Bharat Intern

Name - Shiva D. Mehenge

In [1]:
```
import numpy as np
import pandas as pd
```

NumPy

NumPy is a powerful Python library for numerical computing. It stands for "Numerical Python" and provides support for large, multi-dimensional arrays and matrices, along with a collection of high-level mathematical functions to operate on these arrays efficiently. NumPy is an essential library for scientific and data-related tasks in Python.

Pandas

Pandas is a popular open-source Python library for data manipulation, analysis, and cleaning. It provides powerful data structures, mainly the DataFrame and Series, that are designed to handle and process structured data efficiently. Pandas is widely used in data science, machine learning, and other data-related fields due to its ease of use and versatility.

In [2]:
```
df = pd.read_csv("E:\CSV Data\movie_success_rate (1).csv")
```

In [3]:
```
df.shape
```

Out[3]:
```
(839, 33)
```

In [4]:
```
df.head()
```

Out[4]:

	Rank	Title	Genre	Description	Director	Actors	Yea
0	1.0	Guardians of the Galaxy	Action,Adventure,Sci-Fi	A group of intergalactic criminals are forced ...	James Gunn	Chris Pratt, Vin Diesel, Bradley Cooper, Zoe S...	2014
1	2.0	Prometheus	Adventure,Mystery,Sci-Fi	Following clues to the origin of mankind, a te...	Ridley Scott	Noomi Rapace, Logan Marshall-Green, Michael Fa...	2012
2	3.0	Split	Horror,Thriller	Three girls are kidnapped by a man with a diag...	M. Night Shyamalan	James McAvoy, Anya Taylor-Joy, Haley Lu Richar...	2016
3	4.0	Sing	Animation,Comedy,Family	In a city of humanoid animals, a hustling thea...	Christophe Lourdelet	Matthew McConaughey,Reese Witherspoon, Seth Ma...	2016
4	5.0	Suicide Squad	Action,Adventure,Fantasy	A secret government agency recruits some of th...	David Ayer	Will Smith, Jared Leto, Margot Robbie, Viola D...	2016

5 rows × 33 columns

In [5]:
```
df.columns
```

Out[5]:
```
Index(['Rank', 'Title', 'Genre', 'Description', 'Director', 'Actors', 'Year',
       'Runtime (Minutes)', 'Rating', 'Votes', 'Revenue (Millions)',
       'Metascore', 'Action', 'Adventure', 'Aniimation', 'Biography', 'Comed
y',
       'Crime', 'Drama', 'Family', 'Fantasy', 'History', 'Horror', 'Music',
       'Musical', 'Mystery', 'Romance', 'Sci-Fi', 'Sport', 'Thriller', 'War',
       'Western', 'Success'],
      dtype='object')
```

In [6]:
```
df['Genre'].value_counts()
```

Out[6]:
```
Action,Adventure,Sci-Fi    50
Comedy,Drama,Romance       30
Drama                      29
Drama,Romance              27
Comedy                     26
                           ..
Adventure,Drama,History     1
Action,Crime,Fantasy        1
Comedy,Mystery              1
Adventure,Comedy,Horror     1
Comedy,Family,Fantasy       1
Name: Genre, Length: 189, dtype: int64
```

