

```
In [1]: import numpy as np
import pandas as pd
import os
import random
import json
import ast
import nltk
import re
import cv2
from skimage import io
import matplotlib
import matplotlib.pyplot as plt
from nltk.corpus import sentiwordnet as swn
from nltk.corpus import stopwords
from nltk.tokenize import RegexpTokenizer
from nltk.stem import WordNetLemmatizer
from nltk.corpus import wordnet
from sklearn import linear_model
from sklearn.metrics import r2_score
from sklearn.metrics import accuracy_score
from sklearn.model_selection import cross_val_score
from sklearn.preprocessing import normalize
from requests.exceptions import HTTPError
```

```
In [2]: plt.rcParams["figure.figsize"] = [12,8]
def yearlist(start, end):
    ans = []
    for x in xrange(start, end + 1):
        ans.append(x)
    return ans
obj_dic = {}
def add_dic(x):
    for ele in x.split(","):
        if ele in obj_dic:
            obj_dic[ele] += 1
        else:
            obj_dic[ele] = 1
    return 0

#Sentiment Analysis Help Functions
def get_wordnet_pos(treebank_tag):
    string, tag = treebank_tag
    string = string.lower()
    treebank_tag = tag.upper()
    if treebank_tag.startswith('J'):
        return wordnet.ADJ
    elif treebank_tag.startswith('V'):
        return wordnet.VERB
    elif treebank_tag.startswith('N'):
        return wordnet.NOUN
    elif treebank_tag.startswith('R'):
        return wordnet.ADV
    else:
```

```

        return 'X'
def wordnet_sanitizeword):
    string, tag = word
    string = string.lower()
    tag = tag.lower()
    if tag.startswith('v'):    tag = 'v'
    elif tag.startswith('n'):  tag = 'n'
    elif tag.startswith('j'):  tag = 'a'
    elif tag.startswith('rb'): tag = 'r'
    if tag in ('a', 'n', 'r', 'v'):
        return (string, tag)
    else:
        return (string, None)

#Box Plot Coloring
def boxplot_color(bp):
    for box in bp['boxes']:
        # change outline color
        box.set( color='#7570b3', linewidth=2)
        # change fill color
        box.set( facecolor = '#1b9e77' )

    ## change color and linewidth of the whiskers
    for whisker in bp['whiskers']:
        whisker.set(color='#7570b3', linewidth=2)

    ## change color and linewidth of the caps
    for cap in bp['caps']:
        cap.set(color='#7570b3', linewidth=2)

    ## change color and linewidth of the medians
    for median in bp['medians']:
        median.set(color='#b2df8a', linewidth=2)

    ## change the style of fliers and their fill
    for flier in bp['fliers']:
        flier.set(marker='o', color='#e7298a', alpha=0.5)

```

```

In [3]: oscar = pd.read_csv("./oscars.csv", header = 0)
oscar = map(lambda x : str(re.sub(r'[\x00-\x7F]+' , "" ,str(x.lower()).replace(" ", "")))) ,list(oscar['Movie_Name']))

```

```

In [4]: #Load All Scrapped Data
google_movies_folder = "./google-movies"
facebook_movie_folder = "./facebook-movies"
imdb_movie_folder = "./imdb-movies"
years = yearlist(2008,2016)
dic_google, dic_fb, dic_imdb, dic_people = {}, {}, {}, {}
for year in years:
    dic_google[year] = pd.read_csv(google_movies_folder + "/movie-data-" +
    str(year) + ".csv", header = 0)
    dic_fb[year] = pd.read_csv(facebook_movie_folder + "/movie-data-" + str(
    year) + ".csv", header = 0)
    dic_imdb[year] = pd.read_csv(imdb_movie_folder + "/movie-data-" + str(
    year) + ".csv", header = None)
    dic_people[year] = pd.read_csv(imdb_movie_folder + "/movie-info-" + str

```

```
r(year) + ".csv", header = 0)
dic_imdb[year].columns = ['id','title','score','rating','review']
```

```
In [5]: #Find All Genres from movies - Refine
list_genres = []
years = yearlist(2008,2016)
for year in years:
    obj = dic_google[year]
    for line in xrange(len(obj)):
        if len(ast.literal_eval(obj[line:line + 1]['genres'][line])) > 0:
            elements = ast.literal_eval(str(obj[line:line + 1]['genres'][line]))
            gen = ""
            for ele in elements:
                gen = gen + ele['name'] + ","
            if ele['name'] not in list_genres:
                list_genres.append(ele['name'])
            obj.loc[line,"genres"] = str(gen[:-1])
list_genres = map(str, list_genres)
```

```
In [6]: #Find All production Houses from movies - Refine
list_prod = []
years = yearlist(2008,2016)
for year in years:
    obj = dic_google[year]
    for line in xrange(len(obj)):
        if len(ast.literal_eval(obj[line:line + 1]['production'][line])) > 0:
            elements = ast.literal_eval(str(obj[line:line + 1]['production'][line]))
            gen = ""
            for ele in elements:
                gen = gen + ele['name'] + ","
            if ele['name'] not in list_genres:
                list_prod.append(ele['name'].encode("utf8"))
            obj.loc[line,"production"] = str(gen[:-1].encode("utf8"))
list_prod = map(str, list_prod)
```

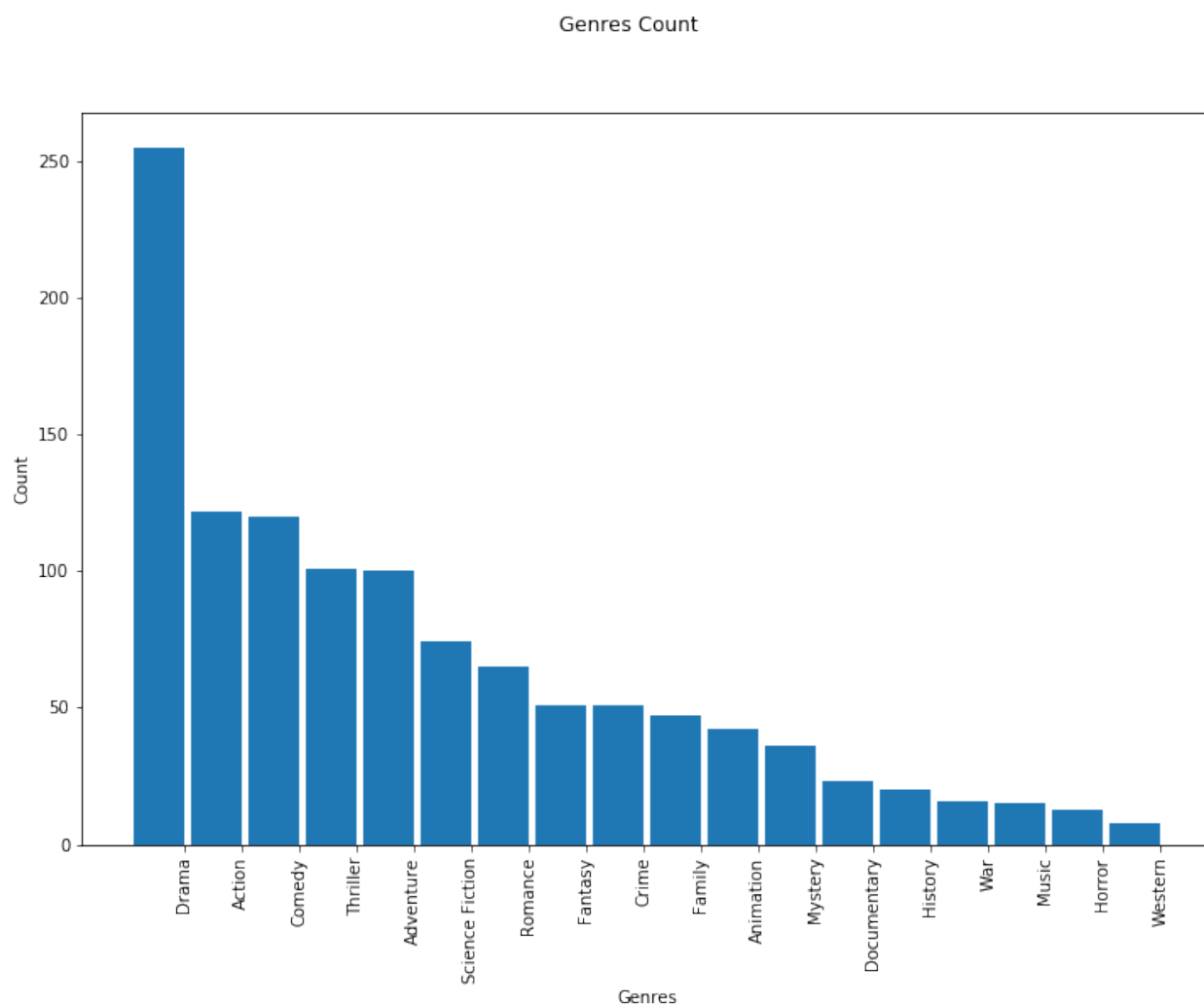
```
In [7]: #Genres Hot Encoding
years = yearlist(2008,2016)
for year in years:
    dic_google[year]['year'] = year
for year in years:
    for ls in list_genres:
        dic_google[year][ls] = dic_google[year]['genres'].apply(lambda x:
1 if ls in x.split(",") else 0)
```

```
In [8]: #Count all production houses and genres
dic_prod, dic_genre = {}, {}
obj_dic = dic_prod
years = yearlist(2008,2016)
for year in years:
    dic_google[year]['production'].apply(add_dic)
del dic_prod[' The']
dic_prod = sorted(dic_prod.iteritems(), key=lambda (k,v): (v,k), reverse=
```

```
True)
obj_dic = dic_genre
for year in years:
    dic_google[year]['genres'].apply(add_dic)
dic_genre = sorted(dic_genre.iteritems(), key=lambda (k,v): (v,k), reverse
= True)
dic_genre = dic_genre[:-1]
```

```
In [9]: keys = map(lambda x : str(x[0]), dic_genre)
values = map(lambda x : str(x[1]), dic_genre)
indexes = np.arange(len(keys))
width = .9
plt.suptitle("Genres Count")
plt.bar(indexes, values, width)
plt.xticks(indexes + width * 0.5, keys, rotation='vertical')
plt.ylabel('Count')
plt.xlabel('Genres')
fig_size = plt.rcParams["figure.figsize"]
plt.rcParams["figure.figsize"] = fig_size
print "Figure Size",fig_size
plt.show()
```

