Movie Revenue

Production of movies is a combination of multiple factors to create a successful, high revenue, movie. This project is to determine the effect of budget and summer releases on the overall revenue. Do either of these factors show a direct impact on the success of movies? More specifically, does a higher budget result in higher success? Do summer movies have an impact on revenue? These are concerns for both the director (budget) and the production companies.

Two statistical tests will be conducted to determine if there are any significant relationship between budget, season of release, and revenue. The first analysis will calculate any correlation between budget and revenue. The second analysis will calculate the significance of summer releases compared to non-summer releases.

The data set used is “Movie Dataset: Budgets, Genres, Insights” provided by Kaggle.com. Data provided includes budget, genre (multiple types per movie), date released, movie name, cast, director, and movie details such as original language, overview, tagline, runtime, and much more. Specifically this project will use budget and revenue, both integers, and release date which is an object.

```
import pandas as pd
import numpy as np
import seaborn as sns
sns.set()
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
from scipy import stats
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Mounted at /content/gdrive
```

```
# Database import from Google Drive

movie_df = pd.read_csv('/content/gdrive/My Drive/Colab Datasets/movie_dataset.csv')
```

```
# Basic information of the dataset

movie_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4803 entries, 0 to 4802
Data columns (total 24 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   index                 4803 non-null   int64  
 1   budget                4803 non-null   int64  
 2   genres                4775 non-null   object 
 3   homepage              1712 non-null   object 
 4   id                    4803 non-null   int64  
 5   keywords              4391 non-null   object 
 6   original_language     4803 non-null   object 
 7   original_title        4803 non-null   object 
 8   overview              4800 non-null   object 
 9   popularity            4803 non-null   float64
 10  production_companies  4803 non-null   object 
 11  production_countries  4803 non-null   object 
 12  release_date          4802 non-null   object 
 13  revenue               4803 non-null   int64  
 14  runtime               4801 non-null   float64
 15  spoken_languages      4803 non-null   object 
 16  status                4803 non-null   object 
 17  tagline               3959 non-null   object 
 18  title                 4803 non-null   object 
 19  vote_average          4803 non-null   float64
 20  vote_count            4803 non-null   int64  
 21  cast                  4760 non-null   object 
 22  crew                  4803 non-null   object 
 23  director              4773 non-null   object 
dtypes: float64(3), int64(5), object(16)
memory usage: 900.7+ KB
```

The dataframe now needs to be cleaned before we start performing our analysis. First, eliminate all movies released before 2011, then delete movies whose budget and revenue are set to 0. Finally, create a new column to determine if the movie was released in the summer.

```
# Create a new data frame with only movies between 2011 and 2017
# Convert the value from object to datetime to properly filter by date
```

```
movie_df['release_date'] = pd.to_datetime(movie_df['release_date'])
recent_movie_df = movie_df[(movie_df['release_date'] >= '2011-01-01')]
```

```
#Check current size of data frame

recent_movie_df.shape
```

```
(1221, 24)
```

```
# Delete all movies whose budget and revenue listed as 0

usable_movie_df = recent_movie_df[(recent_movie_df['budget'] > 0) & (recent_movie_df['revenue'] > 0)]
```

```
# Check number movies in the updated data frame

usable_movie_df.shape
```

```
(785, 24)
```

```
# Add a new column called "summer_movie" with a yes/no value depending on the month of release. Summer constitutes months from June to August

usable_movie_df['summer_movie'] = 'no'

usable_movie_df.loc[usable_movie_df['release_date'].dt.month == 6 | 7, 'summer_movie'] = 'yes'
usable_movie_df.loc[usable_movie_df['release_date'].dt.month == 8, 'summer_movie'] = 'yes'
```

```
<ipython-input-9-d36c74bc17a9>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-cc
  usable_movie_df['summer_movie'] = 'no'
```

```
# Verify information in new column is correct inputted with yes/no for a total of 785

usable_movie_df['summer_movie'].value_counts()
```

```
no     652
yes    133
Name: summer_movie, dtype: int64
```

Now its time to conduct the analysis of our data. The first analysis is to create a scatterplot and caluculating the correlation coefficient to validate the following hypotheses.

Ho: There is no positive, direct variation between budget and revenue.

Ha: There is a positive, direct variation between budget and revenue.

