# imdb_top_250_movies_wikipedia_revisions_analysis

February 3, 2019

## 1 IMDB TOP 250 MOVIES WITH WIKIPEDIA REVISIONS

### 1.1 Main objective

- Analyze the correlation between the number of votes and revisions and present your findings

```
In [1]: import pandas as pd
        import matplotlib
        import matplotlib.pyplot as plt
        %matplotlib inline
```

```
In [2]: data = pd.read_csv("./files/top250_imdb_movies_with_wikipedia_revisions.csv")
```

#### 1.1.1 1. Imported dataset contains no null values

```
In [3]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 250 entries, 0 to 249
Data columns (total 7 columns):
rank         250 non-null int64
title        250 non-null object
year         250 non-null int64
votes        250 non-null int64
kind         250 non-null object
rating       250 non-null float64
revisions    250 non-null int64
dtypes: float64(1), int64(4), object(2)
memory usage: 13.8+ KB
```

#### 1.1.2 2. Samples of the data set

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

```
Out[4]:    rank                     title  year    votes   kind  rating  revisions
        0     1  The Shawshank Redemption  1994  2048674  movie     9.2       6559
```

```
1     2              The Godfather  1972  1405104  movie     9.2       9562
2     3     The Godfather: Part II  1974   974263  movie     9.0       3833
3     4            The Dark Knight  2008  2015822  movie     9.0      13984
4     5               12 Angry Men  1957   577125  movie     8.9        307
```

```
In [5]: data.tail()
```

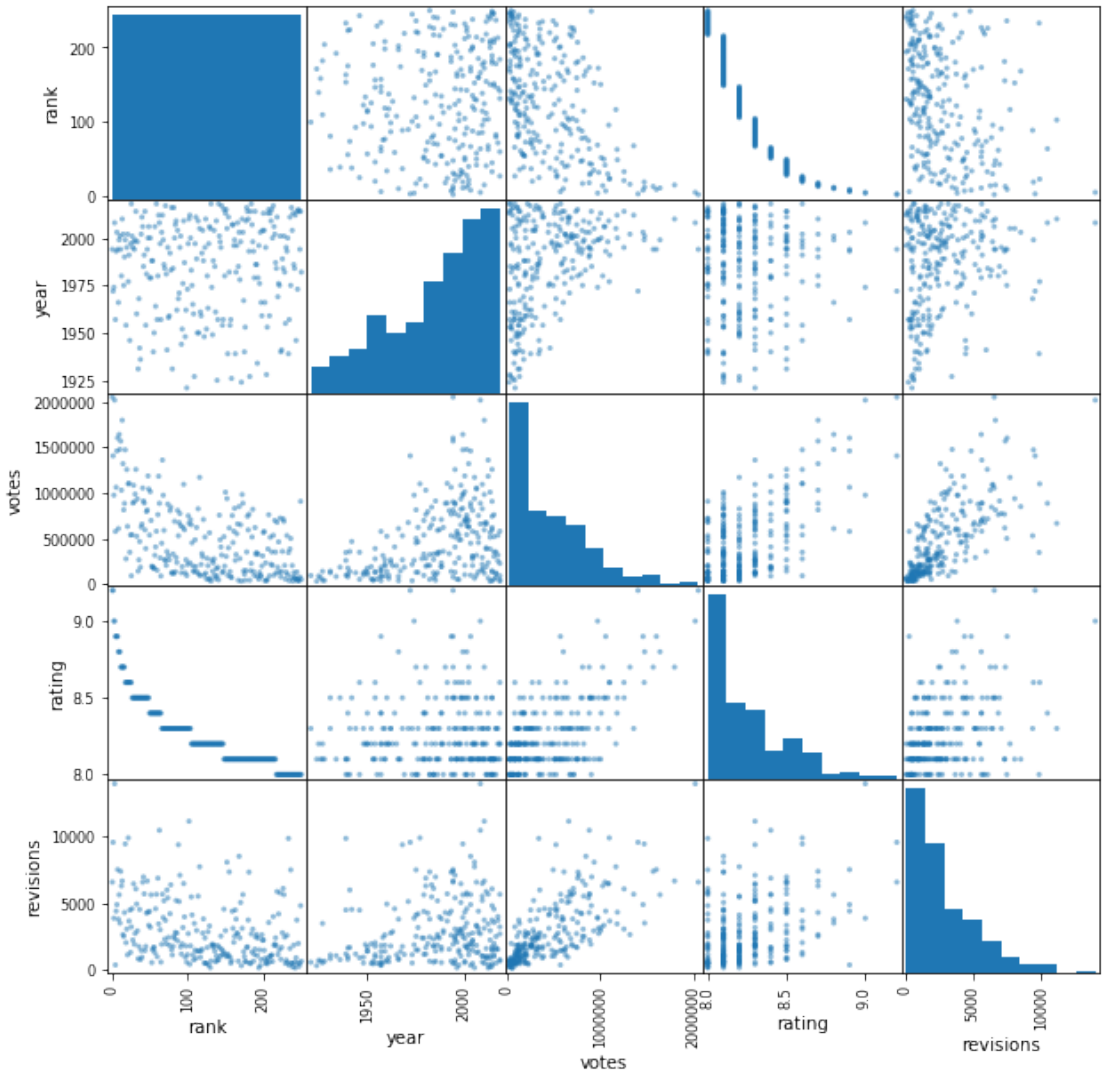
```
Out[5]:      rank                    title  year   votes   kind  rating  revisions
245   246                  Drishyam  2015   52577  movie     8.0       1929
246   247              Winter Sleep  2014   35650  movie     8.0        204
247   248         Three Colors: Red  1994   77986  movie     8.0        421
248   249  Guardians of the Galaxy  2014  906557  movie     8.0       4781
249   250       Fanny and Alexander  1982   51462  movie     8.0        587
```