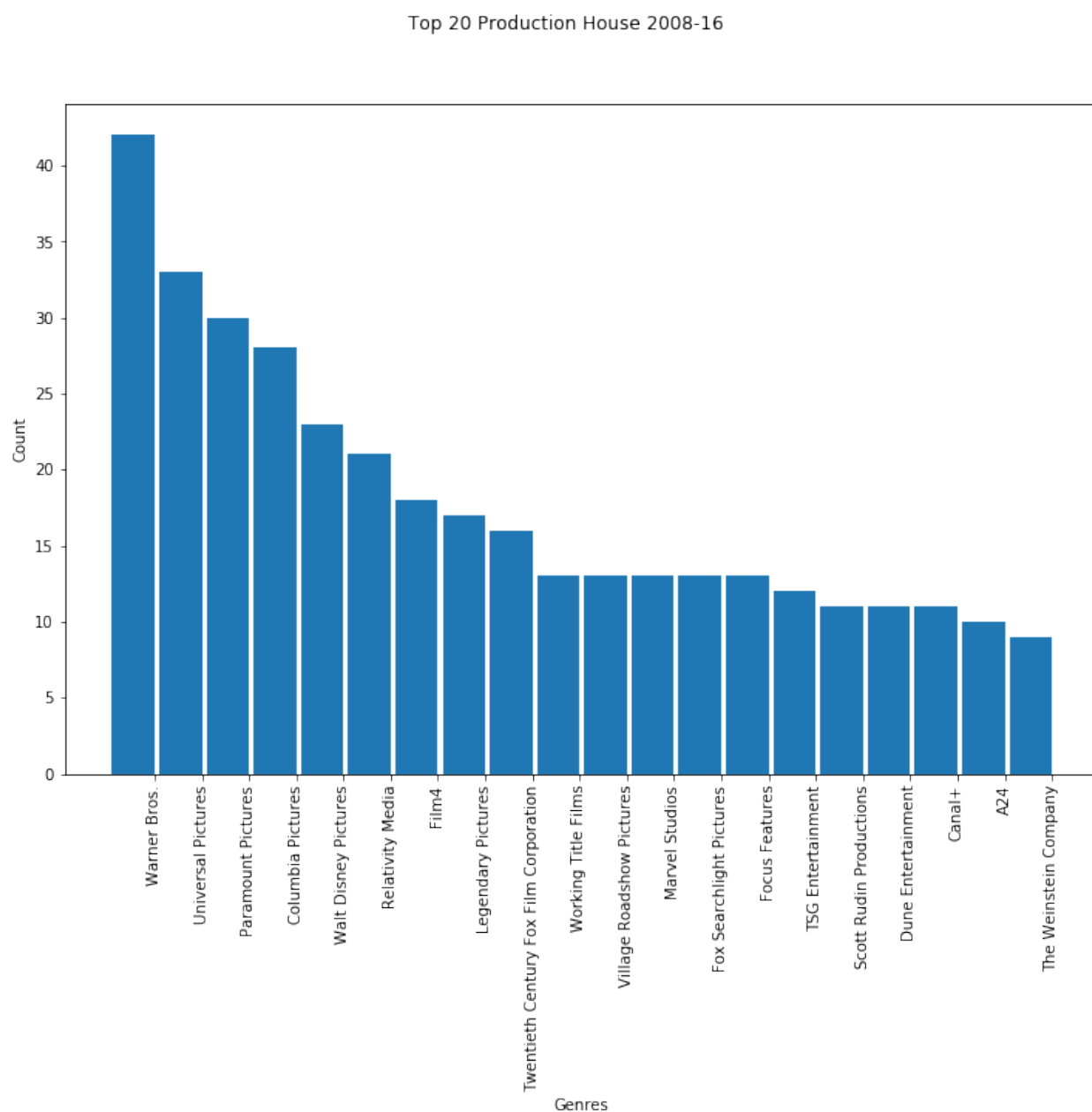
Figure Size [12.0, 8.0]



```
In [303]: keys = map(lambda x : str(re.sub(r'^\x00-\x7F'+',', str(x[0]))), dic_pro
```

```
d[:20])
values = map(lambda x : x[1], dic_prod[:20])
indexes = np.arange(len(keys))
width = .9
plt.suptitle("Top 20 Production House 2008-16")
plt.bar(indexes, values, width)
plt.xticks(indexes + width * 0.5, keys, rotation='vertical')
plt.ylabel('Count')
plt.xlabel('Genres')
fig_size = plt.rcParams["figure.figsize"]
plt.rcParams["figure.figsize"] = fig_size
print "Figure Size",fig_size
plt.show()
```

Figure Size [12.0, 8.0]



```
In [10]: #Sentiment Analysis of Overview
tokenizer = RegexpTokenizer(r'\w+')
stemmer = WordNetLemmatizer()
```

```

years = yearlist(2008,2016)
for year in years:
    obj = dic_google[year]
    obj['overview_pos_score'] = 0
    obj['overview_neg_score'] = 0
    for line in xrange(len(obj)):
        sentence = re.sub(r'^\x00-\x7F+', '', str(obj.loc[line, "overview"
]))
        sentence = re.sub("\d+", "", sentence)
        sentence = tokenizer.tokenize(sentence)
        sentence = [word.lower() for word in sentence if word not in stopw
ords.words('english')]
        sentence = [str(stemmer.lemmatize(plural, get_wordnet_pos(nltk.pos_
tag([plural])[0]))) if get_wordnet_pos(nltk.pos_tag([plural])[0]) != 'X' e
lse str(stemmer.lemmatize(plural)) for plural in sentence]
        sentiment = map(lambda x : swin.senti_synsets(str(x), wordnet_sanit
ize(nltk.pos_tag([x])[0])[1]), sentence)
        obj.loc[line, 'overview_pos_score'] = sum(map(lambda x : x[0].pos_s
core() if len(x) > 0 else 0, sentiment))/len(sentiment)
        obj.loc[line, 'overview_neg_score'] = sum(map(lambda x : x[0].neg_s
core() if len(x) > 0 else 0, sentiment))/len(sentiment)

```

```

In [11]: #Face Information
movie_path = "https://image.tmbd.org/t/p/w500"
face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml'
)
profile_cascade = cv2.CascadeClassifier('haarcascade_profileface.xml')
years = yearlist(2008,2016)
for year in years:
    obj = dic_google[year]
    obj['faces'] = 0
    for line in xrange(len(obj)):
        if str(obj.loc[line, 'poster']) != "nan":
            url = movie_path + str(obj.loc[line, 'poster'])
            try:
                image = io.imread(url)
                img = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
                gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
                front_faces = face_cascade.detectMultiScale(gray, 1.05, 5)
                profile_faces = profile_cascade.detectMultiScale(gray, 1.0
5, 5)
                obj.loc[line, 'faces'] = len(front_faces) + len(profile_fa
ces)
            except HTTPError:
                print url
    print "Posters for",year,"Done!"

```

```

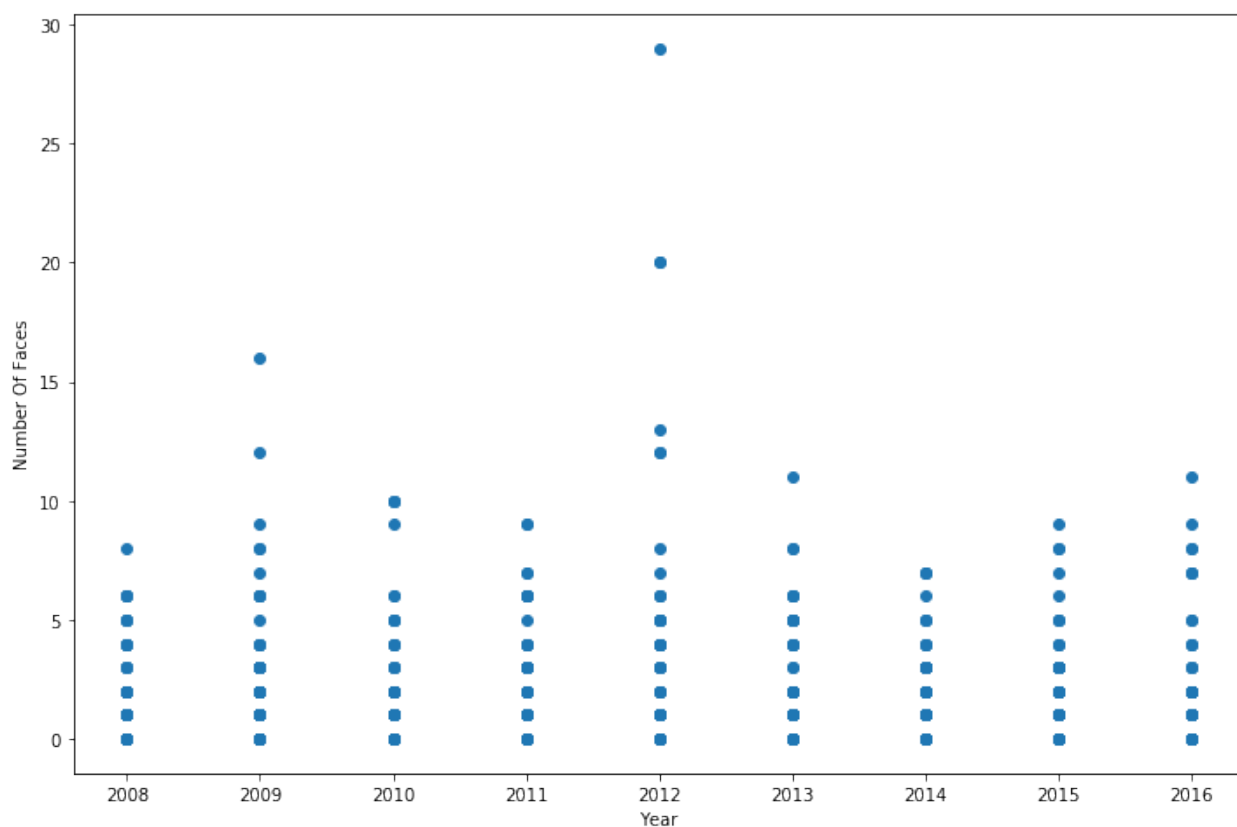
Posters for 2008 Done!
Posters for 2009 Done!
Posters for 2010 Done!
Posters for 2011 Done!
Posters for 2012 Done!
Posters for 2013 Done!
Posters for 2014 Done!
Posters for 2015 Done!
Posters for 2016 Done!

```

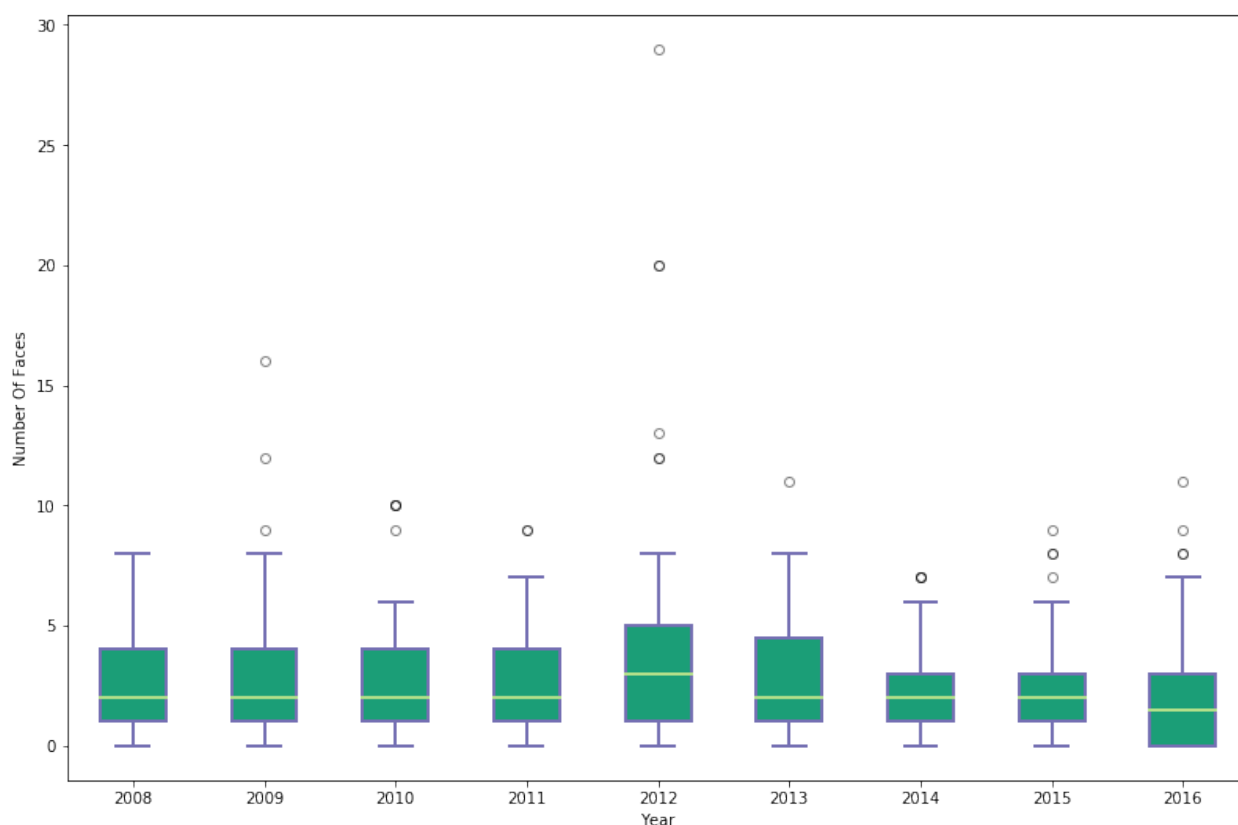
```
In [12]: #Number Of Faces Per Year Plot
x = []
y = []
years = yearlist(2008,2016)
for year in years:
    obj = dic_google[year]
    x += list(obj['year'])
    y += list(obj['faces'])
plt.plot(x, y, "o")
plt.suptitle("Faces In Facebook Display Picture From 2008-16")
plt.ylabel('Number Of Faces')
plt.xlabel('Year')
plt.show()

y = []
for year in years:
    obj = dic_google[year]
    y.append(list(obj['faces']))
fig = plt.figure(1, figsize=(13.5, 9))
ax = fig.add_subplot(111)
bp = ax.boxplot(y, patch_artist=True)
boxplot_color(bp)
plt.suptitle("Faces In Facebook Display Picture From 2008-16")
ax.set_xticklabels(years)
plt.ylabel('Number Of Faces')
plt.xlabel('Year')
plt.show()
```

