Untitled8

November 9, 2018

In [ ]:
```
Project: Investigate a Dataset (TMDb movie Database)
Table of Contents

    Introduction
    Data Wrangling
    Exploratory Data Analysis
    Conclusions

Introduction

    In this section of the report, I'll provide a brief introduction to the dataset I'
    Introduction to dataset:

    I will be using TMDB movie dataset, This data set contains information about 10,000
```

In [1]:
```
import numpy as np
import pandas as pd
import os
import csv
from datetime import datetime
import matplotlib.pyplot as plt
%matplotlib inline
```

In [ ]:
```
    Let's load the data and check some rows from the dataset to identify the questions
```

In [2]:
```
os.chdir('/home/raghusharma/Downloads')
tmdb_data = pd.read_csv('tmdb-movies.csv')
tmdb_data.head()
```

Out[2]:

	id	imdb_id	popularity	budget	revenue	original_title
0	135397	tt0369610	32.985763	150000000	1513528810	Jurassic World
1	76341	tt1392190	28.419936	150000000	378436354	
2	262500	tt2908446	13.112507	110000000	295238201	
3	140607	tt2488496	11.173104	200000000	2068178225	
4	168259	tt2820852	9.335014	190000000	1506249360	

```
1           Mad Max: Fury Road
2                     Insurgent
3  Star Wars: The Force Awakens
4                     Furious 7

                                                cast  \
0  Chris Pratt|Bryce Dallas Howard|Irrfan Khan|Vi...
1  Tom Hardy|Charlize Theron|Hugh Keays-Byrne|Nic...
2  Shailene Woodley|Theo James|Kate Winslet|Ansel...
3  Harrison Ford|Mark Hamill|Carrie Fisher|Adam D...
4  Vin Diesel|Paul Walker|Jason Statham|Michelle ...

                                          homepage          director  \
0                    http://www.jurassicworld.com/   Colin Trevorrow
1                      http://www.madmaxmovie.com/     George Miller
2    http://www.thedivergentseries.movie/#insurgent  Robert Schwentke
3  http://www.starwars.com/films/star-wars-episod...       J.J. Abrams
4                          http://www.furious7.com/         James Wan

                       tagline      ...       \
0             The park is open.      ...
1            What a Lovely Day.      ...
2    One Choice Can Destroy You      ...
3  Every generation has a story.     ...
4           Vengeance Hits Home      ...

                                          overview runtime  \
0  Twenty-two years after the events of Jurassic ...     124
1  An apocalyptic story set in the furthest reach...     120
2  Beatrice Prior must confront her inner demons ...     119
3  Thirty years after defeating the Galactic Empi...     136
4  Deckard Shaw seeks revenge against Dominic Tor...     137

                                     genres  \
0  Action|Adventure|Science Fiction|Thriller
1  Action|Adventure|Science Fiction|Thriller
2         Adventure|Science Fiction|Thriller
3   Action|Adventure|Science Fiction|Fantasy
4                     Action|Crime|Thriller

                              production_companies release_date vote_count  \
0  Universal Studios|Amblin Entertainment|Legenda...       6/9/15       5562
1  Village Roadshow Pictures|Kennedy Miller Produ...      5/13/15       6185
2  Summit Entertainment|Mandeville Films|Red Wago...      3/18/15       2480
3          Lucasfilm|Truenorth Productions|Bad Robot     12/15/15       5292
4  Universal Pictures|Original Film|Media Rights ...       4/1/15       2947

   vote_average  release_year    budget_adj   revenue_adj
```

```
0          6.5          2015  1.379999e+08  1.392446e+09
1          7.1          2015  1.379999e+08  3.481613e+08
2          6.3          2015  1.012000e+08  2.716190e+08
3          7.5          2015  1.839999e+08  1.902723e+09
4          7.3          2015  1.747999e+08  1.385749e+09

[5 rows x 21 columns]
```

