Movie_Budget_and_Financial_Records

November 21, 2025

1 Movie Budgets and Financial Performance

1.1 Do higher film budgets lead to more box office revenue? Let's find out if there's a relationship using the movie budgets and financial performance data scraped from the-numbers.com

2 Import Statements

[54]:
```
import pandas as pd
import matplotlib.pyplot as plt
```

3 Notebook Presentation

[55]:
```
pd.options.display.float_format = '{:,.2f}'.format

from pandas.plotting import register_matplotlib_converters
register_matplotlib_converters()
```

4 Read the Data

[56]:
```
data = pd.read_csv('cost_revenue_dirty.csv')
```

5 Explore and Clean the Data

[57]:
```
data
```

[57]:
```
      Rank Release_Date                     Movie_Title USD_Production_Budget  \
0     5293     8/2/1915           The Birth of a Nation              $110,000
1     5140     5/9/1916                     Intolerance              $385,907
2     5230   12/24/1916    20,000 Leagues Under the Sea              $200,000
3     5299    9/17/1920  Over the Hill to the Poorhouse              $100,000
4     5222     1/1/1925                  The Big Parade              $245,000
…    …          …                             …                    …
5386  2950    10/8/2018                             Meg           $15,000,000
```

```
5387   126   12/18/2018                          Aquaman          $160,000,000
5388    96   12/31/2020                      Singularity          $175,000,000
5389  1119   12/31/2020            Hannibal the Conqueror           $50,000,000
5390  2517   12/31/2020  Story of Bonnie and Clyde, The           $20,000,000

     USD_Worldwide_Gross USD_Domestic_Gross
0            $11,000,000        $10,000,000
1                     $0                 $0
2             $8,000,000         $8,000,000
3             $3,000,000         $3,000,000
4            $22,000,000        $11,000,000
…                    …                  …
5386                  $0                 $0
5387                  $0                 $0
5388                  $0                 $0
5389                  $0                 $0
5390                  $0                 $0

[5391 rows x 6 columns]
```

5.1 Rows and columns

[58]:

```
data.shape
```

[58]:

```
(5391, 6)
```

5.2 NaN values

[59]:

```
data.isna().values.any()
```

[59]:

```
False
```

5.3 Duplicate rows

[60]:

```
data.duplicated().any()  #will return True id there are duplicates
```

[60]:

```
False
```

[61]:

```
data[data.duplicated()] #show the duplicated rows
```

[61]:

```
Empty DataFrame
Columns: [Rank, Release_Date, Movie_Title, USD_Production_Budget,
USD_Worldwide_Gross, USD_Domestic_Gross]
Index: []
```

5.4 Columns Data Types

[62]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5391 entries, 0 to 5390
Data columns (total 6 columns):
 #   Column                 Non-Null Count  Dtype 
---  ------                 --------------  ----- 
 0   Rank                   5391 non-null   int64 
 1   Release_Date           5391 non-null   object
 2   Movie_Title            5391 non-null   object
 3   USD_Production_Budget  5391 non-null   object
 4   USD_Worldwide_Gross    5391 non-null   object
 5   USD_Domestic_Gross     5391 non-null   object
dtypes: int64(1), object(5)
memory usage: 252.8+ KB
```

5.5 Data Type Conversions

5.5.1 Will convert the USD_Production_Budget, USD_Worldwide_Gross, and USD_Domestic_Gross columns to a numeric format by removing $ signs and ,.

Note that *domestic* in this context refers to the United States.

[63]:
```
data.USD_Domestic_Gross = data.USD_Domestic_Gross.astype(str).str.replace('$',␣
 ↪"")
```

[64]:
```
data.USD_Domestic_Gross = data.USD_Domestic_Gross.astype(str).str.replace(',',␣
 ↪"")
```

[65]:
```
data.USD_Domestic_Gross = pd.to_numeric(data.USD_Domestic_Gross)
```

[66]:
```
data.USD_Worldwide_Gross = data.USD_Worldwide_Gross.astype(str).str.
 ↪replace('$', "")
```

[67]:
```
data.USD_Worldwide_Gross = data.USD_Worldwide_Gross.astype(str).str.
 ↪replace(',', "")
```

[68]:
```
data.USD_Worldwide_Gross = pd.to_numeric(data.USD_Worldwide_Gross)
```

[69]:
```
data.USD_Production_Budget = data.USD_Production_Budget.astype(str).str.
 ↪replace('$', "")
```

[70]:
```
data.USD_Production_Budget = data.USD_Production_Budget.astype(str).str.
 ↪replace(',', "")
```

[71]:
```
data.USD_Production_Budget = pd.to_numeric(data.USD_Production_Budget)
```

[72]:

```
data
```

[72]:

```
      Rank Release_Date                     Movie_Title  \
0     5293     8/2/1915          The Birth of a Nation
1     5140     5/9/1916                    Intolerance
2     5230   12/24/1916    20,000 Leagues Under the Sea
3     5299    9/17/1920  Over the Hill to the Poorhouse
4     5222     1/1/1925                  The Big Parade
…      …          …                          …
5386  2950    10/8/2018                             Meg
5387   126   12/18/2018                        Aquaman
5388    96   12/31/2020                    Singularity
5389  1119   12/31/2020          Hannibal the Conqueror
5390  2517   12/31/2020  Story of Bonnie and Clyde, The