Faces In Facebook Display Picture From 2008-16



Faces In Facebook Display Picture From 2008-16



```
In [13]: #Concat FB Likes
years = yearlist(2008,2016)
for year in years:
    obj = dic_google[year]
    fb = dic_fb[year]
    obj['fb_likes'] = 0
    for line in xrange(len(obj)):
        find_movie = fb[fb['title'] == str(obj.loc[line,'title'])]['likes'
    ]

    if len(find_movie) > 0:
        obj.loc[line, 'fb_likes'] = int(find_movie.sum() / len(find_movie))
```

```
In [14]: #FB Likes Plot
years = yearlist(2008,2016)
fig = {}
count = 1
width = .9
for year in years:
    obj = dic_google[year]
    result = obj[obj['fb_likes'] != 0][['title','fb_likes']]
    keys = list(result['title'])
    values = list(result['fb_likes'])
    colors = []
    for k in keys:
        if str(re.sub(r'^\x00-\x7F+', "", str(k.lower().replace(" ", ""))))
in oscar:
```



```
        colors.append('r')
    else:
        colors.append('b')
    indexes = np.arange(len(keys))
    fig[year] = plt.figure(count)
    plt.bar(indexes, values, width, color=colors)
    plt.xticks(indexes + width * 0.5, keys, rotation='vertical')
    plt.ylabel('Count ' + str(year))
    plt.xlabel('Facebook Likes')
    fig_size = plt.rcParams["figure.figsize"]
    print "Figure Size",fig_size
    print "OSCAR MOVIES ARE HIGHLIGHTED WITH RED"
    count += 1
    plt.show()
```