In [ ]:

```
Questions that can be answered by looking at the datasets are:
Some general questions that can be answered are:

    Which movie had the highest and lowest profit?
    Which movie had the greatest and least runtime?
    What is the average runtime of all movies?
    Which movie had the highest and lowest budget?
    Which movie had the highest and lowest revenue?
```

Data Wrangling

```
In this section of the report, I will check for cleanliness, and then trim and clea
Observations from above dataset are:

    The dataset has not provided the currency for columns we will be dealing with
    Even the vote count is not same for all the movies and hence this affects the
```

General Properties

```
Let's check the dataset and see what cleaning does it requires.
```

In [3]:
```
tmdb_data.head()
```

Out[3]:
```
       id    imdb_id  popularity     budget     revenue  \
0  135397  tt0369610   32.985763  150000000  1513528810
1   76341  tt1392190   28.419936  150000000   378436354
2  262500  tt2908446   13.112507  110000000   295238201
3  140607  tt2488496   11.173104  200000000  2068178225
4  168259  tt2820852    9.335014  190000000  1506249360

                original_title  \
0                Jurassic World
1           Mad Max: Fury Road
2                     Insurgent
3  Star Wars: The Force Awakens
4                     Furious 7

                                                cast  \
0  Chris Pratt|Bryce Dallas Howard|Irrfan Khan|Vi...
```

```
1  Tom Hardy|Charlize Theron|Hugh Keays-Byrne|Nic...
2  Shailene Woodley|Theo James|Kate Winslet|Ansel...
3  Harrison Ford|Mark Hamill|Carrie Fisher|Adam D...
4  Vin Diesel|Paul Walker|Jason Statham|Michelle ...

                                          homepage          director  \
0                    http://www.jurassicworld.com/   Colin Trevorrow
1                      http://www.madmaxmovie.com/     George Miller
2    http://www.thedivergentseries.movie/#insurgent  Robert Schwentke
3  http://www.starwars.com/films/star-wars-episod...       J.J. Abrams
4                          http://www.furious7.com/         James Wan

                        tagline      ...       \
0              The park is open.      ...
1             What a Lovely Day.      ...
2    One Choice Can Destroy You      ...
3  Every generation has a story.      ...
4            Vengeance Hits Home      ...

                                          overview runtime  \
0  Twenty-two years after the events of Jurassic ...     124
1  An apocalyptic story set in the furthest reach...     120
2  Beatrice Prior must confront her inner demons ...     119
3  Thirty years after defeating the Galactic Empi...     136
4  Deckard Shaw seeks revenge against Dominic Tor...     137

                                     genres  \
0  Action|Adventure|Science Fiction|Thriller
1  Action|Adventure|Science Fiction|Thriller
2         Adventure|Science Fiction|Thriller
3   Action|Adventure|Science Fiction|Fantasy
4                     Action|Crime|Thriller

                               production_companies release_date vote_count  \
0  Universal Studios|Amblin Entertainment|Legenda...       6/9/15       5562
1  Village Roadshow Pictures|Kennedy Miller Produ...      5/13/15       6185
2  Summit Entertainment|Mandeville Films|Red Wago...      3/18/15       2480
3          Lucasfilm|Truenorth Productions|Bad Robot     12/15/15       5292
4  Universal Pictures|Original Film|Media Rights ...       4/1/15       2947

   vote_average  release_year    budget_adj   revenue_adj
0           6.5          2015  1.379999e+08  1.392446e+09
1           7.1          2015  1.379999e+08  3.481613e+08
2           6.3          2015  1.012000e+08  2.716190e+08
3           7.5          2015  1.839999e+08  1.902723e+09
4           7.3          2015  1.747999e+08  1.385749e+09

[5 rows x 21 columns]
```

```
In [4]: # lets us check some statistics of the data
        tmdb_data.describe()
```

