In [2]:
```
# import libraries
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
plt.style.use('ggplot')
from matplotlib.pyplot import figure

%matplotlib inline
matplotlib.rcParams['figure.figsize']=(12,8) # Adjusts the configuration of the plots

# read in the data

df = pd.read_csv(r'C:\Users\abdul\OneDrive\Desktop\PortfolioProject\working materials\
```

In [3]:
```
# Overview of the data
df.head()
```

Out[3]:

	name	rating	genre	year	released	score	votes	director	writer	star
0	The Shining	R	Drama	1980	June 13, 1980 (United States)	8.4	927000.0	Stanley Kubrick	Stephen King	Jack Nicholson
1	The Blue Lagoon	R	Adventure	1980	July 2, 1980 (United States)	5.8	65000.0	Randal Kleiser	Henry De Vere Stacpoole	Brooke Shields
2	Star Wars: Episode V - The Empire Strikes Back	PG	Action	1980	June 20, 1980 (United States)	8.7	1200000.0	Irvin Kershner	Leigh Brackett	Mark Hamill
3	Airplane!	PG	Comedy	1980	July 2, 1980 (United States)	7.7	221000.0	Jim Abrahams	Jim Abrahams	Robert Hays
4	Caddyshack	R	Comedy	1980	July 25, 1980 (United States)	7.3	108000.0	Harold Ramis	Brian Doyle-Murray	Chevy Chase

In [4]:
```
# Screening for missing data
for col in df.columns:
        pct_missing= np.mean(df[col].isnull())
        print('{}-{}%'.format(col,pct_missing))
```

```
name-0.0%
rating-0.010041731872717789%
genre-0.0%
year-0.0%
released-0.00026082420448617 63%
score-0.0003912363067292645%
votes-0.0003912363067292645%
director-0.0%
writer-0.0003912363067292645%
star-0.00013041210224308815%
country-0.0003912363067292645%
budget-0.2831246739697444%
gross-0.02464788732394366%
company-0.002217005738132499%
runtime-0.0005216484089723526%
```

In [5]:
```
df = df.dropna()
```

In [6]:
```
for col in df.columns:
        pct_missing= np.mean(df[col].isnull())
        print('{}-{}%'.format(col,pct_missing))
```

```
name-0.0%
rating-0.0%
genre-0.0%
year-0.0%
released-0.0%
score-0.0%
votes-0.0%
director-0.0%
writer-0.0%
star-0.0%
country-0.0%
budget-0.0%
gross-0.0%
company-0.0%
runtime-0.0%
```

In [101…
```
# Ifentifying the data types of our columns
df.dtypes
```

Out[101]:
```
name             int16
rating            int8
genre             int8
year             int64
released         int16
score          float64
votes          float64
director         int16
writer           int16
star             int16
country           int8
budget         float64
gross          float64
company          int16
runtime        float64
yearcorrect        int8
dtype: object
```

In [7]:
```
# changing data type of columns
```

```
df['budget']=df['budget'].astype('int64')
df['gross']=df['gross'].astype('int64')
```

In [8]:
```
#creating Correct Year column
df['yearcorrect'] = df['released'].astype(str).str.split(', ').str[-1].astype(str).str
df.head()
```

Out[8]:

	name	rating	genre	year	released	score	votes	director	writer	star
0	The Shining	R	Drama	1980	June 13, 1980 (United States)	8.4	927000.0	Stanley Kubrick	Stephen King	Jack Nicholson
1	The Blue Lagoon	R	Adventure	1980	July 2, 1980 (United States)	5.8	65000.0	Randal Kleiser	Henry De Vere Stacpoole	Brooke Shields
2	Star Wars: Episode V - The Empire Strikes Back	PG	Action	1980	June 20, 1980 (United States)	8.7	1200000.0	Irvin Kershner	Leigh Brackett	Mark Hamill
3	Airplane!	PG	Comedy	1980	July 2, 1980 (United States)	7.7	221000.0	Jim Abrahams	Jim Abrahams	Robert Hays
4	Caddyshack	R	Comedy	1980	July 25, 1980 (United States)	7.3	108000.0	Harold Ramis	Brian Doyle-Murray	Chevy Chase

In [9]:
```
df=df.sort_values(by=['gross'], inplace=False, ascending=False)
```

In [10]:
```
pd.set_option('display.max_rows',None)
```

In [11]:
```
#eliminating duplicates, highlighting issues of data quality and need to aggregate dat
df['company'].drop_duplicates().sort_values(ascending=False).head()
```