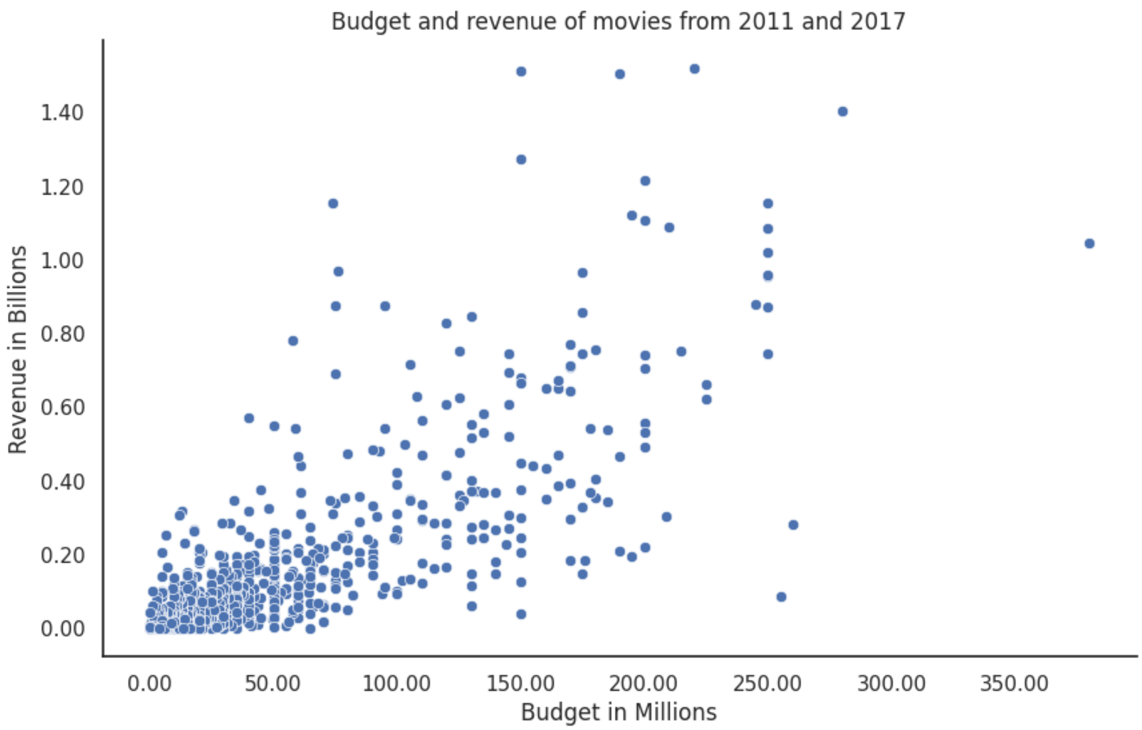
```
# Look at the descriptive statistics for general information

usable_movie_df[['budget', 'revenue']].describe()
```

	budget	revenue
count	7.850000e+02	7.850000e+02
mean	5.174688e+07	1.635123e+08
std	5.682072e+07	2.352975e+08
min	1.000000e+01	1.100000e+01
25%	1.300000e+07	2.400000e+07
50%	3.000000e+07	7.863626e+07
75%	6.600000e+07	1.884416e+08
max	3.800000e+08	1.519558e+09

```
# Construct a scatterplot with proper labels and budget on the x-axis

sns.set_style("white")
```

```
plt.figure(figsize = (10,6))
scatplt = sns.scatterplot(x = 'budget', y = 'revenue', data = usable_movie_df)
sns.despine()
xlabels = ['{:,.2f}'.format(x) for x in scatplt.get_xticks()/1000000]
scatplt.set_xticklabels(xlabels)
ylabels = ['{:,.2f}'.format(y) for y in scatplt.get_yticks()/1000000000]
scatplt.set_yticklabels(ylabels)
plt.xlabel('Budget in Millions')
plt.ylabel('Revenue in Billions')
plt.title('Budget and revenue of movies from 2011 and 2017')
```

```
<ipython-input-12-355d497b2098>:8: UserWarning: FixedFormatter should only be used together with FixedLocator
  scatplt.set_xticklabels(xlabels)
<ipython-input-12-355d497b2098>:10: UserWarning: FixedFormatter should only be used together with FixedLocator
  scatplt.set_yticklabels(ylabels)
Text(0.5, 1.0, 'Budget and revenue of movies from 2011 and 2017')
```




```
# Calculate the correlation coefficient and the likelihood of this relationship to exist

stats.pearsonr(usable_movie_df['budget'], (usable_movie_df['revenue']))
```

```
PearsonRResult(statistic=0.7784406371087922, pvalue=1.633532676370216e-160)
```

The correlation coefficient is 0.778 which does support a moderate positive correlation. However, the p-value is < 0.05 which signifies rejection of the null hypothesis. Statistically, there is no significant relation between budget and revenue. This is evident as the budget increases, revenue is positive yet scattered on various levels.