### 1.1.3 3. Scatter Plot Matrix of Each Featured Compared to All Others

**Initial observations**

1. Rating and rank value decrease correspondingly. This is a good way to validate our data makes sense.
2. Revisions and rank does not have a noticeable linear relationship, but the correlation may prove otherwise.
3. Not specifically part of the assignment, but several other relationships have been discovered: revisions - votes, and votes - rating.

```
In [6]: _ = pd.plotting.scatter_matrix(data, figsize=(10, 10))
```




### 1.1.4 4. Calculate the 3 Main Ways of Correlation: Pearson, Spearman, and Kendall

**Observations**

1. Revisions and rank: weak negative correlation with all three calculations. Spearman rank and Kendall are used for rank correlations which is the test we are observing. Both are similar, although the Kendall correllation is lower. This is generally the case when comparing Spearman vs. Kendall. However, with a smaller sample size, I would tend to choose the Kendall correlation.
2. Rating and rank: As noted above, a good control to see that our calculations make sense. All three correlations show a strong negative relationship, Spearman being the highest. This relationship shows that your rank lowers as your rating lowers.
3. Ratings and votes: Surprising positive correlation. It could mean that people who really love their movie of choice are more willing to vote.

4. Revisions and votes: Similiar surprising positive correlation. Again it is possible that people who are really involved with the movie are more willing to keep it's wikipedia page up to date. Another possiblity is that it's a popular movie, therefore more fans, therefore more people who are picky in how it's page is authored and will edit the page to fit their narrative.

```
In [7]: data.corr().style.format("{:.2}").background_gradient(cmap=plt.get_cmap('coolwarm'), ax
```

```
Out[7]: <pandas.io.formats.style.Styler at 0x115290a20>
```

```
In [8]: data.corr("spearman").style.format("{:.2}").background_gradient(cmap=plt.get_cmap('coo
```