      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross
0                    110000             11000000            10000000
1                    385907                    0                   0
2                    200000              8000000             8000000
3                    100000              3000000             3000000
4                    245000             22000000            11000000
…                      …                    …                   …
5386               15000000                    0                   0
5387              160000000                    0                   0
5388              175000000                    0                   0
5389               50000000                    0                   0
5390               20000000                    0                   0

[5391 rows x 6 columns]
```

5.5.2 Seek and Destroy

[73]:

```python
chars_to_remove = [',', '$']
columns_to_clean = ['USD_Production_Budget',
                    'USD_Worldwide_Gross',
                    'USD_Domestic_Gross']

for col in columns_to_clean:
    for char in chars_to_remove:
        # Replace each character with an empty string
        data[col] = data[col].astype(str).str.replace(char, "")
    # Convert column to a numeric data type
    data[col] = pd.to_numeric(data[col])
```

[74]:

```
data
```

[74]:
```
      Rank Release_Date                      Movie_Title  \
0     5293     8/2/1915           The Birth of a Nation
1     5140     5/9/1916                     Intolerance
2     5230   12/24/1916    20,000 Leagues Under the Sea
3     5299    9/17/1920  Over the Hill to the Poorhouse
4     5222     1/1/1925                  The Big Parade
…      …           …                              …
5386  2950    10/8/2018                             Meg
5387   126   12/18/2018                         Aquaman
5388    96   12/31/2020                     Singularity
5389  1119   12/31/2020           Hannibal the Conqueror
5390  2517   12/31/2020  Story of Bonnie and Clyde, The

      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross
0                    110000             11000000            10000000
1                    385907                    0                   0
2                    200000              8000000             8000000
3                    100000              3000000             3000000
4                    245000             22000000            11000000
…                       …                    …                   …
5386               15000000                    0                   0
5387              160000000                    0                   0
5388              175000000                    0                   0
5389               50000000                    0                   0
5390               20000000                    0                   0

[5391 rows x 6 columns]
```

5.5.3 Will convert the Release_Date column to a Pandas Datetime type.

[75]:
```
data.Release_Date = pd.to_datetime(data.Release_Date)
```

[76]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5391 entries, 0 to 5390
Data columns (total 6 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Rank                   5391 non-null   int64
 1   Release_Date           5391 non-null   datetime64[ns]
 2   Movie_Title            5391 non-null   object
 3   USD_Production_Budget  5391 non-null   int64
 4   USD_Worldwide_Gross    5391 non-null   int64
 5   USD_Domestic_Gross     5391 non-null   int64
dtypes: datetime64[ns](1), int64(4), object(1)
memory usage: 252.8+ KB
```

5.6 Descriptive Statistics

5.6.1 What is the average production budget of the films in the data set?

[77]:

```
avg_budget = data.USD_Production_Budget.mean()
avg_budget
```

[77]:

```
31113737.57837136
```

[78]:

```
print(f"The average production budget of the films is {round(avg_budget, 2)}")
```

```
The average production budget of the films is 31113737.58
```

5.6.2 What is the average worldwide gross revenue of films?

[79]:

```
avg_gross = data.USD_Worldwide_Gross.mean()
avg_gross
```

[79]:

```
88855421.96271564
```

[80]:

```
    print(f"The average worldwide gross revenue of the films is␣
 ↪{round(avg_gross, 2)}")
```

```
The average worldwide gross revenue of the films is 88855421.96
```

5.6.3 Are the bottom 25% of films actually profitable or do they lose money?

5.6.4 What are the highest production budget and highest worldwide gross revenue of any film?

5.6.5 How much revenue did the lowest and highest budget films make?

[81]:

```
data.describe()
```

[81]:

```
          Rank                  Release_Date  USD_Production_Budget  \
count 5,391.00                          5391               5,391.00
mean  2,696.00 2003-09-19 15:02:02.203672704         31,113,737.58
min       1.00           1915-08-02 00:00:00               1,100.00
25%   1,348.50           1999-12-02 12:00:00           5,000,000.00
50%   2,696.00           2006-06-23 00:00:00          17,000,000.00
75%   4,043.50           2011-11-23 00:00:00          40,000,000.00
max   5,391.00           2020-12-31 00:00:00         425,000,000.00
std   1,556.39                           NaN          40,523,796.88

       USD_Worldwide_Gross  USD_Domestic_Gross
count             5,391.00            5,391.00
mean         88,855,421.96       41,235,519.44
min                   0.00                0.00
25%           3,865,206.00        1,330,901.50
50%          27,450,453.00       17,192,205.00
```

```
75%          96,454,455.00       52,343,687.00
max       2,783,918,982.00      936,662,225.00
std         168,457,757.00       66,029,346.27
```