Figure Size [12.0, 8.0]
OSCAR MOVIES ARE HIGHLIGHTED WITH RED

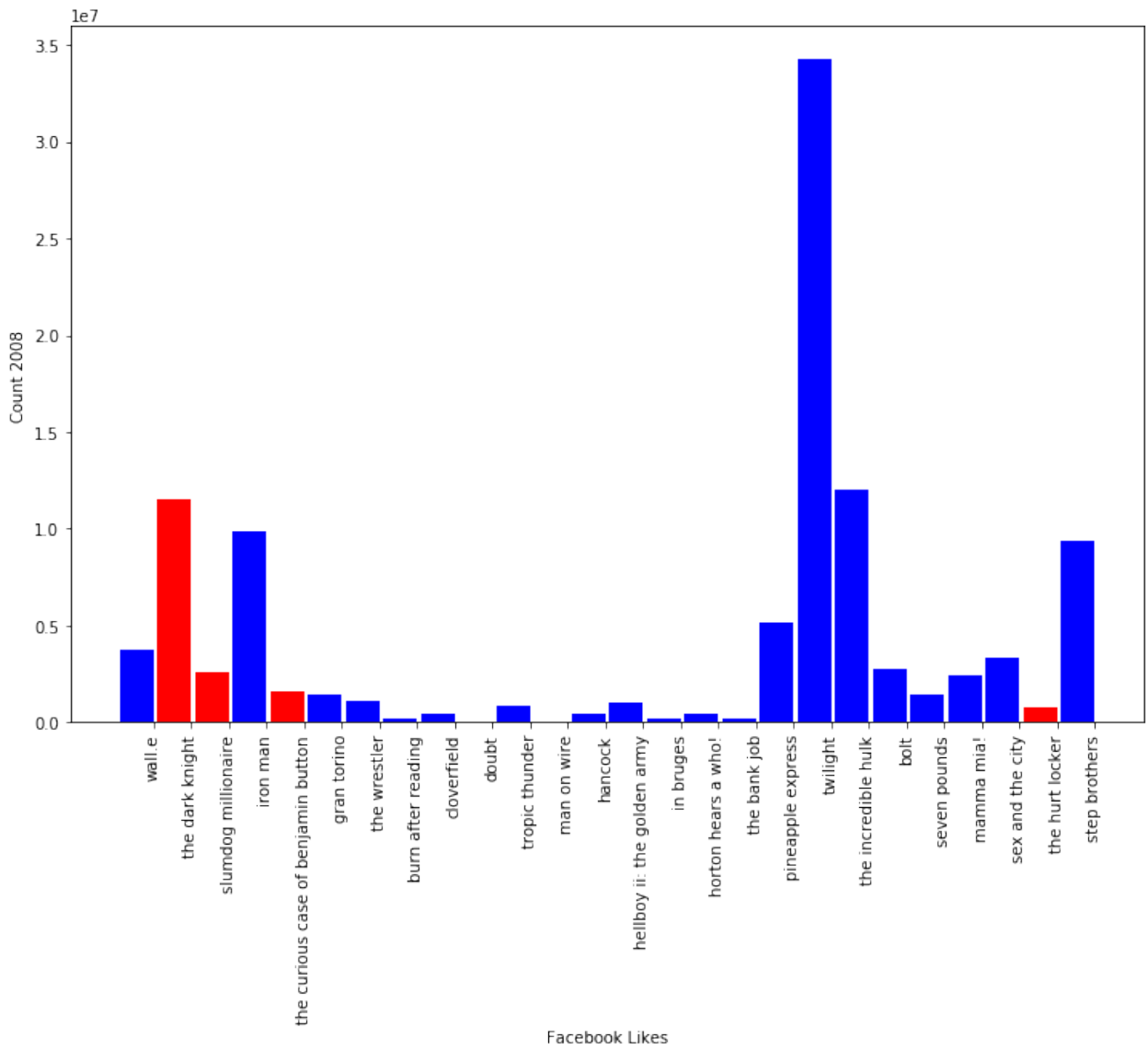


Figure Size [12.0, 8.0]
OSCAR MOVIES ARE HIGHLIGHTED WITH RED

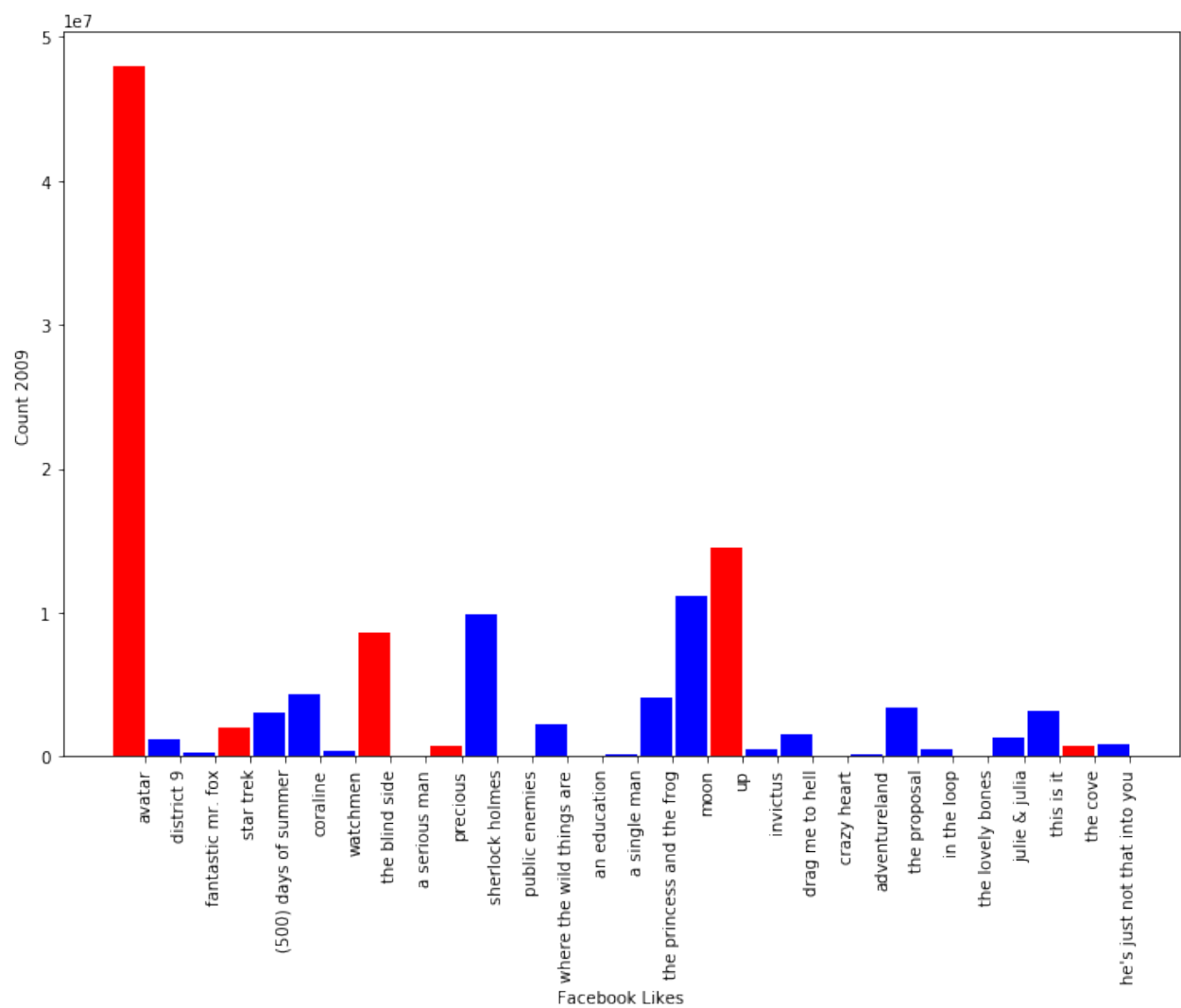


Figure Size [12.0, 8.0]
OSCAR MOVIES ARE HIGHLIGHTED WITH RED

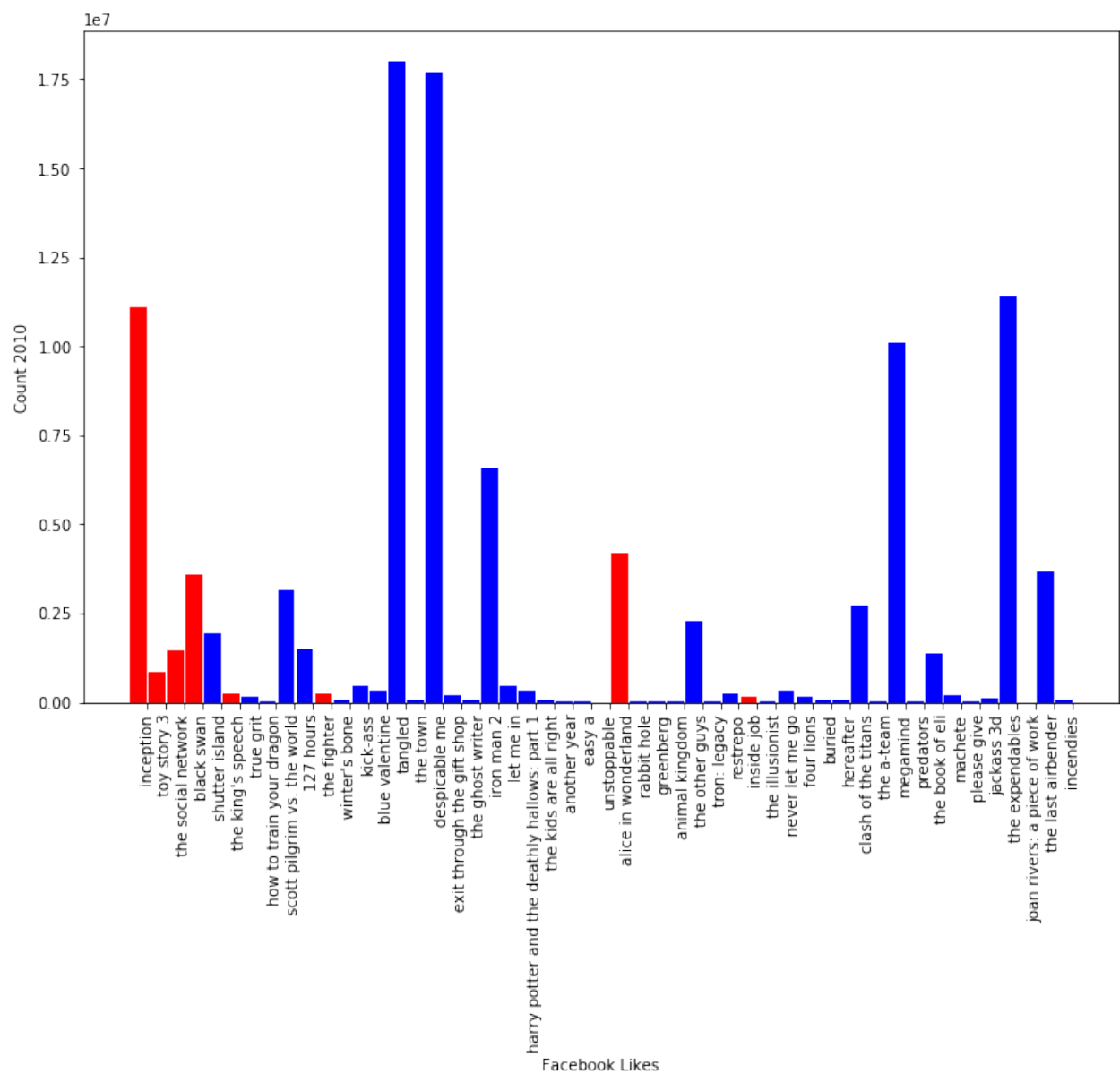


Figure Size [12.0, 8.0]
OSCAR MOVIES ARE HIGHLIGHTED WITH RED

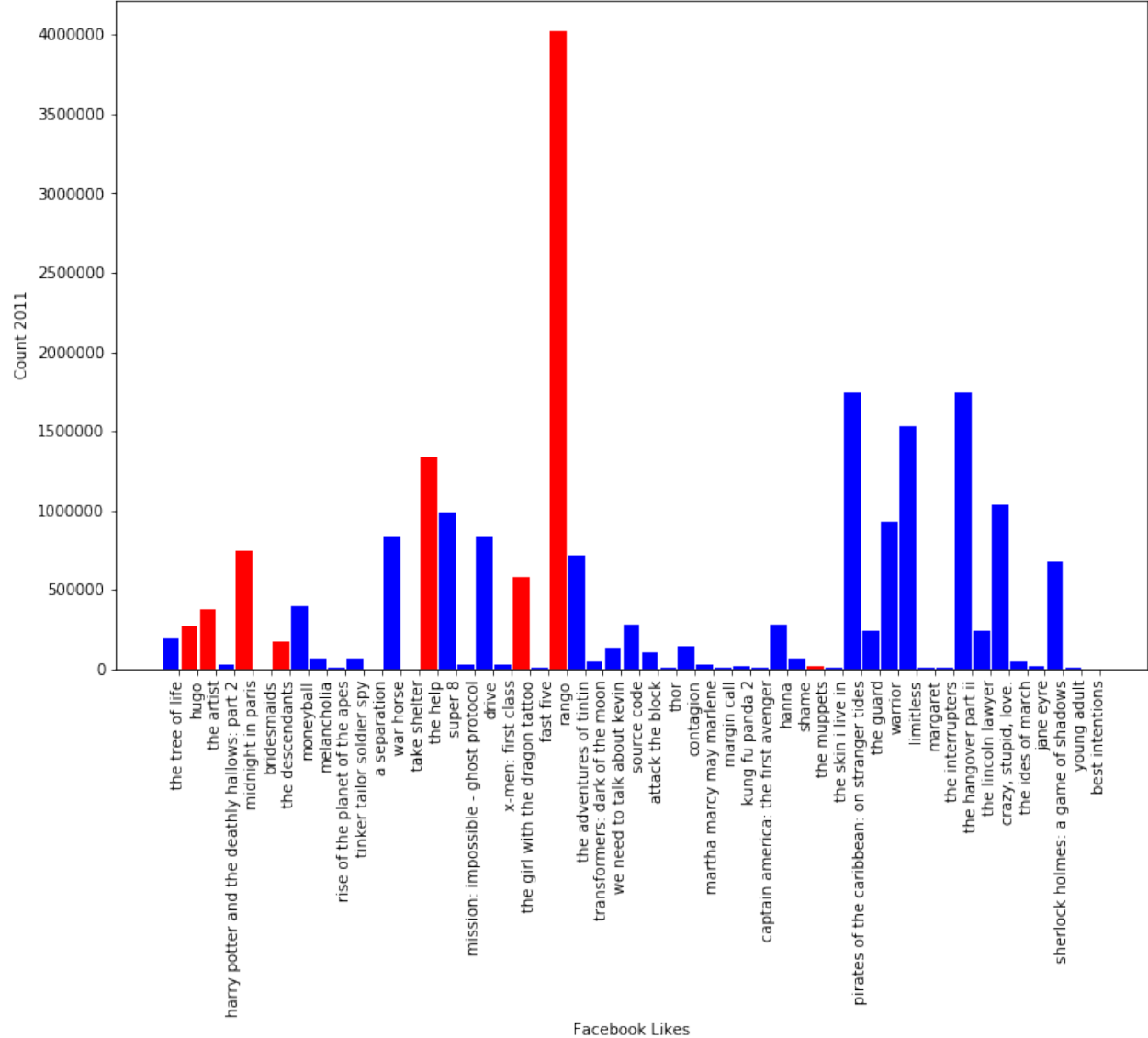


Figure Size [12.0, 8.0]
