Disney Plus Movies/Shows

Capstone Project

David Ariza

University North Florida

Why Disney+ ?


NATIONAL
GEOGRAPHIC


LUCASFILM
Ltd


MARVEL


20th
CENTURY
FOX


PIXAR

Audience

Non-technical audience

More opportunity of discussing

Summarized information

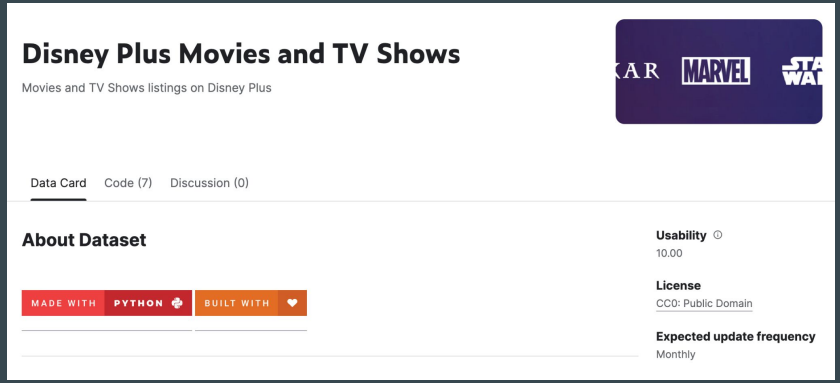
More opportunity to work between different departments and environments.

Data Source

Found in Kaggle, on Movies Datasets.

This dataset contains a couple of shows and series available on Disney+ stream service.

Also, this dataset contains Internet Movie Database (IMDb) ratings that can provide many interesting insights.



Data

workstation : computer and from jupyter notebook.

Columns:

```
Index(['imdb_id', 'title', 'plot', 'type', 'rated', 'year', 'released_at',
       'added_at', 'runtime', 'genre', 'director', 'writer', 'actors',
       'language', 'country', 'awards', 'metascore', 'imdb_rating',
       'imdb_votes'],
      dtype='object')
```

Total of 252 rows and 19 columns after drop the na values out of the dataframe.

Project questions

As a data scientist at Disney+, we have been given a dataset containing information on movies and shows available on the platform. The task is to analyze the data and answer the following questions:

- Which are the top 10 highest rated movies/shows on DisneyPlus?
- Which year had the most releases on DisneyPlus?
- What is the most common genre of movies/shows on DisneyPlus?
- Is there a correlation between the movie/show runtime and its IMDB rating?
- Can we predict the IMDB rating of a movie/show on DisneyPlus based on its genre, director, writer, and actors?

Target

- Title.
- Year.
- Runtime.
- Genre.
- Director.
- Writer.
- Actors.
- Imdb_rating.

-Our target variable is the IMDB rating columns in the disney_plus dataframe.

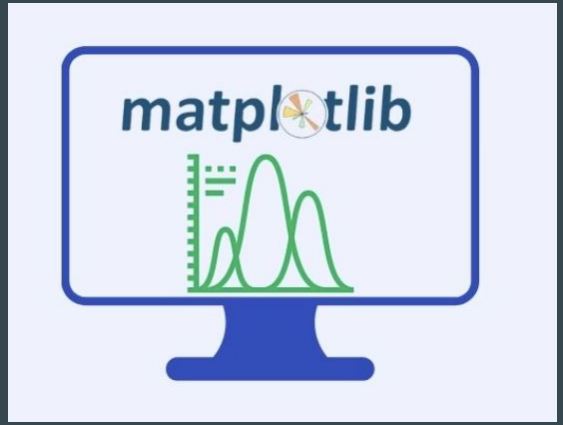
-IMDB rating will be the dependent variable or target variable, while genre, director, writer, and actors will be independent variables or features.

Method

Visualization tools:

Matplotlib.pyplot & seaborn:

- Bar plots.
- Line plots.
- Histograms.
- pie charts.
- heatmaps.
- scatter plots.

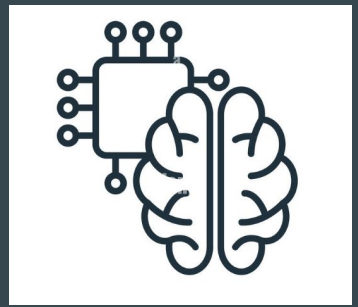


Modeling tools:

Support Vector Machine model SVM for Machine Learning process.