[82]:

```
data[data.USD_Production_Budget == 1100]
```

[82]:

```
      Rank Release_Date        Movie_Title  USD_Production_Budget  \
2427  5391   2005-05-08  My Date With Drew                   1100

      USD_Worldwide_Gross  USD_Domestic_Gross
2427               181041              181041
```

[83]:

```
data[data.USD_Production_Budget == 425000000]
```

[83]:

```
      Rank Release_Date Movie_Title  USD_Production_Budget  \
3529     1   2009-12-18      Avatar              425000000

      USD_Worldwide_Gross  USD_Domestic_Gross
3529           2783918982           760507625
```

6 Investigating the Zero Revenue Films

[84]:

```
zero_domestic = data[data.USD_Domestic_Gross == 0]
zero_domestic
```

[84]:

```
      Rank Release_Date                       Movie_Title  \
1     5140   1916-05-09                       Intolerance
6     4630   1927-12-08                             Wings
8     4240   1930-01-01                     Hell's Angels
17    4814   1936-10-20  Charge of the Light Brigade, The
27    4789   1941-10-28           How Green Was My Valley
…    …          …                                …
5386  2950   2018-10-08                               Meg
5387   126   2018-12-18                           Aquaman
5388    96   2020-12-31                       Singularity
5389  1119   2020-12-31            Hannibal the Conqueror
5390  2517   2020-12-31    Story of Bonnie and Clyde, The

      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross
1                    385907                    0                   0
6                   2000000                    0                   0
8                   4000000                    0                   0
17                  1200000                    0                   0
27                  1250000                    0                   0
…                     …                  …                 …
5386               15000000                    0                   0
5387              160000000                    0                   0
```

```
5388              175000000                    0                   0
5389               50000000                    0                   0
5390               20000000                    0                   0

[512 rows x 6 columns]
```

[85]:
```
zero_domestic.sort_values('USD_Production_Budget', ascending = False)
```

[85]:
```
      Rank Release_Date                         Movie_Title  \
5388    96   2020-12-31                         Singularity
5387   126   2018-12-18                             Aquaman
5384   321   2018-09-03                   A Wrinkle in Time
5385   366   2018-10-08                      Amusement Park
5090   556   2015-12-31  Don Gato, el inicio de la pandilla
…     …          …                                  …
4787  5371   2014-12-31                Stories of Our Lives
3056  5374   2007-12-31                         Tin Can Man
4907  5381   2015-05-19                    Family Motocross
5006  5389   2015-09-29             Signed Sealed Delivered
5007  5390   2015-09-29                A Plague So Pleasant

      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross
5388              175000000                    0                   0
5387              160000000                    0                   0
5384              103000000                    0                   0
5385              100000000                    0                   0
5090               80000000              4547660                   0
…                      …                  …                  …
4787                  15000                    0                   0
3056                  12000                    0                   0
4907                  10000                    0                   0
5006                   5000                    0                   0
5007                   1400                    0                   0

[512 rows x 6 columns]
```

[86]:
```
zero_worldwide = data[data.USD_Worldwide_Gross == 0]
zero_worldwide
```

[86]:
```
      Rank Release_Date                       Movie_Title  \
1     5140   1916-05-09                       Intolerance
6     4630   1927-12-08                             Wings
8     4240   1930-01-01                     Hell's Angels
17    4814   1936-10-20  Charge of the Light Brigade, The
27    4789   1941-10-28           How Green Was My Valley
…     …          …                                …
5386  2950   2018-10-08                               Meg
```

```
5387   126   2018-12-18                           Aquaman
5388    96   2020-12-31                       Singularity
5389  1119   2020-12-31            Hannibal the Conqueror
5390  2517   2020-12-31    Story of Bonnie and Clyde, The

      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross
1                    385907                    0                   0
6                   2000000                    0                   0
8                   4000000                    0                   0
17                  1200000                    0                   0
27                  1250000                    0                   0
…                     …                  …                 …
5386               15000000                    0                   0
5387              160000000                    0                   0
5388              175000000                    0                   0
5389               50000000                    0                   0
5390               20000000                    0                   0

[357 rows x 6 columns]
```

[87]:
```python
zero_worldwide.sort_values('USD_Production_Budget', ascending = False)
```

[87]:
```
      Rank Release_Date              Movie_Title  USD_Production_Budget  \
5388    96   2020-12-31              Singularity              175000000
5387   126   2018-12-18                  Aquaman              160000000
5384   321   2018-09-03        A Wrinkle in Time              103000000
5385   366   2018-10-08           Amusement Park              100000000
5058   880   2015-11-12         The Ridiculous 6               60000000
…    …          …                      …                    …
4787  5371   2014-12-31     Stories of Our Lives                  15000
3056  5374   2007-12-31              Tin Can Man                  12000
4907  5381   2015-05-19         Family Motocross                  10000
5006  5389   2015-09-29  Signed Sealed Delivered                   5000
5007  5390   2015-09-29     A Plague So Pleasant                   1400

      USD_Worldwide_Gross  USD_Domestic_Gross
5388                    0                   0
5387                    0                   0
5384                    0                   0
5385                    0                   0
5058                    0                   0
…                   …                 …
4787                    0                   0
3056                    0                   0
4907                    0                   0
5006                    0                   0
5007                    0                   0
```

```
[357 rows x 6 columns]
```