Out[11]:
```
7129        thefyzz
5664    micro_scope
4007       i5 Films
6793     i am OTHER
6420           erbp
Name: company, dtype: object
```

In [84]:
```
df.head()
```

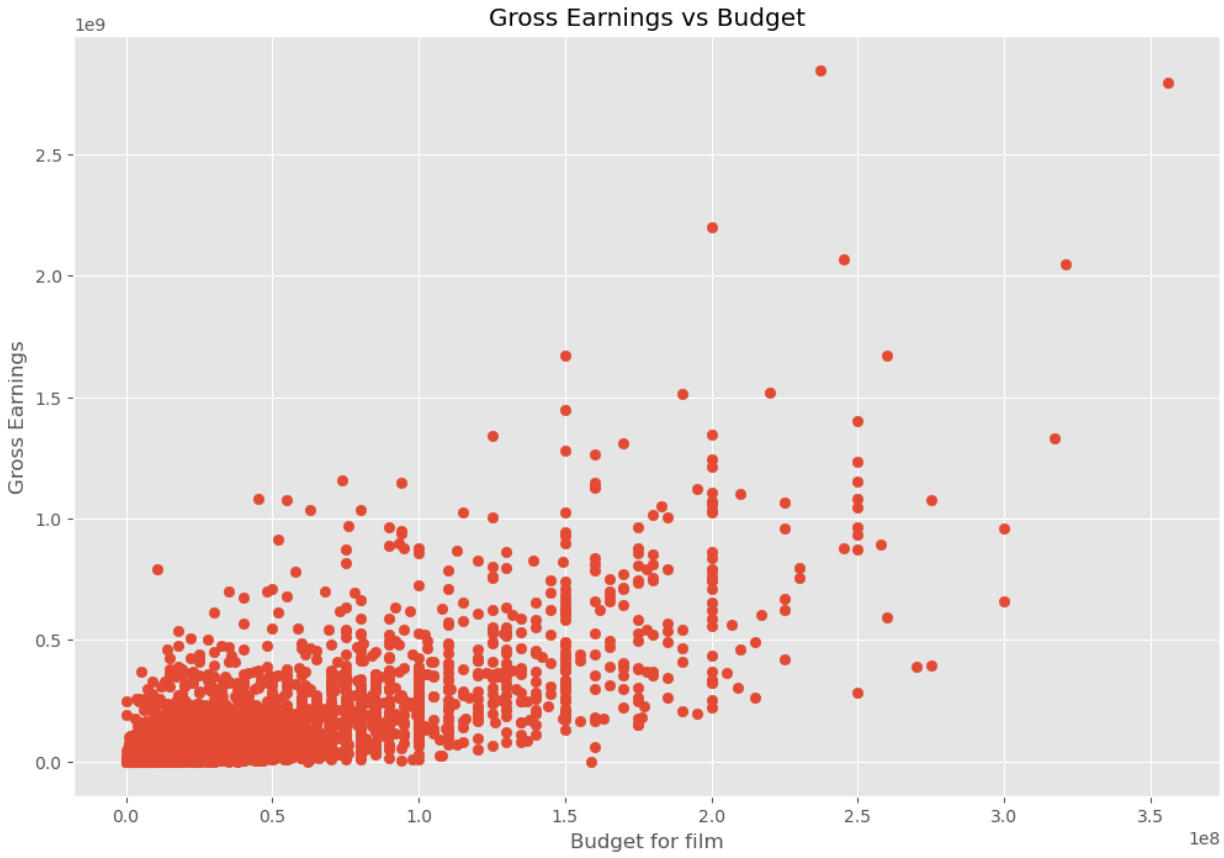
Out[84]:

	name	rating	genre	year	released	score	votes	director	writer	star
0	The Shining	R	Drama	1980	June 13, 1980 (United States)	8.4	927000.0	Stanley Kubrick	Stephen King	Jack Nicholson
1	The Blue Lagoon	R	Adventure	1980	July 2, 1980 (United States)	5.8	65000.0	Randal Kleiser	Henry De Vere Stacpoole	Brooke Shields
2	Star Wars: Episode V - The Empire Strikes Back	PG	Action	1980	June 20, 1980 (United States)	8.7	1200000.0	Irvin Kershner	Leigh Brackett	Mark Hamill
3	Airplane!	PG	Comedy	1980	July 2, 1980 (United States)	7.7	221000.0	Jim Abrahams	Jim Abrahams	Robert Hays
4	Caddyshack	R	Comedy	1980	July 25, 1980 (United States)	7.3	108000.0	Harold Ramis	Brian Doyle-Murray	Chevy Chase

