Movie_Rating_Prediction_With_Python

Importing Libraries

In [3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
from sklearn.linear_model import SGDRegressor
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
```

In [4]:
```
df = pd.read_csv("E:\DATA SCIENCE(GRASS)\DATA SCIENCE-Projects\IMDb Movies India.csv\IMDb Movies India.csv", encoding='ISO-8859-1'
df.head()
```

Out[4]:

	Name	Year	Duration	Genre	Rating	Votes	Director	Actor 1	Actor 2	Actor 3
0		NaN	NaN	Drama	NaN	NaN	J.S. Randhawa	Manmauji	Birbal	Rajendra Bhatia
1	#Gadhvi (He thought he was Gandhi)	(2019)	109 min	Drama	7.0	8	Gaurav Bakshi	Rasika Dugal	Vivek Ghamande	Arvind Jangid
2	#Homecoming	(2021)	90 min	Drama, Musical	NaN	NaN	Soumyajit Majumdar	Sayani Gupta	Plabita Borthakur	Roy Angana
3	#Yaaram	(2019)	110 min	Comedy, Romance	4.4	35	Ovais Khan	Prateik	Ishita Raj	Siddhant Kapoor
4	...And Once Again	(2010)	105 min	Drama	NaN	NaN	Amol Palekar	Rajat Kapoor	Rituparna Sengupta	Antara Mali

Data Preprocessing:

In [6]:
```
def dataoverview (df,message):
    print(f'{message}:\n')
    print("Rows:", df.shape[0])
    print("\nNumber of features:", df.shape[1])
    print("\nFeatures:")
    print(df.columns.tolist())
    print("\nMissing values:", df.isnull().sum().values.sum())
    print("\nUnique values:")
    print(df.nunique())
```

In [7]:
```
dataoverview(df, 'Overview of the training dataset')
```

```
Overview of the training dataset:

Rows: 15509

Number of features: 10

Features:
['Name', 'Year', 'Duration', 'Genre', 'Rating', 'Votes', 'Director', 'Actor 1', 'Actor 2', 'Actor 3']

Missing values: 33523

Unique values:
Name        13838
Year          102
Duration      182
Genre         485
Rating         84
Votes        2034
Director     5938
Actor 1      4718
Actor 2      4891
Actor 3      4820
dtype: int64
```

In [10]:
```
df.isna().sum()
```

Out[10]:
```
Name           0
Year         528
Duration    8269
Genre       1877
Rating      7590
Votes       7589
Director     525
Actor 1     1617
Actor 2     2384
Actor 3     3144
dtype: int64
```

In [11]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15509 entries, 0 to 15508
Data columns (total 10 columns):
Name        15509 non-null object
Year        14981 non-null object
Duration    7240 non-null object
Genre       13632 non-null object
Rating      7919 non-null float64
Votes       7920 non-null object
Director    14984 non-null object
Actor 1     13892 non-null object
Actor 2     13125 non-null object
Actor 3     12365 non-null object
dtypes: float64(1), object(9)
memory usage: 666.4+ KB
```

In [12]:
```
#genre, director, and actors values counts
df['Genre'].value_counts()
```

Out[12]:
```
Drama                              2780
Action                             1289
Thriller                            779
Romance                             708
Drama, Romance                      524
Comedy                              495
Action, Crime, Drama                455
Drama, Family                       418
Horror                              322
Action, Drama                       316
Documentary                         283
Comedy, Drama                       264
Comedy, Drama, Romance              224
Fantasy                             170
Action, Comedy, Drama               149
Action, Thriller                    142
Comedy, Romance                     140
Action, Drama, Romance              134
Family                              112
Drama, Musical, Romance             108
Action, Comedy, Crime                95
Drama, Thriller                      94
Mystery                              92
Crime                                91
Comedy, Drama, Family                82
Crime, Drama                         81
Action, Drama, Thriller              81
Animation                            74
Crime, Drama, Thriller               72
Action, Adventure, Drama             69
                                   ...
Adventure, Romance, Thriller          1
Drama, Action, Family                 1
Action, Fantasy, Musical              1
Comedy, Crime, Horror                 1
Comedy, Drama, Sci-Fi                 1
Romance, Action, Crime                1
Action, Crime, War                    1
Action, Crime, Horror                 1
History, Musical, Romance             1
Mystery, Musical, Romance             1
Comedy, Horror, Thriller              1
Comedy, History                       1
Fantasy, Sci-Fi                       1
History, Romance                      1
Drama, Family, Horror                 1
Drama, Crime, Family                  1
Documentary, Sport                    1
Horror, Drama, Mystery                1
Action, Crime, Fantasy                1
Fantasy, Mystery, Romance             1
Crime, Fantasy, Mystery               1
Biography, History, War               1
Crime, Musical, Romance               1
Horror, Musical                       1
Action, Family, Thriller              1
Drama, Family, Adventure              1
Action, Romance, Western              1
Drama, Action, Musical                1
Adventure, Horror                     1
Action, Drama, Western                1
Name: Genre, Length: 485, dtype: int64
```