Hyperparameter GridSearch (SVR) Support Vector Regression.

Dummy model (Mean of the training model)



Data wrangling

We found some issues with columns and values.

But first we cleaned the new values dropping the N/A values on the Dataframe.

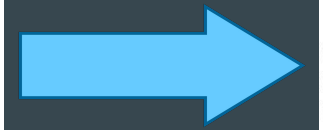
The genre column had different genres per movie, same as writers, actors, directors and year columns, when the values are together almost every movie was a value completely different, to fix it, we organized each value on a new dataframe looping each value separating it from the commas and store it with the title of the movie.

Was the same process for all the non-numeric values on my target, for the year columns I decided to use the range and make a mean of the total of years for each movie.

In order to fix the columns I changed some columns that were on the wrong type such as:

runtime, released_at, added_at & imdb_votes.

```
imdb_id          object
title            object
plot             object
type             object
rated            object
year             object
released_at      object
added_at         object
runtime          object
genre            object
director         object
writer           object
actors           object
language         object
country          object
awards           object
metascore       float64
imdb_rating     float64
imdb_votes       object
dtype: object
```

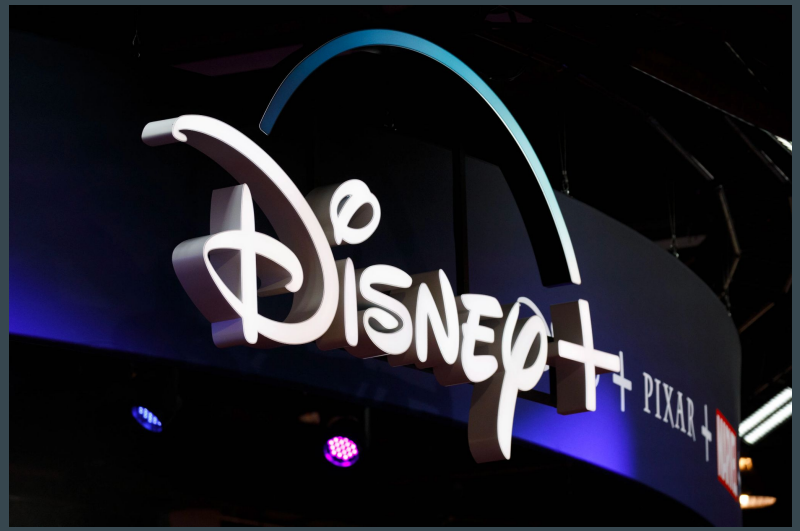


```
imdb_id                  object
title                    object
plot                     object
type                     object
rated                    object
year                     object
released_at      datetime64[ns]
added_at         datetime64[ns]
runtime                   int64
genre                    object
director                 object
writer                   object
actors                   object
language                 object
country                  object
awards                   object
metascore               float64
imdb_rating             float64
imdb_votes                int64
dtype: object
```

Df of genres by title (the same was made by all the non numeric columns on our target).

	title	genre
0	0	Comedy
1	0	Drama
2	0	Romance
3	2	Adventure
4	2	Comedy
...	...	...
1043	991	Adventure
1044	991	Comedy
1045	991	Crime
1046	991	Family
1047	991	Mystery