In [12]:
```
# chief hypothesis:budget and gross earning are positively correlated
# Secondary hypothesis: Company and gross earning are also correlated
# Method:Scatter plot of gross earning vs budget and heatmap analysis of correlation v

plt.scatter(x=df['budget'],y=df['gross'])
plt.title('Gross Earnings vs Budget')
plt.xlabel('Budget for film')
plt.ylabel('Gross Earnings')
```

Out[12]:
```
Text(0, 0.5, 'Gross Earnings')
```



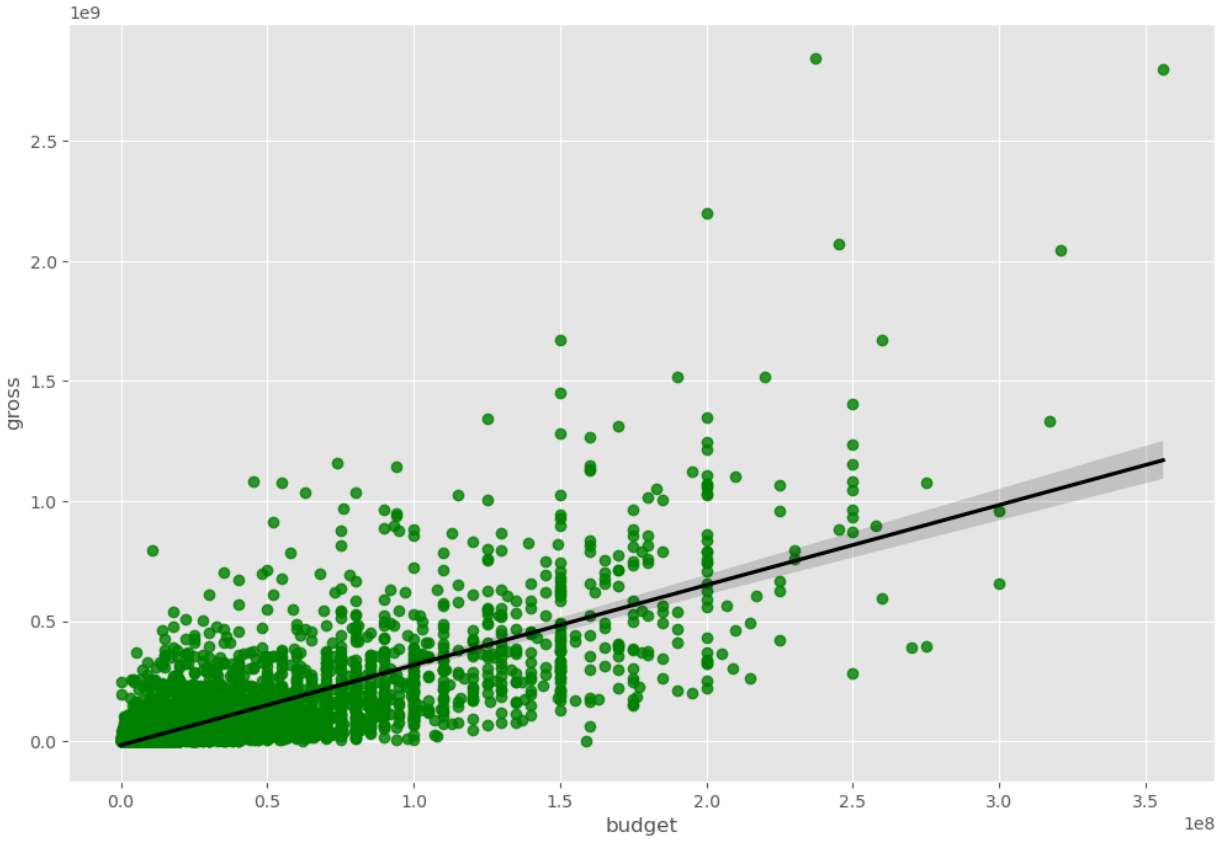
In [13]:
```
df.head()
```

Out[13]:

	name	rating	genre	year	released	score	votes	director	writer	sta
5445	Avatar	PG-13	Action	2009	December 18, 2009 (United States)	7.8	1100000.0	James Cameron	James Cameron	San Worthingtor
7445	Avengers: Endgame	PG-13	Action	2019	April 26, 2019 (United States)	8.4	903000.0	Anthony Russo	Christopher Markus	Rober Downey Jr
3045	Titanic	PG-13	Drama	1997	December 19, 1997 (United States)	7.8	1100000.0	James Cameron	James Cameron	Leonard DiCapri
6663	Star Wars: Episode VII - The Force Awakens	PG-13	Action	2015	December 18, 2015 (United States)	7.8	876000.0	J.J. Abrams	Lawrence Kasdan	Daisy Ridley
7244	Avengers: Infinity War	PG-13	Action	2018	April 27, 2018 (United States)	8.4	897000.0	Anthony Russo	Christopher Markus	Rober Downey Jr