Out[4]:

	id	popularity	budget	revenue	runtime
count	10866.000000	10866.000000	1.086600e+04	1.086600e+04	10866.000000
mean	66064.177434	0.646441	1.462570e+07	3.982332e+07	102.070863
std	92130.136561	1.000185	3.091321e+07	1.170035e+08	31.381405
min	5.000000	0.000065	0.000000e+00	0.000000e+00	0.000000
25%	10596.250000	0.207583	0.000000e+00	0.000000e+00	90.000000
50%	20669.000000	0.383856	0.000000e+00	0.000000e+00	99.000000
75%	75610.000000	0.713817	1.500000e+07	2.400000e+07	111.000000
max	417859.000000	32.985763	4.250000e+08	2.781506e+09	900.000000

	vote_count	vote_average	release_year	budget_adj	revenue_adj
count	10866.000000	10866.000000	10866.000000	1.086600e+04	1.086600e+04
mean	217.389748	5.974922	2001.322658	1.755104e+07	5.136436e+07
std	575.619058	0.935142	12.812941	3.430616e+07	1.446325e+08
min	10.000000	1.500000	1960.000000	0.000000e+00	0.000000e+00
25%	17.000000	5.400000	1995.000000	0.000000e+00	0.000000e+00
50%	38.000000	6.000000	2006.000000	0.000000e+00	0.000000e+00
75%	145.750000	6.600000	2011.000000	2.085325e+07	3.369710e+07
max	9767.000000	9.200000	2015.000000	4.250000e+08	2.827124e+09

```
In [5]: tmdb_data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10866 entries, 0 to 10865
Data columns (total 21 columns):
id                      10866 non-null int64
imdb_id                 10856 non-null object
popularity              10866 non-null float64
budget                  10866 non-null int64
revenue                 10866 non-null int64
original_title          10866 non-null object
cast                    10790 non-null object
homepage                2936 non-null object
director                10822 non-null object
tagline                 8042 non-null object
keywords                9373 non-null object
overview                10862 non-null object
runtime                 10866 non-null int64
genres                  10843 non-null object
production_companies    9836 non-null object
release_date            10866 non-null object
vote_count              10866 non-null int64
vote_average            10866 non-null float64
release_year            10866 non-null int64
budget_adj              10866 non-null float64
```

```
revenue_adj             10866 non-null float64
dtypes: float64(4), int64(6), object(11)
memory usage: 1.7+ MB
```

In [ ]:
```
Data Cleaning (Replace this with more specific notes!)
```

In [6]:
```
# Columns that needs to be deleted
deleted_columns = [ 'id', 'imdb_id', 'popularity', 'budget_adj', 'revenue_adj', 'homepa
# Drop the columns from the database
tmdb_data.drop(deleted_columns, axis=1, inplace=True)
# Lets look at the new dataset
tmdb_data.head()
```

Out[6]:

	budget	revenue	original_title	\
0	150000000	1513528810	Jurassic World	
1	150000000	378436354	Mad Max: Fury Road	
2	110000000	295238201	Insurgent	
3	200000000	2068178225	Star Wars: The Force Awakens	
4	190000000	1506249360	Furious 7	

	cast	runtime	\
0	Chris Pratt|Bryce Dallas Howard|Irrfan Khan|Vi...	124	
1	Tom Hardy|Charlize Theron|Hugh Keays-Byrne|Nic...	120	
2	Shailene Woodley|Theo James|Kate Winslet|Ansel...	119	
3	Harrison Ford|Mark Hamill|Carrie Fisher|Adam D...	136	
4	Vin Diesel|Paul Walker|Jason Statham|Michelle ...	137	

	genres	release_date	release_year
0	Action|Adventure|Science Fiction|Thriller	6/9/15	2015
1	Action|Adventure|Science Fiction|Thriller	5/13/15	2015
2	Adventure|Science Fiction|Thriller	3/18/15	2015
3	Action|Adventure|Science Fiction|Fantasy	12/15/15	2015
4	Action|Crime|Thriller	4/1/15	2015

In [7]:
```
rows, col = tmdb_data.shape
#since rows includes count of a header, we need to remove its count.
print('We have {} total rows and {} columns.'.format(rows-1, col))
```

```
We have 10865 total rows and 8 columns.
```