In [13]:
```
df['Director'].value_counts()
```

Out[13]:
```
Jayant Desai               58
Kanti Shah                 57
Babubhai Mistry            50
Mahesh Bhatt               48
Master Bhagwan             47
Dhirubhai Desai            46
Nanabhai Bhatt             46
David Dhawan               44
Mohammed Hussain           44
B.R. Ishara                44
Hrishikesh Mukherjee       42
Ram Gopal Varma            39
Shakti Samanta             39
Basu Chatterjee            36
Rama Rao Tatineni          36
Shibu Mitra                36
Kedar Kapoor               35
Raj N. Sippy               34
K. Raghavendra Rao         34
Vikram Bhatt               33
Kishan Shah                33
Phani Majumdar             32
Priyadarshan               32
K. Bapaiah                 32
Dwarka Khosla              32
Jagatrai Pesumal Advani    32
Aspi Irani                 32
Mohan Sinha                32
Nanubhai Vakil             32
Abdul Rashid Kardar        31
                           ..
Ganpati Bohra               1
S.N. Azhar                  1
Anita Udeep                 1
Narugopal Mandal            1
Ram Kumar                   1
Sadiq Nizami                1
Satish Shah                 1
Spencer Mathew              1
Sakthi Chidambaram          1
Victor Mukherjee            1
M.D. Baig                   1
Gauravv K. Chawla           1
Rajkannu                    1
Nandamuri Balakrishna       1
Ashwin Rai Shetty           1
Neville Shah                1
Prasenjit Chatterjee        1
Prashant Sehgal             1
Lakshmi Musari              1
Premji                      1
Pradeep R. Sharma           1
Jai Tank                    1
Shashikant Doiphode         1
Ravi Khanna                 1
Prabhuraj                   1
K. Kant                     1
S. Rai                      1
Poonam Balan                1
B.N. Chowhan                1
Jehangir Surti              1
Name: Director, Length: 5938, dtype: int64
```

In [14]:
```
df.head(10)
```

Out[14]:

	Name	Year	Duration	Genre	Rating	Votes	Director	Actor 1	Actor 2	Actor 3
0		NaN	NaN	Drama	NaN	NaN	J.S. Randhawa	Manmauji	Birbal	Rajendra Bhatia
1	#Gadhvi (He thought he was Gandhi)	(2019)	109 min	Drama	7.0	8	Gaurav Bakshi	Rasika Dugal	Vivek Ghamande	Arvind Jangid
2	#Homecoming	(2021)	90 min	Drama, Musical	NaN	NaN	Soumyajit Majumdar	Sayani Gupta	Plabita Borthakur	Roy Angana
3	#Yaaram	(2019)	110 min	Comedy, Romance	4.4	35	Ovais Khan	Prateik	Ishita Raj	Siddhant Kapoor
4	...And Once Again	(2010)	105 min	Drama	NaN	NaN	Amol Palekar	Rajat Kapoor	Rituparna Sengupta	Antara Mali
5	...Aur Pyaar Ho Gaya	(1997)	147 min	Comedy, Drama, Musical	4.7	827	Rahul Rawail	Bobby Deol	Aishwarya Rai Bachchan	Shammi Kapoor
6	...Yahaan	(2005)	142 min	Drama, Romance, War	7.4	1,086	Shoojit Sircar	Jimmy Sheirgill	Minissha Lamba	Yashpal Sharma
7	.in for Motion	(2008)	59 min	Documentary	NaN	NaN	Anirban Datta	NaN	NaN	NaN
8	?: A Question Mark	(2012)	82 min	Horror, Mystery, Thriller	5.6	326	Allyson Patel	Yash Dave	Muntazir Ahmad	Kiran Bhatia
9	@Andheri	(2014)	116 min	Action, Crime, Thriller	4.0	11	Biju Bhaskar Nair	Augustine	Fathima Babu	Byon

In [15]:
```
# As we are going to predict movie ratings based on fdf.dropna(subset=['Name','Year','Duration','Votes','Rating'],inplace=True)eat
df.dropna(subset=['Name','Year','Duration','Votes','Rating'],inplace=True)
df.isna().sum()
```