OSCAR MOVIES ARE HIGHLIGHTED WITH RED

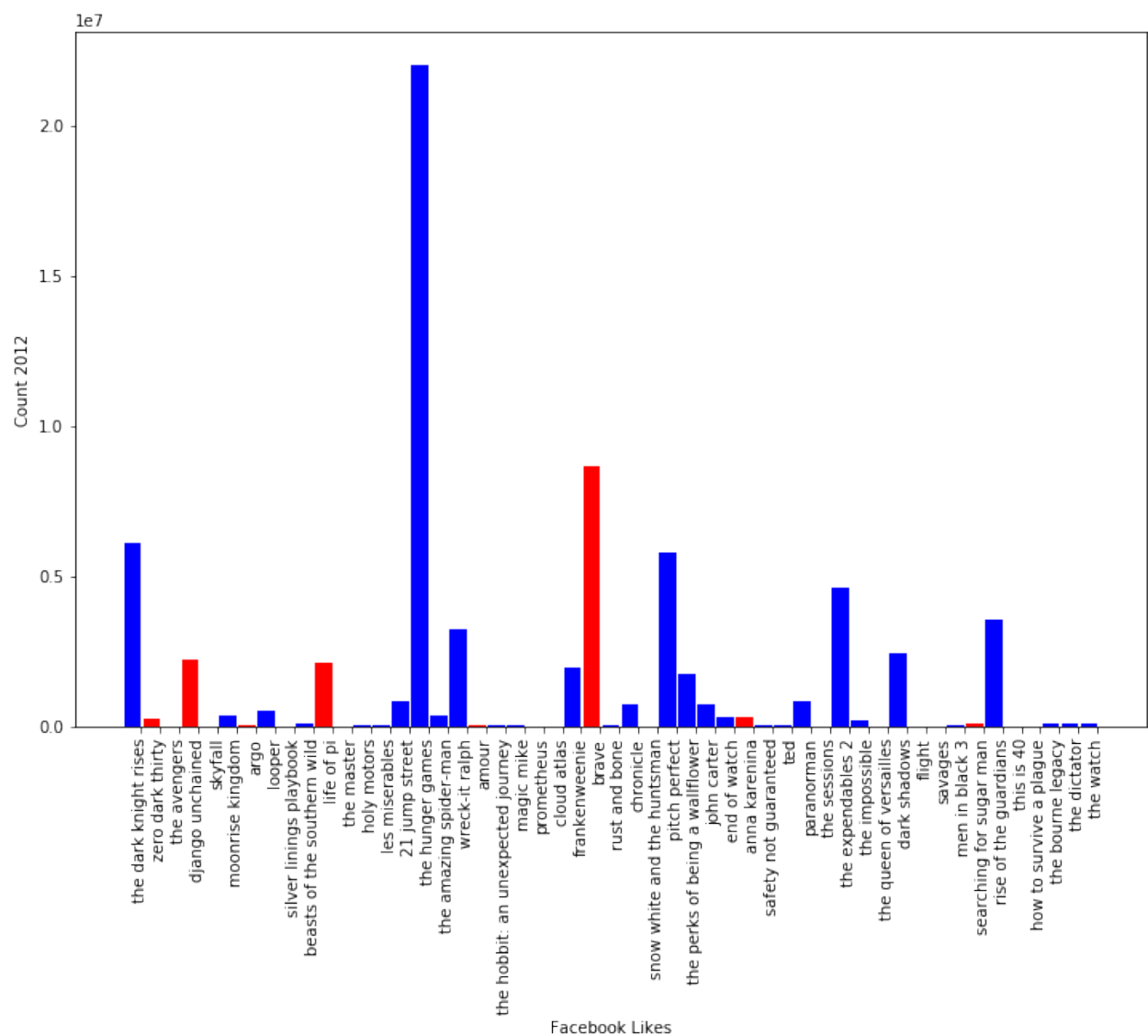


Figure Size [12.0, 8.0]
OSCAR MOVIES ARE HIGHLIGHTED WITH RED

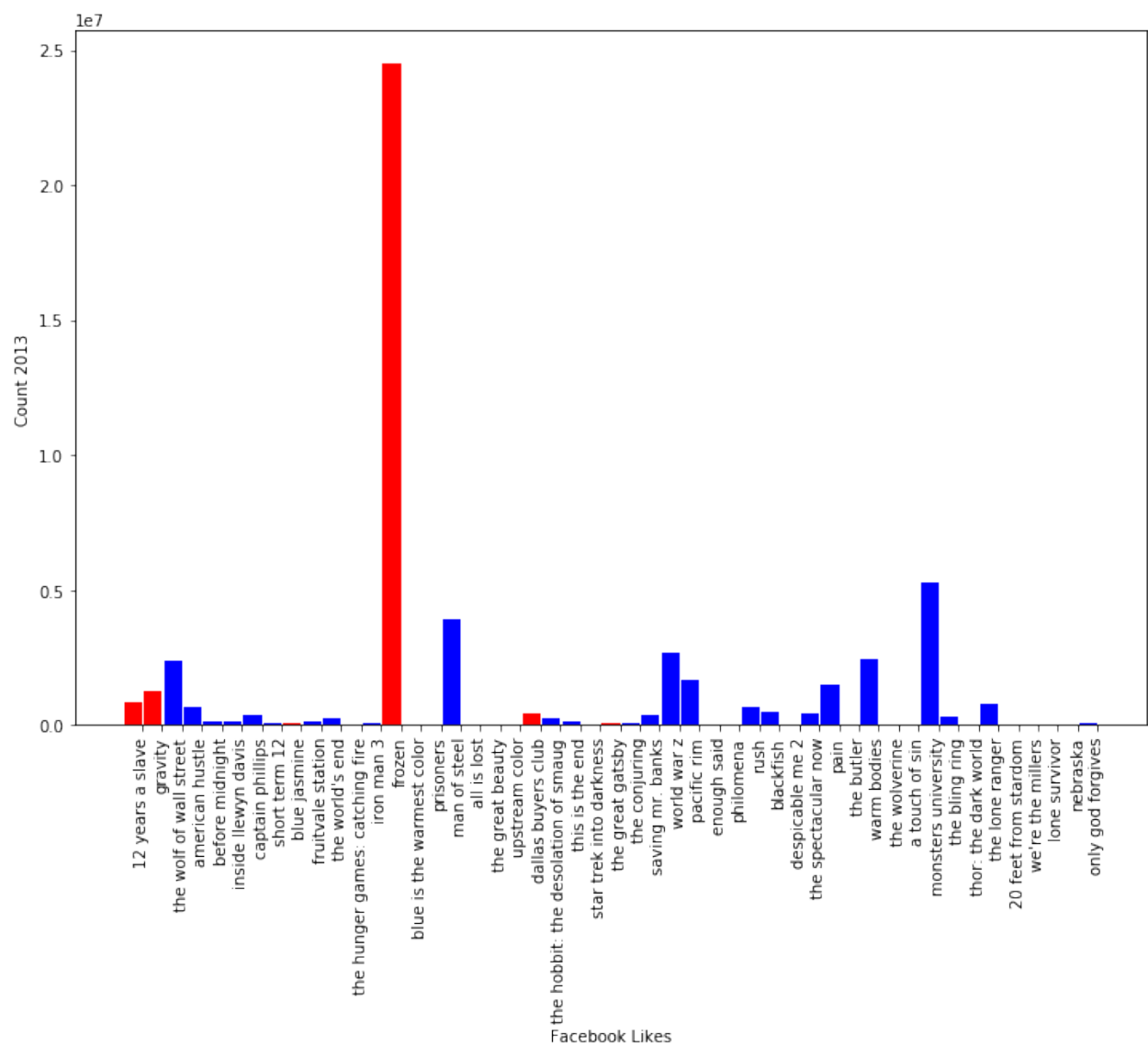


Figure Size [12.0, 8.0]
OSCAR MOVIES ARE HIGHLIGHTED WITH RED

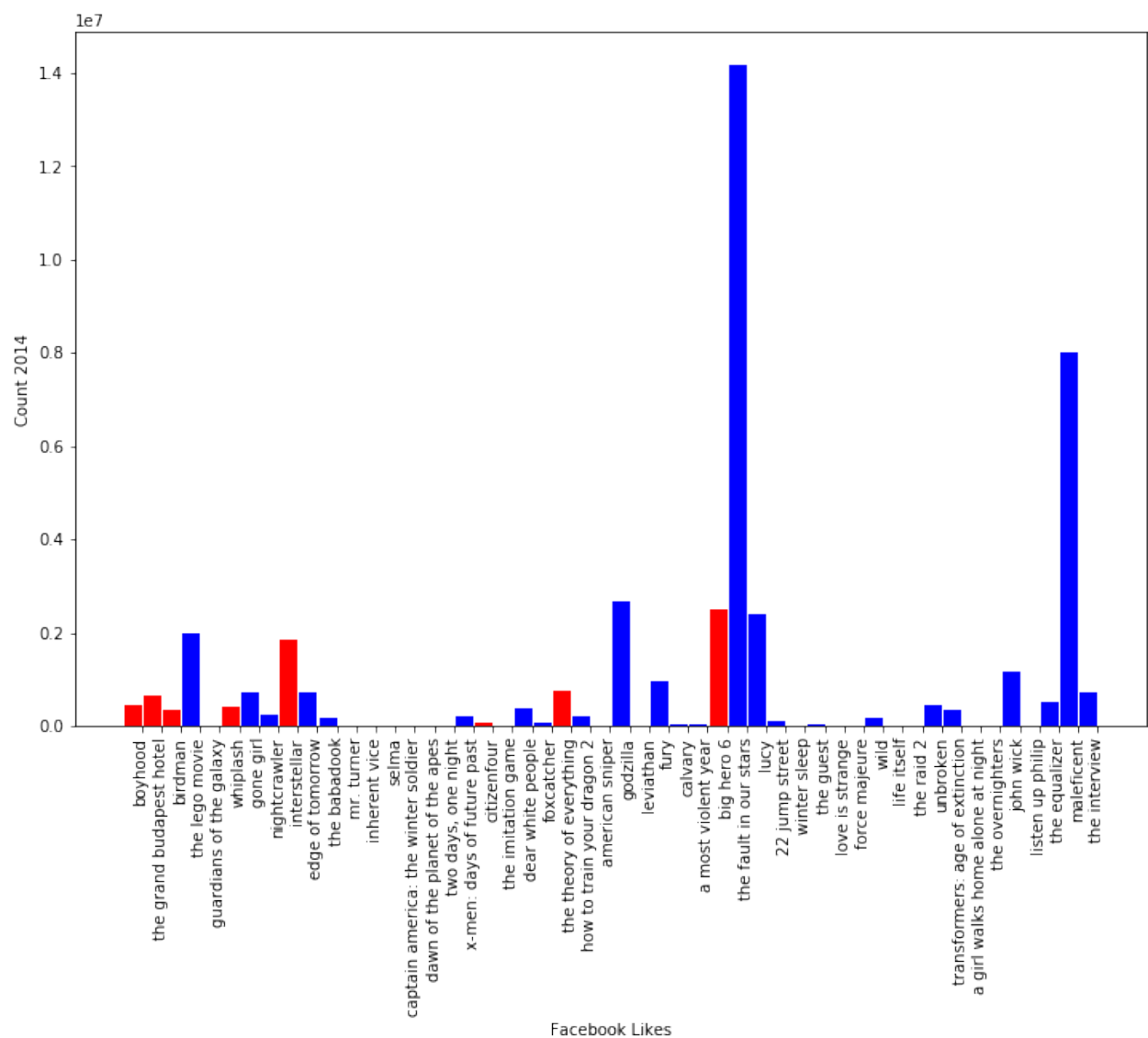


Figure Size [12.0, 8.0]
OSCAR MOVIES ARE HIGHLIGHTED WITH RED

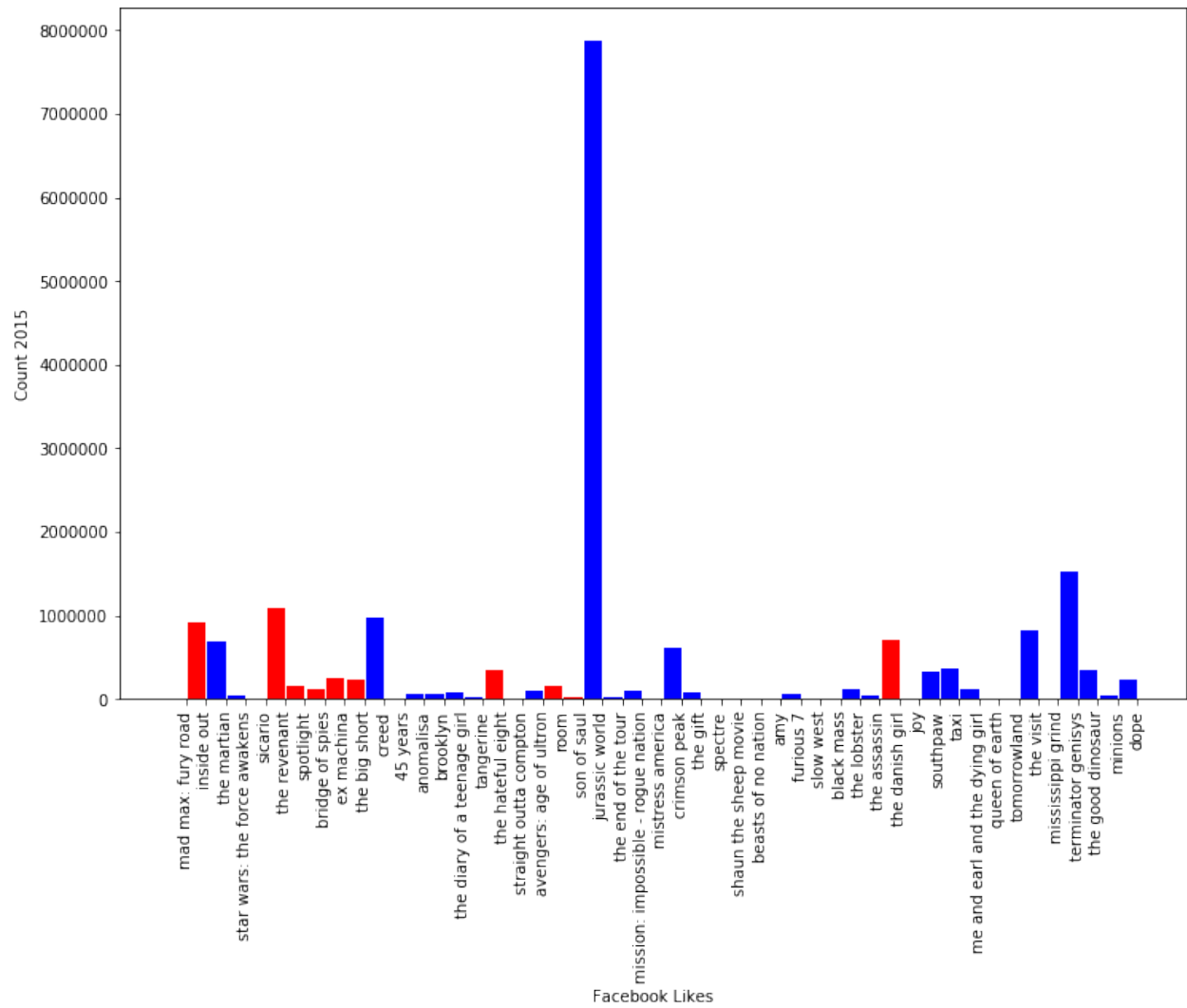
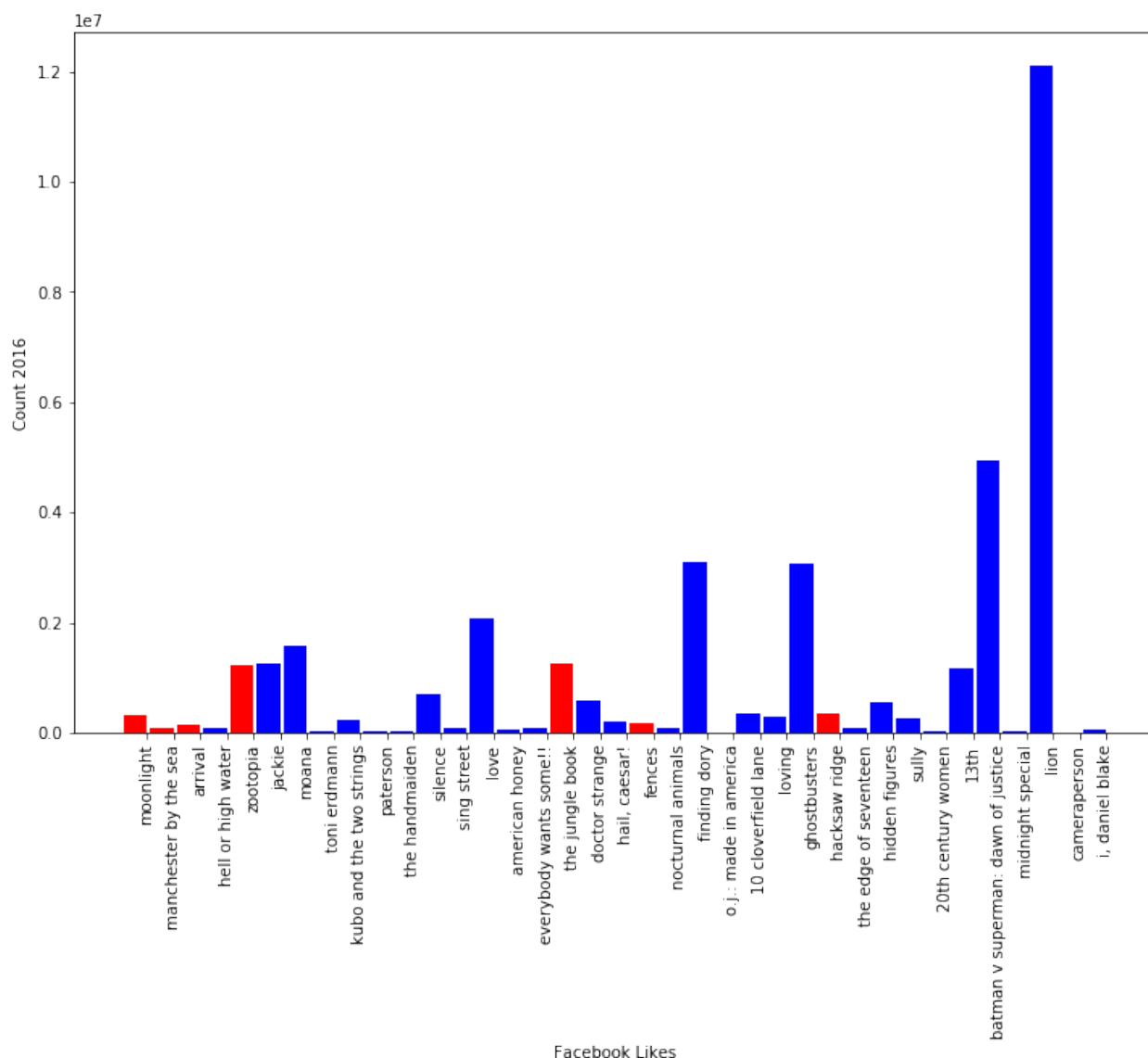