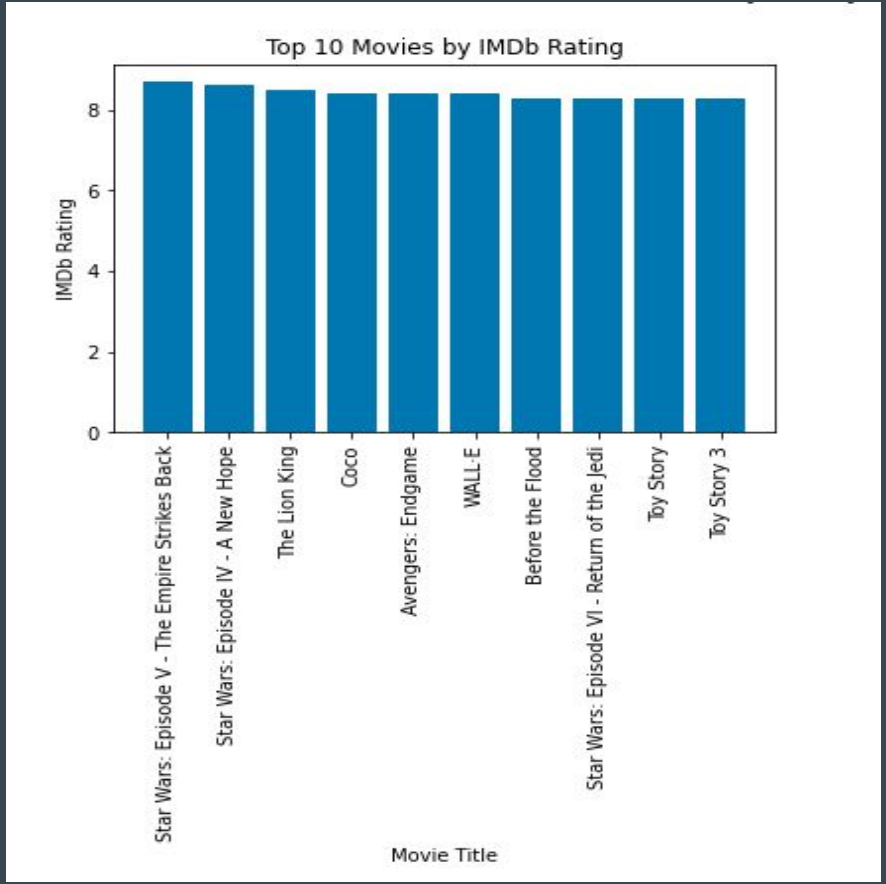
Let’s answer the questions!



Which are the top 10 highest rated movies/shows on DisneyPlus?

To find the top 10 highest rated movies, I decided to use the nlargest attribute using the DataFrame as a Object and select the 10 highest.

	title	imdb_rating
719	Star Wars: Episode V - The Empire Strikes Back	8.7
712	Star Wars: Episode IV - A New Hope	8.6
821	The Lion King	8.5
128	Coco	8.4
486	Avengers: Endgame	8.4
956	WALL·E	8.4
57	Before the Flood	8.3
715	Star Wars: Episode VI - Return of the Jedi	8.3
923	Toy Story	8.3
925	Toy Story 3	8.3



On the plot we could see some movies for Star Wars, other made by pixar such as Toy Story and Wall-E.

Toy Story - by John Lasseter

Stars Wars: The Empire Strikes Back - by Irvin Kershner Lasseter

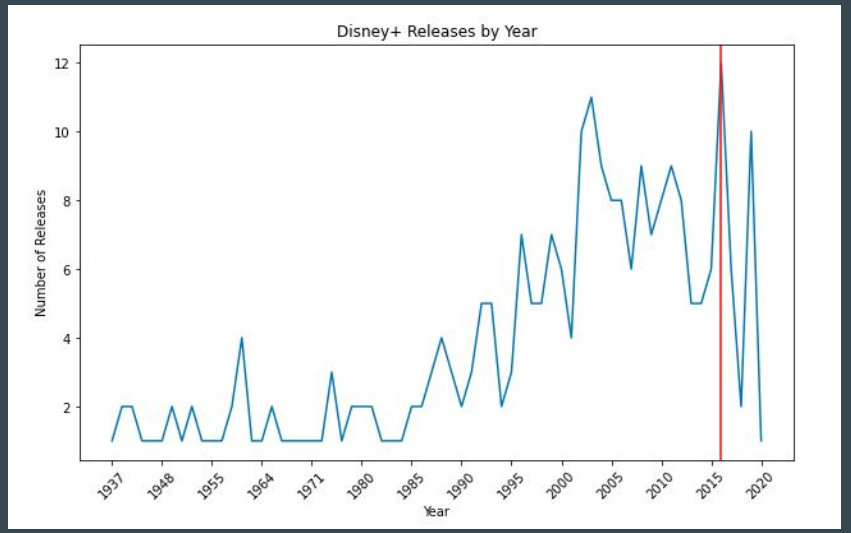
Which year had the most releases on DisneyPlus?

To answer this question we used the year columns and the value_counts attribute.