In [7]:
```
df['Director'].value_counts()
```

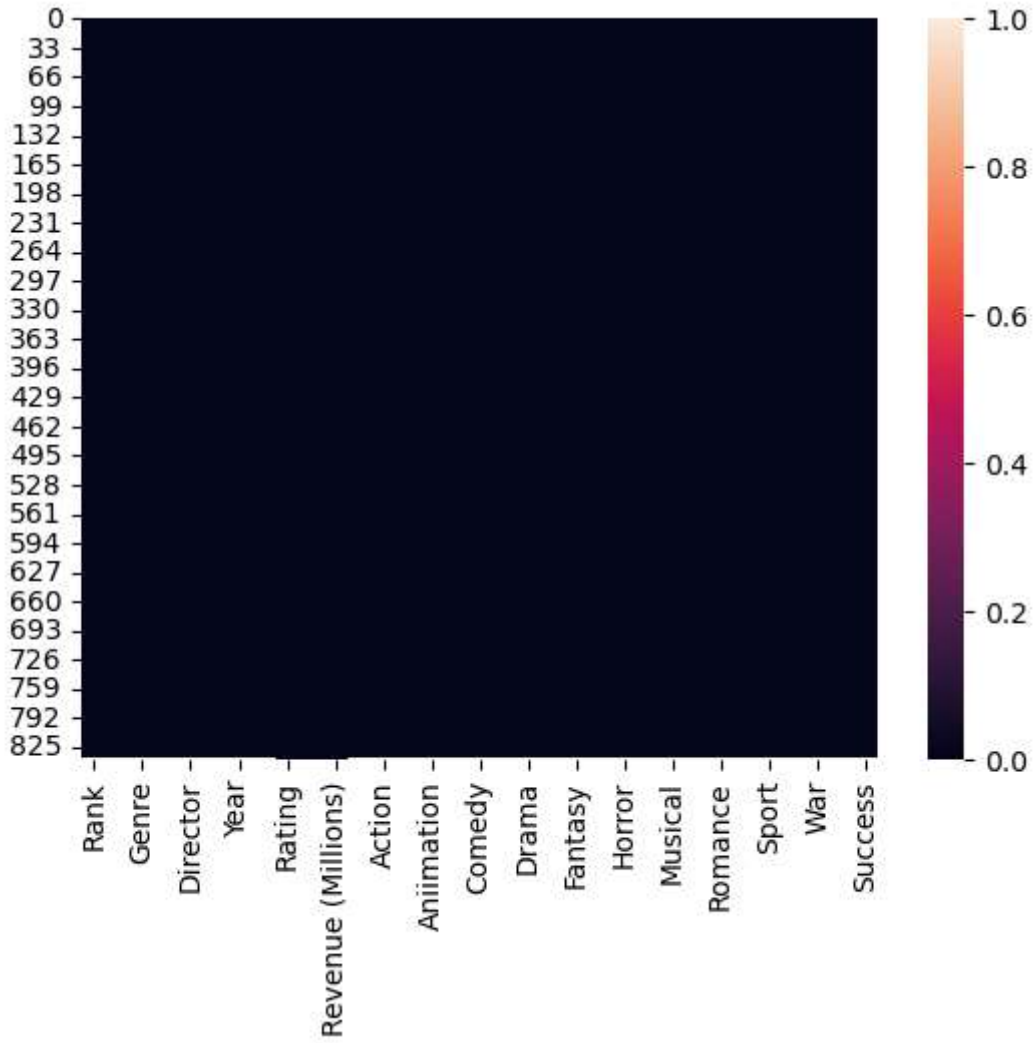
Out[7]:
```
Ridley Scott          8
Paul W.S. Anderson    6
David Yates           6
Michael Bay           6
Antoine Fuqua         5
                     ..
Kyle Balda            1
Chris Renaud          1
Peter Billingsley     1
Lee Toland Krieger    1
Nima Nourizadeh       1
Name: Director, Length: 524, dtype: int64
```

In [8]:
```
df['Actors'].value_counts()
```

Out[8]:
```
Jennifer Lawrence, Josh Hutcherson, Liam Hemsworth, Woody Harrelson    2
Daniel Radcliffe, Emma Watson, Rupert Grint, Michael Gambon            2
Shia LaBeouf, Megan Fox, Josh Duhamel, Tyrese Gibson                   2
Gerard Butler, Aaron Eckhart, Morgan Freeman,Angela Bassett            2
Chris Pratt, Vin Diesel, Bradley Cooper, Zoe Saldana                   1
                                                                      ..
Chris Evans, Jamie Bell, Tilda Swinton, Ed Harris                      1
Chloë Grace Moretz, Matthew Zuk, Gabriela Lopez,Bailey Anne Borders    1
Olivia DeJonge, Ed Oxenbould, Deanna Dunagan, Peter McRobbie           1
Vin Diesel, Paul Walker, Dwayne Johnson, Jordana Brewster              1
Kevin Spacey, Jennifer Garner, Robbie Amell,Cheryl Hines               1
Name: Actors, Length: 834, dtype: int64
```