Figure Size [12.0, 8.0]
OSCAR MOVIES ARE HIGHLIGHTED WITH RED



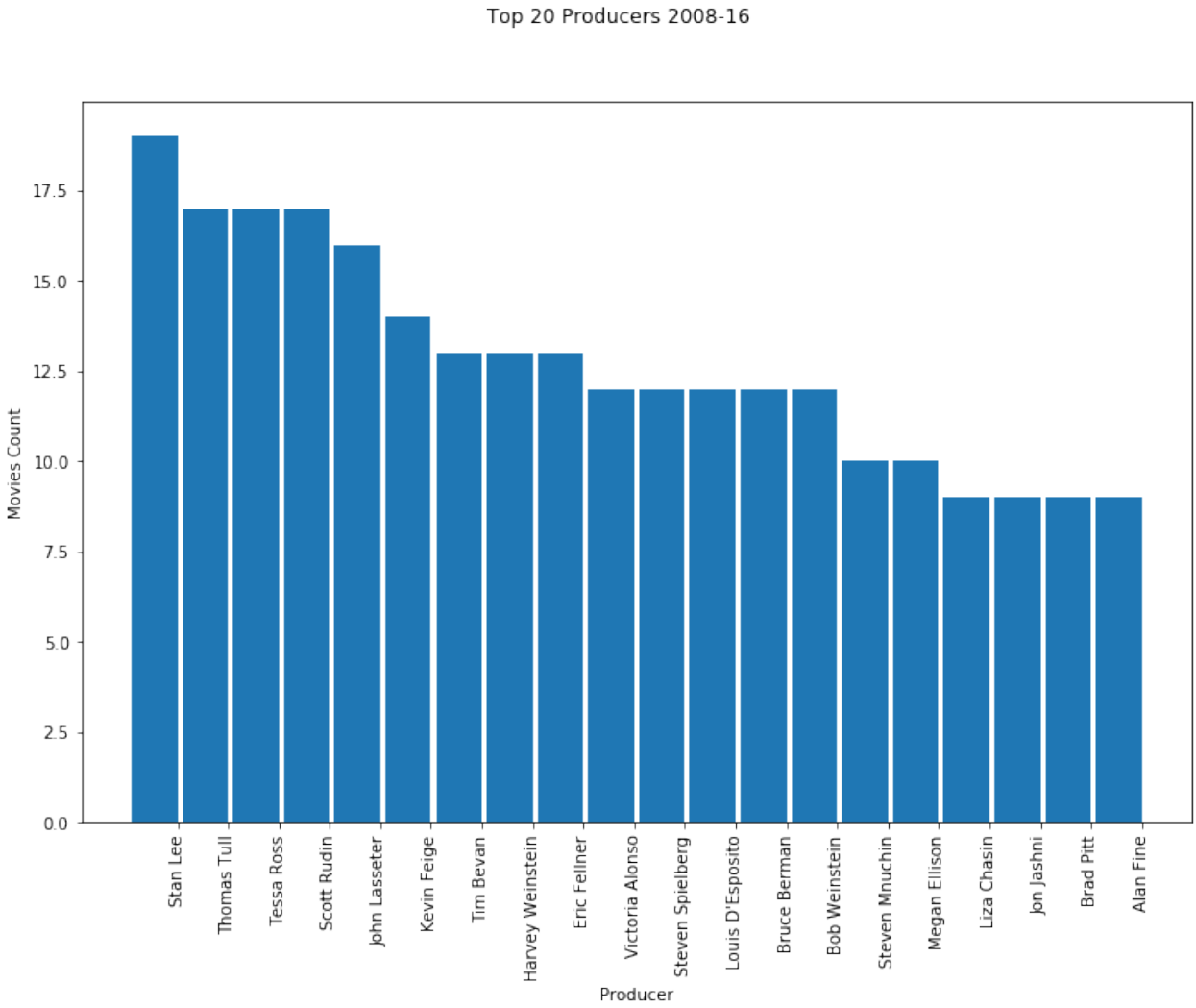
```
In [15]: #Concat IMDB score and ratings
years = yearlist(2008,2016)
for year in years:
    dic_google[year][['score','rating']] = dic_imdb[year][['score','rating']]
```

```
In [16]: years = yearlist(2008,2016)
for year in years:
    obj = dic_google[year]
    people = dic_people[year]
    obj['director'], obj['cast'], obj['producer'] = "", "", ""
    try:
        obj['director'] = people['people'].apply(lambda x : ",".join(map(lambda y : str(y[0].encode("utf8")), ast.literal_eval(x)['directors'] if 'directors' in ast.literal_eval(x) else [])))
        obj['cast'] = people['people'].apply(lambda x : ",".join(map(lambda y : str(y[0].encode("utf8")), ast.literal_eval(x)['cast'] if 'cast' in ast.literal_eval(x) else [])))
        obj['producer'] = people['people'].apply(lambda x : ",".join(map(lambda y : str(y[0].encode("utf8")), ast.literal_eval(x)['producers'] if 'producers' in ast.literal_eval(x) else [])))
```

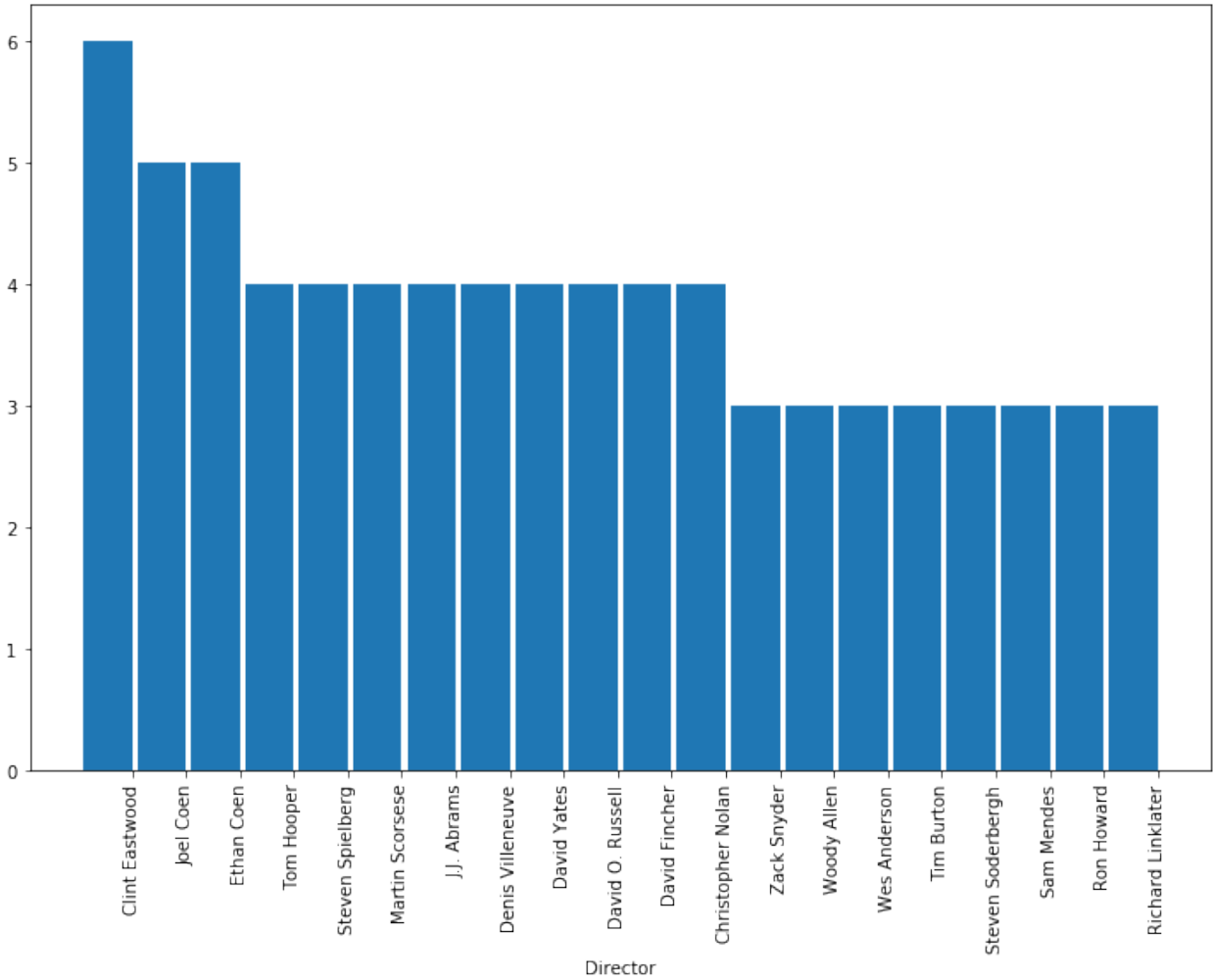
```
except KeyError:
    print year, "Key Error"
```

```
In [17]: #Top Actors, Producers and Directors
dic_actors, dic_directors, dic_producers = {}, {}, {}
width = .9
plt.ylabel('Movies Count')
years = yearlist(2008,2016)
obj_dic = dic_actors
for year in years:
    dic_google[year]['cast'].apply(add_dic)
dic_actors = sorted(dic_actors.iteritems(), key=lambda (k,v): (v,k), reverse=True)
obj_dic = dic_directors
for year in years:
    dic_google[year]['director'].apply(add_dic)
dic_directors = sorted(dic_directors.iteritems(), key=lambda (k,v): (v,k), reverse=True)
obj_dic = dic_producers
for year in years:
    dic_google[year]['producer'].apply(add_dic)
dic_producers = sorted(dic_producers.iteritems(), key=lambda (k,v): (v,k), reverse=True)

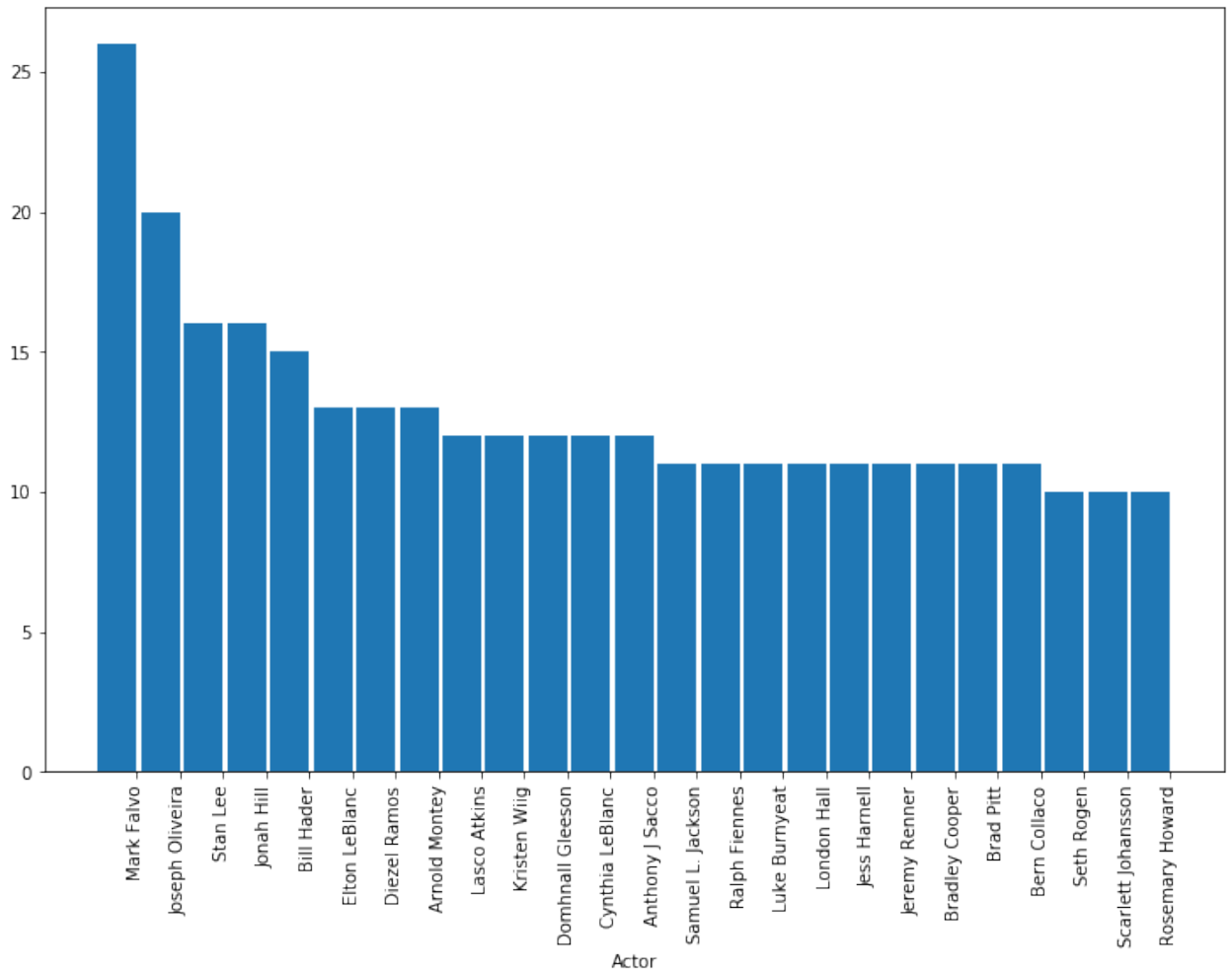
keys = map(lambda x : str(re.sub(r'^\x00-\x7F+', '', str(x[0]))), dic_producers[:20])
values = map(lambda x : x[1], dic_producers[:20])
indexes = np.arange(len(keys))
plt.suptitle("Top 20 Producers 2008-16")
plt.bar(indexes, values, width)
plt.xticks(indexes + width * 0.5, keys, rotation='vertical')
plt.xlabel('Producer')
plt.show()
keys = map(lambda x : str(re.sub(r'^\x00-\x7F+', '', str(x[0]))), dic_directors[:20])
values = map(lambda x : x[1], dic_directors[:20])
indexes = np.arange(len(keys))
plt.suptitle("Top 20 Directors 2008-16")
plt.bar(indexes, values, width)
plt.xticks(indexes + width * 0.5, keys, rotation='vertical')
plt.xlabel('Director')
plt.show()
keys = map(lambda x : str(re.sub(r'^\x00-\x7F+', '', str(x[0]))), dic_actors[:25])
values = map(lambda x : x[1], dic_actors[:25])
indexes = np.arange(len(keys))
plt.suptitle("Top 25 Actors 2008-16")
plt.bar(indexes, values, width)
plt.xticks(indexes + width * 0.5, keys, rotation='vertical')
plt.xlabel('Actor')
plt.show()
```