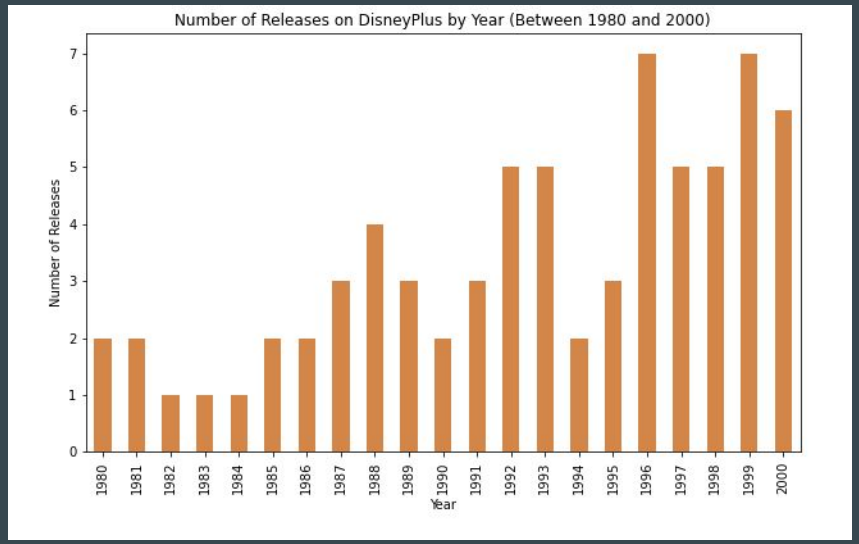
A red line on the X-axis on the year with most releases.

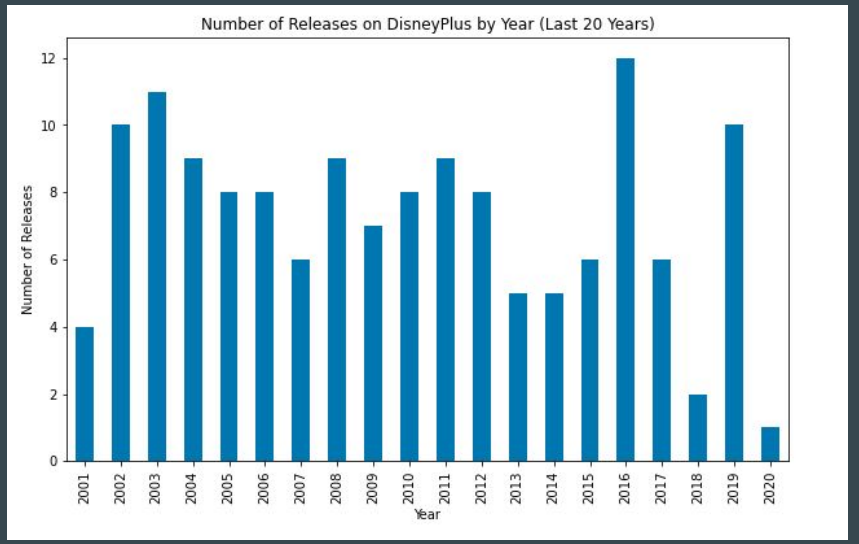
2016 was the year with more releases for disney, with some big projects like:

Avengers: age of Ultron, Zootopia, Finding Dory and more other movies with a total of 12 movies/shows released



Cause of the big amount of years I decided to use bars plots with the last 40 years divided in two (20 years each).





Here we can see that indeed the production of films/shows have increased in the last 30 years for Disney, that can be explained for the big change on technology and special effects.

What is the most common genre of movies/shows on DisneyPlus?