In [14]:
```
#Plot gross earning vs budget using seaborn
sns.regplot(x='budget',y='gross',data=df,scatter_kws={"color":"green"},line_kws={"colo
```

Out[14]:
```
<AxesSubplot:xlabel='budget', ylabel='gross'>
```



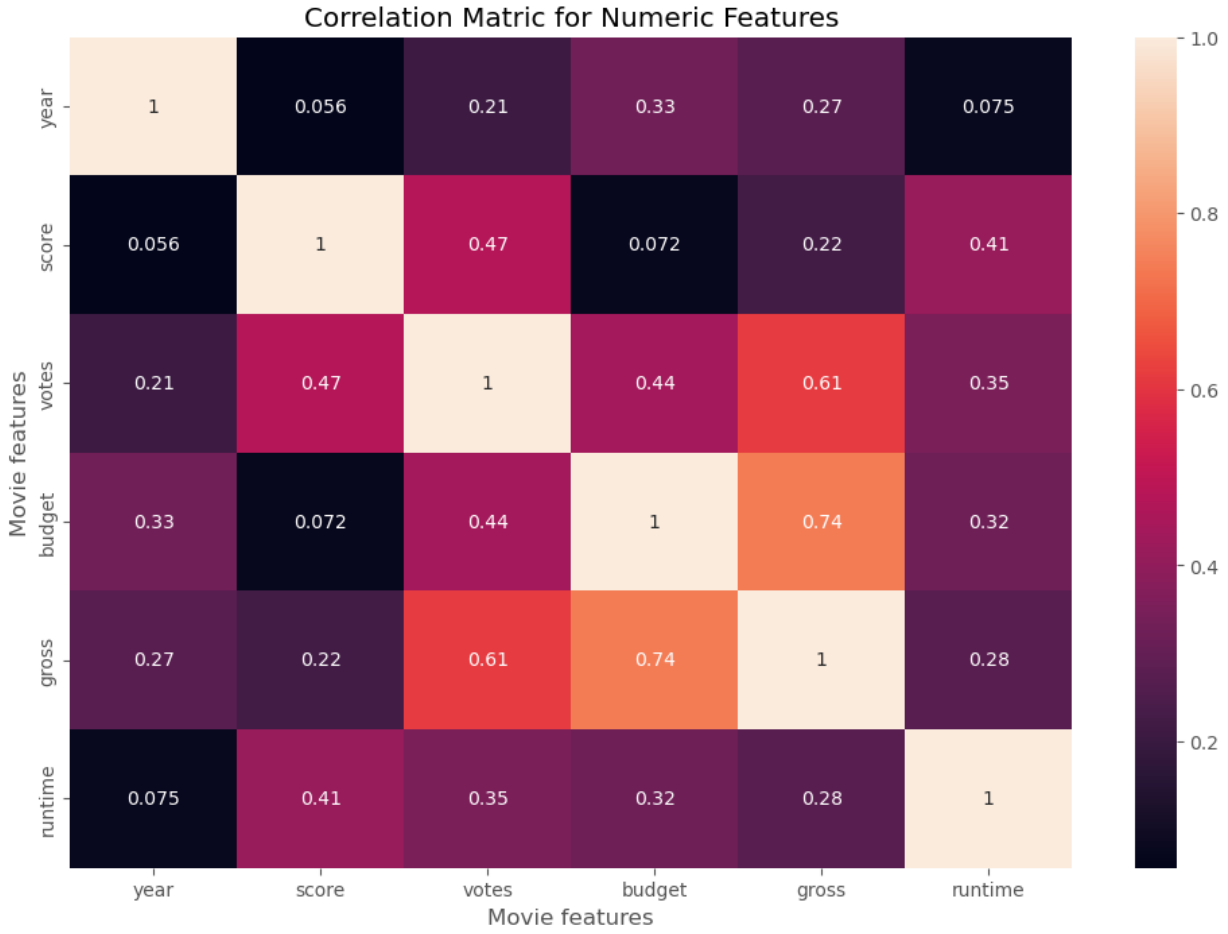
In [112…
```
# analysis of correlation using pearson
df.corr().head()
```

