Ed-Odhiambo Finished Project to submit Latest commit 50ff02e 15 hours ago History

2 contributors

2876 lines (2876 sloc) | 798 KB

Raw Blame

Final Project Submission

Please fill out:

- Student name: ED JOEL OMONDI
- Student pace: Full time
- Scheduled project review date/time: 12/03/2023
- Instructor name: Antonny Muiko, William Okomba, Nikita Njoroge, Lucille Kaleha, Samuel Karu
- Blog post URL:

BOX OFFICE ANALYSIS FOR SUCCESSFUL VENTURES TO MOVIE PRODUCTION

AUTHOR: JOEL OMONDI

Project Overview

The Project is aimed to help Microsoft with strategies to creating a studio for original video content production similar to other big companies. Their aim studio that will be able to compete with the other companies. There are available records of the best films in box office and the project aims to use this analysis of what would be the best course of action for Microsoft since it lacks expertise in the field of movie production. Microsoft needs to identify key such as genre, storyline, and production techniques, that contribute to the success of these movies. The data used comes from TheMovieDB and The show data on genre preferences, budgets, gross revenues, release dates and movie titles available in studios. This data has been cleaned and transfo produce visualizations to determine success factors and other factors that would help advice Microsoft. The methods used are majorly the removal of irrelevant data from our main data set and the transformation through ranking using different columns to determine reasonable results based on a part also helped to remove rows of data that would otherwise burden our analysis due to their size. Based on the observations you have provided, it is seen factors for movies are not limited to genres, storyline, and release date. For example, it is seen that recent releases tend to have higher grossing return releases. In order to produce successful films that will appeal to audiences and generate revenue, it is important to consider a variety of factors, as we the film. Movies with high budgets may not necessarily be successful, and vice versa. It is also important to consider the target audience and ensure th to them. Finally, it is important to ensure that the film has a strong storyline and engaging characters.

Business Understanding

This section discusses the business problems and questions associated with the project. The company wants to create successful movies that can co box office market, it thus needs to identify and understand the critical elements that contribute to the success of current box office hits and how to impl elements in its own movies to ensure success in the movie industry. This section is to handle the business perspective of the project. What are the key questions to consider and their importance for the business?

The pain points of Microsoft should revolve around the lack of experience in the creation of movies and their understanding of the film industry. Anothe trying to develop strategies for the successful production of movies that will appeal to the audience and generate adequate revenue. This would neces and analysis of the data to determine highest grossing movies in recent years and the common genres of them. Some of the questions to be asked ar

1. Which are the key success factors e.g genres and storyline?
2. What types of movies have been top rated and successful in recent years?
3. What are the top grossing films in recent years and what genres do they fall under?
4. Are there any patterns and trends in terms of themes and genres?
5. What strategy can produce successful films that will appeal to audiences and generate revenue?

These questions are important from a business perspective because they provide insights that can inform decision-making around what types of films to market those films effectively. By understanding what has worked in the past, Microsoft's movie studio can increase their chances of creating succes generate revenue and appeal to audiences.

Data Understanding

This section is for the purpose of describing our data. The data is collected from popular sites used to analyse various aspects of movie production. Th Box Office Mojo, IMDB, Rotten Tomatoes, TheMovieDB, and The NUmbers. They are used to track box office revenue margins, provide information ab their cast as well as to provide review from critics about the movies. This project will use dta from TheNovieDB and, The Numbers. For the analysis th from two of such sources that give data on the popularity and profitability of such movies.

TMDB and The Numbers data can be used to identify trends and patterns in the movie industry. There are factors such as budget, performance, genre reception/popularity to consider analyzing. The Numbers data can provide insights into the marketing strategies used for successful films, such as the channels, release dates, and promotional campaigns. There can be opportunities to take advantage of in the data such as dates that have no diversity are in demand but not produced. The data will help analyze feasibility and profitability of the movie industry and identify ideas that would have the bigg data we will focus on factors such as budget, production timelines and gross margins. The next step would be to use these analyses to recommend in

Microsoft for strategies.

Data Analysisis

Data Preparation

For analysis, I have used libraries that use python language as the base such as pandas, matplotlib, seaborn and even numpy. I will use these to clea transform our data to ensure accuracy and consistency of data. These libraries will help to do cleaning, transformation, analysis and enable visualizati necessary so that any errors, inaccuracies and inconsistencies can be identified and corrected. This prevents incorrect analysis from being carried out improve accuracy of the analysis and help in the answering of the questions available in a number of ways.

In [2]:

```
# Importing the necessary standard library packages
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import csv

%matplotlib inline
```

In [3]:

```
# Connecting and reading the data for TMDB
TMDB_data = pd.read_csv("tmdb.movies.csv")
TMDB_data
```

Out[3]:

	Unnamed: 0	genre_ids	id	original_language	original_title	popularity	release_date	title	vote_
0	0	[12, 14, 10751]	12444	en	Harry Potter and the Deathly Hallows: Part 1	33.533	2010-11-19	Harry Potter and the Deathly Hallows: Part 1	
1	1	[14, 12, 16, 10751]	10191	en	How to Train Your Dragon	28.734	2010-03-26	How to Train Your Dragon	
2	2	[12, 28, 878]	10138	en	Iron Man 2	28.515	2010-05-07	Iron Man 2	
3	3	[16, 35, 10751]	862	en	Toy Story	28.005	1995-11-22	Toy Story	
4	4	[28, 878, 12]	27205	en	Inception	27.920	2010-07-16	Inception	
...	...	...	...	...	...	...	...	...	

26512	26512	[27, 18]	488143	en	Laboratory Conditions	0.600	2018-10-13	Laboratory Conditions
26513	26513	[18, 53]	485975	en	_EXHIBIT_84xxx_	0.600	2018-05-01	_EXHIBIT_84xxx_
26514	26514	[14, 28, 12]	381231	en	The Last One	0.600	2018-10-01	The Last One
26515	26515	[10751, 12, 28]	366854	en	Trailer Made	0.600	2018-06-22	Trailer Made
26516	26516	[53, 27]	309885	en	The Church	0.600	2018-10-05	The Church

26517 rows × 10 columns

In [4]:

```python
# Connecting and reading data from The Numbers
TheNumbers_data = pd.read_csv("tn.movie_budgets.csv")
TheNumbers_data
```

Out[4]:

	id	release_date	movie	production_budget	domestic_gross	worldwide_gross
0	1	Dec 18, 2009	Avatar	$425,000,000	$760,507,625	$2,776,345,279
1	2	May 20, 2011	Pirates of the Caribbean: On Stranger Tides	$410,600,000	$241,063,875	$1,045,663,875
2	3	Jun 7, 2019	Dark Phoenix	$350,000,000	$42,762,350	$149,762,350
3	4	May 1, 2015	Avengers: Age of Ultron	$330,600,000	$459,005,868	$1,403,013,963
4	5	Dec 15, 2017	Star Wars Ep. VIII: The Last Jedi	$317,000,000	$620,181,382	$1,316,721,747
...	...	...	...	...	...	...
5777	78	Dec 31, 2018	Red 11	$7,000	$0	$0
5778	79	Apr 2, 1999	Following	$6,000	$48,482	$240,495
5779	80	Jul 13, 2005	Return to the Land of Wonders	$5,000	$1,338	$1,338
5780	81	Sep 29, 2015	A Plague So Pleasant	$1,400	$0	$0
5781	82	Aug 5, 2005	My Date With Drew	$1,100	$181,041	$181,041

5782 rows × 6 columns

The data that has been retrieved may have inconsistencies and errors such as missing values, outdated values, outliers and irrelevant data for our ana
data more useful it is necessary to clean and transform the data so that only the necessary and accurate data may remain. Data cleaning involves che
values, duplicated entries, invalid and conflicting entries. There are a number of ways to use the data provided as from it, it is possible to get descriptiv

general information of the data that can help with providing insights on the course of action.

In [5]:
```
# Checking whether there is any missing data in our data frame from TheMOvieDB
TMDB_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 26517 entries, 0 to 26516
Data columns (total 10 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Unnamed: 0         26517 non-null  int64  
 1   genre_ids          26517 non-null  object 
 2   id                 26517 non-null  int64  
 3   original_language  26517 non-null  object 
 4   original_title     26517 non-null  object 
 5   popularity         26517 non-null  float64
 6   release_date       26517 non-null  object 
 7   title              26517 non-null  object 
 8   vote_average       26517 non-null  float64
 9   vote_count         26517 non-null  int64  
dtypes: float64(2), int64(3), object(5)
memory usage: 2.0+ MB
```

In [6]:
```
# Checking for missing data in the data from The Numbers
TheNumbers_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5782 entries, 0 to 5781
Data columns (total 6 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   id                 5782 non-null   int64 
 1   release_date       5782 non-null   object
 2   movie              5782 non-null   object
 3   production_budget  5782 non-null   object
 4   domestic_gross     5782 non-null   object
 5   worldwide_gross    5782 non-null   object
dtypes: int64(1), object(5)
memory usage: 271.2+ KB
```

The above method is used to get the general information of our data frames. For example, the RangeIndex tells how many entries were available, the variables are represented/examined, the non-null count tells how many non-null/valid values exist in the data while the 'Dtype' section tells us what typ dealing with dealing with.

From the information, it is seen that both sets have complete data with no missing values but that does not mean all the data can be used. Other forms
transformation may be needed. Some case examples are the columns that are useless to the problem statements. They can be done away with in ger
the column and check the shape to confirm the new data frame that should have less columns after.

In [7]:
```
# Dropping our unnecessary columns in TMDB and confirming through its shape
TMDB_data.drop(['Unnamed: 0','original_language', 'original_title'],axis=1, inplace=True)
print("Shape of IMDB_data: ", TMDB_data.shape)
print("Columns now in IMDB_data: ", list(TMDB_data.columns))
```

```
Shape of IMDB_data:  (26517, 7)
Columns now in IMDB_data:  ['genre_ids', 'id', 'popularity', 'release_date', 'title', 'vote_average', 'vote_count']
```

In [8]:
```
"""
In order to check for duplicated though-out the whole data, we are going to use a method that can draw out a boolean.
This is a True or False response where any mention of True for unique values indicates a duplicate is present.
"""

TMDB_data.duplicated()
set(TMDB_data.duplicated())
```

Out[8]:
```
{False, True}
```

In [9]:
```
# Dropping the duplicates from the data frame
TMDB_data.drop_duplicates(inplace=True)
TMDB_data.shape
```

Out[9]:
```
(25497, 7)
```

In [10]:
```
# Dropping rows in IMDB with less than 2000 vote counts or less that 6.0 average rating
for x in TMDB_data.index:
    if TMDB_data.loc[x, "vote_count"] < 2000 or TMDB_data.loc[x, "vote_average"] < 6.0:
        TMDB_data.drop(x, inplace=True)

TMDB_data
```

Out[10]:

	genre_ids	id	popularity	release_date	title	vote_average	vote_count
0	[12, 14, 10751]	12444	33.533	2010-11-19	Harry Potter and the Deathly Hallows: Part 1	7.7	10788

1	[14, 12, 16, 10751]	10191	28.734	2010-03-26	How to Train Your Dragon	7.7	7610
2	[12, 28, 878]	10138	28.515	2010-05-07	Iron Man 2	6.8	12368
3	[16, 35, 10751]	862	28.005	1995-11-22	Toy Story	7.9	10174
4	[28, 878, 12]	27205	27.920	2010-07-16	Inception	8.3	22186
...	...	...	...	...	...	...	...
23916	[9648, 53]	401981	18.117	2018-03-02	Red Sparrow	6.5	3406
23939	[35, 18, 10749]	462919	16.804	2018-09-07	Sierra Burgess Is a Loser	6.5	2243
23951	[12, 14]	11253	16.266	2008-07-11	Hellboy II: The Golden Army	6.7	2820
23970	[35, 18, 10749]	449176	15.608	2018-03-16	Love, Simon	8.2	3165
23999	[53, 27]	460019	14.354	2018-04-13	Truth or Dare	6.0	2005