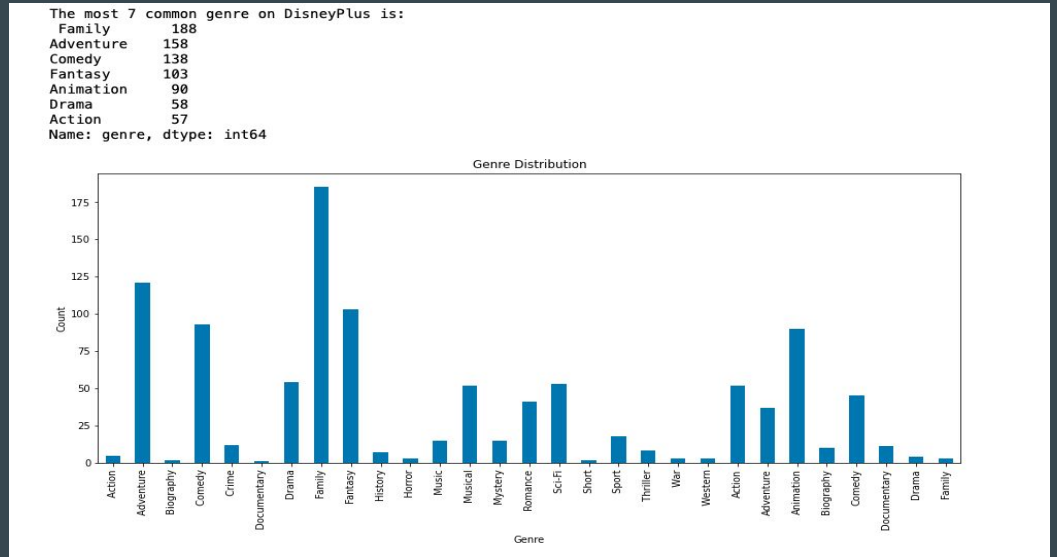
To solve that question I decided to make the Genre column a dummy DataFrame to separate the genres even if a single movie has more than one genre. Once I decided to make the dummy values for the Genre column, I created dummy values with every non numeric column. This permit analyze the values of genre separately and make the visualization better.

	Action	Adventure	Biography	Comedy	Crime	Documentary	Drama	Family	Fantasy	History	...	War	Western	Action	Adventure	Animation	Biograp
0	0	0	0	0	0	0	1	0	0	0	...	0	0	0	0	0	
2	0	0	0	1	1	0	0	1	0	0	...	0	0	0	1	0	
4	0	0	0	1	0	0	0	1	0	0	...	0	0	0	1	0	
6	0	0	0	0	0	0	1	1	1	0	...	0	0	0	1	0	
7	0	1	0	1	0	0	0	1	0	0	...	0	0	0	0	1	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
962	0	1	0	1	1	0	0	1	1	0	...	0	0	0	0	1	
967	0	1	0	0	0	0	1	0	1	0	...	0	0	1	0	0	
970	0	1	0	1	0	0	0	1	0	0	...	0	0	0	0	1	
978	0	1	0	1	0	0	0	1	1	0	...	0	0	0	0	1	
991	0	1	0	1	1	0	0	1	0	0	...	0	0	0	0	1	

#dummy values for the object columns

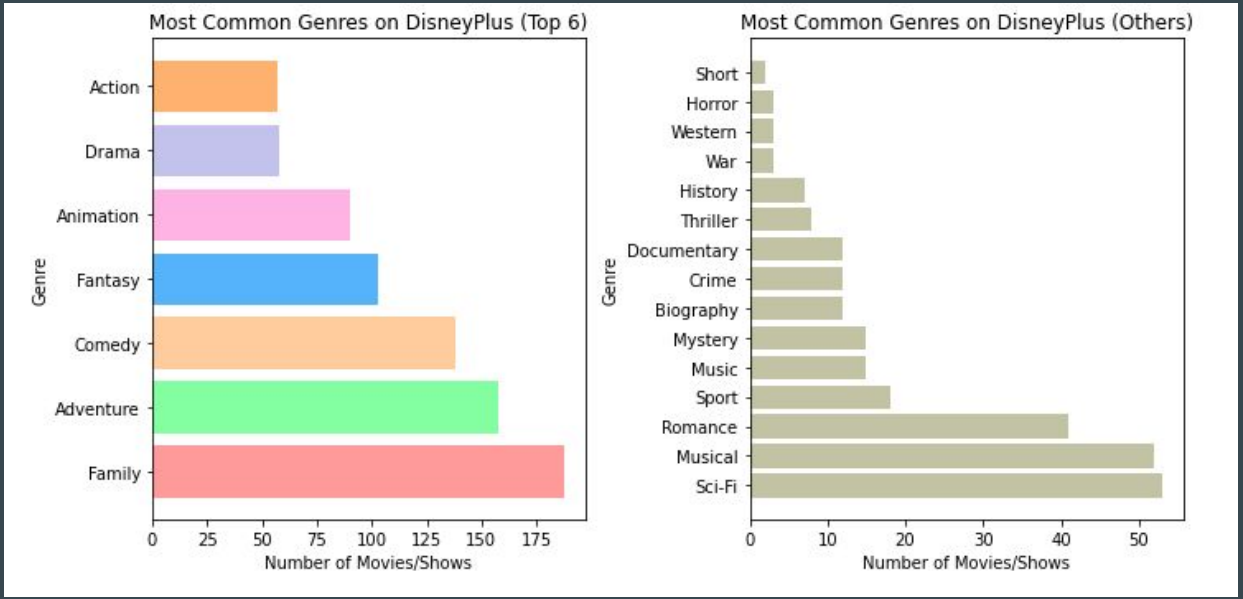
genre_dummies= df['genre'].str.get_dummies(',')

```
The most 7 common genre on DisneyPlus is:
 Family       188
Adventure     158
Comedy        138
Fantasy       103
Animation      90
Drama          58
Action         57
Name: genre, dtype: int64
```