In [9]:
```
import seaborn as sns
sns.heatmap(df.isnull())
```

Out[9]: <Axes: >



Seaborn

Seaborn is a Python data visualization library based on Matplotlib. It is designed to create attractive and informative statistical graphics with ease. Seaborn builds on top of Matplotlib and provides a high-level interface for drawing attractive and informative statistical graphics. It is particularly useful for visualizing complex datasets and gaining insights quickly.

In [10]:
```
df = df.fillna(df.median())
```

```
C:\Users\sanja\AppData\Local\Temp\ipykernel_27664\3493596106.py:1: FutureWarn
ing: The default value of numeric_only in DataFrame.median is deprecated. In
a future version, it will default to False. In addition, specifying 'numeric_
only=None' is deprecated. Select only valid columns or specify the value of n
umeric_only to silence this warning.
  df = df.fillna(df.median())
```

LOGISTIC REGRESSION

Logistic Regression is a widely used statistical method for binary classification problems. It is a type of regression analysis where the dependent variable (the target) is categorical, typically taking two values such as 0 and 1, or "Yes" and "No." Logistic Regression models the probability of the binary outcome by fitting the data to a logistic or sigmoid function.

In [11]:
```
df.columns
```

Out[11]:
```
Index(['Rank', 'Title', 'Genre', 'Description', 'Director', 'Actors', 'Year',
       'Runtime (Minutes)', 'Rating', 'Votes', 'Revenue (Millions)',
       'Metascore', 'Action', 'Adventure', 'Aniimation', 'Biography', 'Comed
y',
       'Crime', 'Drama', 'Family', 'Fantasy', 'History', 'Horror', 'Music',
       'Musical', 'Mystery', 'Romance', 'Sci-Fi', 'Sport', 'Thriller', 'War',
       'Western', 'Success'],
      dtype='object')
```

In [13]:
```
x = df[['Year',
       'Runtime (Minutes)', 'Rating', 'Votes', 'Revenue (Millions)',
       'Metascore', 'Action', 'Adventure', 'Aniimation', 'Biography', 'Comedy'
       'Crime', 'Drama', 'Family', 'Fantasy', 'History', 'Horror', 'Music',
       'Musical', 'Mystery', 'Romance', 'Sci-Fi', 'Sport', 'Thriller', 'War',
       'Western']]
y = df['Success']
```

In [14]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test= train_test_split(x,y,test_size=0.1,stratify=y)
```

SKLEARN

Scikit-learn, often referred to as sklearn, is a popular and widely-used open-source machine learning library for Python. It provides a comprehensive set of tools for various machine learning tasks, including classification, regression, clustering, dimensionality reduction, and more. Scikit-learn is built on top of other Python libraries, such as NumPy, SciPy, and Matplotlib, making it an integral part of the scientific Python ecosystem.

In [15]:
```
from sklearn.linear_model import LogisticRegression
log = LogisticRegression()
log.fit(x_train,y_train)
```

Out[15]:
```
▾ LogisticRegression