In [8]:
```
# Drop duplicate rows but keep the first one
tmdb_data.drop_duplicates(keep = 'first', inplace = True)
# Store rows and columns using shape function.
rows, col = tmdb_data.shape
print('Now we have {} total rows and {} columns.'.format(rows-1, col))
```

```
Now we have 10864 total rows and 8 columns.
```

```
In [9]: # Columns that need to be checked.
        columns = ['budget', 'revenue']
        # Replace 0 with NAN
        tmdb_data[columns] = tmdb_data[columns].replace(0, np.NaN)
        # Drop rows which contains NAN
        tmdb_data.dropna(subset = columns, inplace = True)
        rows, col = tmdb_data.shape
        print('We now have only {} rows.'.format(rows-1))
```

```
We now have only 3853 rows.
```

```
In [10]: tmdb_data.release_date = pd.to_datetime(tmdb_data['release_date'])
         # Lets look at the new dataset
         tmdb_data.head()
```

Out[10]:

	budget	revenue	original_title	cast	runtime	genres	release_date	release_year
0	150000000.0	1.513529e+09	Jurassic World	Chris Pratt|Bryce Dallas Howard|Irrfan Khan|Vi...	124	Action|Adventure|Science Fiction|Thriller	2015-06-09	2015
1	150000000.0	3.784364e+08	Mad Max: Fury Road	Tom Hardy|Charlize Theron|Hugh Keays-Byrne|Nic...	120	Action|Adventure|Science Fiction|Thriller	2015-05-13	2015
2	110000000.0	2.952382e+08	Insurgent	Shailene Woodley|Theo James|Kate Winslet|Ansel...	119	Adventure|Science Fiction|Thriller	2015-03-18	2015
3	200000000.0	2.068178e+09	Star Wars: The Force Awakens	Harrison Ford|Mark Hamill|Carrie Fisher|Adam D...	136	Action|Adventure|Science Fiction|Fantasy	2015-12-15	2015
4	190000000.0	1.506249e+09	Furious 7	Vin Diesel|Paul Walker|Jason Statham|Michelle ...	137	Action|Crime|Thriller	2015-04-01	2015

```
In [11]: # Columns to convert datatype of
         columns = ['budget', 'revenue']
         # Convert budget and revenue column to int datatype
         tmdb_data[columns] = tmdb_data[columns].applymap(np.int64)
         # Lets look at the new datatype
         tmdb_data.dtypes
```

```
Out[11]: budget                int64
         revenue               int64
         original_title       object
         cast                 object
         runtime               int64
```

```
genres                    object
release_date      datetime64[ns]
release_year               int64
dtype: object
```

In [12]:
```
# Replace runtime value of 0 to NAN, Since it will affect the result.
tmdb_data['runtime'] = tmdb_data['runtime'].replace(0, np.NaN)
# Check the stats of dataset
tmdb_data.describe()
```

Out[12]:

	budget	revenue	runtime	release_year
count	3.854000e+03	3.854000e+03	3854.000000	3854.000000
mean	3.720370e+07	1.076866e+08	109.220291	2001.261028
std	4.220822e+07	1.765393e+08	19.922820	11.282575
min	1.000000e+00	2.000000e+00	15.000000	1960.000000
25%	1.000000e+07	1.360003e+07	95.000000	1995.000000
50%	2.400000e+07	4.480000e+07	106.000000	2004.000000
75%	5.000000e+07	1.242125e+08	119.000000	2010.000000
max	4.250000e+08	2.781506e+09	338.000000	2015.000000

In [ ]: Exploratory Data Analysis

Tip: Now that you've trimmed and cleaned your data, you're ready to move on to expl

Research Question 1 (Which movie had the highest and lowest profit?)