500 rows × 7 columns

In [11]:

```
# Getting the top 100 vote averages
TMDB_TopAvgVoted = TMDB_data.sort_values(by='vote_average', ascending=False).head(100)
TMDB_TopAvgVoted
```

Out[11]:

	genre_ids	id	popularity	release_date	title	vote_average	vote_count
17389	[10749, 16, 18]	372058	28.238	2017-04-07	Your Name.	8.6	4161
23861	[18, 36, 10752]	424	25.334	1993-12-15	Schindler's List	8.5	8065
14173	[16, 10751, 14]	129	32.043	2002-09-20	Spirited Away	8.5	7424
5201	[18, 80]	311	17.717	1984-06-01	Once Upon a Time in America	8.4	2243
23812	[28, 12, 16, 878, 35]	324857	60.534	2018-12-14	Spider-Man: Into the Spider-Verse	8.4	4048
...	...	...	...	...	...	...	...
17401	[28, 12, 878]	330459	21.401	2016-12-16	Rogue One: A Star Wars Story	7.5	9296
7895	[27, 53]	138843	18.886	2013-07-19	The Conjuring	7.5	5912
17436	[18]	334541	16.638	2016-11-18	Manchester by the Sea	7.5	3176
20704	[18]	389015	15.407	2017-12-08	I, Tonya	7.5	2904
20822	[12, 18, 878, 28]	387426	10.805	2017-06-28	Okja	7.5	2146

100 rows × 7 columns

In [12]:

```
# Checking for duplicates in The Numbers data
TheNumbers_data.duplicated()
set(TheNumbers_data.duplicated())
```

Out[12]:

```
{False}
```

In [13]:

```
# Converting The Numbers data to integers for descriptive analysis
# Converting for the Production Budget column
TheNumbers_data['production_budget'] = TheNumbers_data['production_budget'].str.replace('[$,]', '', regex=True)
TheNumbers_data = TheNumbers_data.astype({"production_budget": int})

# Converting for the Domestic gross column
TheNumbers_data["domestic_gross"] = TheNumbers_data["domestic_gross"].str.replace('[$,]','', regex=True)
TheNumbers_data["domestic_gross"] = TheNumbers_data["domestic_gross"].astype(str).astype(int)

# Converting for the Worldwide Gross column
TheNumbers_data["worldwide_gross"] = TheNumbers_data["worldwide_gross"].str.replace('[$,]','', regex=True)
TheNumbers_data["worldwide_gross"] = TheNumbers_data["worldwide_gross"].astype(int)

# Printing out the data types
print(TheNumbers_data.dtypes)

# Calling the data frame
TheNumbers_data
```

```
id                    int64
release_date         object
movie                object
production_budget     int64
domestic_gross        int64
worldwide_gross       int64
dtype: object
```

Out[13]:

	id	release_date	movie	production_budget	domestic_gross	worldwide_gross
0	1	Dec 18, 2009	Avatar	425000000	760507625	2776345279
1	2	May 20, 2011	Pirates of the Caribbean: On Stranger Tides	410600000	241063875	1045663875
2	3	Jun 7, 2019	Dark Phoenix	350000000	42762350	149762350
3	4	May 1, 2015	Avengers: Age of Ultron	330600000	459005868	1403013963

4	5	Dec 15, 2017	Star Wars Ep. VIII: The Last Jedi	317000000	620181382	1316721747
...	...	...	...	...	...	...
5777	78	Dec 31, 2018	Red 11	7000	0	0
5778	79	Apr 2, 1999	Following	6000	48482	240495
5779	80	Jul 13, 2005	Return to the Land of Wonders	5000	1338	1338
5780	81	Sep 29, 2015	A Plague So Pleasant	1400	0	0
5781	82	Aug 5, 2005	My Date With Drew	1100	181041	181041

5782 rows × 6 columns

Data Modeling

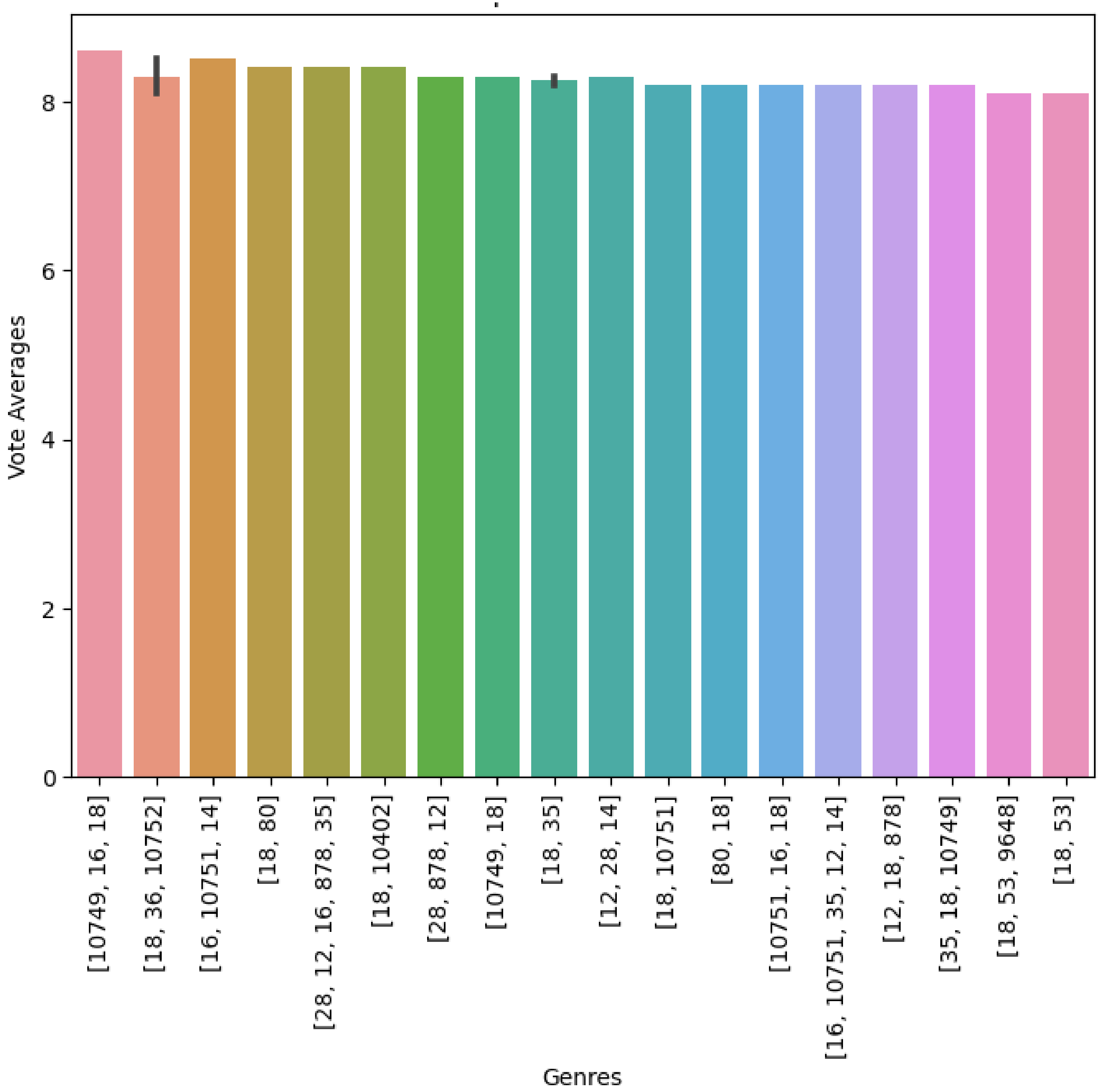
This section is where the cleaned data can be transformed and be used to derive insights from. The data will be analyzed using a number of combinat information using visualizations. This should be able to provide descriptive insights that can help us to be able to answer project questions.An example transformation is above where rows have been removed for movies that have vote counts of less than 2000. This is so that we can remain with data th viewership as shown by vote counts. This provides a good demographic for analysis. Going ahead and limiting the movies to those with ratings of 6.0 we can weed put non-performing movies and remain with those that could help give an idea of those that are doing well since those would be more us would help advice on what can be done to get to the same level.

Given the data from The Numbers, the same approach applies as we can weed out the non-performing movies using the gross margins as well as the will help us see which movies have higher revenue output and well as low budget costs as well as the profitability. Other iterations can be done to prov trends and even popularity with the data available. Using visualization methods we can be able to observe these factors in a non-technical manner and ways forward.

In [14]:

```
# Genre vs vote average:
fig, ax = plt.subplots(figsize=(8,6))
sns.barplot(x=TMDB_TopAvgVoted['genre_ids'].head(20), y=TMDB_TopAvgVoted['vote_average'].head(20))
plt.title('Top Voted Genres')
plt.xlabel('Genres')
plt.ylabel('Vote Averages')
plt.xticks(rotation =90)
plt.show()
```