LogisticRegression()
```

- LogisticRegression worked successfully

In [17]:
```
log.score(x_test,y_test)
```

Out[17]:
```
0.8809523809523809
```

-log.score 0.9047619047619048

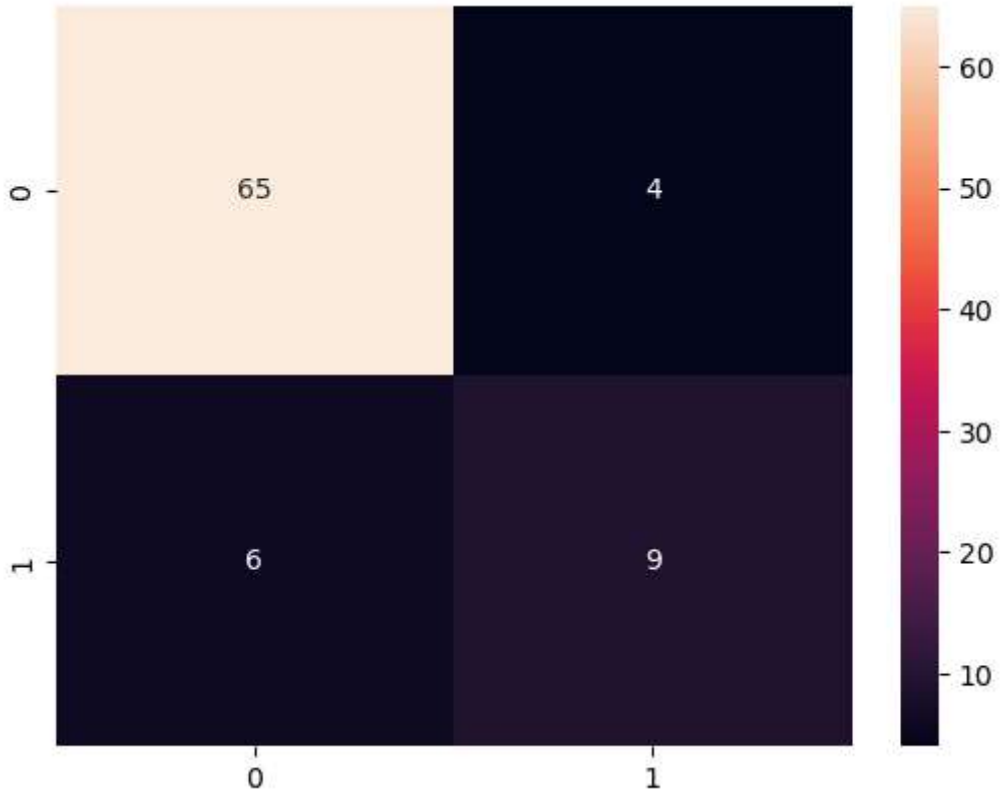
In [18]:
```
from sklearn.metrics import confusion_matrix
clf = confusion_matrix(y_test,log.predict(x_test))
```

Heatmap

In data visualization, a heatmap is a graphical representation of data in the form of a color-coded matrix. It is useful for visualizing 2D data where each cell in the matrix is represented by a color based on its value. Heatmaps are commonly used to display correlations, distributions, and patterns in data.

In [19]:
```
sns.heatmap(clf,annot=True)
```

Out[19]:
```
<Axes: >
```



SOME OPTIMAZTIONS

In [20]:
```
#normalising all columns
x_train_opt = x_train.copy()
x_test_opt = x_test.copy()
```

In [21]:
```
from sklearn.preprocessing import StandardScaler
x_train_opt = StandardScaler().fit_transform(x_train_opt)
x_test_opt = StandardScaler().fit_transform(x_test_opt)
```

In [20]:
```
#fitting again in Logistic Regression
```

In [22]:
```
log.fit(x_train_opt,y_train)
```

Out[22]:
```
LogisticRegression
LogisticRegression()
```

In [20]:
```
log.score(x_test_opt,y_test)
```

Out[20]:
```
0.9166666666666666
```

Model Performance went down so we would not pursuit this more

KNN

In [24]:
```