6.0.1 Filtering on Multiple Conditions

[88]:
```
international_releases = data.loc[(data.USD_Domestic_Gross == 0) & (data.
 ↪USD_Worldwide_Gross != 0)]
international_releases
```

[88]:
```
      Rank Release_Date              Movie_Title  USD_Production_Budget  \
71    4310   1956-02-16                 Carousel                3380000
1579  5087   2001-02-11  Everything Put Together                 500000
1744  3695   2001-12-31                 The Hole                7500000
2155  4236   2003-12-31                  Nothing                4000000
2203  2513   2004-03-31                The Touch               20000000
…     …             …                        …                     …
5340  1506   2017-04-14      Queen of the Desert               36000000
5348  2225   2017-05-05        Chāi dàn zhuānjiā               23000000
5360  4832   2017-07-03                Departure                1100000
5372  1856   2017-08-25                Ballerina               30000000
5374  4237   2017-08-25     Polina danser sa vie                4000000

      USD_Worldwide_Gross  USD_Domestic_Gross
71                   3220                   0
1579                 7890                   0
1744             10834406                   0
2155                63180                   0
2203              5918742                   0
…                       …                  …
5340              1480089                   0
5348             58807172                   0
5360                27561                   0
5372             48048527                   0
5374                36630                   0

[155 rows x 6 columns]
```

7 or …

[89]:
```
international_releases2 = data.query('USD_Domestic_Gross == 0 and␣
 ↪USD_Worldwide_Gross != 0')
international_releases2
```

[89]:
```
      Rank Release_Date              Movie_Title  USD_Production_Budget  \
71    4310   1956-02-16                 Carousel                3380000
1579  5087   2001-02-11  Everything Put Together                 500000
```

```
1744  3695   2001-12-31                 The Hole                 7500000
2155  4236   2003-12-31                  Nothing                 4000000
2203  2513   2004-03-31                The Touch                20000000
…    …          …                      …                    …
5340  1506   2017-04-14      Queen of the Desert                36000000
5348  2225   2017-05-05        Chāi dàn zhuānjiā                23000000
5360  4832   2017-07-03                Departure                 1100000
5372  1856   2017-08-25                Ballerina                30000000
5374  4237   2017-08-25     Polina danser sa vie                 4000000

      USD_Worldwide_Gross  USD_Domestic_Gross
71                   3220                   0
1579                 7890                   0
1744             10834406                   0
2155                63180                   0
2203              5918742                   0
…                   …                 …
5340              1480089                   0
5348             58807172                   0
5360                27561                   0
5372             48048527                   0
5374                36630                   0

[155 rows x 6 columns]
```

7.0.1 Unreleased Films

[90]:
```
# Date of Data Collection
scrape_date = pd.Timestamp('2018-5-1')
```

[91]:
```
future_releases = data[data.Release_Date >= scrape_date]
future_releases
```

[91]:
```
      Rank Release_Date                     Movie_Title  \
5384   321   2018-09-03               A Wrinkle in Time
5385   366   2018-10-08                  Amusement Park
5386  2950   2018-10-08                             Meg
5387   126   2018-12-18                         Aquaman
5388    96   2020-12-31                     Singularity
5389  1119   2020-12-31          Hannibal the Conqueror
5390  2517   2020-12-31  Story of Bonnie and Clyde, The

      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross
5384              103000000                    0                   0
5385              100000000                    0                   0
5386               15000000                    0                   0
5387              160000000                    0                   0
```

```
5388              175000000                    0                   0
5389               50000000                    0                   0
5390               20000000                    0                   0
```

[92]:
```
data_clean = data.drop(future_releases.index)
data_clean
```

[92]:
```
      Rank Release_Date                       Movie_Title   \
0     5293   1915-08-02             The Birth of a Nation
1     5140   1916-05-09                       Intolerance
2     5230   1916-12-24      20,000 Leagues Under the Sea
3     5299   1920-09-17    Over the Hill to the Poorhouse
4     5222   1925-01-01                    The Big Parade
…     …         …                               …
5379  1295   2017-10-02            John Wick: Chapter Two
5380    70   2017-10-03                Kong: Skull Island
5381    94   2017-12-05  King Arthur: Legend of the Sword
5382  1254   2017-12-05                          Snatched
5383  2521   2017-12-31                The Thousand Miles

      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross
0                    110000             11000000            10000000
1                    385907                    0                   0
2                    200000              8000000             8000000
3                    100000              3000000             3000000
4                    245000             22000000            11000000
…                      …                  …                 …
5379               40000000            166893990            92029184
5380              185000000            561137727           168052812
5381              175000000            140012608            39175066
5382               42000000             57850343            45850343
5383               20000000                    0                   0

[5384 rows x 6 columns]
```

