

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [2]: disney_path = '../input/disney-movies-19372016-total-gross/disney_m
ovies_total_gross.csv'

disney_data = pd.read_csv(disney_path)
```

Introduction:

Disney seems to be successfull with children since the first movie, and everytime this company gets bigger ang bigger with their animations and thematic parks, also disney pursue a lot of other animations and movie studios which make this company one of the biggest in the world.

Preprocessing:

```
In [3]: disney_data.head()
```

Out[3]:

	movie_title	release_date	genre	mpaa_rating	total_gross	inflation_adjusted_gross
0	Snow White and the Seven Dwarfs	1937-12-21	Musical	G	184925485	5228953251
1	Pinocchio	1940-02-09	Adventure	G	84300000	2188229052
2	Fantasia	1940-11-13	Musical	G	83320000	2187090808
3	Song of the South	1946-11-12	Adventure	G	65000000	1078510579
4	Cinderella	1950-02-15	Drama	G	85000000	920608730

In [4]: `disney_data.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 579 entries, 0 to 578
Data columns (total 6 columns):
 #   Column                                Non-Null Count  Dtype
---  -
 0   movie_title                          579 non-null    object
 1   release_date                         579 non-null    object
 2   genre                               562 non-null    object
 3   mpaa_rating                         523 non-null    object
 4   total_gross                         579 non-null    int64
 5   inflation_adjusted_gross            579 non-null    int64
dtypes: int64(2), object(4)
memory usage: 27.3+ KB
```

In [5]: `## Converting release_date to date type:`
`disney_data['release_date'] = pd.to_datetime(disney_data['release_date'])`

In [6]: `# Let's create an column named year to visualize later`
`disney_data['year'] = disney_data['release_date'].dt.date.astype(str).str.split('-', expand=True)[0]`

In [7]: `disney_data.dtypes`

```
Out[7]: movie_title                object
release_date                datetime64[ns]
genre                      object
mpaa_rating                object
total_gross                int64
inflation_adjusted_gross    int64
year                      object
dtype: object
```

In [8]: `disney_data.describe().T`

```
Out[8]:
```

	count	mean	std	min	25%	50%
total_gross	579.0	6.470179e+07	9.301301e+07	0.0	12788864.0	30702446.0
inflation_adjusted_gross	579.0	1.187625e+08	2.860853e+08	0.0	22741232.0	55159783.0

Analysing the statistical data:

The best and worst gross:

```
In [9]: print('the best total gross', disney_data['total_gross'].max(),  
            '\n\nThe worse total gross:', disney_data['total_gross'].min())
```

```
the best total gross 936662225  
The worse total gross: 0
```

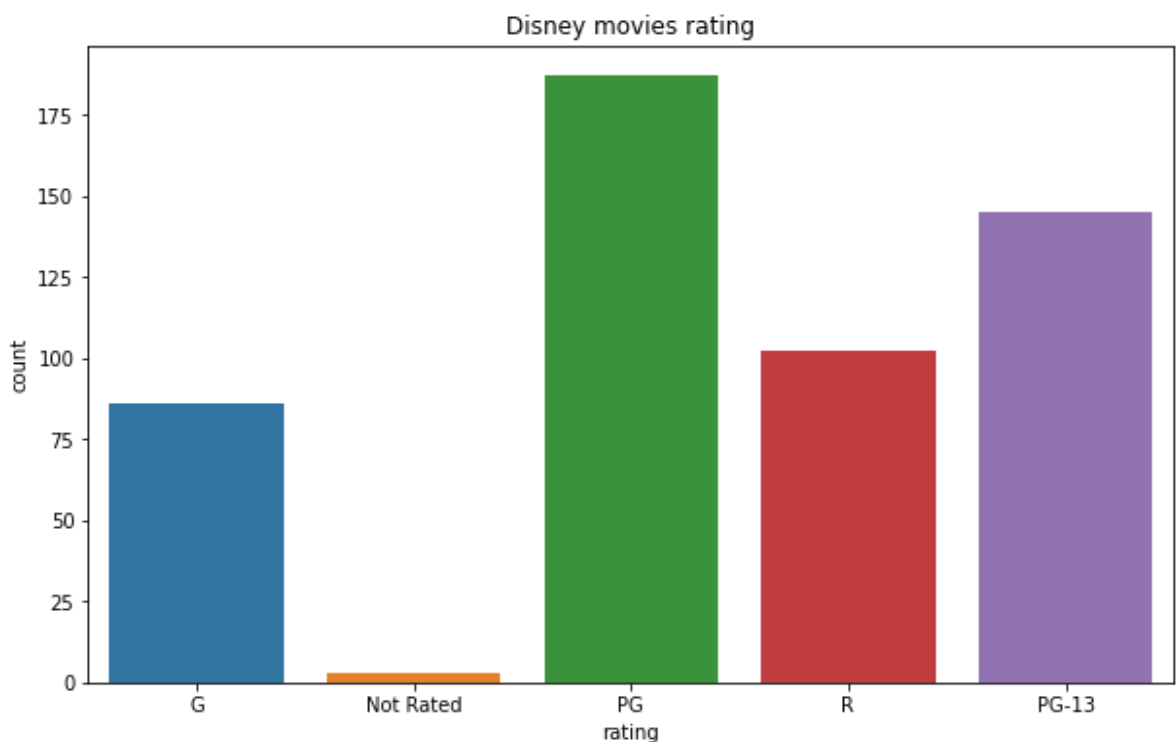
```
In [10]: print('the best inflation adjusted gross', disney_data['inflation_a  
adjusted_gross'].max(), '\n\nThe worse inflation adjusted gross:', disn  
ey_data['inflation_adjusted_gross'].min())
```

```
the best inflation adjusted gross 5228953251  
The worse inflation_adjusted_gross: 0
```

What are the movies ratings and genres?