In [13]:
```
# To calculate profit, we need to substract the budget from the revenue.
tmdb_data['profit'] = tmdb_data['revenue'] - tmdb_data['budget']
# Lets look at the new dataset
tmdb_data.head()
```

Out[13]:

	budget	revenue	original_title	\
0	150000000	1513528810	Jurassic World	
1	150000000	378436354	Mad Max: Fury Road	
2	110000000	295238201	Insurgent	
3	200000000	2068178225	Star Wars: The Force Awakens	
4	190000000	1506249360	Furious 7	

	cast	runtime	\
0	Chris Pratt|Bryce Dallas Howard|Irrfan Khan|Vi...	124	
1	Tom Hardy|Charlize Theron|Hugh Keays-Byrne|Nic...	120	
2	Shailene Woodley|Theo James|Kate Winslet|Ansel...	119	
3	Harrison Ford|Mark Hamill|Carrie Fisher|Adam D...	136	
4	Vin Diesel|Paul Walker|Jason Statham|Michelle ...	137	

	genres	release_date	release_year	\
0	Action|Adventure|Science Fiction|Thriller	2015-06-09	2015	
1	Action|Adventure|Science Fiction|Thriller	2015-05-13	2015	
2	Adventure|Science Fiction|Thriller	2015-03-18	2015	

```
3   Action|Adventure|Science Fiction|Fantasy   2015-12-15          2015
4                     Action|Crime|Thriller   2015-04-01          2015

       profit
0  1363528810
1   228436354
2   185238201
3  1868178225
4  1316249360
```

In [14]:
```
# Movie with highest profit
tmdb_data.loc[tmdb_data['profit'].idxmax()]
```

Out[14]:
```
budget                                                    237000000
revenue                                                  2781505847
original_title                                               Avatar
cast              Sam Worthington|Zoe Saldana|Sigourney Weaver|S...
runtime                                                         162
genres                   Action|Adventure|Fantasy|Science Fiction
release_date                                  2009-12-10 00:00:00
release_year                                                 2009
profit                                                   2544505847
Name: 1386, dtype: object
```

In [15]:
```
# Movie with lowest profit
tmdb_data.loc[tmdb_data['profit'].idxmin()]
```

Out[15]:
```
budget                                                    425000000
revenue                                                    11087569
original_title                                   The Warrior's Way
cast              Kate Bosworth|Jang Dong-gun|Geoffrey Rush|Dann...
runtime                                                         100
genres                  Adventure|Fantasy|Action|Western|Thriller
release_date                                  2010-12-02 00:00:00
release_year                                                 2010
profit                                                   -413912431
Name: 2244, dtype: object
```

In [ ]: Research Question 1.2 (Which movie had the greatest and least runtime?)

In [16]:
```
# Movie with greatest runtime
tmdb_data.loc[tmdb_data['runtime'].idxmax()]
```

Out[16]:
```
budget                                                     18000000
revenue                                                      871279
original_title                                               Carlos
cast              Edgar RamÃ¡rez|Alexander Scheer|Fadi Abi Samra...
runtime                                                         338
genres                                Crime|Drama|Thriller|History
```

```
release_date                                          2010-05-19 00:00:00
release_year                                                         2010
profit                                                           -17128721
Name: 2107, dtype: object
```

```
In [17]: # Movie with least runtime
         tmdb_data.loc[tmdb_data['runtime'].idxmin()]
```

```
Out[17]: budget                                                         10
         revenue                                                         5
         original_title                                        Kid's Story
         cast              Clayton Watson|Keanu Reeves|Carrie-Anne Moss|K...
         runtime                                                        15
         genres                                  Science Fiction|Animation
         release_date                                2003-06-02 00:00:00
         release_year                                               2003
         profit                                                       -5
         Name: 5162, dtype: object
```