7.0.2 Films that Lost Money

[93]:
```
losing_money = data_clean.loc[data_clean.USD_Worldwide_Gross < data_clean.
 ↪USD_Production_Budget]
losing_money
```

[93]:
```
      Rank Release_Date                       Movie_Title   \
1     5140   1916-05-09                       Intolerance
6     4630   1927-12-08                             Wings
8     4240   1930-01-01                     Hell's Angels
15    4738   1936-05-02                      Modern Times
17    4814   1936-10-20  Charge of the Light Brigade, The
```

```
…    …          …                                …
5371  4901   2017-07-28            An Inconvenient Sequel
5373  2161   2017-08-25                       Tulip Fever
5374  4237   2017-08-25              Polina danser sa vie
5381    94   2017-12-05  King Arthur: Legend of the Sword
5383  2521   2017-12-31                The Thousand Miles

      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross
1                    385907                    0                   0
6                   2000000                    0                   0
8                   4000000                    0                   0
15                  1500000               165049              163245
17                  1200000                    0                   0
…                     …                   …                 …
5371                1000000               130874              130874
5373               25000000                    0                   0
5374                4000000                36630                   0
5381              175000000            140012608            39175066
5383               20000000                    0                   0

[2007 rows x 6 columns]
```

[94]:

```
len(losing_money)/len(data_clean)
```

[94]:

```
0.37277117384843983
```

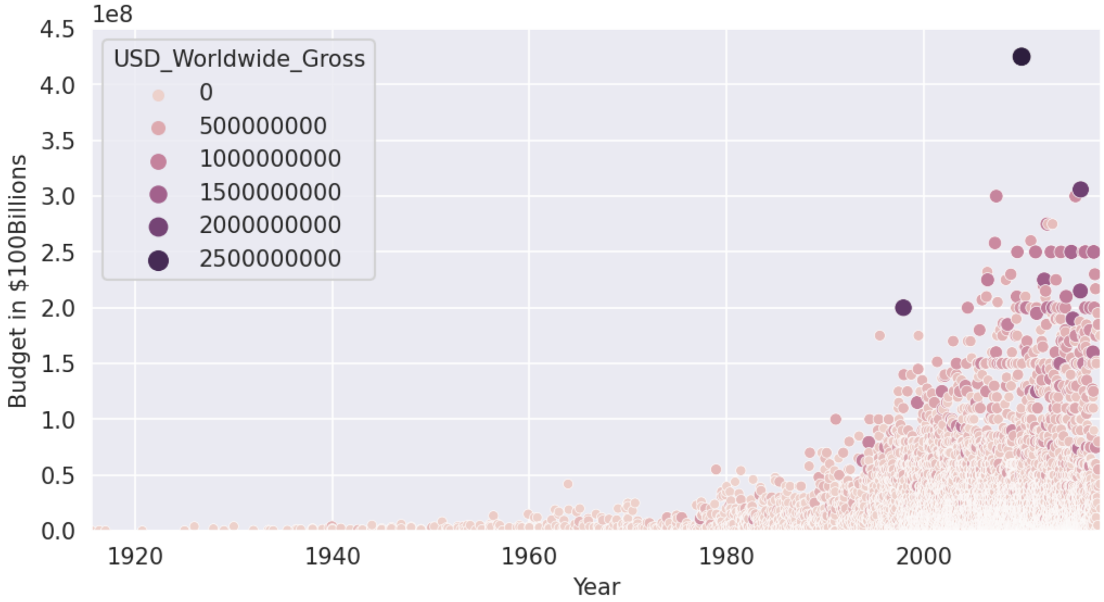
8 Seaborn for Data Viz: Bubble Charts

[95]:

```
import seaborn as sns
```

[96]:

```
plt.figure(figsize = (8,4), dpi=150)

with sns.axes_style('darkgrid'):
    ax = sns.scatterplot(data=data_clean,
                     x='Release_Date',
                     y='USD_Production_Budget',
                     hue = 'USD_Worldwide_Gross',
                     size = 'USD_Worldwide_Gross')
    ax.set(ylim = (0, 450000000),
       xlim = (data_clean.Release_Date.min(), data_clean.Release_Date.max()),
       ylabel = 'Budget in $100Billions',
       xlabel = 'Year')

plt.show()
```



9 Converting Years to Decades

[97]:
```
dt_index = pd.DatetimeIndex(data_clean.Release_Date)
dt_index
```

[97]:
```
DatetimeIndex(['1915-08-02', '1916-05-09', '1916-12-24', '1920-09-17',
               '1925-01-01', '1925-12-30', '1927-12-08', '1929-01-02',
               '1930-01-01', '1931-12-31',
               …
               '2017-08-25', '2017-09-06', '2017-09-06', '2017-10-02',
               '2017-10-02', '2017-10-02', '2017-10-03', '2017-12-05',
               '2017-12-05', '2017-12-31'],
              dtype='datetime64[ns]', name='Release_Date', length=5384,
freq=None)
```