Out[112]:

	name	rating	genre	year	released	score	votes	director	wr
name	1.000000	-0.008069	0.016355	0.011453	-0.011311	0.017097	0.013088	0.009079	0.009
rating	-0.008069	1.000000	0.072423	0.008779	0.016613	-0.001314	0.033225	0.019483	-0.005
genre	0.016355	0.072423	1.000000	-0.081261	0.029822	0.027965	-0.145307	-0.015258	0.006
year	0.011453	0.008779	-0.081261	1.000000	-0.000695	0.097995	0.222945	-0.020795	-0.008
released	-0.011311	0.016613	0.029822	-0.000695	1.000000	0.042788	0.016097	-0.001478	-0.002

In [50]:
```
# confirmation of hypothesis: high correlation between budget and gross
```

In [15]:
```
correlation_matrix = df.corr(method='pearson')
sns.heatmap(correlation_matrix,annot=True)
plt.title('Correlation Matric for Numeric Features')
plt.xlabel('Movie features')
plt.ylabel('Movie features')
plt.show()
```



In [16]:
```
# Analysis of secondary hypothesis: determining relevent correlation between Company c
df.head()
```

Out[16]:

	name	rating	genre	year	released	score	votes	director	writer	sta
5445	Avatar	PG-13	Action	2009	December 18, 2009 (United States)	7.8	1100000.0	James Cameron	James Cameron	San Worthingtor
7445	Avengers: Endgame	PG-13	Action	2019	April 26, 2019 (United States)	8.4	903000.0	Anthony Russo	Christopher Markus	Rober Downey Jr
3045	Titanic	PG-13	Drama	1997	December 19, 1997 (United States)	7.8	1100000.0	James Cameron	James Cameron	Leonard DiCapri
6663	Star Wars: Episode VII - The Force Awakens	PG-13	Action	2015	December 18, 2015 (United States)	7.8	876000.0	J.J. Abrams	Lawrence Kasdan	Daisy Ridle
7244	Avengers: Infinity War	PG-13	Action	2018	April 27, 2018 (United States)	8.4	897000.0	Anthony Russo	Christopher Markus	Rober Downey Jr

In [17]:

```
df_numerized= df
for col_name in df_numerized.columns:
        if(df_numerized[col_name].dtype=='object'):
            df_numerized[col_name]=df_numerized[col_name].astype('category')
            df_numerized[col_name]=df_numerized[col_name].cat.codes

df_numerized.head()
```

Out[17]:

	name	rating	genre	year	released	score	votes	director	writer	star	country	bu
5445	386	5	0	2009	527	7.8	1100000.0	785	1263	1534	47	23700
7445	388	5	0	2019	137	8.4	903000.0	105	513	1470	47	35600
3045	4909	5	6	1997	534	7.8	1100000.0	785	1263	1073	47	20000
6663	3643	5	0	2015	529	7.8	876000.0	768	1806	356	47	24500
7244	389	5	0	2018	145	8.4	897000.0	105	513	1470	47	32100