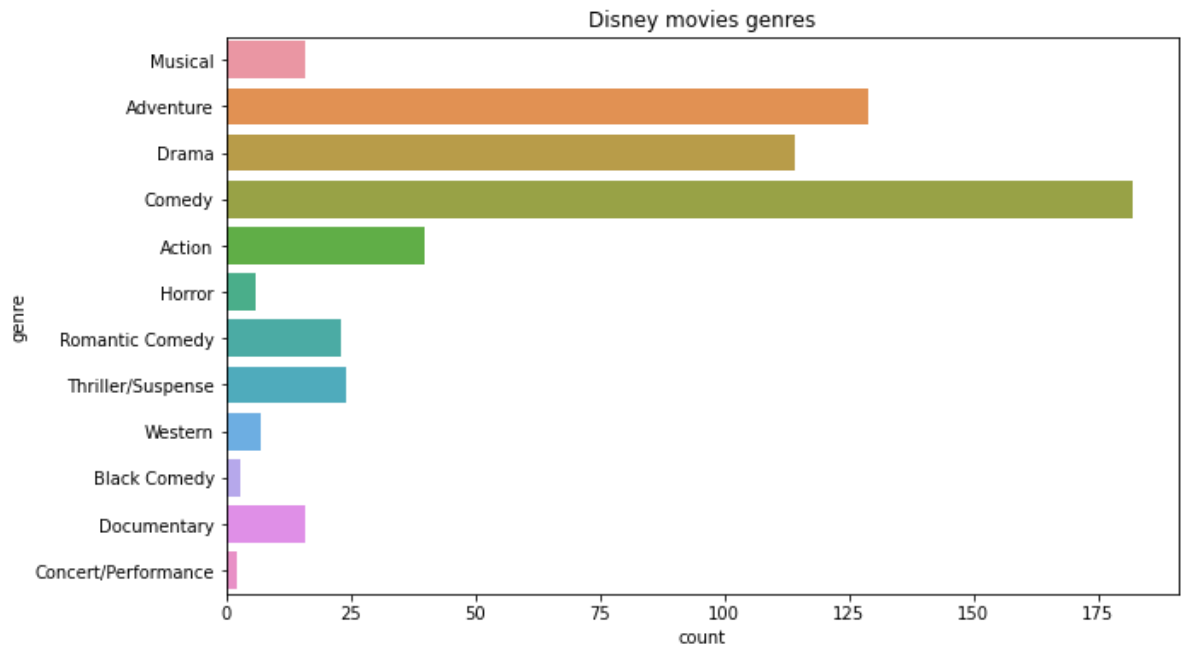
```
In [11]: plt.figure(figsize=(10, 6))  
plt.title('Disney movies rating')  
sns.countplot(x='mpaa_rating', data=disney_data)  
plt.xlabel('rating')
```

```
Out[11]: Text(0.5, 0, 'rating')
```



```
In [12]: plt.figure(figsize=(10, 6))  
plt.title('Disney movies genres')  
sns.countplot(y='genre', data=disney_data)
```

```
Out[12]: <AxesSubplot:title={'center':'Disney movies genres'}, xlabel='count', ylabel='genre'>
```

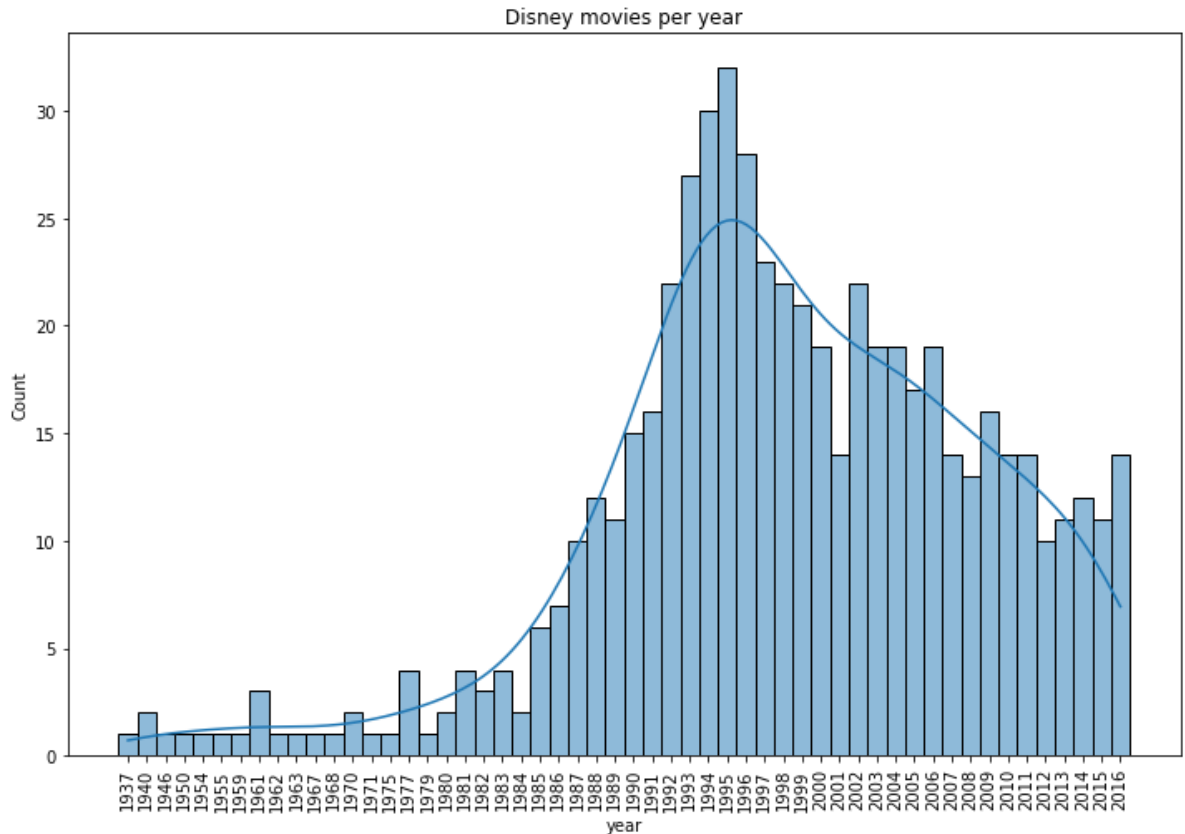


As expected the most of disney movie-genres are comedy, adventure and drama.