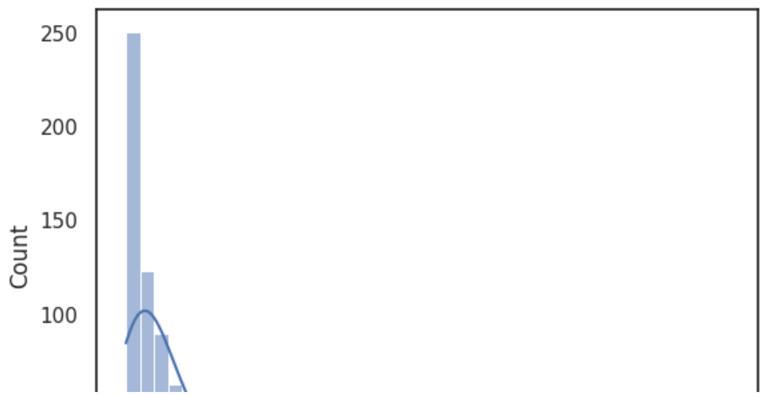
The second analysis is testing the impact of the releasing movies in the summer vice any other season on revenue. Here are the hypotheses.

Ho: Movies released in the summer months have no impact on revenue.

Ha: Movies released in the summer months have an impact on revenue.

```
# Quick visualization of revenue distribution

sns.histplot(usable_movie_df['revenue'], kde=True)
```

```
<Axes: xlabel='revenue', ylabel='Count'>
```



Since the distribution is not evenly distributed, a Mann-Whitney test will be used to calculate the p-value.

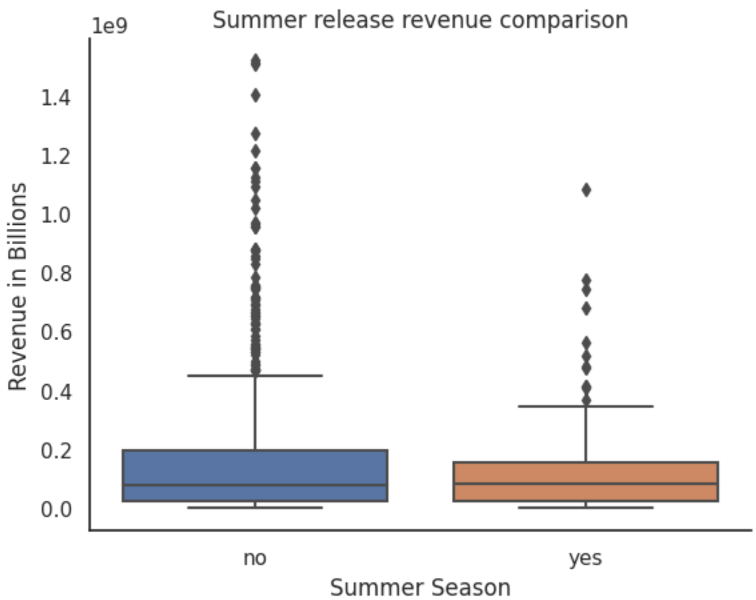
```
# Separate the movies into two groups based on summer movie

sum_yes = usable_movie_df.loc[(usable_movie_df['summer_movie'] == 'yes')]
sum_no = usable_movie_df.loc[(usable_movie_df['summer_movie'] == 'no')]
```

```
# Use the Mann-Whitney test to determine the p-value

stats.mannwhitneyu(sum_yes['revenue'], sum_no['revenue'])
```

```
MannwhitneyuResult(statistic=41798.0, pvalue=0.5128814728425648)
```

```
# Create a box plot to visually see the comparison

bxplt = sns.boxplot(x = "summer_movie", y = "revenue", data = usable_movie_df).set(title = 'Summer release revenue comparison')
sns.despine()
plt.xlabel('Summer Season')
plt.ylabel('Revenue in Billions')
```

```
Text(0, 0.5, 'Revenue in Billions')
```