Using the dummy values to visualize all the values and the genres by the count of movies. This gives a contextualized and complete visual of the data and how the genres are distributed.

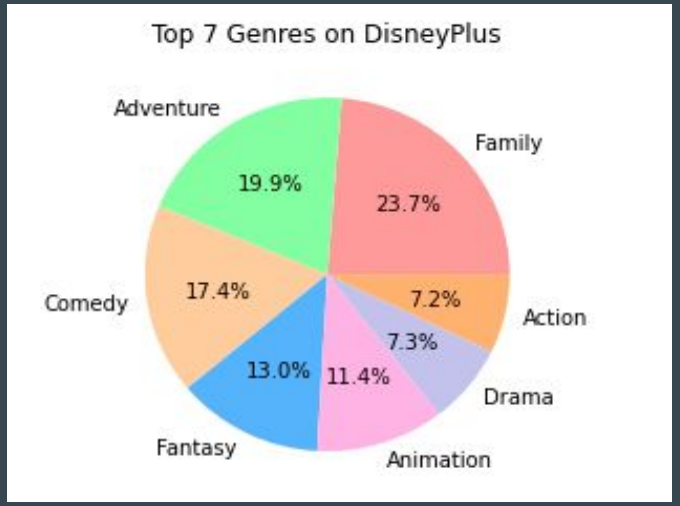
I used a bar chart rotated to visualize two kinds of values, one for the most common genres on movies and the other one with the rest of the data.

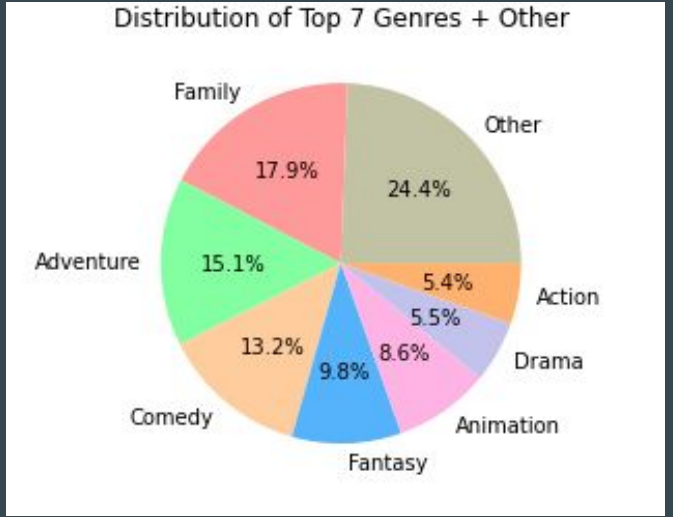


Finally I used pie charts to show how distributed these genres are to each other and I made others to recreate the distribution between the most common genres and the less common genres represented as Other.

From the pie chart we can visualize that the Family Genre is the highest genre used by Disney to make movies and shows, making the majority of the brand and their product Family-Friendly.

The distribution of the most common genres is pretty big between the other genres, being more than the 75% of the genres in total.