from sklearn.neighbors import KNeighborsClassifier
kn = KNeighborsClassifier(n_neighbors=40)
kn.fit(x_train,y_train)
```

Out[24]:
```
KNeighborsClassifier(n_neighbors=40)
```
In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

-KNeighborsClassifier work it!

In [25]:
```
kn.score(x_test,y_test)
```

Out[25]:
```
0.8809523809523809
```

kn.score is 0.8809523809523809

DECISION TREE

A decision tree is a popular machine learning algorithm used for both classification and regression tasks. It is a tree-like model that makes a sequence of decisions based on the input features and arrives at a prediction or decision at the leaf nodes of the tree. Decision trees are simple yet powerful models, and they are widely used due to their interpretability and ease of understanding.

In [26]:
```
from sklearn.tree import DecisionTreeClassifier
tree = DecisionTreeClassifier()
tree.fit(x_train,y_train)
tree.score(x_test,y_test)
```

Out[26]:
```
1.0
```

In [27]:
```
tree.score(x_train,y_train)
```

Out[27]:
```
1.0
```

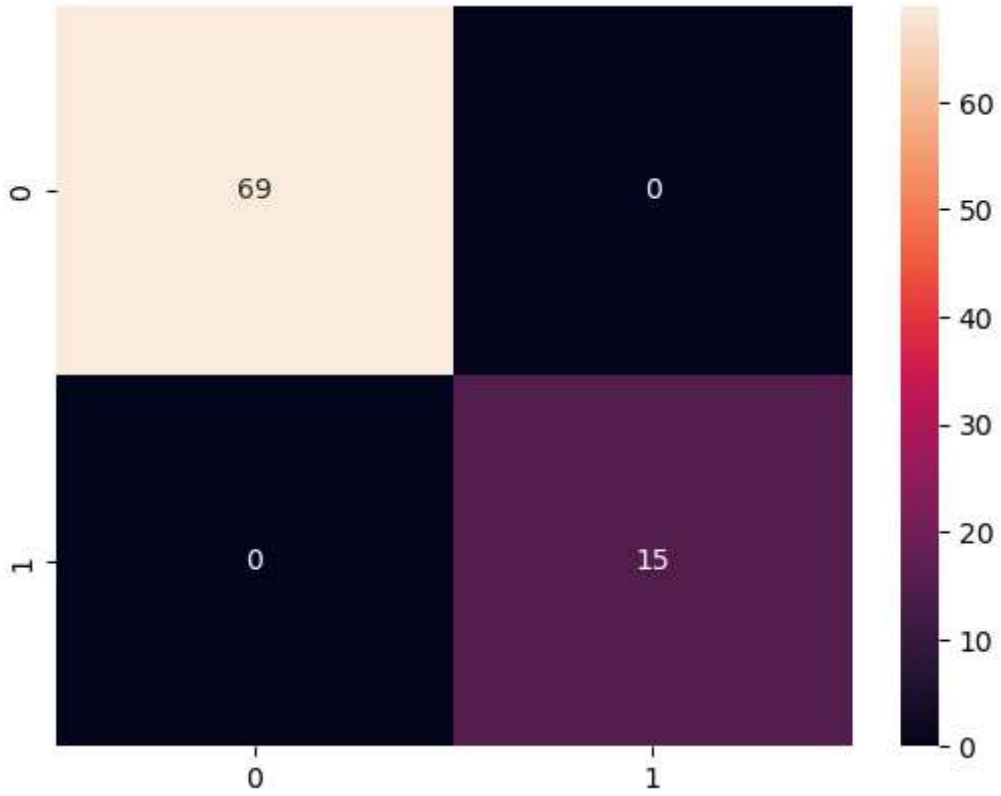
In [28]:
```
from sklearn.metrics import confusion_matrix
clf = confusion_matrix(y_test,tree.predict(x_test))
```

In [29]:
```
clf
```

Out[29]:
```
array([[69,  0],
       [ 0, 15]], dtype=int64)
```

In [30]:
```
sns.heatmap(clf,annot=True)
```

Out[30]: <Axes: >



Thank You!

In [ ]:
```

```