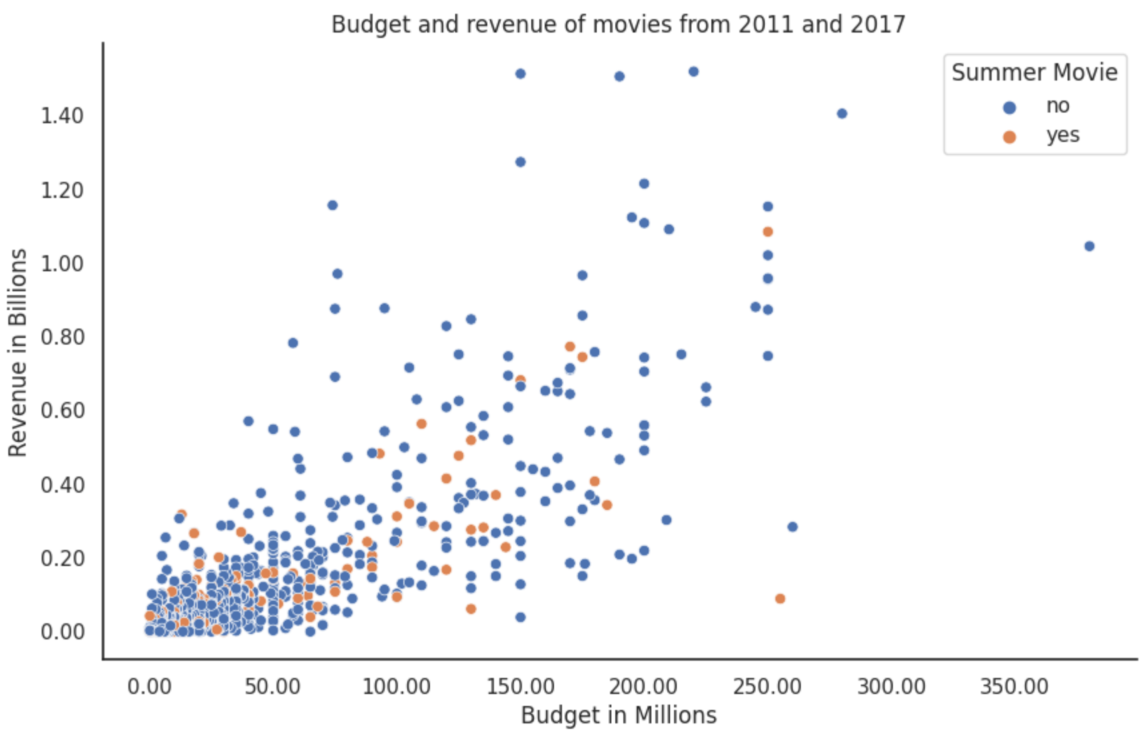
Since the p-value is > 0.05, we can accept the null hypothesis where there is no significant difference between releasing a movie during the summer or not. The box plot also illustrates an overlap of the center 50% of non-summer movies over summer movies.

Added bonus. Combine all three variables, budget, revenue, and summer movie, into one graphic to see a bigger picture.

```
# Combination of all three variables, budget, revenue, and summer movie

#sns.scatterplot(x="budget", y="revenue", hue="summer_movie", data = usable_movie_df)
```

```
plt.figure(figsize = (10,6))
bonus = sns.scatterplot(x = "budget", y = "revenue", hue = "summer_movie", data = usable_movie_df)
sns.despine()
xlabels = ['{:,.2f}'.format(x) for x in bonus.get_xticks()/1000000]
bonus.set_xticklabels(xlabels)
ylabels = ['{:,.2f}'.format(y) for y in bonus.get_yticks()/1000000000]
bonus.set_yticklabels(ylabels)
plt.xlabel('Budget in Millions')
plt.ylabel('Revenue in Billions')
plt.title('Budget and revenue of movies from 2011 and 2017')
plt.legend(title = 'Summer Movie')
```

```
<ipython-input-28-9b625f879612>:9: UserWarning: FixedFormatter should only be used together with FixedLocator
  bonus.set_xticklabels(xlabels)
<ipython-input-28-9b625f879612>:11: UserWarning: FixedFormatter should only be used together with FixedLocator
  bonus.set_yticklabels(ylabels)
<matplotlib.legend.Legend at 0x78e9c02ad6f0>
```




```
stats.pearsonr(sum_yes['budget'], (sum_yes['revenue']))
```

```
PearsonRResult(statistic=0.7799436621189224, pvalue=1.9315826199256704e-28)
```

```
stats.pearsonr(sum_no['budget'], (sum_no['revenue']))
```

```
PearsonRResult(statistic=0.7804929743529091, pvalue=9.973442370642119e-135)
```

In conclusion. For the first analysis it was determined that budget does have a positive correlation with revenue. However, the probability that it is a direct variation is not supported. As the budget increases, there is a wide range of revenue from breaking even to 6 or 8 times the budget. Not a reliable measure of revenue.

The second analysis conlcuded with no significant revenue difference in movies releaed in the summer vice any other season. The box plot shows the summer movies as a slightly condensed version of the non-summer movies with a slightly lower median. Summer releases is not a factor of increased revenue.

Overall, when combining the three variables, budget, summer movies, and non-summer movies have roughly the same correlation coefficient at 0.77, 0.78, and 0.78, repectively. They all statistically have similar patterns.

The recommendation is not to use budget or summer release as a measure for predicting revenue.