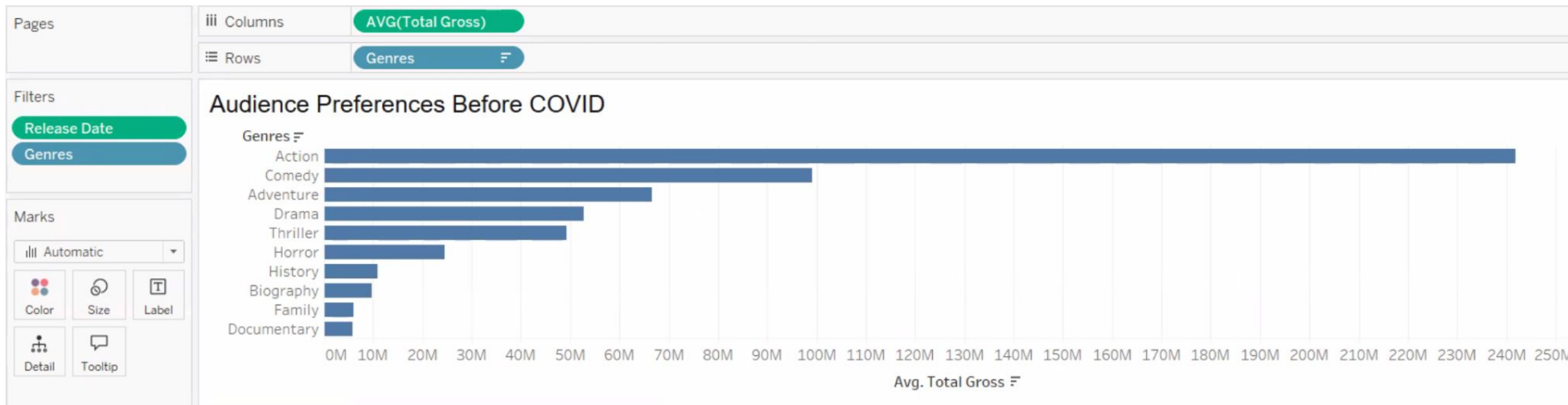
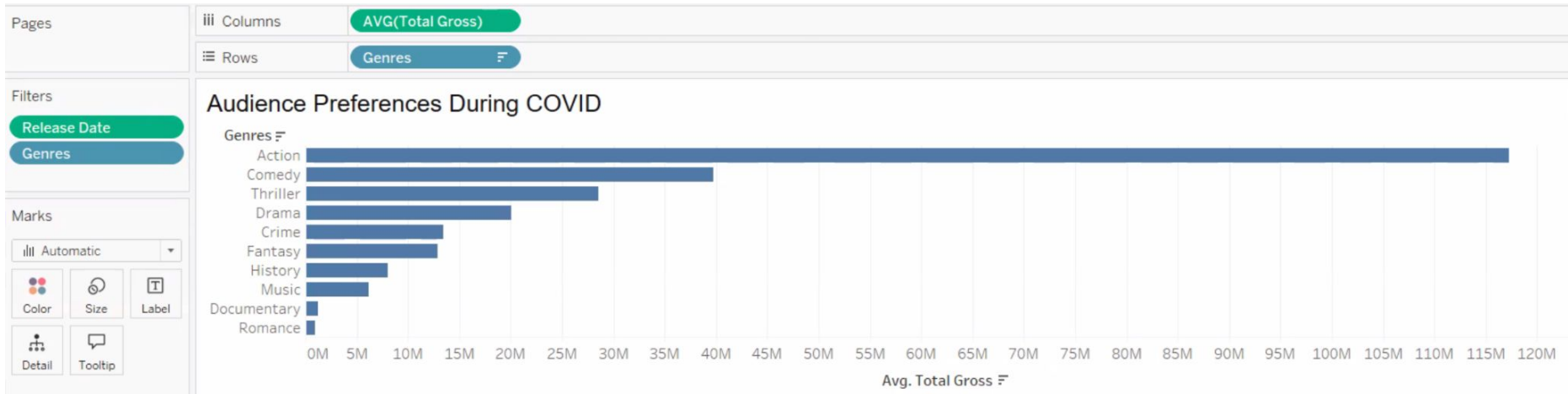
Path

Revenue Trends by Release Month



Genre and Rating (all)



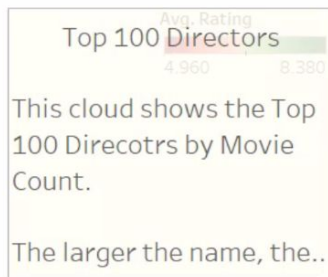






Imdb Movie Project Findings

Top 20 Grossing Movies	Top Distributors	Distributor by Revenue	Revenue by Release Month	Total Revenue YoY	Budget YoY	Budget vs. Revenue	Theatres and Revenue	Genre and Rating	Genre and Rating (all)	Revenue Covid
------------------------	------------------	------------------------	--------------------------	-------------------	------------	--------------------	----------------------	------------------	------------------------	---------------



**Now Let's Move to
Tableau for our
Interactive Analysis**