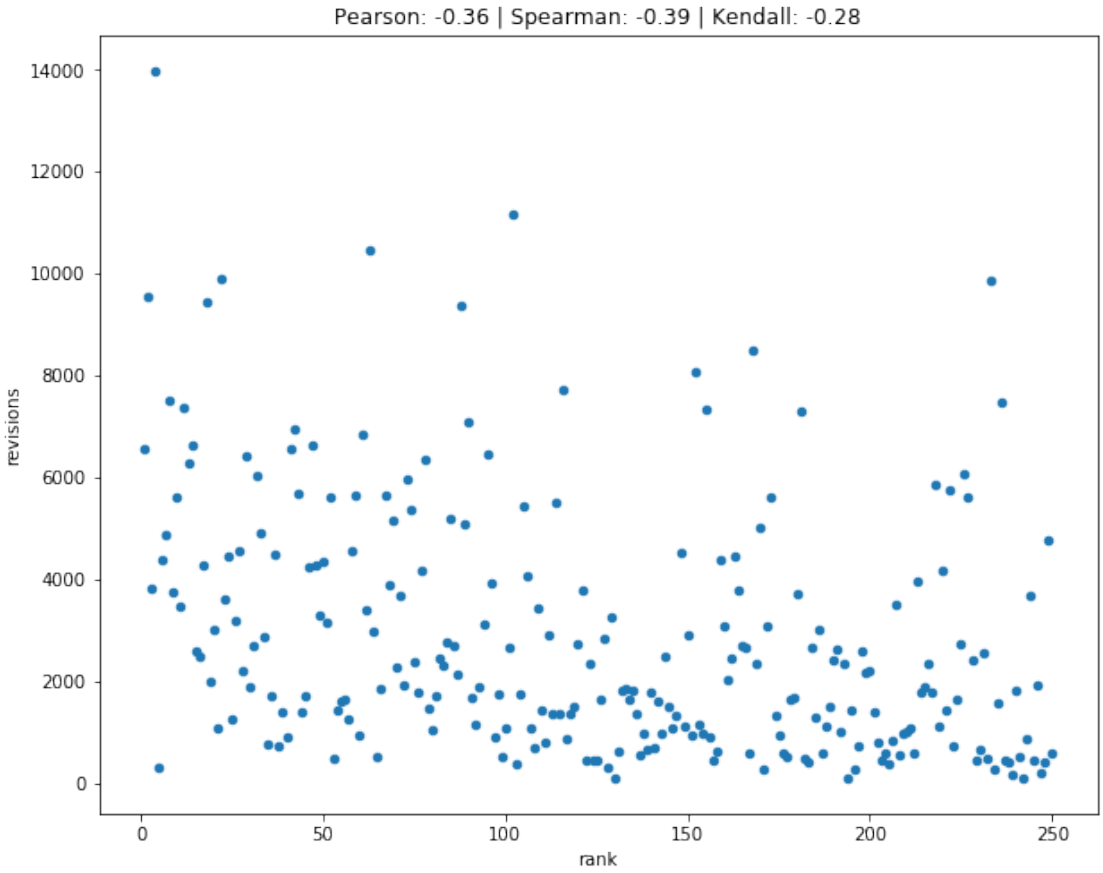
```
Out[8]: <pandas.io.formats.style.Styler at 0x115bb9cf8>
```

```
In [9]: data.corr("kendall").style.format("{:.2}").background_gradient(cmap=plt.get_cmap('coolw
```

```
Out[9]: <pandas.io.formats.style.Styler at 0x115be0390>
```

### 1.1.5 5. Revisions and Rank Scatter Plot with Correlation Calculations

```
In [10]: pearson_values = data.corr()['rank']['revisions']
         spearman_values = data.corr('spearman')['rank']['revisions']
         kendall_values = data.corr('kendall')['rank']['revisions']
         _ = data.plot.scatter(
             "rank",
             "revisions",
             title=f"Pearson: {pearson_values:.2} | Spearman: {spearman_values:.2} | Kendall:
             figsize=(10, 8))
```




### 1.1.6 6. Conclusion

Overall, the comparision of how a movie is ranked in the IMDB top 250 and the number of revisions for it's wikipedia page are weakly negative correlated. This means that there is possibly a small relationship between a movies rank towards the top and the number of revisions, but not entirely so. Ideally, one would obtain more data, possibly the top 1000 movies in IMDB, to see if the relationship changes.

In [ ]: