Disney

December 6, 2021

1 Exploring Disney's Box Office and Revenue Performance

1.1 Jesse Sallis Bcom, MIB

1.1.1 Dec 2021

Introduction: Purpose of notebook is to explore Disney datasets to evaluate multiple performance metrics over time. Analysis will consist of two sections: Revenue and Box Office

1.1.2 Question's that will be explored:

- Has Disney's box office performance improved over time?
- Has Disney's revenue streams diversified or consolidated over time? Have they grown?

The main driver for these questions is to evaluate Disney's ability to maintain success through time. Has Disney been able to evolve/diversify?

1.1.3 Datasets

For the purpose of this excerise, 2 datasets will be used Data can be found here https://data.world/kgarrett/disney-character-success-00-16 and follows a https://creativecommons.org/licenses/by/4.0/ license.

- **Disney_movies_total_gross.csv**
  - historical view on Disney movie releases and their gross box office performance (older titles are inflation adjusted)
- **disney_revenue_1991-2016.csv**
  - revenue streams by Disney department over time ex: Disney Parks, Disney Studio

1.1.4 Method and Results

Only the above two datasets are needed to explore and answer initial questions. The analysis process will flow like: * Load in datasets as dataframe * Run some initial diagnoisis on data quality (NaNs, missing values, incorect dtypes) * Wrangle/tidy up dataframes * Calculate important summary statistics (mean, min, max) * Create simple visualizations to better understand data and recognize any noticable trends * Summarize and group data in an appropiate way that will help answer initial questions

To assist with excuting the above, I will develop a function to clean/tidy data. This will help with creating reproducible code and enable tidy dataframes for future analysis.

[1]:
```
!set PATH=/Library/TeX/texbin:$PATH
```

[2]:
```
#importing required packages\
import pandas as pd
import altair as alt
import melt_script as ms

#importing datasets
revenue_df = pd.read_csv('data/disney_movies_total_gross.
 ↪csv',parse_dates=['release_date'])
movies_df = pd.read_csv('data/disney_revenue_1991-2016.
 ↪csv',parse_dates=['Year'])
```

1.2 Revenue Analysis

Let's look at movies df - initial thoughts… there are some NaN values that we will need to correct. NaN values in this dataset seem to = zero. All columns aside from Year (which will need to be datetime) will need to be converted to float dtype.

[3]:
```
print(movies_df.info())
print('\n')
print(movies_df.dtypes)
print('\n')
movies_df.head(5)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 26 entries, 0 to 25
Data columns (total 7 columns):
 #   Column                           Non-Null Count  Dtype
---  ------                           --------------  -----
 0   Year                             26 non-null     datetime64[ns]
 1   Studio Entertainment[NI 1]       25 non-null     float64
 2   Disney Consumer Products[NI 2]   24 non-null     float64
 3   Disney Interactive[NI 3][Rev 1]  12 non-null     float64
 4   Walt Disney Parks and Resorts    26 non-null     float64
 5   Disney Media Networks            23 non-null     object
 6   Total                            26 non-null     int64
dtypes: datetime64[ns](1), float64(4), int64(1), object(1)
memory usage: 1.5+ KB
None


Year                                datetime64[ns]
Studio Entertainment[NI 1]                 float64
Disney Consumer Products[NI 2]             float64
Disney Interactive[NI 3][Rev 1]            float64
Walt Disney Parks and Resorts              float64
```

```
Disney Media Networks                         object
Total                                          int64
dtype: object
```

[3]:
```
        Year  Studio Entertainment[NI 1]  Disney Consumer Products[NI 2]  \
0 1991-01-01                      2593.0                           724.0
1 1992-01-01                      3115.0                          1081.0
2 1993-01-01                      3673.4                          1415.1
3 1994-01-01                      4793.0                          1798.2
4 1995-01-01                      6001.5                          2150.0

   Disney Interactive[NI 3][Rev 1]  Walt Disney Parks and Resorts  \
0                              NaN                         2794.0
1                              NaN                         3306.0
2                              NaN                         3440.7
3                              NaN                         3463.6
4                              NaN                         3959.8

  Disney Media Networks  Total
0                   NaN   6111
1                   NaN   7502
2                   NaN   8529
3                   359  10414
4                   414  12525
```

[4]:
```
#tidying up data by fiilling in NaNs, removing commas from columns, converting␣
↪to float and setting index to year
movies_df = movies_df.fillna(method='ffill').fillna(0)
movies_df = movies_df.replace(',','', regex=True)
movies_df['Disney Media Networks'] = movies_df['Disney Media Networks'].
↪astype('float')
movies_df['Year'] = movies_df['Year'].dt.year
movies_df = movies_df.rename(columns={'Studio Entertainment[NI 1]':'Studio␣
↪Entertainment','Disney Consumer Products[NI 2]':'Disney Consumer␣
↪Products','Disney Interactive[NI 3][Rev 1]':'Disney Interactive'})
```