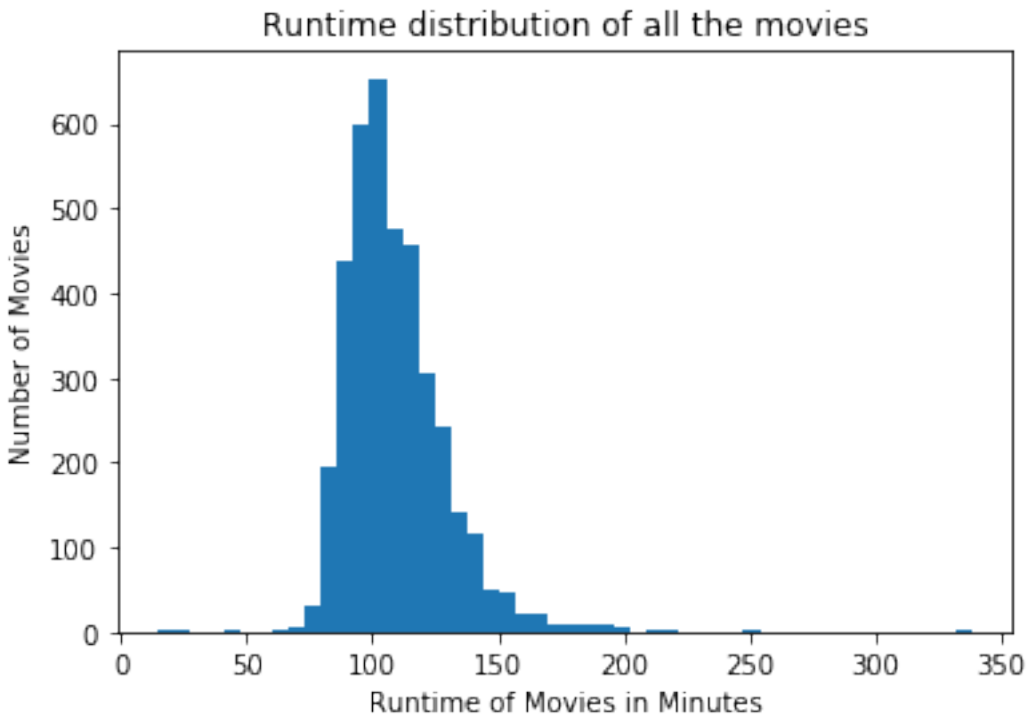
```
In [ ]: Research Question 1.3 (What is the average runtime of all movies?)
```

```
In [18]: # Average runtime of movies
         tmdb_data['runtime'].mean()
```