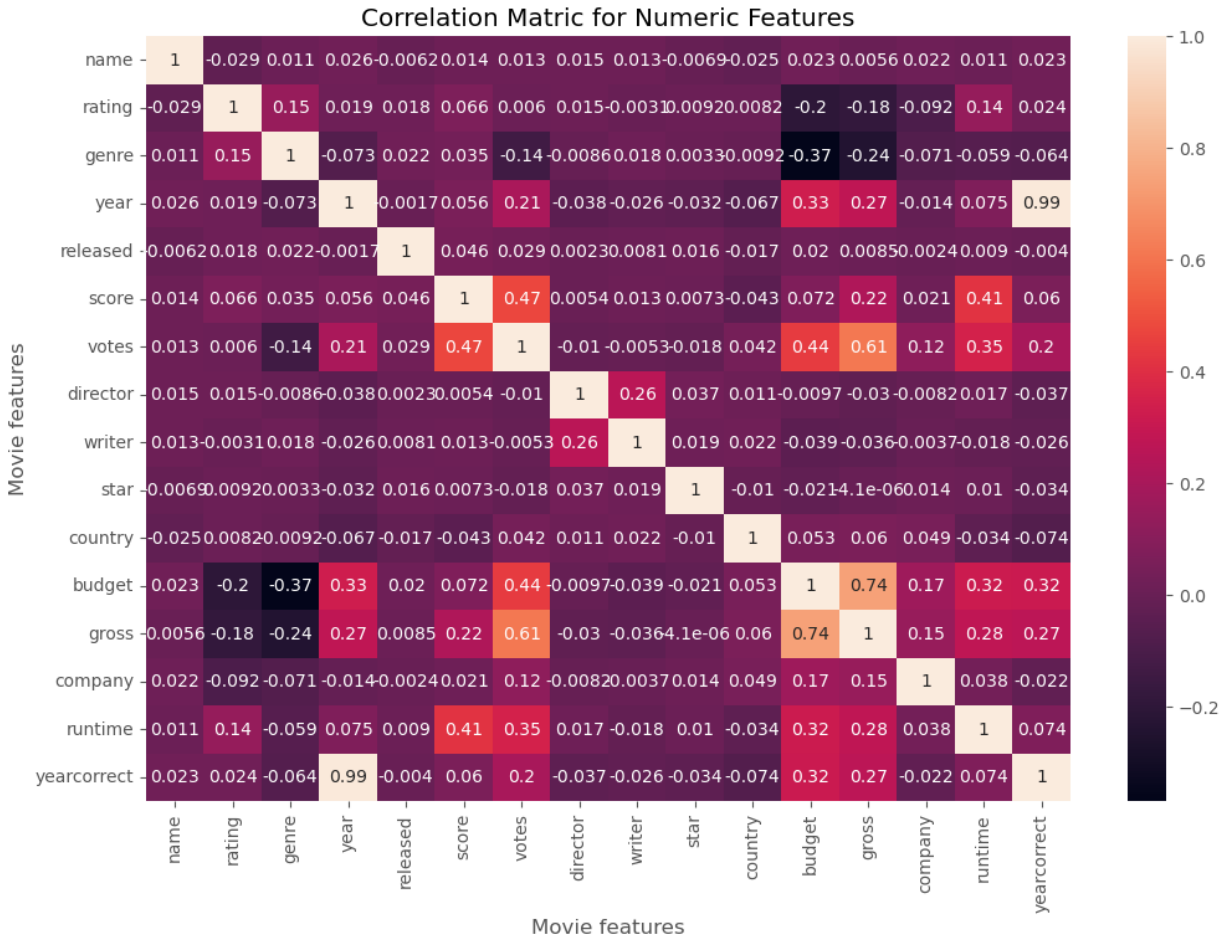
In [97]:

```
df.head()
```

Out[97]:

	name	rating	genre	year	released	score	votes	director	writer	star	country	budge
0	6587	6	6	1980	1705	8.4	927000.0	2589	4014	1047	54	19000000.
1	5573	6	1	1980	1492	5.8	65000.0	2269	1632	327	55	4500000.
2	5142	4	0	1980	1771	8.7	1200000.0	1111	2567	1745	55	18000000.
3	286	4	4	1980	1492	7.7	221000.0	1301	2000	2246	55	3500000.
4	1027	6	4	1980	1543	7.3	108000.0	1054	521	410	55	6000000.

In [18]:
```
correlation_matrix = df_numerized.corr(method='pearson')
sns.heatmap(correlation_matrix,annot=True)
plt.title('Correlation Matric for Numeric Features')
plt.xlabel('Movie features')
plt.ylabel('Movie features')
plt.show()
```



In [19]:
```
correlation_mat = df_numerized.corr()
corr_pairs = correlation_mat.unstack()
corr_pairs.head()
```

Out[19]:
```
name  name        1.000000
      rating     -0.029234
      genre       0.010996
      year        0.025542
      released   -0.006152
dtype: float64
```

In [21]:
```
sorted_pairs = corr_pairs.sort_values()
sorted_pairs.head()
```

Out[21]:
```
genre   budget   -0.368523
budget  genre    -0.368523
gross   genre    -0.244101
genre   gross    -0.244101
rating  budget   -0.203946
dtype: float64
```

In [22]:
```
high_corr = sorted_pairs[(sorted_pairs)>0.5]
high_corr.head()
```

Out[22]:
```
votes        gross     0.614751
gross        votes     0.614751
budget       gross     0.740247
gross        budget    0.740247
yearcorrect  year      0.990417
dtype: float64
```

In [ ]:
```
# Conclusion: votes and budget have the highest correlation to
#gross earnings while company did not significantly correlate.
```