Analysing number per year:

```
In [13]: plt.figure(figsize=(12, 8))
plt.xticks(rotation=90)
plt.title('Disney movies per year')
sns.histplot(disney_data['year'], kde=True)
```

```
Out[13]: <AxesSubplot:title={'center':'Disney movies per year'}, xlabel='year', ylabel='Count'>
```

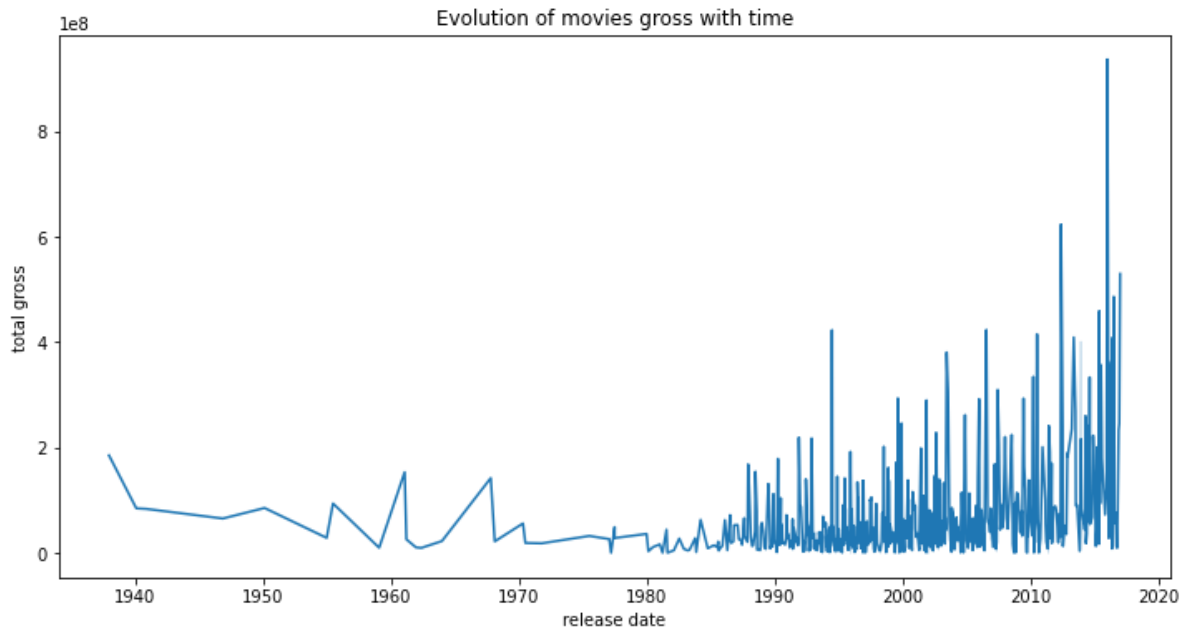


Analysing the gross:

The total gross ignores the inflation, so we can see how much money they've gained in the time, but since the money's values change with time we need to analyse the gross with inflation, the gross with inflation makes possible to compare how much they've gained in the current monetary value.

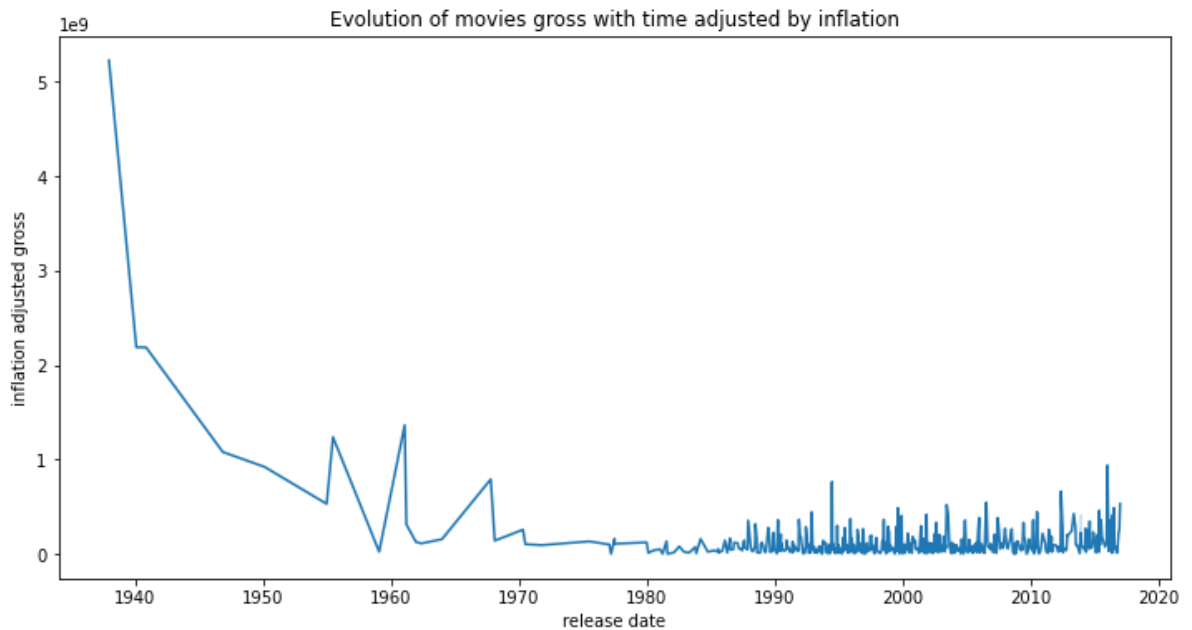
```
In [14]: plt.figure(figsize=(12, 6))  
plt.title('Evolution of movies gross with time')  
plt.xlabel('release date')  
plt.ylabel('total gross')  
sns.lineplot(x='release_date', y='total_gross', data=disney_data)
```

```
Out[14]: <AxesSubplot:title={'center':'Evolution of movies gross with time'},  
xlabel='release date', ylabel='total gross'>
```



```
In [15]: plt.figure(figsize=(12, 6))
plt.title('Evolution of movies gross with time adjusted by inflation')
plt.xlabel('release date')
plt.ylabel('inflation adjusted gross')
sns.lineplot(x='release_date', y='inflation_adjusted_gross', data=disney_data)
```

```
Out[15]: <AxesSubplot:title={'center':'Evolution of movies gross with time adjusted by inflation'}, xlabel='release date', ylabel='inflation adjusted gross'>
```