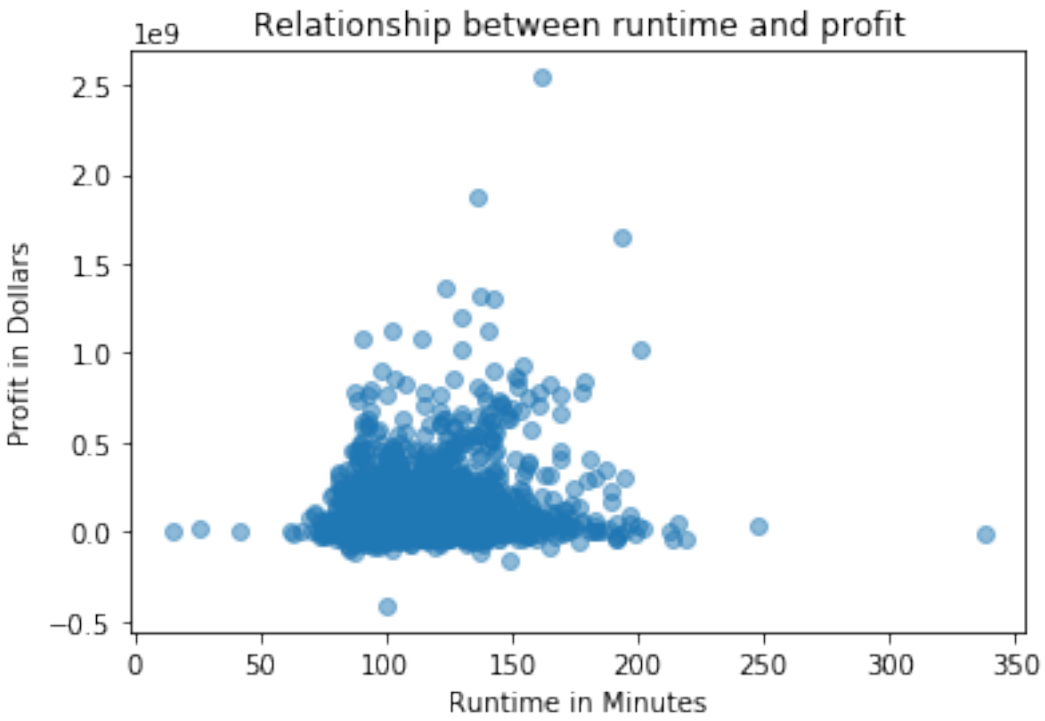
```
Out[18]: 109.22029060716139
```

```
In [19]: # x-axis
         plt.xlabel('Runtime of Movies in Minutes')
         # y-axis
         plt.ylabel('Number of Movies')
         # Title of the histogram
         plt.title('Runtime distribution of all the movies')
         # Plot a histogram
         plt.hist(tmdb_data['runtime'], bins = 50)
```

```
Out[19]: (array([  1.,   1.,   0.,   0.,   1.,   0.,   0.,   3.,   5.,  31., 196.,
                 437., 598., 653., 475., 458., 305., 243., 141., 117.,  49.,  47.,
                  21.,  23.,   9.,  10.,  10.,   8.,   6.,   0.,   2.,   2.,   0.,
                   0.,   0.,   0.,   1.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,
                   0.,   0.,   0.,   0.,   0.,   1.]),
          array([ 15.  ,  21.46,  27.92,  34.38,  40.84,  47.3 ,  53.76,  60.22,
                  66.68,  73.14,  79.6 ,  86.06,  92.52,  98.98, 105.44, 111.9 ,
                 118.36, 124.82, 131.28, 137.74, 144.2 , 150.66, 157.12, 163.58,
                 170.04, 176.5 , 182.96, 189.42, 195.88, 202.34, 208.8 , 215.26,
                 221.72, 228.18, 234.64, 241.1 , 247.56, 254.02, 260.48, 266.94,
                 273.4 , 279.86, 286.32, 292.78, 299.24, 305.7 , 312.16, 318.62,
                 325.08, 331.54, 338.  ]),
          <a list of 50 Patch objects>)
```




```
In [20]: # x-axis
         plt.xlabel('Runtime in Minutes')
         # y-axis
         plt.ylabel('Profit in Dollars')
         # Title of the histogram
         plt.title('Relationship between runtime and profit')
         plt.scatter(tmdb_data['runtime'], tmdb_data['profit'], alpha=0.5)
         plt.show()
```



In [ ]: Research Question 1.4 (Which movie had the highest and lowest budget?)

In [21]:
```
# Movie with lowest budget
tmdb_data.loc[tmdb_data['budget'].idxmin()]
```

Out[21]:
```
budget                                                          1
revenue                                                       100
original_title                                        Lost & Found
cast             David Spade|Sophie Marceau|Ever Carradine|Step...
runtime                                                        95
genres                                           Comedy|Romance
release_date                                1999-04-23 00:00:00
release_year                                               1999
profit                                                       99
Name: 2618, dtype: object
```

In [22]:
```
# Movie with highest budget
tmdb_data.loc[tmdb_data['budget'].idxmax()]
```

Out[22]:
```
budget                                                  425000000
revenue                                                  11087569
original_title                                The Warrior's Way
cast             Kate Bosworth|Jang Dong-gun|Geoffrey Rush|Dann...
runtime                                                      100
```

```
genres                  Adventure|Fantasy|Action|Western|Thriller
release_date                                  2010-12-02 00:00:00
release_year                                                 2010
profit                                                   -413912431
Name: 2244, dtype: object
```

In [ ]:
```
Conclusions
```

In [ ]:
```
The limitations associated with the conclusions are:
The conclusion is not full proof that given the above requirement the movie will be a l
Also, we also lost some of the data in the data cleaning steps where we dont know the r
This conclusion is not error proof.
```

In [23]:
```
from subprocess import call
call(['python', '-m', 'nbconvert', 'Investigate_a_Dataset.ipynb'])
```

Out[23]:
```
255
```