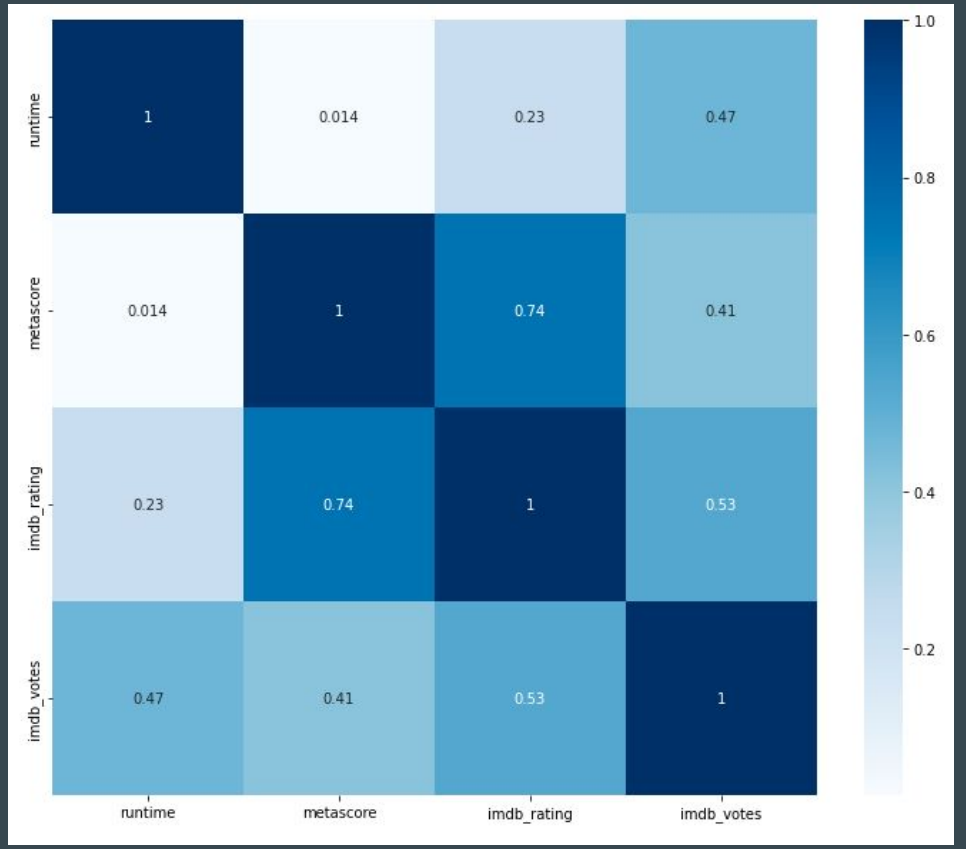




Is there a correlation between the movie/show runtime and its IMDB rating?

We can make correlations between all columns using a Heatmap.

That allows us the ability not just to find the correlation between the movie/show and it's IMDB rating, also it allows me to find any correlation between values. being 1.0 a high correlation and 0.0 a lower expectation.



We can conclude that the **runtime** has **no correlation** between the **Imdb rating**.

- We could find some correlation according to the heat map between the **'metascore'** and the '**Imdb rating**'.

  This suggests that there is no linear relationship between Imdb rating and the runtime, but it does not necessarily mean that there is no relationship at all. Further analysis and investigation may be necessary to fully understand the nature of the relationship between the variables.

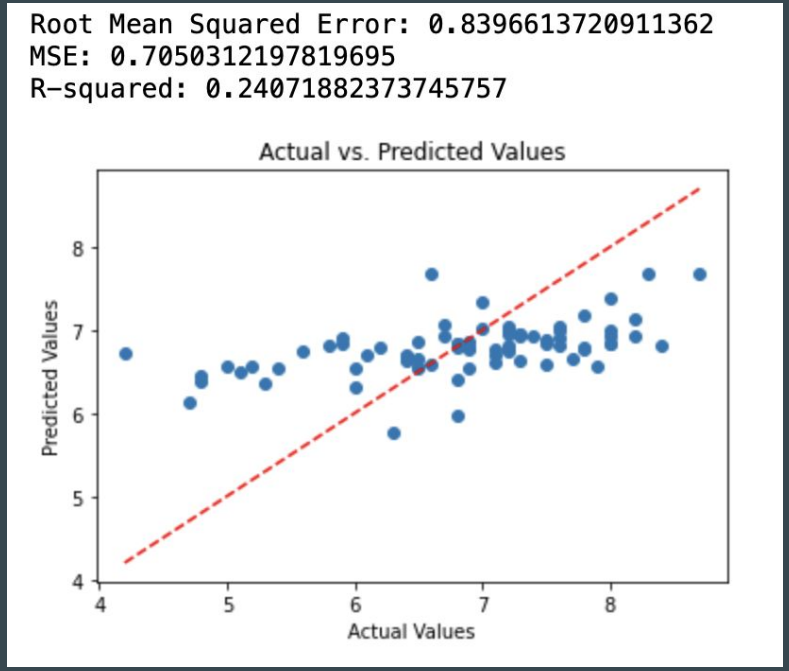
```
correlation = df['runtime'].corr(df['imdb_rating'])
print( 'The correlation between runtime and runtime is: ', correlation)
```

```
The correlation between runtime and runtime is:  0.23492840711156165
```