Out[15]:
```
Name          0
Year          0
Duration      0
Genre        31
Rating        0
Votes         0
Director      1
Actor 1      75
Actor 2     117
Actor 3     163
dtype: int64
```

In [16]:
```
# Remove parentheses from 'Year' column and convert to integer
df['Year'] = df['Year'].str.strip('()').astype(int)
```

In [17]:
```
# Remove commas from 'Votes' column and convert to integer
df['Votes'] = df['Votes'].str.replace(',', '').astype(int)
```

In [18]:
```
# Remove min from 'Duration' column andDurationonvert to integer
df['Duration'] = df['Duration'].str.replace('min', '').astype(int)
```

In [19]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 5851 entries, 1 to 15508
Data columns (total 10 columns):
Name        5851 non-null object
Year        5851 non-null int32
Duration    5851 non-null int32
Genre       5820 non-null object
Rating      5851 non-null float64
Votes       5851 non-null int32
Director    5850 non-null object
Actor 1     5776 non-null object
Actor 2     5734 non-null object
Actor 3     5688 non-null object
dtypes: float64(1), int32(3), object(6)
memory usage: 297.1+ KB
```

In [20]:
```
df.describe()
```

Out[20]:

	Year	Duration	Rating	Votes
count	5851.000000	5851.000000	5851.000000	5851.000000
mean	1996.416852	132.294480	5.931875	2611.273116
std	19.914640	26.555826	1.389942	13433.828528
min	1931.000000	21.000000	1.100000	5.000000
25%	1983.000000	117.000000	5.000000	28.000000
50%	2002.000000	134.000000	6.100000	119.000000
75%	2013.000000	150.000000	7.000000	862.500000
max	2021.000000	321.000000	10.000000	591417.000000

In [21]:
```
# Drop Genre column
df.drop('Genre',axis=1,inplace=True)
```

In [22]:
```
df.head()
```

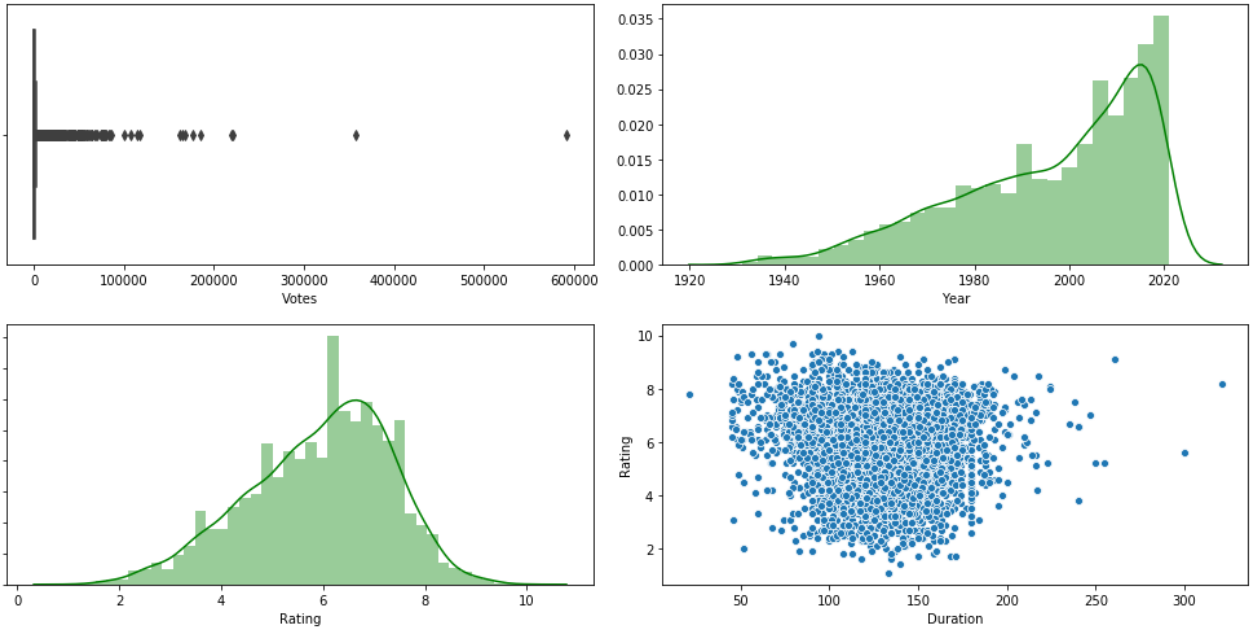
Out[22]:

	Name	Year	Duration	Rating	Votes	Director	Actor 1	Actor 2	Actor 3
1	#Gadhvi (He thought he was Gandhi)	2019	109	7.0	8	Gaurav Bakshi	Rasika Dugal	Vivek Ghamande	Arvind Jangid
3	#Yaaram	2019	110	4.4	35	Ovais Khan	Prateik	Ishita Raj	Siddhant Kapoor
5	...Aur Pyaar Ho Gaya	1997	147	4.7	827	Rahul Rawail	Bobby Deol	Aishwarya Rai Bachchan	Shammi Kapoor
6	...Yahaan	2005	142	7.4	1086	Shoojit Sircar	Jimmy Sheirgill	Minissha Lamba	Yashpal Sharma
8	?: A Question Mark	2012	82	5.6	326	Allyson Patel	Yash Dave	Muntazir Ahmad	Kiran Bhatia

Exploratory Data Analysis

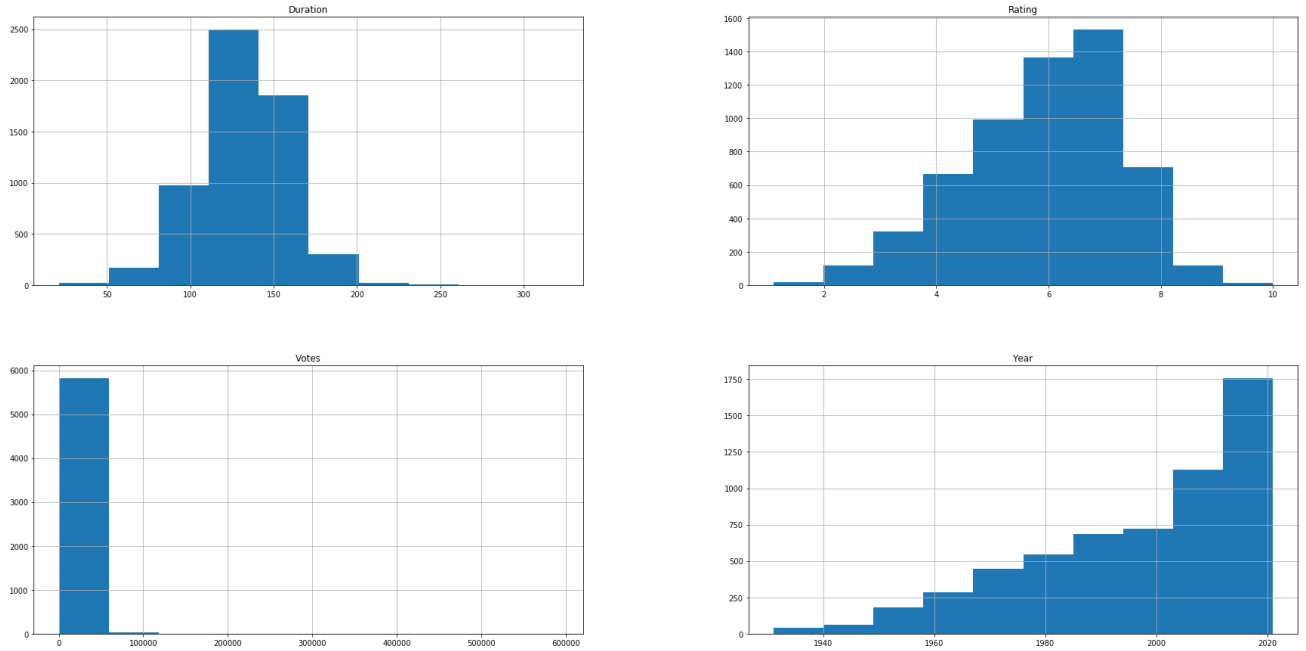
In [23]:
```
plt.figure(figsize=(14,7))
plt.subplot(2,2,1)
sns.boxplot(x='Votes', data=df)