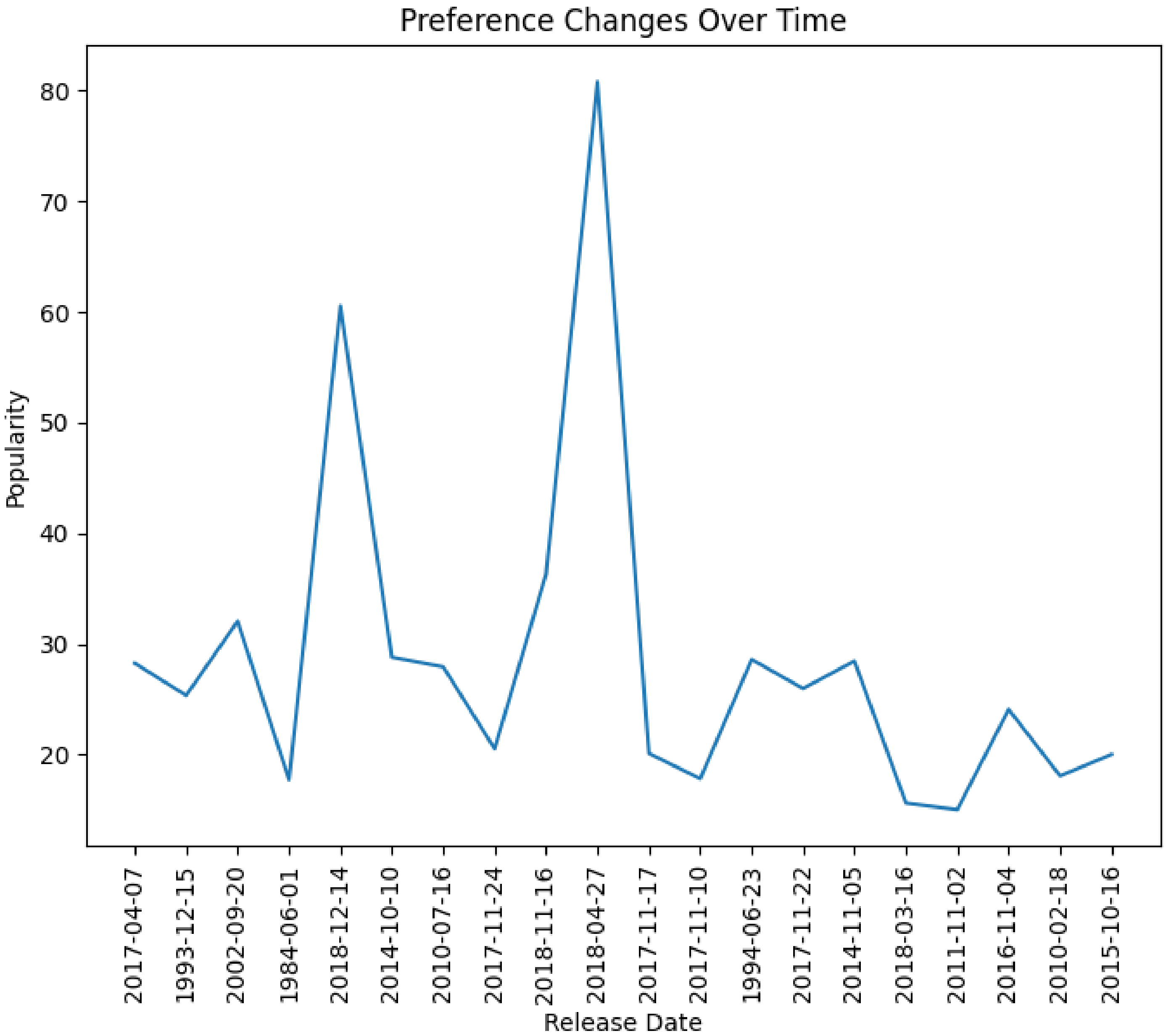
Top Voted Genres



In [33]:

```
# Release date vs popularity:
fig, ax = plt.subplots(figsize=(8,6))
```

```
plt.plot(IMDB_TopAvgVoted['release_date'].head(20), IMDB_TopAvgVoted['popularity'].head(20))
plt.title('Preference Changes Over Time')
plt.xlabel('Release Date')
plt.ylabel('Popularity')
plt.xticks(rotation=90)
plt.show()
```



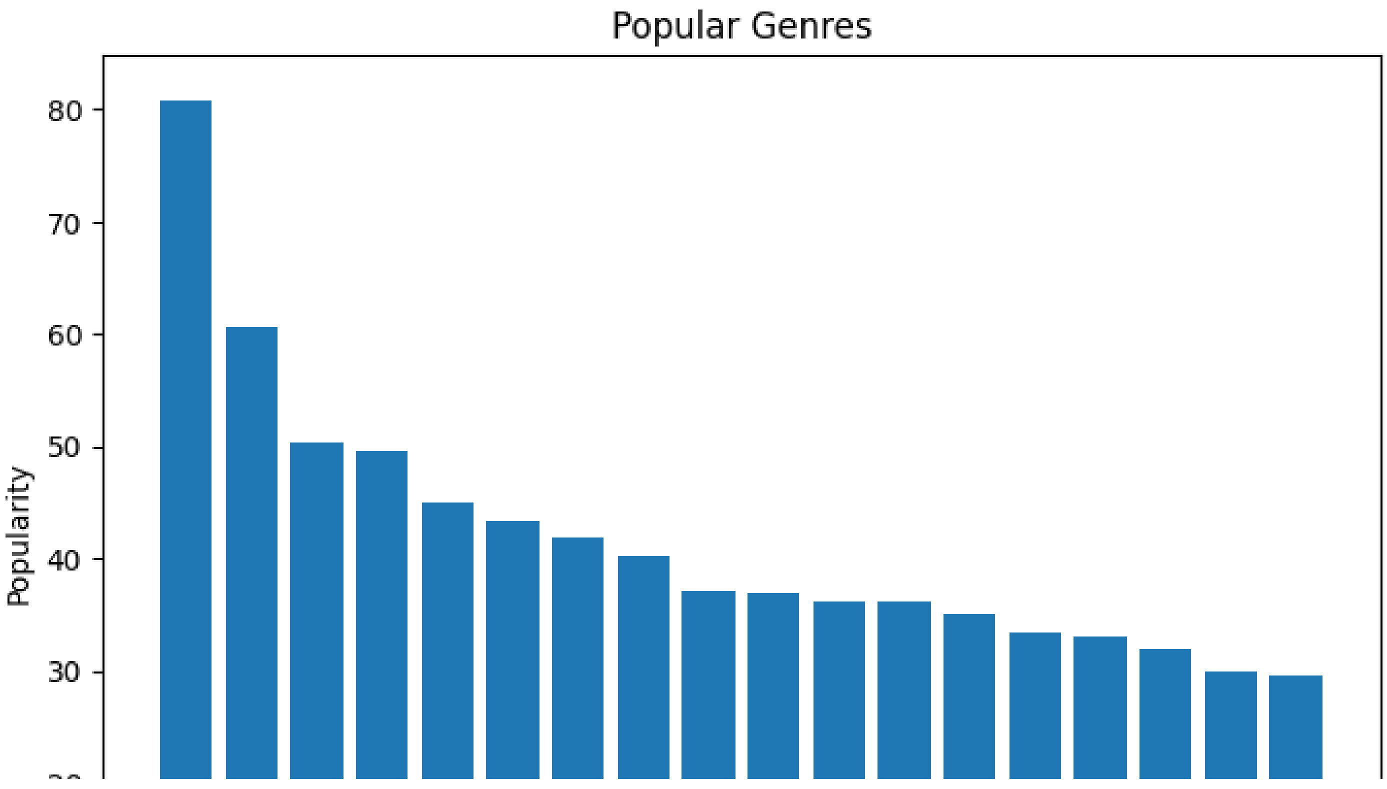
From the list of top 20 voted movies, we observe that there have been peaks of popularity in recent years for these top voted movies which indicates tl
production methods as well as audience availability has increased. This is a good sign as it indicates that the business venture is moving on the right c

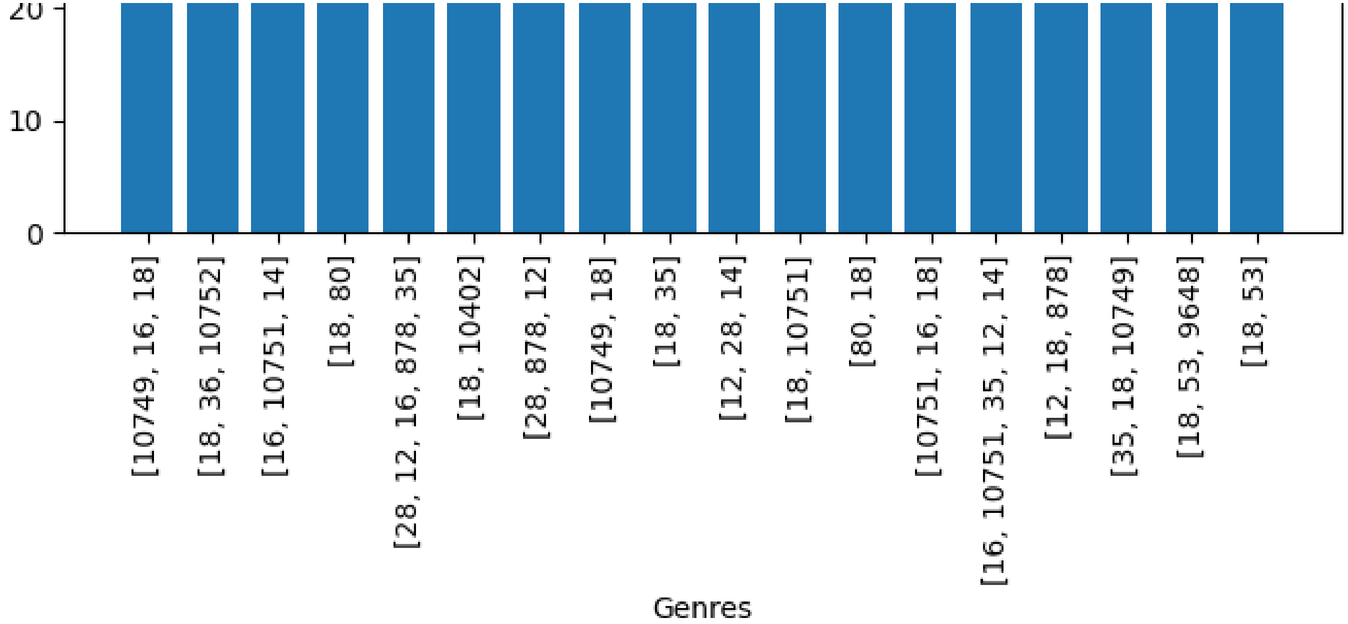
TMDB and The Numbers can provide insights into audience demographics, ratings, and reviews. By analyzing these factors, the head of the movie stu
trends and preferences in audience behavior and tailor their content accordingly.

Overall, using data from TMDB and The Numbers can help the head of Microsoft's new movie studio make informed decisions about which types of m
how to market them for maximum success.

In [37]:

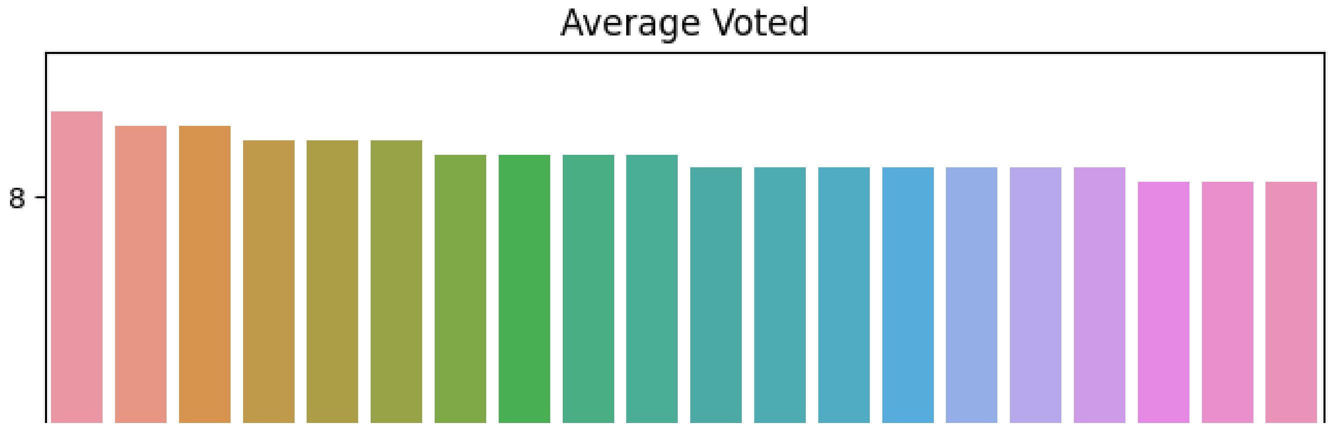
```
# Genre vs popularity:
fig, ax = plt.subplots(figsize=(8,6))
plt.bar(TMDB_TopAvgVoted['genre_ids'].head(20), TMDB_TopAvgVoted['popularity'].sort_values(ascending=False).head(20))
plt.title('Popular Genres')
plt.xlabel('Genres')
plt.ylabel('Popularity')
plt.xticks(rotation=90)
plt.show()
```