When we visualize the first graph looks like the gross is increasing, which is not true when we see the inflation adjusted graph

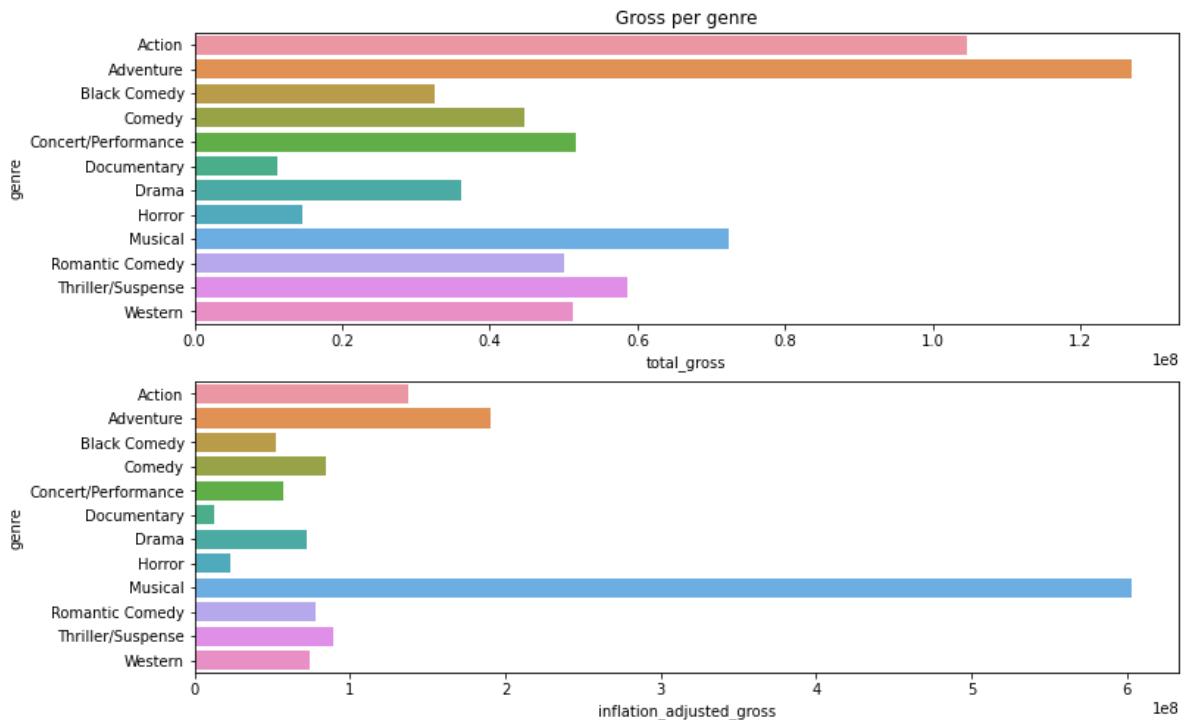
Analysing values:

Analysing movie genres:

```
In [16]: avg_genre = disney_data.groupby('genre').mean()
```

```
In [17]: plt.figure(figsize=(12, 8))
plt.subplot(2, 1, 1).set_title('Gross per genre')
sns.barplot(x=avg_genre['total_gross'], y=avg_genre.index)
plt.subplot(2, 1, 2)
sns.barplot(x=avg_genre['inflation_adjusted_gross'], y=avg_genre.index)
```

```
Out[17]: <AxesSubplot:xlabel='inflation_adjusted_gross', ylabel='genre'>
```