[5]:
```
movies_df.head()
```

[5]:
```
   Year  Studio Entertainment  Disney Consumer Products  Disney Interactive  \
0  1991                2593.0                     724.0                 0.0
1  1992                3115.0                    1081.0                 0.0
2  1993                3673.4                    1415.1                 0.0
3  1994                4793.0                    1798.2                 0.0
4  1995                6001.5                    2150.0                 0.0
```

	Walt Disney Parks and Resorts	Disney Media Networks	Total
0	2794.0	0.0	6111
1	3306.0	0.0	7502
2	3440.7	0.0	8529
3	3463.6	359.0	10414
4	3959.8	414.0	12525

Now that we have a tidy df, let's start by graphing total revenue over time with each business unit a unique color in stacked bar. This will visualize if total revenue has increased and provide some insight into each business unit's contribution. Melt may be needed to get df is appropiate format for a stacked bar chart

[6]:
```
#melting df so we are able to stack business lines in bar chart
stacked_df = ms.custom_melt(movies_df,['Year'],['Studio Entertainment','Disney␣
↪Consumer Products','Disney Interactive','Walt Disney Parks and␣
↪Resorts','Disney Media␣
↪Networks'],var_name='Business_Line',value_name='Revenue')
stacked_df.head()
```

[6]:

	Year	Business_Line	Revenue
0	1991	Studio Entertainment	2593.0
1	1992	Studio Entertainment	3115.0
2	1993	Studio Entertainment	3673.4
3	1994	Studio Entertainment	4793.0
4	1995	Studio Entertainment	6001.5

[7]:
```
stacked = alt.Chart(stacked_df).mark_bar().encode(
                x=alt.X("Year:O", title="Year"),
                y=alt.Y('sum(Revenue):Q',title='Revenue $100M',axis=alt.
↪Axis(format='$,.0f')), color='Business_Line',
                ).properties(title="Disney Revenue by Business Line")
stacked
```

[7]:
```
alt.Chart(…)
```

1.2.1 Revenue discussion

Looking at the above visualization, it is clear that Disney revenues have increased steadily over time, up ~6x during the assessed time period. In terms of business line contribution, the most notable change is the massive growth in Media Networks over time. This makes sense, as Disney grows it's content offerings (ESPN) - expected revenue would grow as well. In 1991, Parks and Studio Entertaonment consisted of the majoriity of generated revenue. In 2016, they only made up ~50% - with Studio Entertainment revenue remaining flat over time and it's share of total revenue declining. From the above, Disney has been able to diversify and grow revenue streams over time as was initially expected.

- Some other questions that would interesting to explore:
  - Has revenue growth outpaced inflation?

– How is revenue growth correlated with net earnings?

1.3 Box Office Analysis

Let's look at revenue df - initial thoughts… the dollar sign could cause issues as the box office columns are object format (we will need float/int formatting for analysis on values). A few NaN values that will need to be evaluated to determine if they should be corrected.

[8]:
```
print(revenue_df.info())
print('\n')
print(revenue_df.dtypes)
print('\n')
revenue_df.head(5)
```

```
<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    datetime64[ns]
 2   genre                    562 non-null    object
 3   MPAA_rating              523 non-null    object
 4   total_gross              579 non-null    object
 5   inflation_adjusted_gross  579 non-null    object
dtypes: datetime64[ns](1), object(5)
memory usage: 27.3+ KB
None


movie_title                        object
release_date               datetime64[ns]
genre                              object
MPAA_rating                        object
total_gross                        object
inflation_adjusted_gross           object
dtype: object
```

[8]:

	movie_title	release_date	genre	MPAA_rating	total_gross	inflation_adjusted_gross
0	Snow White and the Seven Dwarfs	1937-12-21	Musical	G		
1	Pinocchio	1940-02-09	Adventure	G		
2	Fantasia	1940-11-13	Musical	G		
3	Song of the South	1946-11-12	Adventure	G		
4	Cinderella	1950-02-15	Drama	G		

```
0  $184,925,485           $5,228,953,251
1   $84,300,000           $2,188,229,052
2   $83,320,000           $2,187,090,808
3   $65,000,000           $1,078,510,579
4   $85,000,000             $920,608,730
```