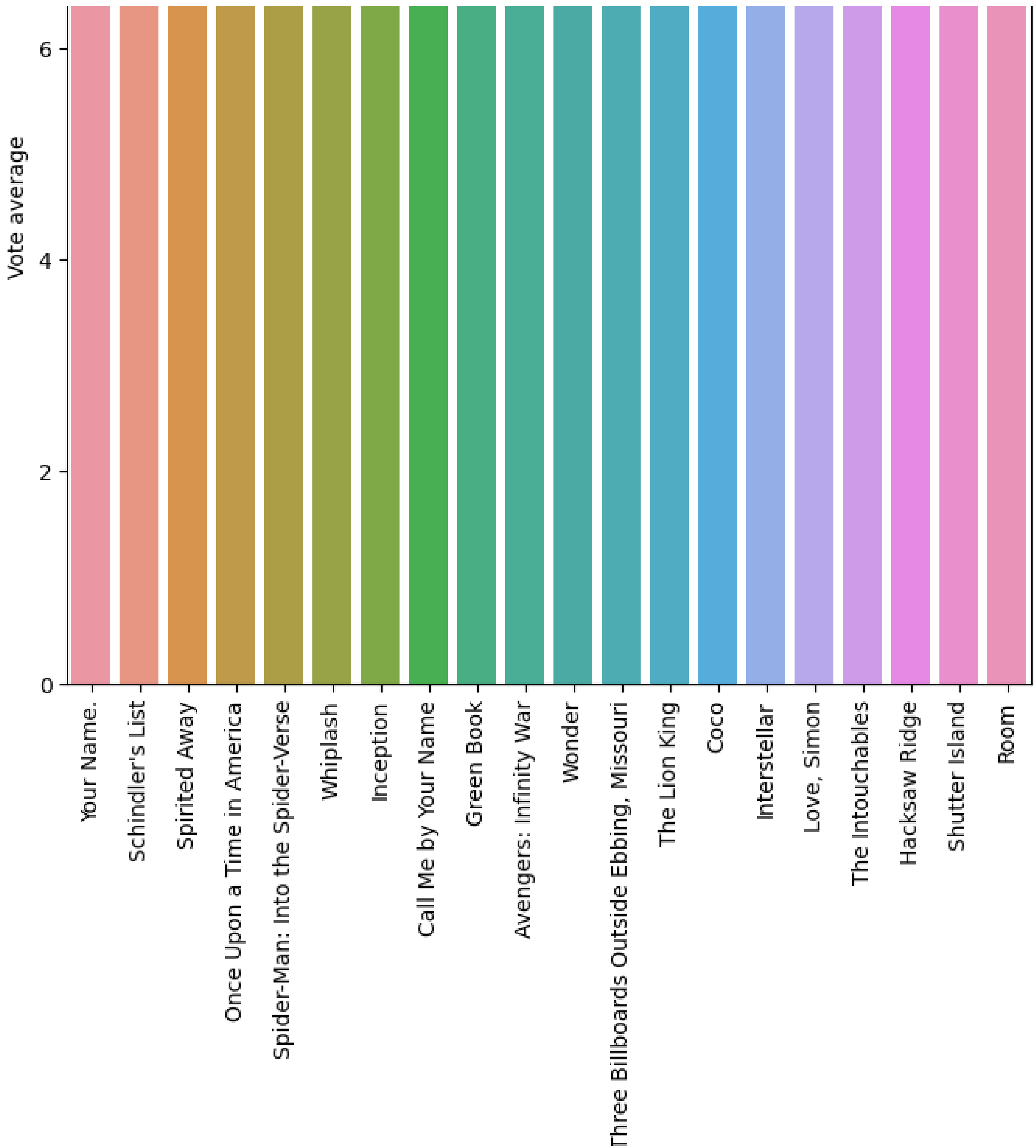




In [35]:

```
# Plotting for Top 20 Average Voted
fig, axes = plt.subplots(figsize=(8,6))

sns.barplot(x=TMDB_TopAvgVoted['title'].head(20), y=TMDB_TopAvgVoted['vote_average'].head(20))
plt.title('Average Voted')
plt.xlabel('Movie')
plt.ylabel('Vote average')
plt.xticks(rotation = 90)

plt.show()
```





Vote average
0
2
4
6
Your Name.
Schindler's List
Spirited Away
Once Upon a Time in America
Spider-Man: Into the Spider-Verse
Whiplash
Inception
Call Me by Your Name
Green Book
Avengers: Infinity War
Wonder
Three Billboards Outside Ebbing, Missouri
The Lion King
Coco
Interstellar
Love, Simon
The Intouchables
Hacksaw Ridge
Shutter Island
Room

Movie

In [18]:

```python
# Sorting data frame according to popularity
TMDB_TopPopular = TMDB_data.sort_values(by='popularity', ascending=False).head(100)
TMDB_TopPopular
```

Out[18]:

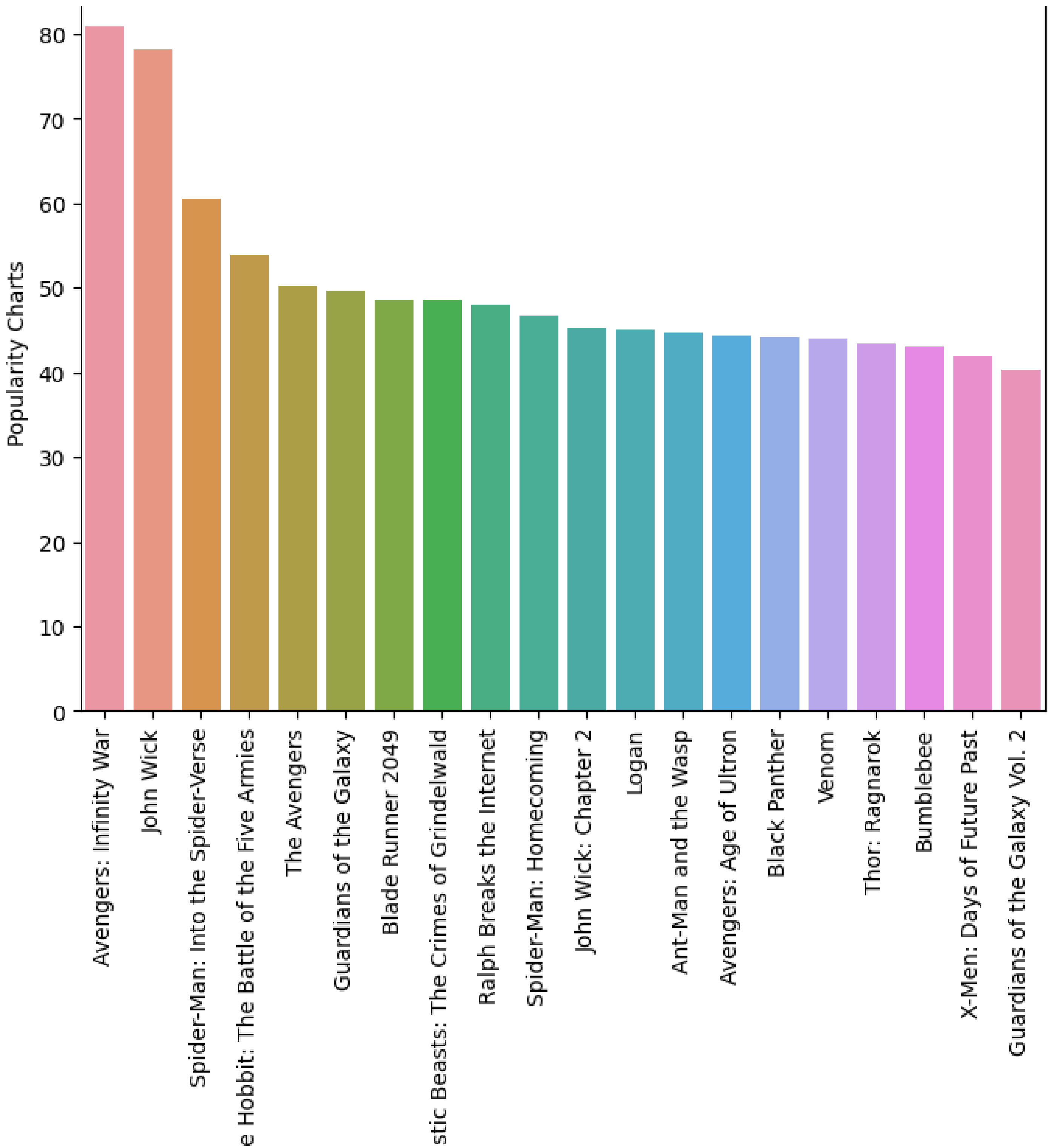
	genre_ids	id	popularity	release_date	title	vote_average	vote_count
23811	[12, 28, 14]	299536	80.773	2018-04-27	Avengers: Infinity War	8.3	13948
11019	[28, 53]	245891	78.123	2014-10-24	John Wick	7.2	10081
23812	[28, 12, 16, 878, 35]	324857	60.534	2018-12-14	Spider-Man: Into the Spider-Verse	8.4	4048
11020	[28, 12, 14]	122917	53.783	2014-12-17	The Hobbit: The Battle of the Five Armies	7.3	8392
5179	[878, 28, 12]	24428	50.289	2012-05-04	The Avengers	7.6	19673
...	...	...	...	...	...	...	...
23858	[80, 35, 28, 53]	402900	26.009	2018-06-08	Ocean's Eight	6.9	3709
20635	[16, 10751, 35, 12, 14]	354912	25.961	2017-11-22	Coco	8.2	8669
2474	[28, 12, 878]	1771	25.808	2011-07-22	Captain America: The First Avenger	6.9	12810
17392	[28, 35, 53]	291805	25.805	2016-06-10	Now You See Me 2	6.8	6744
17393	[27, 53]	381288	25.783	2016-09-26	Split	7.2	10375

100 rows × 7 columns

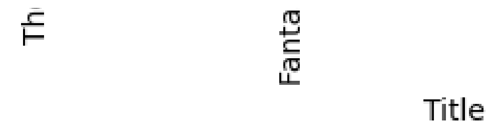
In [38]:

```python
# Title vs Popularity
fig, axes = plt.subplots(figsize=(8,6))

sns.barplot(x=TMDB_TopPopular['title'].head(20), y=TMDB_TopPopular['popularity'].head(20))
plt.title('Popularity')
plt.ylabel('Popularity Charts')
plt.xlabel('Title')
plt.xticks(rotation = 90)

plt.show()
```

Popularity



Popularity Charts
0
10
20
30
40
50
60
70
80
Avengers: Infinity War
John Wick
Spider-Man: Into the Spider-Verse
e Hobbit: The Battle of the Five Armies
The Avengers
Guardians of the Galaxy
Blade Runner 2049
stic Beasts: The Crimes of Grindelwald
Ralph Breaks the Internet
Spider-Man: Homecoming
John Wick: Chapter 2
Logan
Ant-Man and the Wasp
Avengers: Age of Ultron
Black Panther
Venom
Thor: Ragnarok
Bumblebee
X-Men: Days of Future Past
Guardians of the Galaxy Vol. 2



In [20]:

```
# Filter for movies with reasonable budgets with high returns
TheNumbers_profitable = TheNumbers_data.loc[TheNumbers_data['production_budget'] > 100000]
TheNumbers_profitable
```