9.0.1 Turn dates into years

[98]:
```
years = dt_index.year
years
```

[98]:
```
Index([1915, 1916, 1916, 1920, 1925, 1925, 1927, 1929, 1930, 1931,
       …
       2017, 2017, 2017, 2017, 2017, 2017, 2017, 2017, 2017, 2017],
      dtype='int32', name='Release_Date', length=5384)
```

9.0.2 Turn years into decades with floor division. 1995 / 10 = 199.5 , but 1995 // 10 = 199. So 199 * 10 = 1990

[99]:
```
decades = (years // 10)*10
decades
```

[99]:
```
Index([1910, 1910, 1910, 1920, 1920, 1920, 1920, 1920, 1930, 1930,
        …
       2010, 2010, 2010, 2010, 2010, 2010, 2010, 2010, 2010, 2010],
      dtype='int32', name='Release_Date', length=5384)
```

9.0.3 Add a “Decades” column in the data frame

[100]:
```
data_clean['Decade'] = decades
data_clean
```

[100]:
```
      Rank Release_Date                       Movie_Title  \
0     5293   1915-08-02             The Birth of a Nation
1     5140   1916-05-09                       Intolerance
2     5230   1916-12-24      20,000 Leagues Under the Sea
3     5299   1920-09-17    Over the Hill to the Poorhouse
4     5222   1925-01-01                    The Big Parade
…    …          …                                   …
5379  1295   2017-10-02            John Wick: Chapter Two
5380    70   2017-10-03                Kong: Skull Island
5381    94   2017-12-05  King Arthur: Legend of the Sword
5382  1254   2017-12-05                          Snatched
5383  2521   2017-12-31                The Thousand Miles

      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross  Decade
0                    110000             11000000            10000000    1910
1                    385907                    0                   0    1910
2                    200000              8000000             8000000    1910
3                    100000              3000000             3000000    1920
4                    245000             22000000            11000000    1920
…                       …                  …                 …     …
5379               40000000            166893990            92029184    2010
5380              185000000            561137727           168052812    2010
5381              175000000            140012608            39175066    2010
5382               42000000             57850343            45850343    2010
5383               20000000                    0                   0    2010

[5384 rows x 7 columns]
```

9.0.4 Separate the "old" (before 1969) and "New" (1970s onwards) Films

[101]:

```
old_films = data_clean[data_clean.Decade <= 1969]
new_films = data_clean[data_clean.Decade > 1969]
old_films
```

[101]:

```
     Rank Release_Date                         Movie_Title  \
0    5293   1915-08-02              The Birth of a Nation
1    5140   1916-05-09                        Intolerance
2    5230   1916-12-24        20,000 Leagues Under the Sea
3    5299   1920-09-17       Over the Hill to the Poorhouse
4    5222   1925-01-01                      The Big Parade
..    …          …                                  …
148  2375   1969-10-15                    Paint Your Wagon
149  3831   1969-10-24  Butch Cassidy and the Sundance Kid
150  2175   1969-12-16                        Hello, Dolly
151  3613   1969-12-18     On Her Majesty's Secret Service
152  4195   1969-12-19                               Topaz

     USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross  Decade
0                   110000             11000000            10000000    1910
1                   385907                    0                   0    1910
2                   200000              8000000             8000000    1910
3                   100000              3000000             3000000    1920
4                   245000             22000000            11000000    1920
..                     …                    …                   …     …
148               20000000             31678778            31678778    1960
149                6000000            102308900           102308900    1960
150               24000000             33208099            33208099    1960
151                8000000             82000000            22800000    1960
152                4000000              6000000             6000000    1960

[153 rows x 7 columns]
```

[102]:

```
new_films
```

[102]:

```
      Rank Release_Date                        Movie_Title  \
153   2159   1970-01-01                           Waterloo
154   2270   1970-01-01                       Darling Lili
155   3136   1970-01-01                             Patton
156   3277   1970-01-01                 The Molly Maguires
157   4265   1970-01-01                              M*A*S*H
…      …          …                                  …
5379  1295   2017-10-02             John Wick: Chapter Two
5380    70   2017-10-03                 Kong: Skull Island
5381    94   2017-12-05  King Arthur: Legend of the Sword
5382  1254   2017-12-05                           Snatched
5383  2521   2017-12-31                 The Thousand Miles
```

```
      USD_Production_Budget  USD_Worldwide_Gross  USD_Domestic_Gross  Decade
153                25000000                    0                   0    1970
154                22000000              5000000             5000000    1970
155                12000000             62500000            62500000    1970
156                11000000              2200000             2200000    1970
157                 3500000             81600000            81600000    1970
…                       …                   …                 …     …
5379               40000000            166893990            92029184    2010
5380              185000000            561137727           168052812    2010
5381              175000000            140012608            39175066    2010
5382               42000000             57850343            45850343    2010
5383               20000000                    0                   0    2010

[5231 rows x 7 columns]
```

[103]:

```
old_films.describe()
```

[103]:

```
          Rank                   Release_Date  USD_Production_Budget  \
count   153.00                            153                 153.00
mean  4,274.77  1954-06-10 04:04:42.352941184           4,611,297.65
min   1,253.00            1915-08-02 00:00:00             100,000.00
25%   3,973.00            1946-01-01 00:00:00           1,250,000.00
50%   4,434.00            1956-12-23 00:00:00           2,900,000.00
75%   4,785.00            1964-10-22 00:00:00           5,000,000.00
max   5,299.00            1969-12-19 00:00:00          42,000,000.00
std     742.14                            NaN           5,713,648.85

       USD_Worldwide_Gross  USD_Domestic_Gross   Decade
count               153.00              153.00   153.00
mean         30,419,634.38       22,389,473.87 1,949.15
min                   0.00                0.00 1,910.00
25%           5,273,000.00        5,000,000.00 1,940.00
50%          10,000,000.00       10,000,000.00 1,950.00
75%          33,208,099.00       28,350,000.00 1,960.00
max         390,525,192.00      198,680,470.00 1,960.00
std          54,931,828.93       32,641,752.41    12.72
```

[104]:

```
old_films.sort_values('USD_Production_Budget', ascending = False).head(10)
```

[104]:

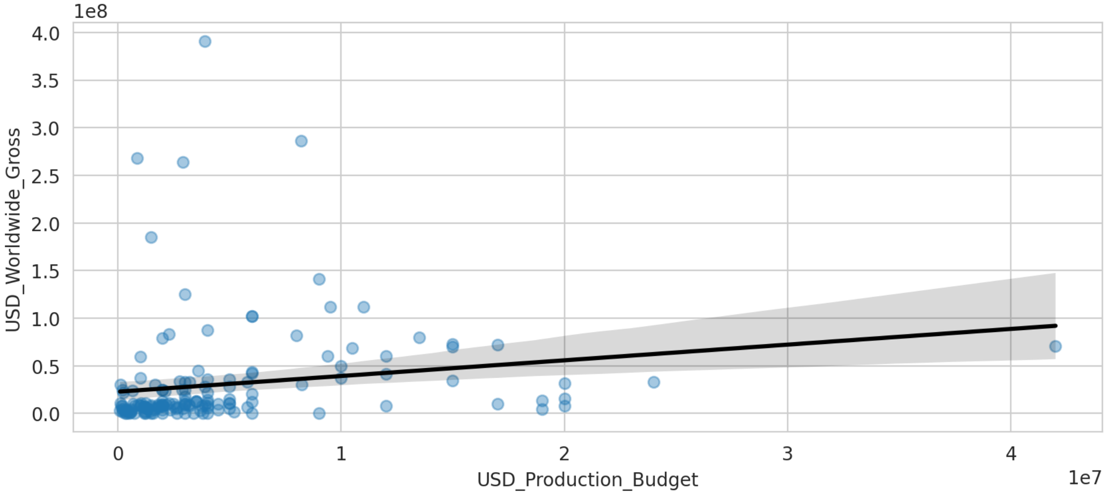
```
     Rank Release_Date                     Movie_Title  USD_Production_Budget  \
109  1253   1963-12-06                       Cleopatra               42000000
150  2175   1969-12-16                    Hello, Dolly               24000000
143  2465   1969-01-01                  Sweet Charity               20000000
118  2425   1965-02-15  The Greatest Story Ever Told               20000000
148  2375   1969-10-15                Paint Your Wagon               20000000
110  2552   1964-01-01    The Fall of the Roman Empire               19000000
```

```
98   2546   1962-08-11          Mutiny on The Bounty               19000000
114  2670   1964-10-22                  My Fair Lady               17000000
102  2698   1963-01-01              55 Days at Peking              17000000
125  2831   1966-10-10                        Hawaii               15000000

     USD_Worldwide_Gross  USD_Domestic_Gross  Decade
109             71000000            57000000    1960
150             33208099            33208099    1960
143              8000000             8000000    1960
118             15473333            15473333    1960
148             31678778            31678778    1960
110              4750000             4750000    1960
98              13680000            13680000    1960
114             72070955            72000000    1960
102             10000000            10000000    1960
125             34562222            34562222    1960
```