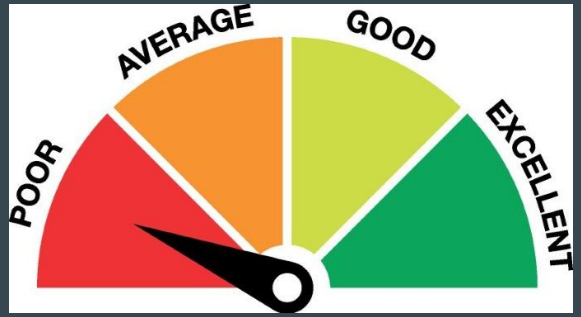
Can we predict the IMDB rating of a movie/show on DisneyPlus based on its genre, director, writer, and actors?

And finally the last question. I wanted to predict the IMDB rating using the SVM (Support Vector Machine) model for this Machine Learning project.

```
Root Mean Squared Error: 0.8396613720911362
MSE: 0.7050312197819695
R-squared: 0.24071882373745757
```



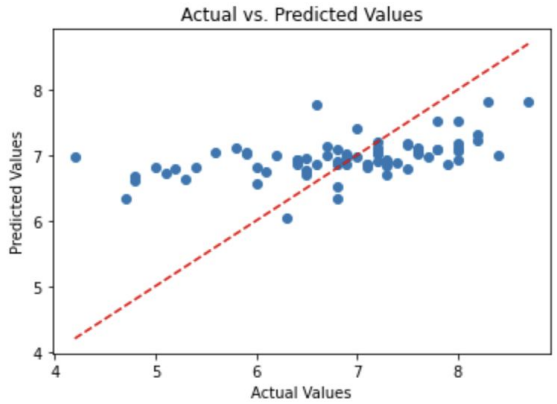
I found that the performance for the model was not good and I decided to use a hyperparameter model to find the best parameters to improve my model.



Hyperparameter GridSearch Model SVR (Support Vector Regression)

I used the Hyperparameter GridSearch and I found the model SVR (Support Vector Regression), that gives the opportunity to use a Hyperplane in a High-Dimensional Space. However, the performance was not as expected and the performance of the model was inaccurate

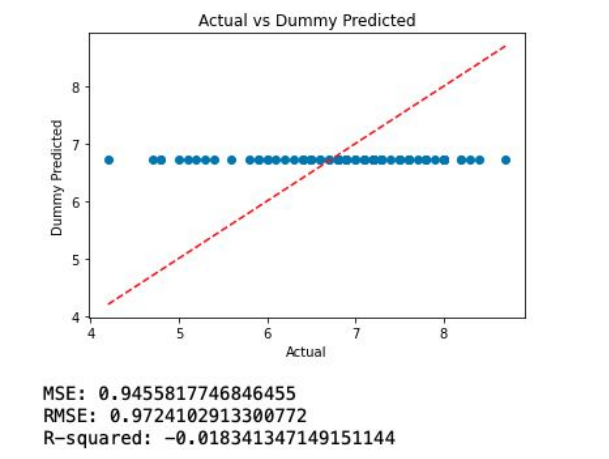
Hyperparameter GridSearch (SVR model used)




```
Root Mean Squared Error: 0.8580537841607551
MSE: 0.7362562965125917
R-squared: 0.20709107460565757
```

Dummy Model

With a high Root Mean Squared Error, a high Mean Squared Error and a low R-squared showed on the model and the prediction, I confirm the low performance that my model can have a low validity, So to confirm that I used a Dummy Model that just uses the mean of the values to make the prediction.



Conclusions

The Dummy model showed a lower performance than the SVR model, however, both models have a low performance and we can conclude that with the data given, we cannot predict the IMDB ratings with the genre, the director, the writers and the actors.

And with the question if there was possible to make prediction of IMDB rating with the genre, actors, directors and writers, we couldn’t find a good modeling for this dataset.