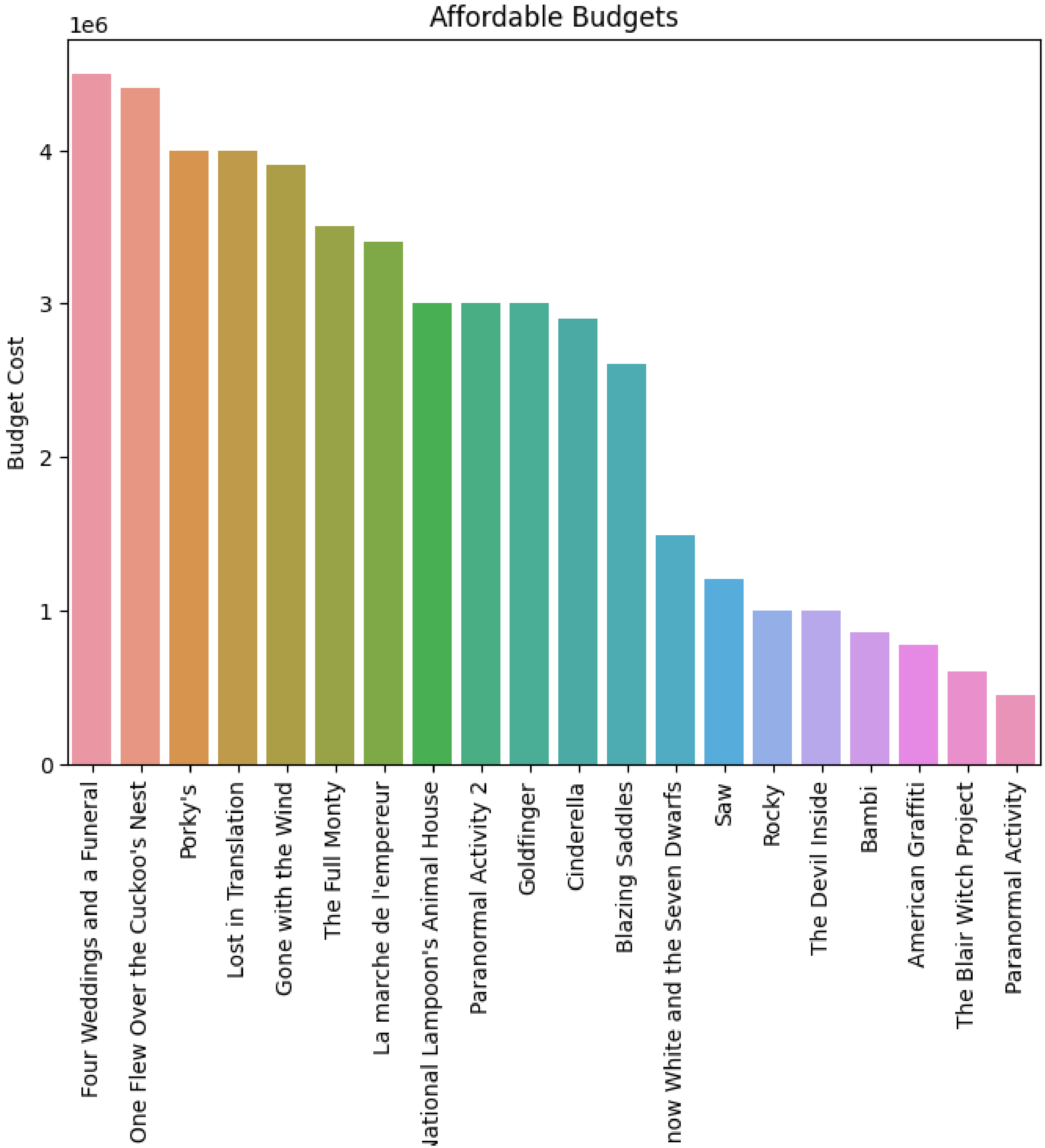
Out[20]:

	id	release_date	movie	production_budget	domestic_gross	worldwide_gross
0	1	Dec 18, 2009	Avatar	425000000	760507625	2776345279
1	2	May 20, 2011	Pirates of the Caribbean: On Stranger Tides	410600000	241063875	1045663875
2	3	Jun 7, 2019	Dark Phoenix	350000000	42762350	149762350
3	4	May 1, 2015	Avengers: Age of Ultron	330600000	459005868	1403013963
4	5	Dec 15, 2017	Star Wars Ep. VIII: The Last Jedi	317000000	620181382	1316721747
...	...	...	...	...	...	...
5674	75	Dec 31, 2007	A Dog's Breakfast	120000	0	0
5675	76	May 24, 2016	Une Femme MariÃ©e	120000	0	0
5676	77	Oct 1, 1968	Night of the Living Dead	114000	12087064	30087064
5677	78	Feb 8, 1915	The Birth of a Nation	110000	10000000	11000000
5678	79	Oct 3, 2003	The Work and the Story	103000	16137	16137

5679 rows × 6 columns

In [39]:

```
# Showing movies with cheapest budget costs
fig, axes = plt.subplots(figsize=(8,6))

sns.barplot(x=TheNumbers_profitable['movie'].tail(20), y=TheNumbers_profitable['production_budget'].tail(20))
plt.title('Affordable Budgets')
plt.xlabel('Movie Title')
plt.ylabel('Budget Cost')
plt.xticks(rotation = 90)

plt.show()
```


Affordable Budgets
1e6
Budget Cost
0
1
2
3
4
Four Weddings and a Funeral
One Flew Over the Cuckoo's Nest
Porky's
Lost in Translation
Gone with the Wind
The Full Monty
La marche de l'empereur
National Lampoon's Animal House
Paranormal Activity 2
Goldfinger
Cinderella
Blazing Saddles
Snow White and the Seven Dwarfs
Saw
Rocky
The Devil Inside
Bambi
American Graffiti
The Blair Witch Project
Paranormal Activity

Movie Title

In [22]:
```python
TheNumbers_profitable = TheNumbers_profitable.loc[TheNumbers_profitable["worldwide_gross"] > 100000000]
TheNumbers_profitable
```

Out[22]:

	id	release_date	movie	production_budget	domestic_gross	worldwide_gross
0	1	Dec 18, 2009	Avatar	425000000	760507625	2776345279
1	2	May 20, 2011	Pirates of the Caribbean: On Stranger Tides	410600000	241063875	1045663875
2	3	Jun 7, 2019	Dark Phoenix	350000000	42762350	149762350
3	4	May 1, 2015	Avengers: Age of Ultron	330600000	459005868	1403013963
4	5	Dec 15, 2017	Star Wars Ep. VIII: The Last Jedi	317000000	620181382	1316721747
...	...	...	...	...	...	...
5211	12	Jan 6, 2012	The Devil Inside	1000000	53262945	101759490
5346	47	Aug 13, 1942	Bambi	858000	102797000	268000000
5372	73	Aug 11, 1973	American Graffiti	777000	115000000	140000000
5406	7	Jul 14, 1999	The Blair Witch Project	600000	140539099	248300000
5492	93	Sep 25, 2009	Paranormal Activity	450000	107918810	194183034

1414 rows × 6 columns

In [23]:
```python
TheNum_Budgetable = TheNumbers_profitable.sort_values(by='production_budget').head(100)
TheNum_Budgetable
```

Out[23]:

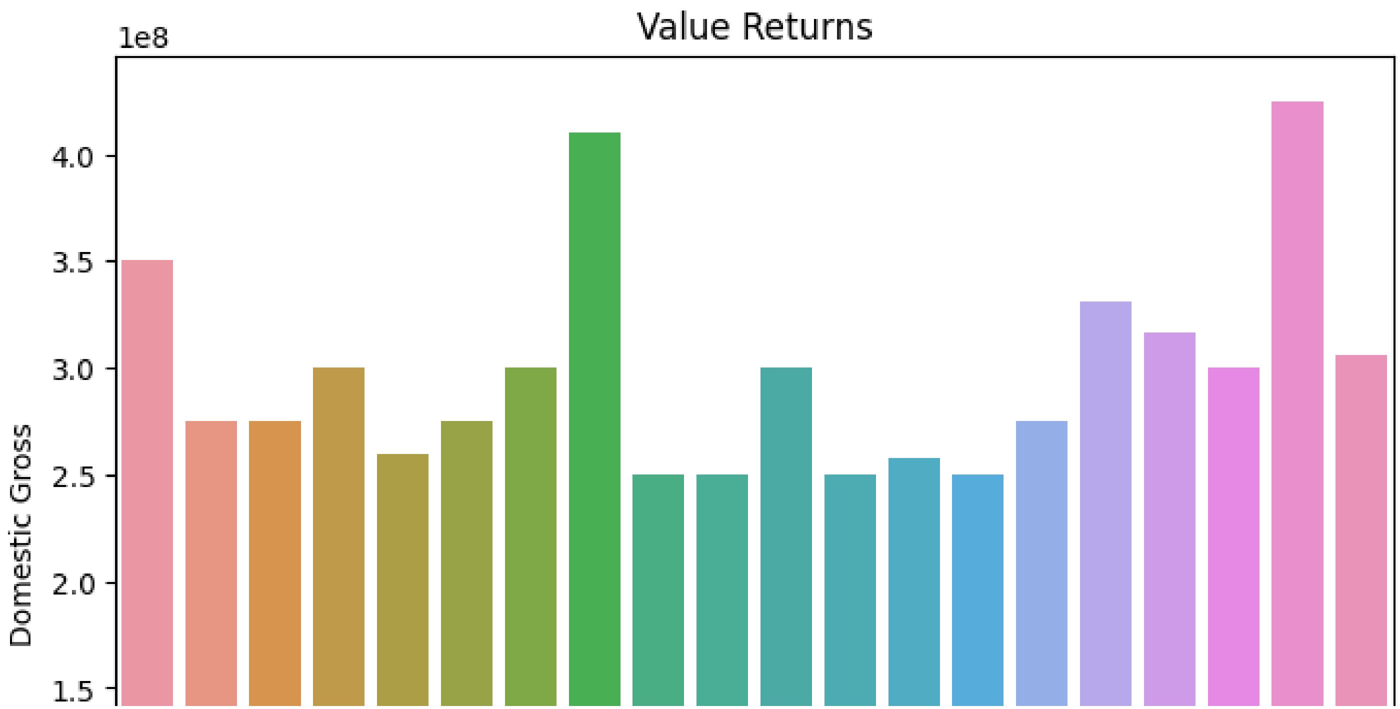
	id	release_date	movie	production_budget	domestic_gross	worldwide_gross
5492	93	Sep 25, 2009	Paranormal Activity	450000	107918810	194183034
5406	7	Jul 14, 1999	The Blair Witch Project	600000	140539099	248300000
5372	73	Aug 11, 1973	American Graffiti	777000	115000000	140000000
5346	47	Aug 13, 1942	Bambi	858000	102797000	268000000
5210	11	Nov 21, 1976	Rocky	1000000	117235147	225000000
...	...	...	...	...	...	...