[9]:
```
#cleaning up data, dropping uneeded columns, converting columns to appropiate␣
↪dtype
revenue_df = revenue_df.dropna(axis='columns')
revenue_df['total_gross'] = revenue_df['total_gross'].str.
↪replace('$','',regex=False).str.replace(',', '',regex=False).astype(float)
revenue_df['inflation_adjusted_gross'] = revenue_df['inflation_adjusted_gross'].
↪str.replace('$','',regex=False).str.replace(',', '').astype(float)
```

[10]:
```
revenue_df.head()
```

[10]:
```
                      movie_title release_date  total_gross  \
0  Snow White and the Seven Dwarfs   1937-12-21  184925485.0
1                        Pinocchio   1940-02-09   84300000.0
2                         Fantasia   1940-11-13   83320000.0
3                Song of the South   1946-11-12   65000000.0
4                       Cinderella   1950-02-15   85000000.0

   inflation_adjusted_gross
0              5.228953e+09
1              2.188229e+09
2              2.187091e+09
3              1.078511e+09
4              9.206087e+08
```

Now that we have a clean df, I will create a decade column and groupby decade to get average box-office per film, total box office and number of film releases. Then will create a bar chart to show trend over time. I will use inflation adjusted value in order to get a like for like comparison.

[11]:
```
#create decade column, thanks to https://towardsdatascience.com/
↪two-pandas-functions-you-must-know-for-easy-data-manipulation-in-python-2f6d0a2ef3e5
revenue_df['Decade'] = revenue_df.release_date.dt.year.apply(lambda x: str(x)[:
↪3]+'0s')

# apply summary statistics to groupby and creating new df to be graphed
decade_df = revenue_df.groupby('Decade').
↪agg(total_gross=('inflation_adjusted_gross','sum'),mean_gross=('inflation_adjusted_gross','
↪reset_index()
decade_df
```

[11]:
```
  Decade   total_gross    mean_gross  movie_count
0  1930s  5.228953e+09  5.228953e+09            1
```

1	1940s	5.453830e+09	1.817943e+09	3
2	1950s	2.706430e+09	6.766075e+08	4
3	1960s	2.989484e+09	4.270692e+08	7
4	1970s	1.062951e+09	1.181057e+08	9
5	1980s	4.636550e+09	7.600902e+07	61
6	1990s	1.774330e+10	7.518349e+07	236
7	2000s	1.579150e+10	9.181107e+07	172
8	2010s	1.315049e+10	1.529127e+08	86

Quickly looking at the above, we can see Disney has dramtically increased the number of films released in the 1990's with a slight drop off in 2000's. Seems that average gross per film peaked in 1930's (however small sample size). Let's graph it

[12]:
```
# First lets see yearly release count distrubution because im curious
movie_hist = alt.Chart(revenue_df.groupby(revenue_df.release_date.dt.year).
 ↪agg({'movie_title':'count'}).reset_index(), width=500, height=300).
 ↪mark_bar().encode(
                x=alt.X('movie_title:Q', bin=True),
                y=alt.Y('count():Q')
                ).properties(title="Movie Releases per year")

movie_hist
```

[12]: alt.Chart(…)

[13]:
```
base = alt.Chart(decade_df,width=500, height=300).encode(
                alt.X('Decade:O', axis=alt.Axis(title=None))
                )
area = base.mark_bar(color='#57A44C').encode(
                alt.Y('total_gross:Q',title='Revenue $M',axis=alt.
 ↪Axis(format='$,.0f')))
line = base.mark_line(stroke='#5276A7', interpolate='monotone').encode(
                alt.Y('mean_gross:Q',axis=alt.Axis(title='Avg Box␣
 ↪Office',format='$,.0f')))

alt.layer(area, line).resolve_scale(y = 'independent')
```

[13]: alt.LayerChart(…)

1.3.1 Box Office discussion

Looking at the above, it clear that Disney had massive box office success in 1990's with it somewhat dropping off the following two decades. Avg box office movie actually increased in the 2000s/2010s as it seems Disney prioritized quality over quantity. Early movies had significant success when adjusted for inflation. So has Disney's box office improved over time? From the 1970s low, yes there is a large increase over time in total box office performance, however, this more seems driven by number of releases rather than single movie box office performance. As total box office revenue

has come down in recent decades, the slight uptick in avgerage box office revenue per movie shows fewer release performing better. Seems Disney's box office strategy has evolved over time. Makes sense given their rich history and engrained part of Western pop culture.

Some additional questions that would be interesting to explore: * What % of box office revenue comes from animated shows vs live action? * How have acquiations like Marvel/Star Wars delivered on box office earnings?

Resources referrenced throughout body of work.