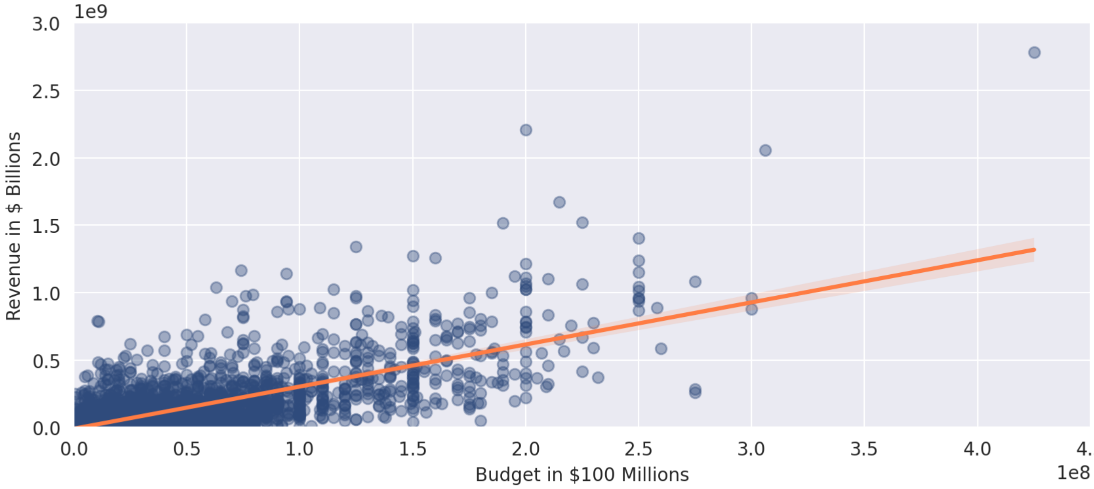
10 Seaborn Regression Plots

[105]:

```
plt.figure(figsize = (10,4), dpi=200)


with sns.axes_style('whitegrid'):
    ax = sns.regplot(data = old_films,
                     x = 'USD_Production_Budget',
                     y = 'USD_Worldwide_Gross',
                     scatter_kws = {'alpha' : 0.4},
                     line_kws = {'color' : 'black'})
```



```
[106]: plt.figure(figsize = (10,4), dpi=200)


       with sns.axes_style('darkgrid'):
           ax = sns.regplot(data = new_films,
                            x = 'USD_Production_Budget',
                            y = 'USD_Worldwide_Gross',
                            color = '#2f4b7c',
                            scatter_kws = {'alpha' : 0.4},
                            line_kws = {'color' : '#ff7c43'})

           ax.set(ylim = (0, 3000000000),
                  xlim = (0, 450000000),
                  ylabel = 'Revenue in $ Billions',
                  xlabel = 'Budget in $100 Millions')
```



11 Running Regression with scikit-learn

$$REV\hat{E}NUE = \theta_0 + \theta_1 BUDGET$$

```
[107]: from sklearn.linear_model import LinearRegression
```

11.0.1 Will run a linear regression for the old_films and Calculate the intercept, slope and r-squared.

11.0.2 How much of the variance in movie revenue does the linear model explain in this case?

```
[108]: regression = LinearRegression()
```

11.0.3 Explanatory variable (Feature in ML)

```
[109]: X = pd.DataFrame(new_films, columns = ['USD_Production_Budget'])
```

11.0.4 Response variable (Target in ML)

```
[110]: y = pd.DataFrame(new_films, columns = ['USD_Worldwide_Gross'])
```

11.0.5 Creating DataFrames because LinearRegression doesn't like to receive Pandas Series

12 Will find the best fit line

```
[111]: regression.fit(X, y)
```

```
[111]: LinearRegression()
```

```
[163]: regression.intercept_    #Theta zero
```

```
[163]: array([-8650768.00661042])
```

```
[164]: regression.coef_         #Theta one
```

```
[164]: array([[3.12259592]])
```

```
[168]: regression.score(X, y)   #R²
```

```
[168]: 0.5577032617720403
```

Our model explains about 56% of the variance in movie revenue.

12.0.1 Regression

```
[113]: X = pd.DataFrame(old_films, columns = ['USD_Production_Budget'])
       y = pd.DataFrame(old_films, columns = ['USD_Worldwide_Gross'])
```

```
[114]: regression.fit(X, y)
```

```
[114]: LinearRegression()
```

```
[115]: regression.intercept_
```

```
[115]: array([22821538.63508039])
```

```
[116]: regression.coef_
```

```
[116]: array([[1.64771314]])
```

```
[117]: regression.score(X, y)
```

```
[117]: 0.02937258620576877
```

```
[118]: print(f'The intercept is : {regression.intercept_}')
       print(f'The slope is : {regression.coef_}')
       print(f'The R² is : {regression.score(X, y)}')
       print(f'That means our model explains {round(regression.score(X, y) * 100)}% of␣
        ↪the variance in movie revenue.')
```

```
The intercept is : [22821538.63508039]
The slope is : [[1.64771314]]
The R² is : 0.02937258620576877
That means our model explains 3% of the variance in movie revenue.
```

12.0.2 How much global revenue does our model estimate for a film with a budget of i.e. $350 million?

```
[119]: budget = 350000000
```

```
[120]: revenue_estimate = regression.intercept_[0] + regression.coef_[0,0] * budget
```

```
[121]: revenue_estimate
```

```
[121]: 599521139.0388364
```

```
[122]: revenue_estimate = round(revenue_estimate, -6)    # me to -6 can round stin 6h␣
        ↪taxi aristera tis ypodiastolis. Sto ekatommyrio dld, 10^6
```

```
[123]: revenue_estimate
```

```
[123]: 600000000.0
```

```
[124]: print(f'The revenue estimate for a movie with a budget of ${budget} is expected␣
        ↪to be around ${revenue_estimate:.10}')
```

```
The revenue estimate for a movie with a budget of $350000000 is expected to be
around $600000000.0
```