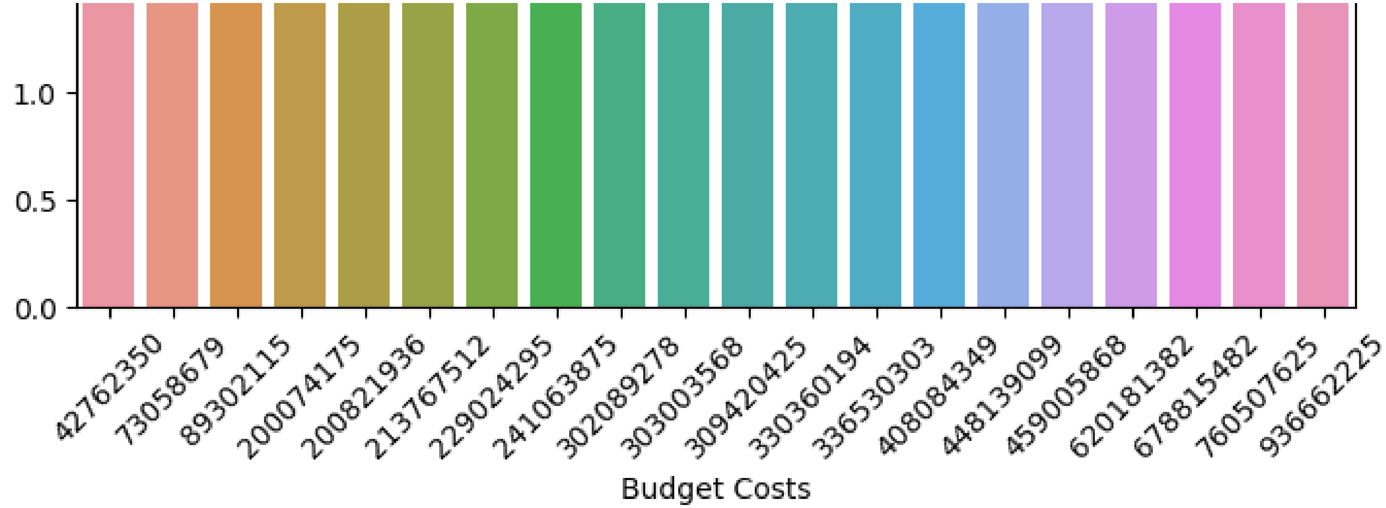
3246	47	Jan 12, 2001	Save the Last Dance	13000000	91038276	122244329
3253	54	Jun 24, 2016	The Shallows	13000000	55121623	118763442
3282	83	May 29, 2009	MÃ¤n som hatar kvinnor	13000000	12749992	109421911
3285	86	Jul 17, 2015	Bajrangi Bhaijaan	13000000	8178001	121778347
3248	49	Jul 4, 2018	The First Purge	13000000	69488745	136617305

100 rows × 6 columns

In [40]:

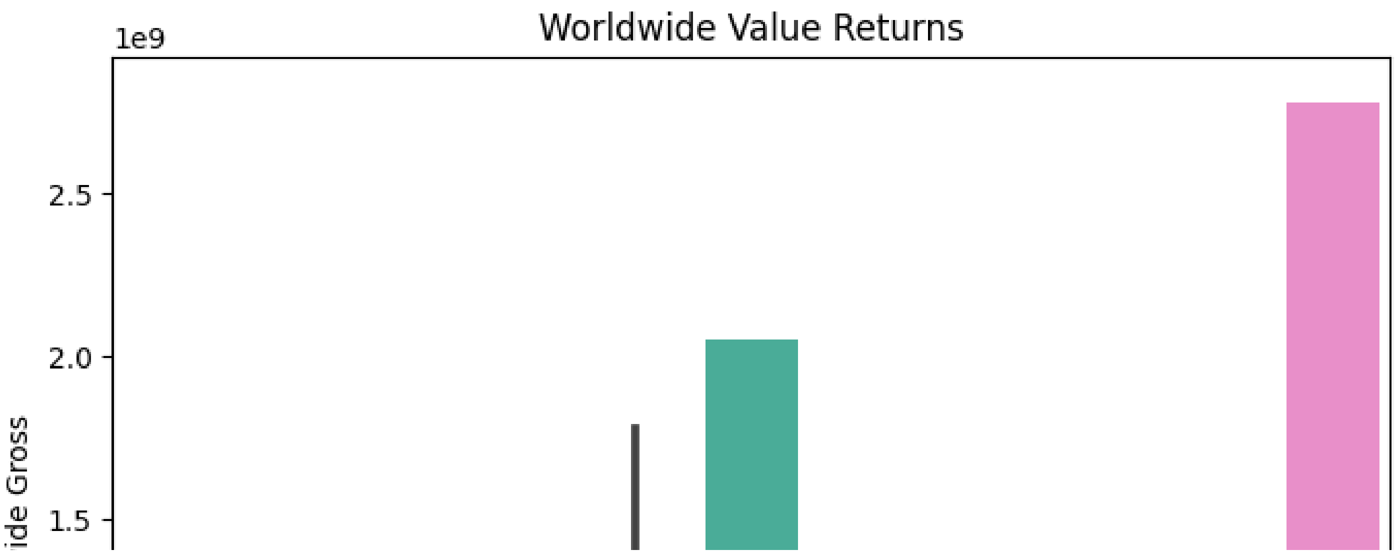
```
# Production Budget vs Domestic gross
fig, axes = plt.subplots(figsize=(8,6))

sns.barplot(x=TheNumbers_profitable['domestic_gross'].head(20), y=TheNumbers_profitable['production_budget'].head(20))
plt.title('Value Returns')
plt.xlabel('Budget Costs')
plt.ylabel('Domestic Gross')
plt.xticks(rotation = 45)

plt.show()
```

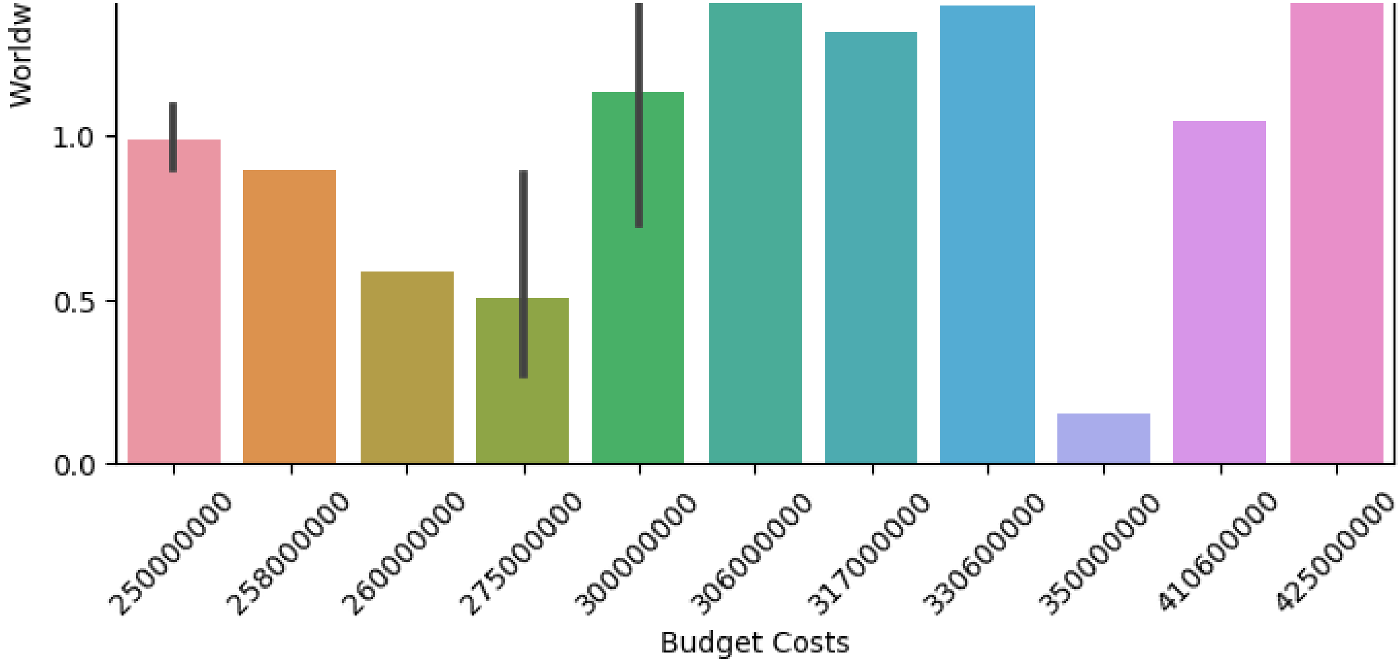




In [41]:

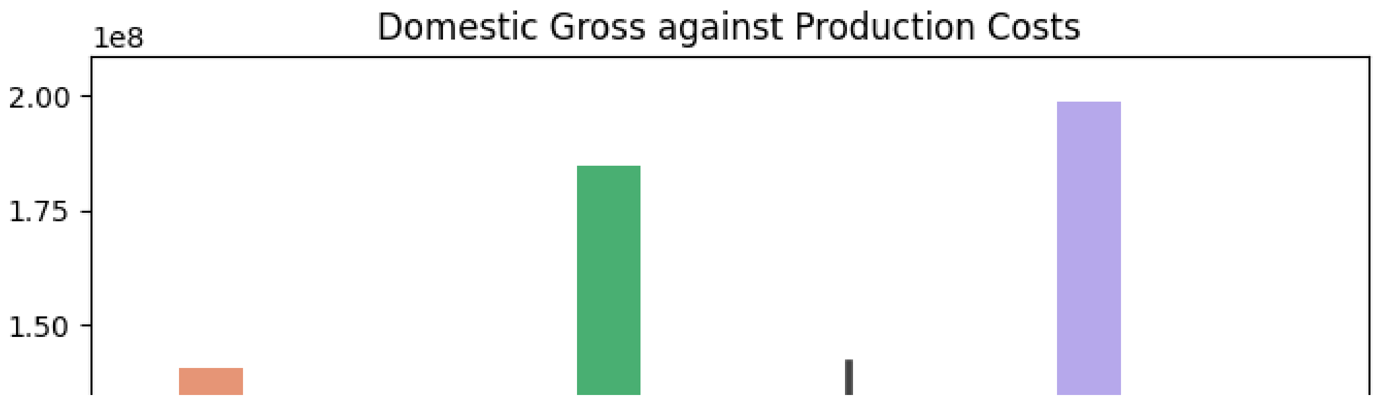
```
# Production Budget vs Worldwide Gross
fig, axes = plt.subplots(figsize=(8,6))

sns.barplot(x=TheNumbers_profitable['production_budget'].head(20), y=TheNumbers_profitable['worldwide_gross'].head(20)
plt.title('Worldwide Value Returns')
plt.xlabel('Budget Costs')
plt.ylabel('Worldwide Gross')
plt.xticks(rotation = 45)

plt.show()
```

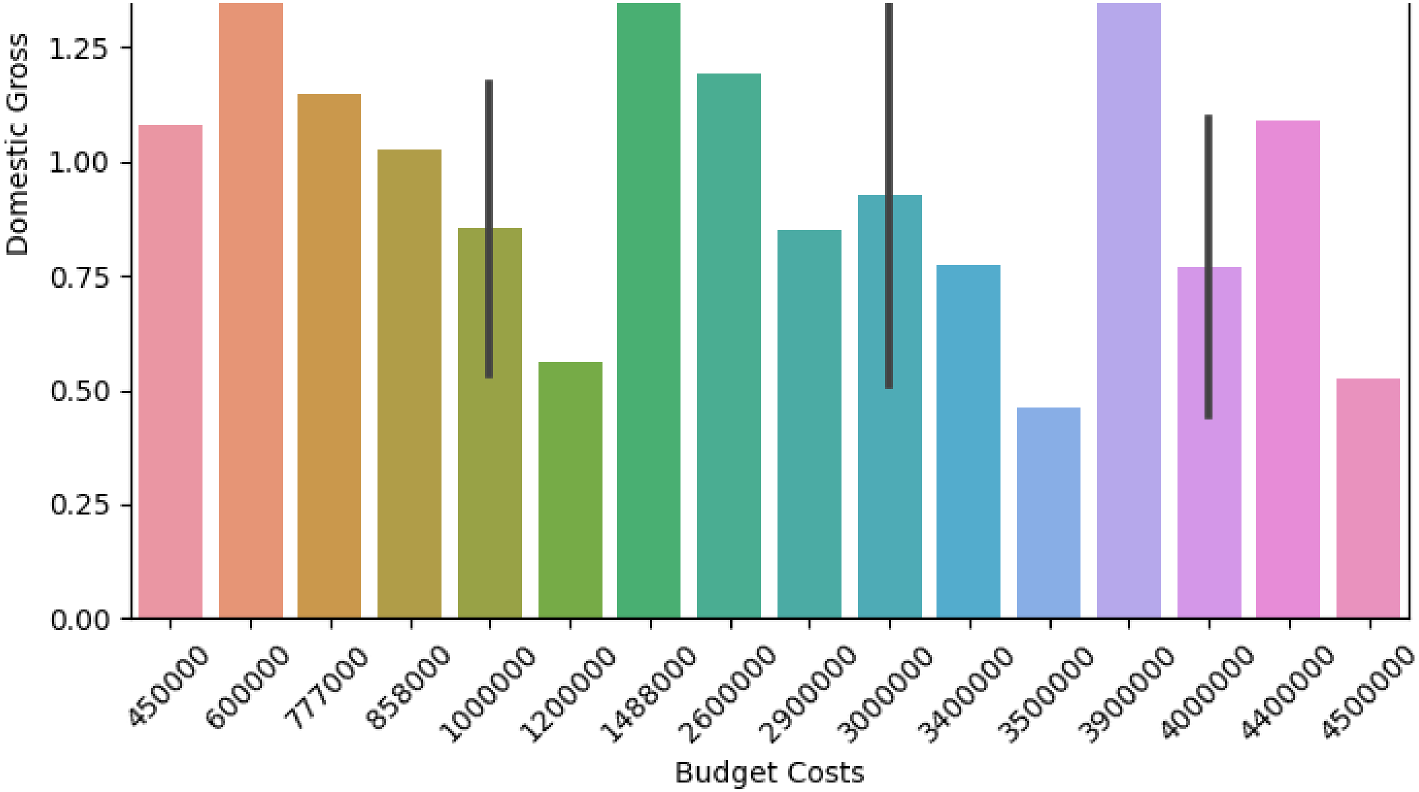




In [42]:

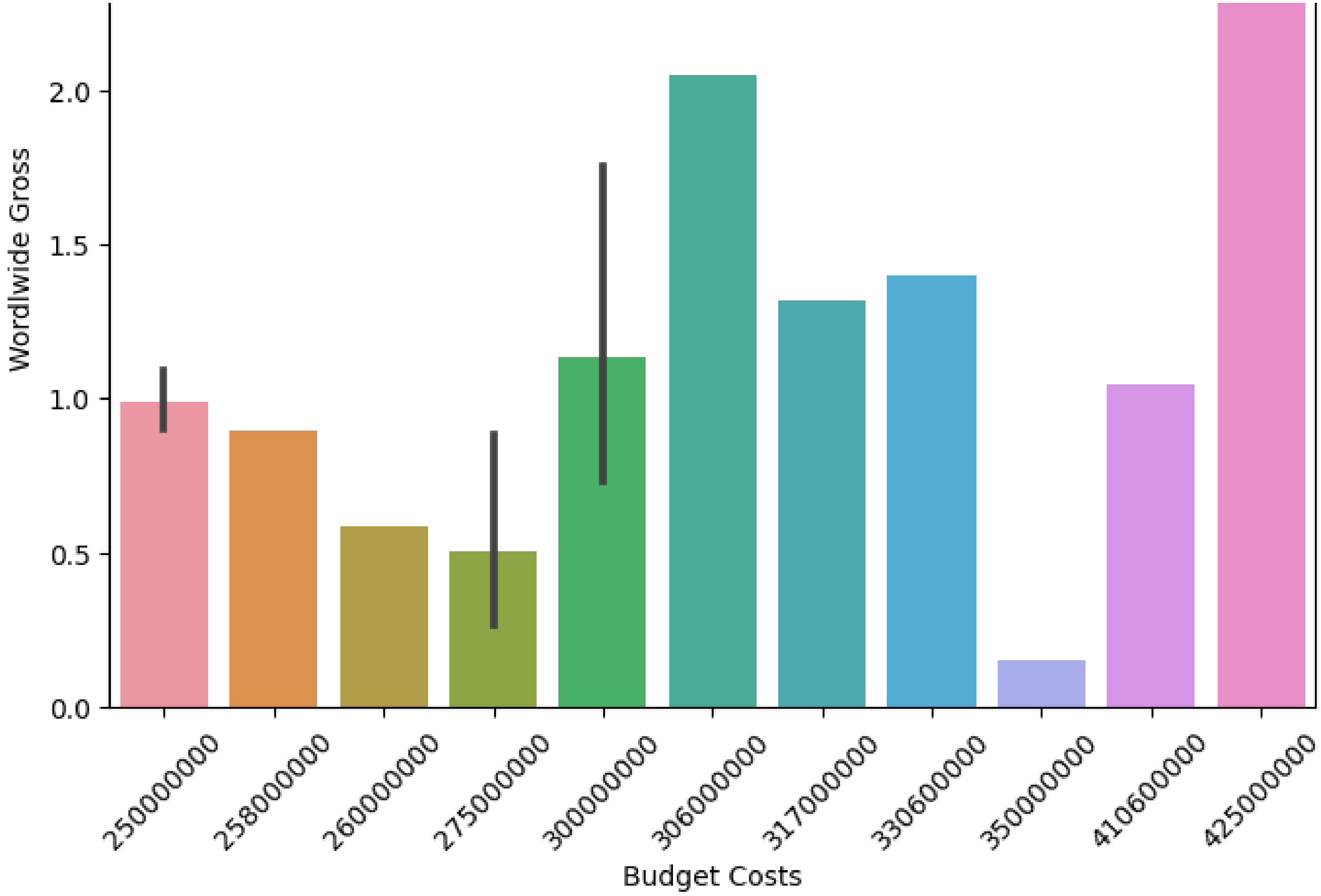
```
# Production Budget vs Domestic Gross
fig, axes = plt.subplots(figsize=(8,6))

sns.barplot(x=TheNum_Budgetable['production_budget'].head(20), y=TheNum_Budgetable['domestic_gross'].head(20))
plt.title('Domestic Gross against Production Costs')
plt.xlabel('Budget Costs')
plt.ylabel('Domestic Gross')
plt.xticks(rotation = 45)

plt.show()
```





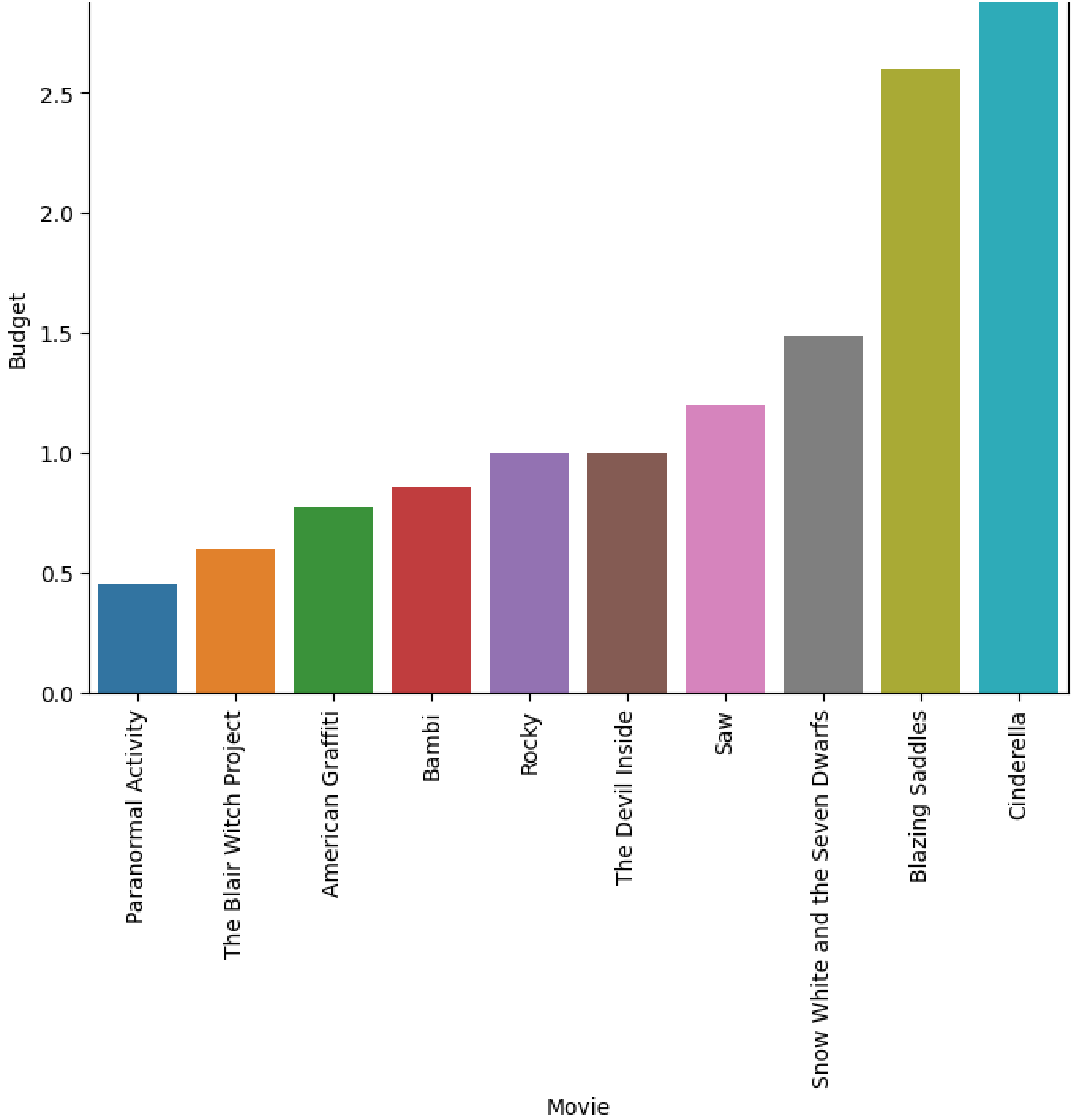
In [43]:

```
# Production Budget vs Worldwide Gross
fig, axes = plt.subplots(figsize=(8,6))

sns.barplot(x=TheNumbers_profitable['production_budget'].head(20), y=TheNumbers_profitable['worldwide_gross'].head(20)
plt.title('Worldwide Gross against Production Costs')
plt.xlabel('Budget Costs')
plt.ylabel('Wordlwide Gross')
plt.xticks(rotation = 45)

plt.show()
```



In [44]:

```
# Movie vs Production Budget
fig, axes = plt.subplots(figsize=(8,6))

sns.barplot(x=TheNum_Budgetable['movie'].head(10), y=TheNum_Budgetable['production_budget'].head(10))
plt.title('Budget vs Movie')
plt.ylabel('Budget')
plt.xlabel('Movie')
plt.xticks(rotation = 90)

plt.show()
```



Budget
2.5
2.0
1.5
1.0
0.5
0.0
Paranormal Activity
The Blair Witch Project
American Graffiti
Bambi
Rocky
The Devil Inside
Saw
Snow White and the Seven Dwarfs
Blazing Saddles
Cinderella
Movie

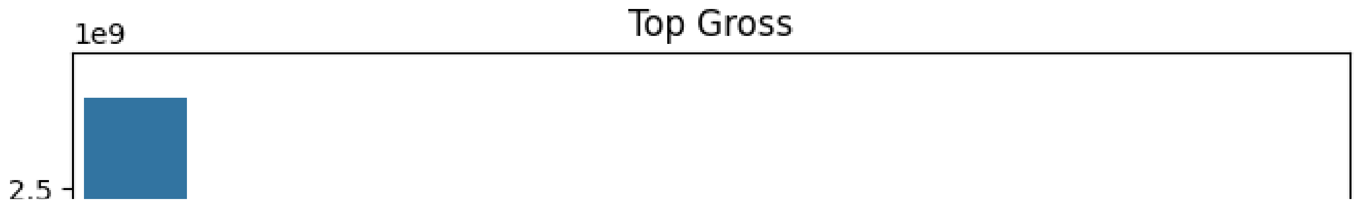
In [29]:
```python
TheNum_TopGross = TheNumbers_profitable.sort_values(by='worldwide_gross', ascending=False).head(100)
TheNum_TopGross
```