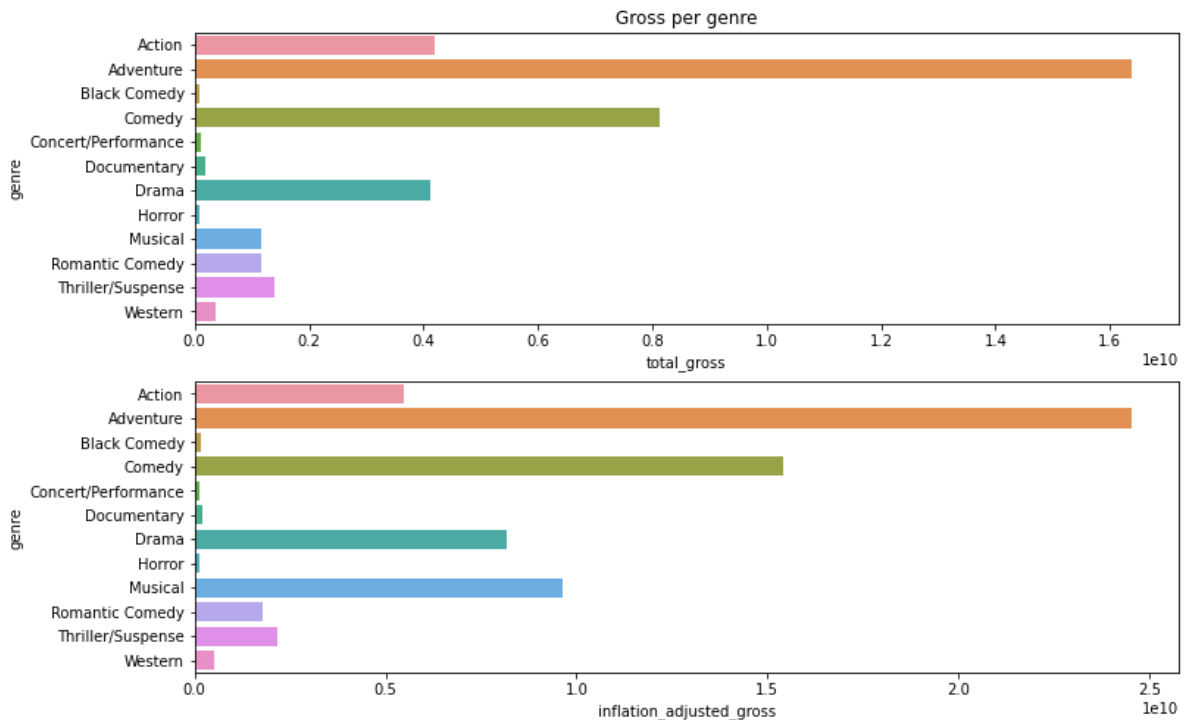
Even though the genres that had more average total gross are action and adventure we can see that when we adjust by inflation musical is the genre that had the highest average gross.

```
In [18]: sum_genre = disney_data.groupby('genre').sum()
```



```
In [19]: plt.figure(figsize=(12, 8))
plt.subplot(2, 1, 1).set_title('Gross per genre')
sns.barplot(x=sum_genre['total_gross'], y=sum_genre.index)
plt.subplot(2, 1, 2)
sns.barplot(x=sum_genre['inflation_adjusted_gross'], y=sum_genre.index)
```

```
Out[19]: <AxesSubplot:xlabel='inflation_adjusted_gross', ylabel='genre'>
```



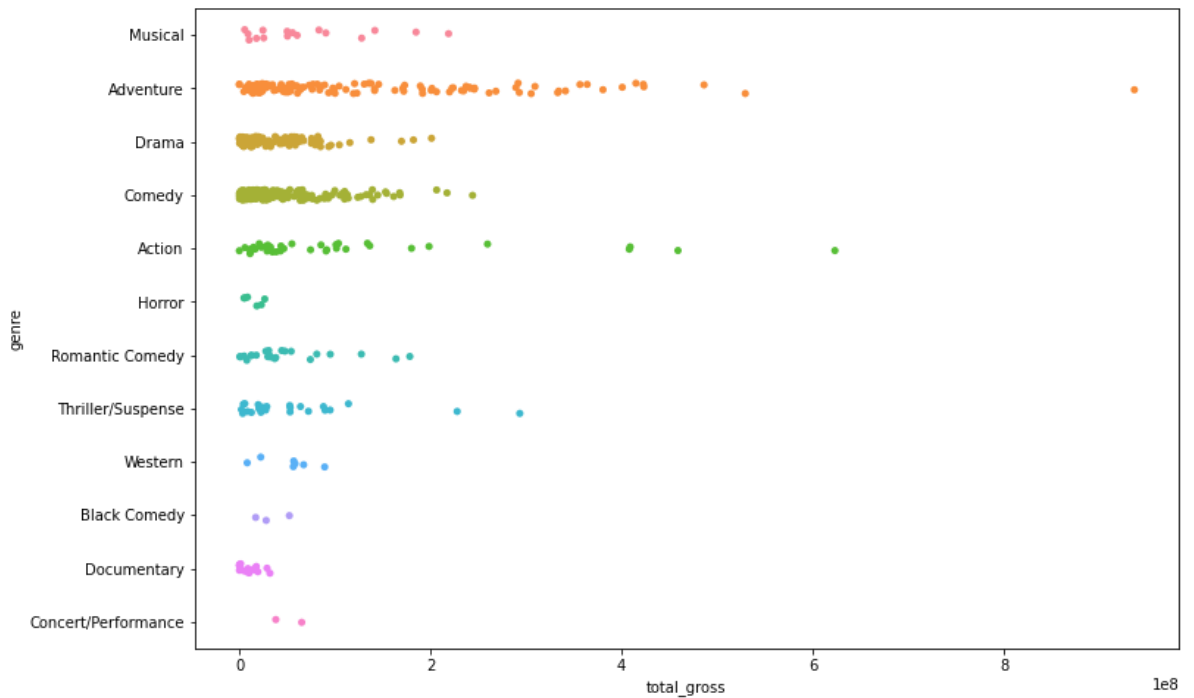
Even though the average gross of Musical movies is high we can see that Adventure and Comedy are the ones with the highest gross sum, it can be explained by the number of movies that every genre had

```
In [20]: disney_data['genre'].value_counts()
```

```
Out[20]: Comedy          182
Adventure       129
Drama          114
Action          40
Thriller/Suspense 24
Romantic Comedy 23
Musical         16
Documentary     16
Western         7
Horror          6
Black Comedy    3
Concert/Performance 2
Name: genre, dtype: int64
```