plt.subplot(2,2,2)
sns.distplot(df['Year'], color='g')

plt.subplot(2,2,3)
sns.distplot(df['Rating'], color='g')

plt.subplot(2,2,4)
sns.scatterplot(x=df['Duration'], y=df['Rating'], data=df)

plt.tight_layout()
plt.show()
```

```
D:\Anaconda\Ana\lib\site-packages\scipy\stats\stats.py:1713: FutureWarning: Using a non-tuple sequence for multidimensional in
dexing is deprecated; use `arr[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpreted as an array index, `
arr[np.array(seq)]`, which will result either in an error or a different result.
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```

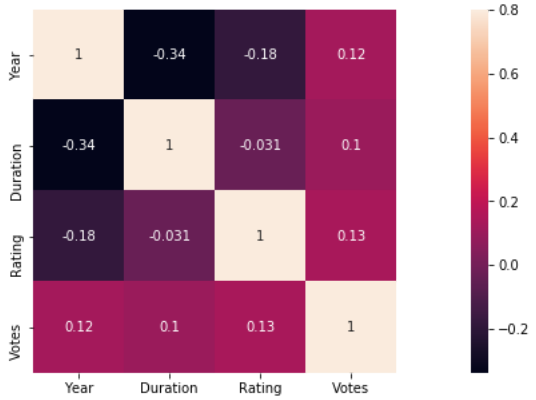


In [24]:
```
df.hist(figsize=(30,15))
```

```
None
```



In [25]:
```
# Heatmap
corrmat = df.corr()
fig = plt.figure(figsize=(20,5))

sns.heatmap(corrmat, vmax=.8, square=True, annot=True)
plt.show()
```



In [26]:
```
df.head()
```

Out[26]:

	Name	Year	Duration	Rating	Votes	Director	Actor 1	Actor 2	Actor 3
1	#Gadhvi (He thought he was Gandhi)	2019	109	7.0	8	Gaurav Bakshi	Rasika Dugal	Vivek Ghamande	Arvind Jangid
3	#Yaaram	2019	110	4.4	35	Ovais Khan	Prateik	Ishita Raj	Siddhant Kapoor
5	...Aur Pyaar Ho Gaya	1997	147	4.7	827	Rahul Rawail	Bobby Deol	Aishwarya Rai Bachchan	Shammi Kapoor
6	...Yahaan	2005	142	7.4	1086	Shoojit Sircar	Jimmy Sheirgill	Minissha Lamba	Yashpal Sharma
8	?: A Question Mark	2012	82	5.6	326	Allyson Patel	Yash Dave	Muntazir Ahmad	Kiran Bhatia

Feature Engineering:

In [27]:
```
df.drop(['Name','Director','Actor 1','Actor 2','Actor 3'], axis=1,inplace=True)
df.head()
```

Out[27]:

	Year	Duration	Rating	Votes
1	2019	109	7.0	8
3	2019	110	4.4	35
5	1997	147	4.7	827
6	2005	142	7.4	1086
8	2012	82	5.6	326

In [28]:
```
X = df[['Year','Duration','Votes']]
y = df['Rating']
```

In [29]:
```
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,random_state=1000)
```

Model Buiding:

In [30]:
```
# Create a pipeline with standard scaling and SGD regression
pipeline = Pipeline([
    ('scaler', StandardScaler()),
    ('sgd', SGDRegressor(max_iter=10000, random_state=1000))
])
```

In [34]:
```
pipeline.fit(X_train, y_train)
```

Out[34]:
```
Pipeline(memory=None,
     steps=[('scaler', StandardScaler(copy=True, with_mean=True, with_std=True)), ('sgd', SGDRegressor(alpha=0.0001, average=Fa
lse, epsilon=0.1, eta0=0.01,
       fit_intercept=True, l1_ratio=0.15, learning_rate='invscaling',
       loss='squared_loss', max_iter=10000, n_iter=None, penalty='l2',
       power_t=0.25, random_state=1000, shuffle=True, tol=None, verbose=0,
       warm_start=False))])
```

In [35]:
```
# Predict ratings on the test set
y_pred_pipeline = pipeline.predict(X_test)
```

Model Evaluation:

In [36]:
```
# Evaluation Metrics for the Pipeline
mae_pipeline = mean_absolute_error(y_test, y_pred_pipeline)
mse_pipeline = mean_squared_error(y_test, y_pred_pipeline)
r2_pipeline = r2_score(y_test, y_pred_pipeline)
```

In [37]:
```
print("Pipeline Mean Absolute Error:", mae_pipeline)
print("Pipeline Mean Squared Error:", mse_pipeline)
print("Pipeline R-squared:", r2_pipeline)
```

```
Pipeline Mean Absolute Error: 1.0398241948647298
Pipeline Mean Squared Error: 1.7897862935710085
Pipeline R-squared: 0.019359484203190114
```

Model Deployment:

In [38]:
```
# Take new user input for prediction
new_input = pd.DataFrame({
    'Year': [2023],          # Replace with the desired year
    'Duration': [120],       # Replace with the desired duration in minutes
    'Votes': [10000],        # Replace with the desired number of votes
})

# Use the trained pipeline to make predictions on the input
predicted_rating = pipeline.predict(new_input)

print("Predicted Rating:", predicted_rating)
```

```
Predicted Rating: [5.71243533]
```

Thank You!!!

GitHub: https://github.com/anujtiwari21?tab=repositories (https://github.com/anujtiwari21?tab=repositories)