Top 20 Directors 2008-16



Top 25 Actors 2008-16



```
In [19]: #Review Sentiment Analysis
review_path = "./imdb-movies/reviews/"
tokenizer = RegexpTokenizer(r'\w+')
stemmer = WordNetLemmatizer()
years = yearlist(2008,2016)
for year in years:
    obj = dic_google[year]
    imdb = obj['imdb']
    obj['IMDB_review_sentiment_pos'] = 0
    obj['IMDB_review_sentiment_neg'] = 0
    obj['IMDB_review_helpful_metric'] = 0
    obj['IMDB_review_rating'] = 0
    obj['IMDB_review_neutral_metric'] = 0
    print ""
    print year,"started ..."
    for line in xrange(len(imdb)):
        #line = 47
        print obj.loc[line,'title'], "Review Sentiment Analysis Started"
        rfile = open(review_path + "movie-" + imdb[line], 'r')
        if len(rfile.readline()) > 0:
            review, user_score, helpful, t_people = [], "", 0, 0
            total_score, helpful_ratio, neutral_to_helpful_ratio = 0.0, 0.0, 0.0
```

```

review_status, line_count, review_count = True, 1, 0
fault_score, fault_helpful = 0, 0
for textline in rfile:
    if review_status:
        text = textline
        if "###" in str(text):
            #print text
            review_status = False
            line_count = 1
            '''
            if len(review[-1]) <= 8:
                user_score = review[-1][: -1]
                review = review[: -1]
            '''
            continue
        review.append(text)
    else:
        if line_count == 3:
            potential_score = textline
            if "None" not in potential_score:
                user_score = potential_score[: -1]
        elif line_count == 9:
            if "None" not in textline:
                #print user_score, textline
                helpful = int(textline[: -1])
            else:
                helpful = 0
        elif line_count == 11:
            if "None" not in textline:
                t_people = int(textline[: -1])
            else:
                t_people = 0
        elif line_count == 12:
            review_status = True
            review_count += 1
            #print ""
            #print review_count, review
            if len(user_score) == 0:
                fault_score += 1
            else:
                #print user_score
                total_score += float(user_score.split("/")[0])
            if helpful == 0 or t_people == 0:
                fault_helpful += 1
            else:
                helpful_ratio += float(float(helpful)/t_people
)
                neutral_to_helpful_ratio += float(float(t_people - helpful)/helpful)
                user_score, helpful, t_people = "", 0, 0
            line_count += 1
        #Summarize Reviews For Movie
        total_score += fault_score * (total_score / (review_count - fault_score))
        helpful_ratio += fault_helpful * (helpful_ratio / (review_count - fault_helpful))
        neutral_to_helpful_ratio += fault_helpful * (neutral_to_helpful_ratio / (review_count - fault_helpful))

```

```

l_ratio / (review_count - fault_helpful))
    count = 0
    for text in review:
        text = text.replace('\n','').lower()
        text = re.sub(r'[^x00-x7F]+',' ', text)
        text = re.sub('[^a-zA-Z ]+', ' ', text)
        text = " ".join([word for word in text.split(' ') if word
not in stopwords.words('english')])
        if len(text) > 0:
            review[count] = text
            count += 1
    review = [word for word in review if '\n' not in word]
    total_score = total_score / review_count
    helpful_ratio = helpful_ratio / review_count
    neutral_to_helpful_ratio = neutral_to_helpful_ratio / review_c
ount

    review = re.sub(' +', ' ', " ".join(review))
    review = tokenizer.tokenize(review)
    review = [str(stemmer.lemmatize(plural, get_wordnet_pos(nltk.po
s_tag([plural])[0]))) if get_wordnet_pos(nltk.pos_tag([plural])[0]) != 'X'
    else str(stemmer.lemmatize(plural)) for plural in review]
    review = map(lambda x : swn.senti_synsets(str(x), wordnet_sani
tize(nltk.pos_tag([x])[0])[1]), review)
    #Add New Features After Sentiment Analysis on Movie Reviews
    obj.loc[line, 'IMDB_review_sentiment_pos'] = sum(map(lambda x :
x[0].pos_score() if len(x) > 0 else 0, review))/len(review)
    obj.loc[line, 'IMDB_review_sentiment_neg'] = sum(map(lambda x :
x[0].neg_score() if len(x) > 0 else 0, review))/len(review)
    obj.loc[line, 'IMDB_review_helpful_metric'] = helpful_ratio
    obj.loc[line, 'IMDB_review_rating'] = total_score
    obj.loc[line, 'IMDB_review_neutral_metric'] = neutral_to_helpfu
l_ratio

    #print review, total_score, helpful_ratio, neutral_to_helpful_
ratio

```

2008 started ...
 wall.e Review Sentiment Analysis Started
 the dark knight Review Sentiment Analysis Started
 slumdog millionaire Review Sentiment Analysis Started
 iron man Review Sentiment Analysis Started
 the curious case of benjamin button Review Sentiment Analysis Started
 gran torino Review Sentiment Analysis Started
 indiana jones and the kingdom of the crystal skull Review Sentiment Analys
 is Started
 the wrestler Review Sentiment Analysis Started
 quantum of solace Review Sentiment Analysis Started
 kung fu panda Review Sentiment Analysis Started
 let the right one in Review Sentiment Analysis Started
 burn after reading Review Sentiment Analysis Started
 the black list: volume one Review Sentiment Analysis Started
 cloverfield Review Sentiment Analysis Started
 frost/nixon Review Sentiment Analysis Started
 doubt Review Sentiment Analysis Started
 tropic thunder Review Sentiment Analysis Started
 man on wire Review Sentiment Analysis Started
 hancock Review Sentiment Analysis Started
 vicky cristina barcelona Review Sentiment Analysis Started

milk Review Sentiment Analysis Started
 revolutionary road Review Sentiment Analysis Started
 rachel getting married Review Sentiment Analysis Started
 happy-go-lucky Review Sentiment Analysis Started
 synecdoche, new york Review Sentiment Analysis Started
 hellboy ii: the golden army Review Sentiment Analysis Started
 the reader Review Sentiment Analysis Started
 in bruges Review Sentiment Analysis Started
 changeling Review Sentiment Analysis Started
 horton hears a who! Review Sentiment Analysis Started
 forgetting sarah marshall Review Sentiment Analysis Started
 the bank job Review Sentiment Analysis Started
 pineapple express Review Sentiment Analysis Started
 waltz with bashir Review Sentiment Analysis Started
 twilight Review Sentiment Analysis Started
 the incredible hulk Review Sentiment Analysis Started
 bolt Review Sentiment Analysis Started
 seven pounds Review Sentiment Analysis Started
 mamma mia! Review Sentiment Analysis Started
 frozen river Review Sentiment Analysis Started
 sex and the city Review Sentiment Analysis Started
 speed racer Review Sentiment Analysis Started
 the hurt locker Review Sentiment Analysis Started
 the happening Review Sentiment Analysis Started
 defiance Review Sentiment Analysis Started
 step brothers Review Sentiment Analysis Started
 i've loved you so long Review Sentiment Analysis Started
 dear zachary: a letter to a son about his father Review Sentiment Analysis Started
 role models Review Sentiment Analysis Started
 madagascar: escape 2 africa Review Sentiment Analysis Started
 wanted Review Sentiment Analysis Started

2009 started ...

avatar Review Sentiment Analysis Started
 inglourious basterds Review Sentiment Analysis Started
 district 9 Review Sentiment Analysis Started
 the hangover Review Sentiment Analysis Started
 fantastic mr. fox Review Sentiment Analysis Started
 up in the air Review Sentiment Analysis Started
 star trek Review Sentiment Analysis Started
 zombieland Review Sentiment Analysis Started
 (500) days of summer Review Sentiment Analysis Started
 coraline Review Sentiment Analysis Started
 harry potter and the half-blood prince Review Sentiment Analysis Started
 watchmen Review Sentiment Analysis Started
 the blind side Review Sentiment Analysis Started
 a serious man Review Sentiment Analysis Started
 precious Review Sentiment Analysis Started
 sherlock holmes Review Sentiment Analysis Started
 transformers: revenge of the fallen Review Sentiment Analysis Started
 public enemies Review Sentiment Analysis Started
 where the wild things are Review Sentiment Analysis Started
 an education Review Sentiment Analysis Started
 a single man Review Sentiment Analysis Started
 the princess and the frog Review Sentiment Analysis Started
 moon Review Sentiment Analysis Started

the white ribbon Review Sentiment Analysis Started
up Review Sentiment Analysis Started
invictus Review Sentiment Analysis Started
the twilight saga: new moon Review Sentiment Analysis Started
the informant! Review Sentiment Analysis Started
drag me to hell Review Sentiment Analysis Started
crazy heart Review Sentiment Analysis Started
adventureland Review Sentiment Analysis Started
the proposal Review Sentiment Analysis Started
in the loop Review Sentiment Analysis Started
the lovely bones Review Sentiment Analysis Started
the messenger Review Sentiment Analysis Started
broken embraces Review Sentiment Analysis Started
julie & julia Review Sentiment Analysis Started
ice age: dawn of the dinosaurs Review Sentiment Analysis Started
i love you, man Review Sentiment Analysis Started
x-men origins: wolverine Review Sentiment Analysis Started
this is it Review Sentiment Analysis Started
state of play Review Sentiment Analysis Started
the damned united Review Sentiment Analysis Started
the cove Review Sentiment Analysis Started
funny people Review Sentiment Analysis Started
he's just not that into you Review Sentiment Analysis Started
law abiding citizen Review Sentiment Analysis Started
the bad lieutenant: port of call - new orleans Review Sentiment Analysis S
tarted
the boat that rocked Review Sentiment Analysis Started
angels & demons Review Sentiment Analysis Started

2010 started ...