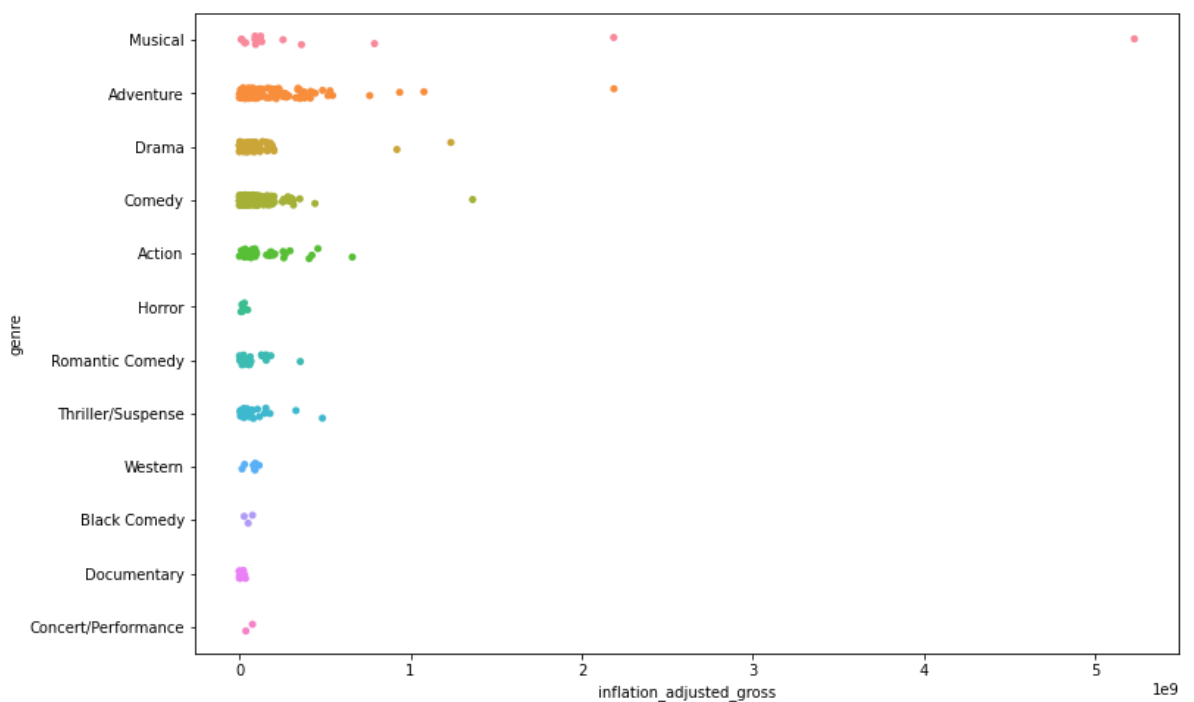
```
In [21]: plt.figure(figsize=(12, 8))  
sns.stripplot(y='genre', x='total_gross', data=disney_data)
```

Out[21]: <AxesSubplot:xlabel='total_gross', ylabel='genre'>



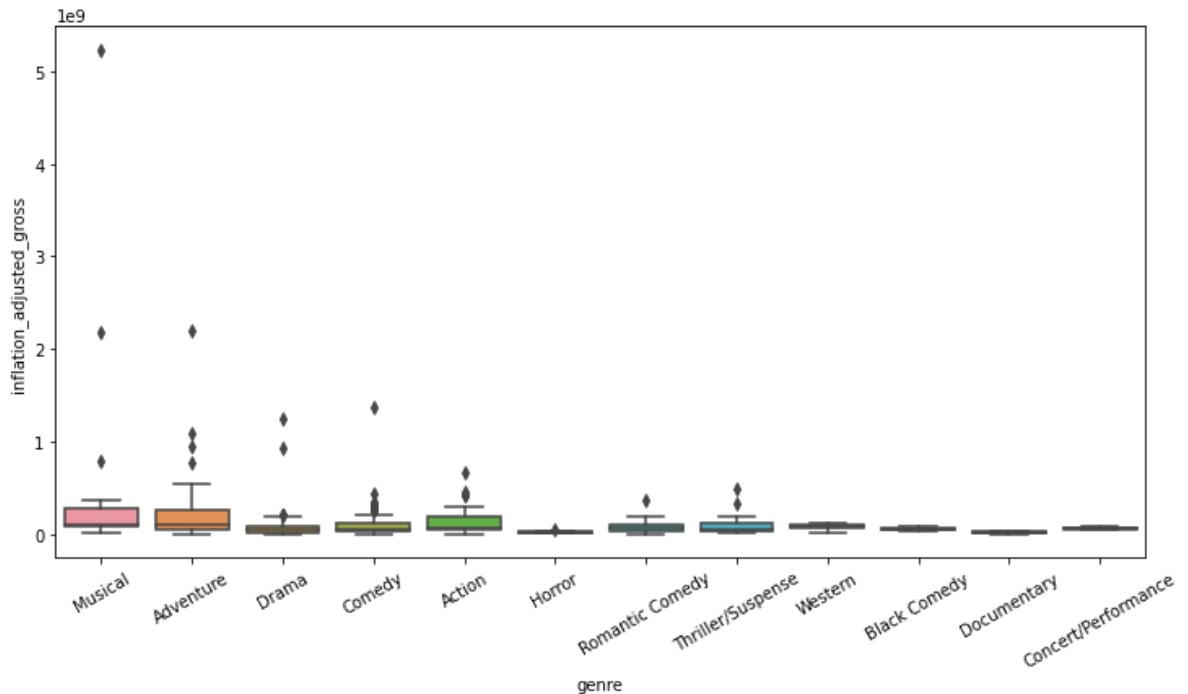
```
In [22]: plt.figure(figsize=(12, 8))  
sns.stripplot(y='genre', x='inflation_adjusted_gross', data=disney_data)
```