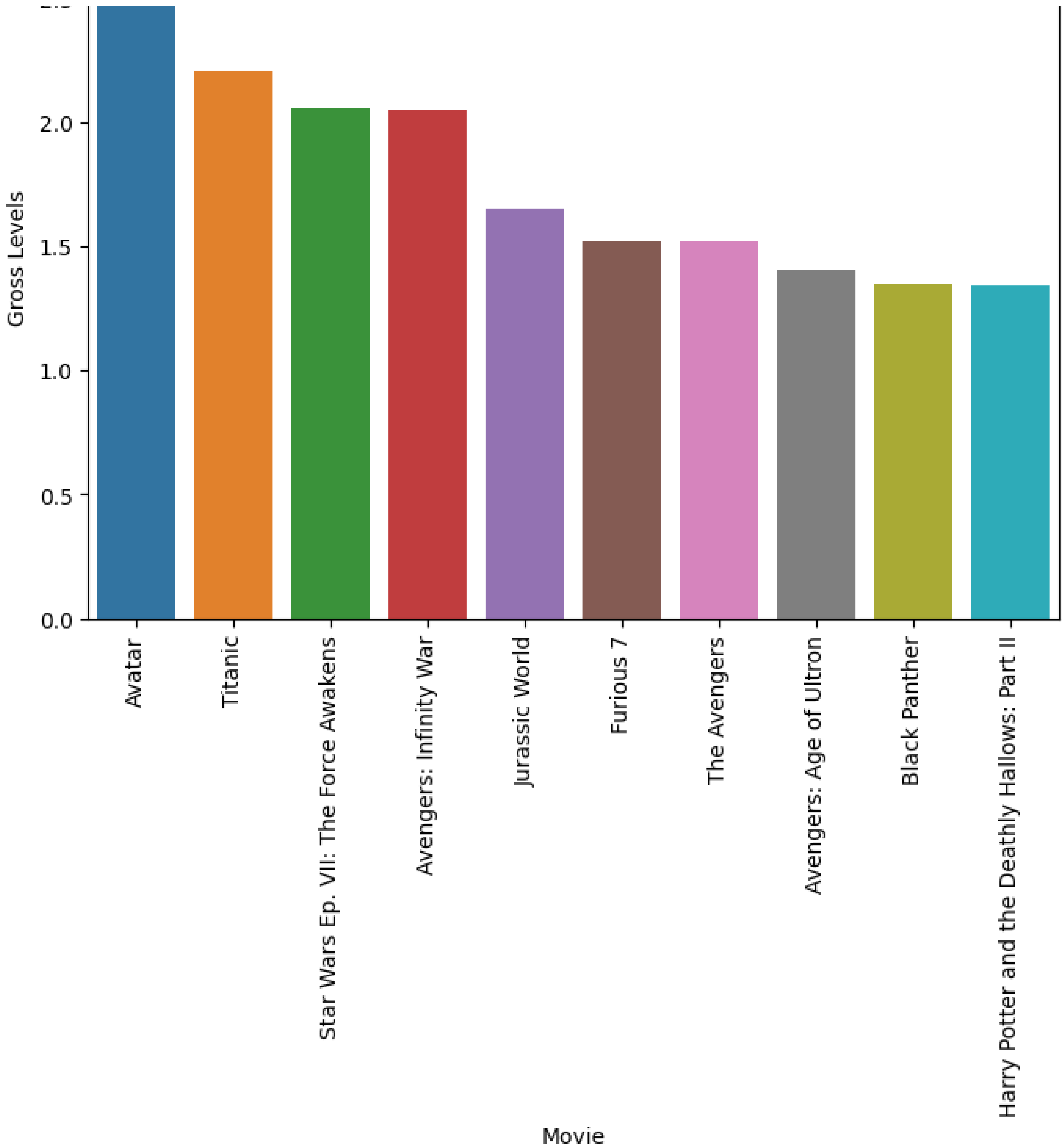
Out[29]:

	id	release_date	movie	production_budget	domestic_gross	worldwide_gross
0	1	Dec 18, 2009	Avatar	425000000	760507625	2776345279
42	43	Dec 19, 1997	Titanic	200000000	659363944	2208208395
5	6	Dec 18, 2015	Star Wars Ep. VII: The Force Awakens	306000000	936662225	2053311220
6	7	Apr 27, 2018	Avengers: Infinity War	300000000	678815482	2048134200
33	34	Jun 12, 2015	Jurassic World	215000000	652270625	1648854864
...	...	...	...	...	...	...
54	55	May 23, 2014	X-Men: Days of Future Past	200000000	233921534	747862775
196	97	Jun 8, 2012	Madagascar 3: Europe's Most Wanted	145000000	216391482	746921271
99	100	Aug 5, 2016	Suicide Squad	175000000	325100054	746059887
52	53	Jun 21, 2013	Monsters University	200000000	268488329	743588329
159	60	May 15, 2003	The Matrix Reloaded	150000000	281553689	738576929

100 rows × 6 columns

In [45]:
```python
# Movie vs Worldwide Gross
fig, axes = plt.subplots(figsize=(8,6))

sns.barplot(x =TheNum_TopGross['movie'].head(10), y =TheNum_TopGross['worldwide_gross'].head(10))
plt.title('Top Gross')
plt.xlabel('Movie')
plt.ylabel('Gross Levels')
plt.xticks(rotation = 90)

plt.show()
```





Gross Levels
0.0
0.5
1.0
1.5
2.0
Avatar
Titanic
Star Wars Ep. VII: The Force Awakens
Avengers: Infinity War
Jurassic World
Furious 7
The Avengers
Avengers: Age of Ultron
Black Panther
Harry Potter and the Deathly Hallows: Part II
Movie

In [31]:
```
# Getting descriptive statistics for tmdb_df data frame
TMDB_data[["popularity", "vote_average", "vote_count"]].describe()
```

Out[31]:

	popularity	vote_average	vote_count
count	500.00000	500.000000	500.000000
mean	19.68794	6.962600	5301.510000
std	9.40349	0.611366	3351.875102
min	0.60000	6.000000	2005.000000
25%	13.88100	6.475000	2912.250000
50%	17.00150	6.900000	4136.500000
75%	24.02075	7.400000	6720.000000
max	80.77300	8.600000	22186.000000

In [32]:
```
TheNumbers_profitable[["production_budget", "domestic_gross", "worldwide_gross"]].describe()
```

Out[32]:

	production_budget	domestic_gross	worldwide_gross
count	1.414000e+03	1.414000e+03	1.414000e+03
mean	7.586693e+07	1.241784e+08	2.995459e+08
std	5.841339e+07	9.508621e+07	2.555141e+08
min	4.500000e+05	3.276600e+04	1.000038e+08
25%	3.400000e+07	6.502977e+07	1.426717e+08
50%	6.000000e+07	1.000136e+08	2.084565e+08
75%	1.000000e+08	1.504101e+08	3.515901e+08
max	4.250000e+08	9.366622e+08	2.776345e+09

Evaluation

This section goes through the data results and visualizations to see how well the business problem can be tackled. From the visualizations above, we

on some matters. In the bar graphs of Top Voted Genres and average voted, we are able to see that the similar ranges suggest success in the industry
multiple factors beyond just movie genres as there may not be a specific genre or storyline that consistently leads to success. Key success factors cou
production quality, marketing,and even audience reception. Microsoft Studio should not rely solely on particular movies and associated genres, and ne
overall film quality and appeal. Instead, it may be more important to focus on creating a high-quality, engaging film regardless of the genre.

The irregular but peaking line chart in the bar chart of Preference Changes Over Time shows that recent releases tend to be more popular. This sugge
may be more interested in newer movies thus focusing on latest trends and techniques to maximize popularity should work. The reason for this may be
marketing efforts and a greater focus on creating visually stunning movies that appeal to modern audiences. Therefore, it may be wise to prioritize the
appealing movies with engaging storylines to maximize their chances of success.

Specific genres and movies having high popularity indicates that certain types of content resonate more with audiences. Identifying the popular genres
help in developing films with market potential. Emulating the successful films can also be a strategy.The fact that they have higher popularity and gross
suggests that it may be beneficial to focus on producing movies within those genres. However, it is important to note that this may change over time, s
stay up to date on current trends and audience preferences.

The budget vs gross returns data shows that high cost does not guarantee high returns and vice versa. This means that an expensive, large-scale pro
the only path to success. It is important to focus on creating a high-quality film that is engaging and resonates with audiences, regardless of budget or
story and strong marketing may be able to achieve good results even with a lower budget. Risks should be managed and returns should be considered
investment decisions.

Generally, in answering our problem questions; success depends on multiple factors like story, production, and marketing; genre alone is not sufficient
techniques can boost popularity as shown by peak popularity of recent releases in the visualization. Certain genres/themes and emulating successful f
strategies due to specific high-popularity content. The budget is not the only driver of returns, as good story and marketing can be effective with lower
high investments with shallow content.

Conclusion

Recommendations

From the analyses and insights of the data, it can be seen that it is possible for Microsoft to set up a movie studio for original content as soon as they a
only constraint being money and information. Provided information of current trends and available production methods, Microsoft may be able to set up
can compete with other big companies in th foreseeable future. I would recommend that:

1. Microsoft should focus on story, production quality, and marketing in addition to genre so as to be sure of success in the venture. Microsoft can co

market research to identify current trends and audience preferences.
2. Microsoft should apply latest trends and techniques to maximize relevance and interest from the audience and increase popularity to 'make it big/l
This can be done by prioritizing producing and promoting new releases to capture audience attention.
3. Microsoft should manage risks and consider returns when making investment decisions. Not all hugh budget films bring high returns and vice-vers producing movies within genres that have higher popularity and gross returns would help, but staying up to date on current trends and audience p preferable.
4. Microsoft should identify current popular genres/themes and successful films to emulate in their production process. The market is very flexible an invest in researching current trends for their target audience.
5. Microsoft should have a strategy could be to combine a popular genre/theme with a compelling story and strong marketing, while leveraging latest and controlling costs. Monitoring industry patterns and top films can help in defining the strategy and execution details.

In general, based on the observations provided, there are several key success factors for movies, including genres, storyline, and release date. Recen trend towards movies with popular genres, such as action, adventure, and comedy, being more successful than those with less popular genres. Additio releases tend to have higher grossing returns than older releases. Furthermore, there appears to be a pattern in the themes and genres of successful genres being more popular than others.

In order to produce successful films that will appeal to audiences and generate revenue, it is important to consider the above factors, as well as the bu Movies with high budgets may not necessarily be successful, and vice versa. Additionally, it is important to consider the target audience and ensure tha them. Finally, it is important to ensure that the film has a strong storyline and engaging characters.

Limitations

The analysis, insight, and recommendations are based on the data that is limited and may not be as recent or insightful as those from other sources. T analysis and presentation used may also differ from the most suitable ones available. This may result in this proposal falling short of the best expectati action. However the insights and recommendations should prove useful for the business problem if utilized well.

Next Steps

I will attempt to improve on the data that is used for analysis and be able to provide a more accurate analysis and insight as well as recommendations this project to a higher quality and standard.

Thanks

Contact information:

1. Email: joel.omondi@student.moringaschool.com
2. Linked-in: linkedin.com/in/ed-joel-odhiambo/