Out[22]: <AxesSubplot:xlabel='inflation_adjusted_gross', ylabel='genre'>



```
In [23]: plt.figure(figsize=(12, 6))  
plt.xticks(rotation=30)  
plt.ylabel('inflation adjusted gross')  
sns.boxplot(x='genre', y='inflation_adjusted_gross', data=disney_data)
```

Out[23]: <AxesSubplot:xlabel='genre', ylabel='inflation_adjusted_gross'>

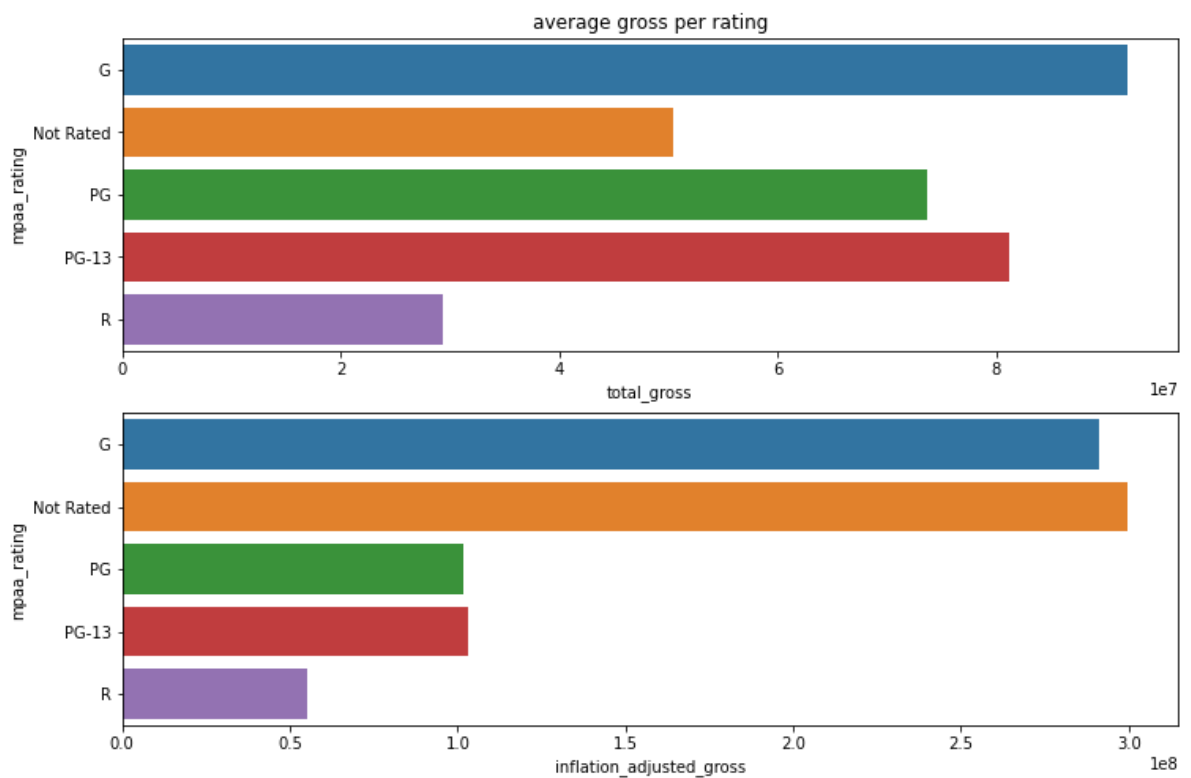


So, adventure movies are probably the most lucrative disney genre, but the musical genre have many outliers and tends to make musicals averagely better than other genres

Analysing rating

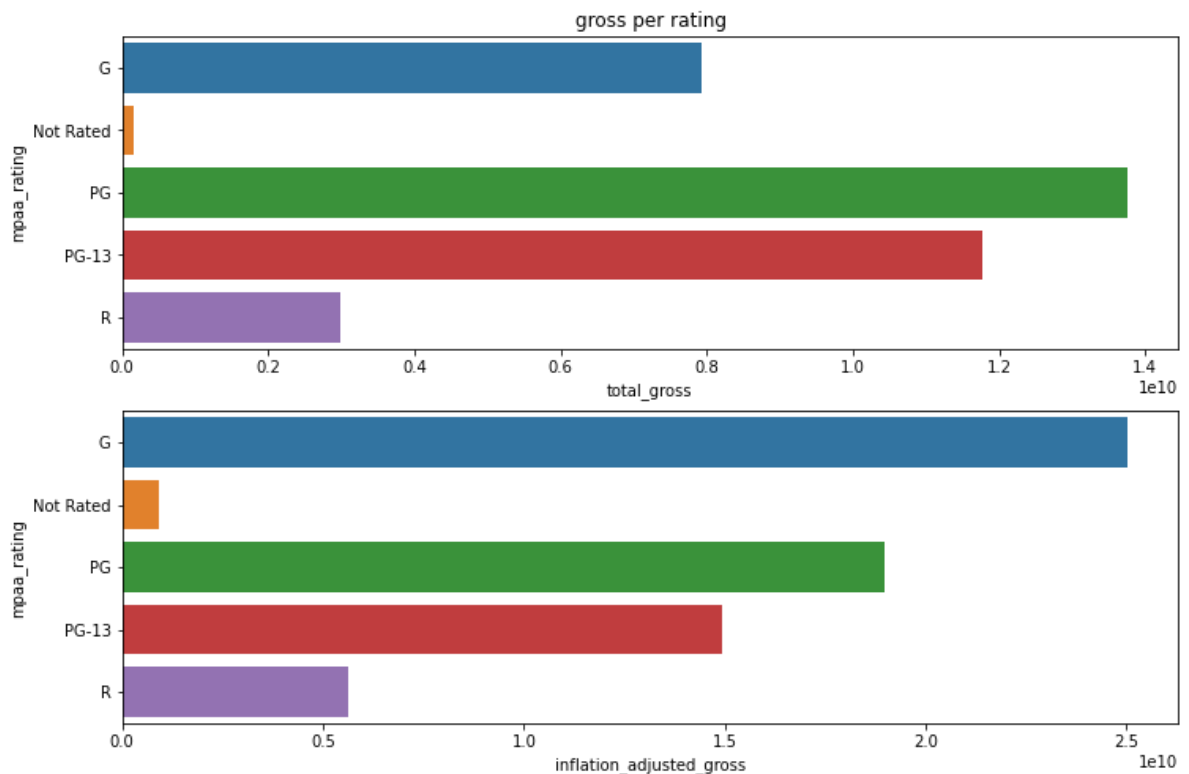
```
In [24]: avg_rating = disney_data.groupby('mpaa_rating').mean()  
plt.figure(figsize=(12, 8))  
plt.subplot(2, 1, 1).set_title('average gross per rating')  
sns.barplot(x=avg_rating['total_gross'], y=avg_rating.index)  
plt.subplot(2, 1, 2)  
sns.barplot(x=avg_rating['inflation_adjusted_gross'], y=avg_rating.  
index)
```

Out[24]: <AxesSubplot:xlabel='inflation_adjusted_gross', ylabel='mpaa_rating'>



```
In [25]: avg_rating = disney_data.groupby('mpaa_rating').sum()
plt.figure(figsize=(12, 8))
plt.subplot(2, 1, 1).set_title('gross per rating')
sns.barplot(x=avg_rating['total_gross'], y=avg_rating.index)
plt.subplot(2, 1, 2)
sns.barplot(x=avg_rating['inflation_adjusted_gross'], y=avg_rating.index)
```

```
Out[25]: <AxesSubplot:xlabel='inflation_adjusted_gross', ylabel='mpaa_rating'>
```

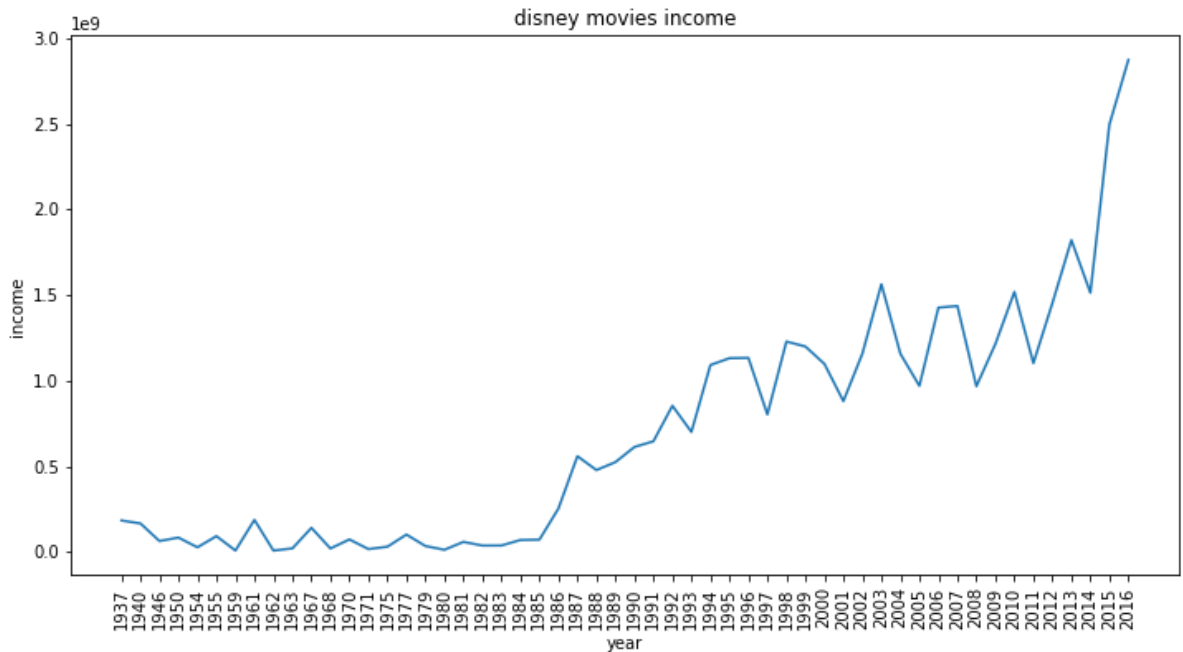


Have disney movies income decreased?

```
In [26]: year_income = disney_data.groupby('year').sum()
```

```
In [27]: plt.figure(figsize=(12, 6))
plt.title('disney movies income')
plt.xlabel('year')
plt.ylabel('income')
plt.xticks(rotation=90)
sns.lineplot(x=year_income.index, y=year_income['total_gross'])
```

```
Out[27]: <AxesSubplot:title={'center':'disney movies income'}, xlabel='year', ylabel='income'>
```



```
In [28]: plt.figure(figsize=(12, 6))
plt.title('disney movies income adjusted')
plt.xlabel('year')
plt.ylabel('inflation adjusted income')
plt.xticks(rotation=90)
sns.lineplot(x=year_income.index, y=year_income['inflation_adjusted_gross'])
```

```
Out[28]: <AxesSubplot:title={'center':'disney movies income adjusted'}, xlabel='year', ylabel='inflation adjusted income'>
```