inception Review Sentiment Analysis Started
toy story 3 Review Sentiment Analysis Started
the social network Review Sentiment Analysis Started
black swan Review Sentiment Analysis Started
shutter island Review Sentiment Analysis Started
the king's speech Review Sentiment Analysis Started
true grit Review Sentiment Analysis Started
how to train your dragon Review Sentiment Analysis Started
scott pilgrim vs. the world Review Sentiment Analysis Started
127 hours Review Sentiment Analysis Started
the fighter Review Sentiment Analysis Started
winter's bone Review Sentiment Analysis Started
kick-ass Review Sentiment Analysis Started
blue valentine Review Sentiment Analysis Started
tangled Review Sentiment Analysis Started
the town Review Sentiment Analysis Started
despicable me Review Sentiment Analysis Started
exit through the gift shop Review Sentiment Analysis Started
the ghost writer Review Sentiment Analysis Started
iron man 2 Review Sentiment Analysis Started
let me in Review Sentiment Analysis Started
harry potter and the deathly hallows: part 1 Review Sentiment Analysis Sta
rted
the kids are all right Review Sentiment Analysis Started
another year Review Sentiment Analysis Started
easy a Review Sentiment Analysis Started
unstoppable Review Sentiment Analysis Started

alice in wonderland Review Sentiment Analysis Started
 rabbit hole Review Sentiment Analysis Started
 greenberg Review Sentiment Analysis Started
 animal kingdom Review Sentiment Analysis Started
 the other guys Review Sentiment Analysis Started
 tron: legacy Review Sentiment Analysis Started
 restrepo Review Sentiment Analysis Started
 inside job Review Sentiment Analysis Started
 the illusionist Review Sentiment Analysis Started
 never let me go Review Sentiment Analysis Started
 four lions Review Sentiment Analysis Started
 buried Review Sentiment Analysis Started
 hereafter Review Sentiment Analysis Started
 clash of the titans Review Sentiment Analysis Started
 the a-team Review Sentiment Analysis Started
 megamind Review Sentiment Analysis Started
 predators Review Sentiment Analysis Started
 the book of eli Review Sentiment Analysis Started
 machete Review Sentiment Analysis Started
 please give Review Sentiment Analysis Started
 jackass 3d Review Sentiment Analysis Started
 the expendables Review Sentiment Analysis Started
 joan rivers: a piece of work Review Sentiment Analysis Started
 the last airbender Review Sentiment Analysis Started
 incendies Review Sentiment Analysis Started

2011 started ...

the tree of life Review Sentiment Analysis Started
 hugo Review Sentiment Analysis Started
 the artist Review Sentiment Analysis Started
 harry potter and the deathly hallows: part 2 Review Sentiment Analysis Started
 midnight in paris Review Sentiment Analysis Started
 bridesmaids Review Sentiment Analysis Started
 the descendants Review Sentiment Analysis Started
 moneyball Review Sentiment Analysis Started
 melancholia Review Sentiment Analysis Started
 rise of the planet of the apes Review Sentiment Analysis Started
 tinker tailor soldier spy Review Sentiment Analysis Started
 a separation Review Sentiment Analysis Started
 war horse Review Sentiment Analysis Started
 take shelter Review Sentiment Analysis Started
 the help Review Sentiment Analysis Started
 super 8 Review Sentiment Analysis Started
 mission: impossible - ghost protocol Review Sentiment Analysis Started
 drive Review Sentiment Analysis Started
 x-men: first class Review Sentiment Analysis Started
 the girl with the dragon tattoo Review Sentiment Analysis Started
 fast five Review Sentiment Analysis Started
 rango Review Sentiment Analysis Started
 the adventures of tintin Review Sentiment Analysis Started
 transformers: dark of the moon Review Sentiment Analysis Started
 we need to talk about kevin Review Sentiment Analysis Started
 source code Review Sentiment Analysis Started
 attack the block Review Sentiment Analysis Started
 thor Review Sentiment Analysis Started
 contagion Review Sentiment Analysis Started

martha marcy may marlene Review Sentiment Analysis Started
margin call Review Sentiment Analysis Started
kung fu panda 2 Review Sentiment Analysis Started
captain america: the first avenger Review Sentiment Analysis Started
hanna Review Sentiment Analysis Started
shame Review Sentiment Analysis Started
the muppets Review Sentiment Analysis Started
the skin i live in Review Sentiment Analysis Started
pirates of the caribbean: on stranger tides Review Sentiment Analysis Star
ted
the guard Review Sentiment Analysis Started
warrior Review Sentiment Analysis Started
limitless Review Sentiment Analysis Started
margaret Review Sentiment Analysis Started
the interrupters Review Sentiment Analysis Started
the hangover part ii Review Sentiment Analysis Started
the lincoln lawyer Review Sentiment Analysis Started
crazy, stupid, love. Review Sentiment Analysis Started
the ides of march Review Sentiment Analysis Started
jane eyre Review Sentiment Analysis Started
sherlock holmes: a game of shadows Review Sentiment Analysis Started
young adult Review Sentiment Analysis Started
best intentions Review Sentiment Analysis Started

2012 started ...

the dark knight rises Review Sentiment Analysis Started
zero dark thirty Review Sentiment Analysis Started
the avengers Review Sentiment Analysis Started
django unchained Review Sentiment Analysis Started
skyfall Review Sentiment Analysis Started
moonrise kingdom Review Sentiment Analysis Started
argo Review Sentiment Analysis Started
looper Review Sentiment Analysis Started
silver linings playbook Review Sentiment Analysis Started
beasts of the southern wild Review Sentiment Analysis Started
life of pi Review Sentiment Analysis Started
the master Review Sentiment Analysis Started
holy motors Review Sentiment Analysis Started
les misrables Review Sentiment Analysis Started
21 jump street Review Sentiment Analysis Started
the hunger games Review Sentiment Analysis Started
the amazing spider-man Review Sentiment Analysis Started
wreck-it ralph Review Sentiment Analysis Started
amour Review Sentiment Analysis Started
the hobbit: an unexpected journey Review Sentiment Analysis Started
magic mike Review Sentiment Analysis Started
prometheus Review Sentiment Analysis Started
cloud atlas Review Sentiment Analysis Started
frankenweenie Review Sentiment Analysis Started
brave Review Sentiment Analysis Started
rust and bone Review Sentiment Analysis Started
chronicle Review Sentiment Analysis Started
snow white and the huntsman Review Sentiment Analysis Started
pitch perfect Review Sentiment Analysis Started
the perks of being a wallflower Review Sentiment Analysis Started
john carter Review Sentiment Analysis Started
end of watch Review Sentiment Analysis Started

anna karenina Review Sentiment Analysis Started
 safety not guaranteed Review Sentiment Analysis Started
 ted Review Sentiment Analysis Started
 paranorman Review Sentiment Analysis Started
 the sessions Review Sentiment Analysis Started
 the expendables 2 Review Sentiment Analysis Started
 the impossible Review Sentiment Analysis Started
 the queen of versailles Review Sentiment Analysis Started
 dark shadows Review Sentiment Analysis Started
 flight Review Sentiment Analysis Started
 savages Review Sentiment Analysis Started
 men in black 3 Review Sentiment Analysis Started
 searching for sugar man Review Sentiment Analysis Started
 rise of the guardians Review Sentiment Analysis Started
 this is 40 Review Sentiment Analysis Started
 how to survive a plague Review Sentiment Analysis Started
 the bourne legacy Review Sentiment Analysis Started
 the dictator Review Sentiment Analysis Started
 the watch Review Sentiment Analysis Started

2013 started ...

12 years a slave Review Sentiment Analysis Started
 gravity Review Sentiment Analysis Started
 the wolf of wall street Review Sentiment Analysis Started
 american hustle Review Sentiment Analysis Started
 before midnight Review Sentiment Analysis Started
 inside llewyn davis Review Sentiment Analysis Started
 captain phillips Review Sentiment Analysis Started
 short term 12 Review Sentiment Analysis Started
 blue jasmine Review Sentiment Analysis Started
 fruitvale station Review Sentiment Analysis Started
 the world's end Review Sentiment Analysis Started
 the hunger games: catching fire Review Sentiment Analysis Started
 iron man 3 Review Sentiment Analysis Started
 frozen Review Sentiment Analysis Started
 blue is the warmest color Review Sentiment Analysis Started
 prisoners Review Sentiment Analysis Started
 man of steel Review Sentiment Analysis Started
 all is lost Review Sentiment Analysis Started
 the great beauty Review Sentiment Analysis Started
 upstream color Review Sentiment Analysis Started
 dallas buyers club Review Sentiment Analysis Started
 the hobbit: the desolation of smaug Review Sentiment Analysis Started
 this is the end Review Sentiment Analysis Started
 star trek into darkness Review Sentiment Analysis Started
 the great gatsby Review Sentiment Analysis Started
 the conjuring Review Sentiment Analysis Started
 saving mr. banks Review Sentiment Analysis Started
 world war z Review Sentiment Analysis Started
 pacific rim Review Sentiment Analysis Started
 enough said Review Sentiment Analysis Started
 philomena Review Sentiment Analysis Started
 rush Review Sentiment Analysis Started
 blackfish Review Sentiment Analysis Started
 despicable me 2 Review Sentiment Analysis Started
 the spectacular now Review Sentiment Analysis Started
 pain Review Sentiment Analysis Started

the butler Review Sentiment Analysis Started
 warm bodies Review Sentiment Analysis Started
 hold fast Review Sentiment Analysis Started
 the wolverine Review Sentiment Analysis Started
 a touch of sin Review Sentiment Analysis Started
 monsters university Review Sentiment Analysis Started
 the bling ring Review Sentiment Analysis Started
 the kings of summer Review Sentiment Analysis Started
 thor: the dark world Review Sentiment Analysis Started
 the lone ranger Review Sentiment Analysis Started
 20 feet from stardom Review Sentiment Analysis Started
 we're the millers Review Sentiment Analysis Started
 lone survivor Review Sentiment Analysis Started
 nebraska Review Sentiment Analysis Started
 only god forgives Review Sentiment Analysis Started

2014 started ...

boyhood Review Sentiment Analysis Started
 the grand budapest hotel Review Sentiment Analysis Started
 birdman Review Sentiment Analysis Started
 the lego movie Review Sentiment Analysis Started
 guardians of the galaxy Review Sentiment Analysis Started
 whiplash Review Sentiment Analysis Started
 gone girl Review Sentiment Analysis Started
 nightcrawler Review Sentiment Analysis Started
 interstellar Review Sentiment Analysis Started
 edge of tomorrow Review Sentiment Analysis Started
 the babadook Review Sentiment Analysis Started
 mr. turner Review Sentiment Analysis Started
 inherent vice Review Sentiment Analysis Started
 selma Review Sentiment Analysis Started
 captain america: the winter soldier Review Sentiment Analysis Started
 dawn of the planet of the apes Review Sentiment Analysis Started
 two days, one night Review Sentiment Analysis Started
 x-men: days of future past Review Sentiment Analysis Started
 citizenfour Review Sentiment Analysis Started
 the imitation game Review Sentiment Analysis Started
 dear white people Review Sentiment Analysis Started
 foxcatcher Review Sentiment Analysis Started
 the theory of everything Review Sentiment Analysis Started
 how to train your dragon 2 Review Sentiment Analysis Started
 american sniper Review Sentiment Analysis Started
 godzilla Review Sentiment Analysis Started
 leviathan Review Sentiment Analysis Started
 fury Review Sentiment Analysis Started
 calvary Review Sentiment Analysis Started
 a most violent year Review Sentiment Analysis Started
 big hero 6 Review Sentiment Analysis Started
 the fault in our stars Review Sentiment Analysis Started
 lucy Review Sentiment Analysis Started
 22 jump street Review Sentiment Analysis Started
 winter sleep Review Sentiment Analysis Started
 the guest Review Sentiment Analysis Started
 love is strange Review Sentiment Analysis Started
 force majeure Review Sentiment Analysis Started
 wild Review Sentiment Analysis Started
 life itself Review Sentiment Analysis Started

the raid 2 Review Sentiment Analysis Started
 unbroken Review Sentiment Analysis Started
 transformers: age of extinction Review Sentiment Analysis Started
 a girl walks home alone at night Review Sentiment Analysis Started
 the overnights Review Sentiment Analysis Started
 john wick Review Sentiment Analysis Started
 listen up philip Review Sentiment Analysis Started
 frank Review Sentiment Analysis Started
 the equalizer Review Sentiment Analysis Started
 maleficent Review Sentiment Analysis Started
 the interview Review Sentiment Analysis Started

2015 started ...

mad max: fury road Review Sentiment Analysis Started
 inside out Review Sentiment Analysis Started
 the martian Review Sentiment Analysis Started
 star wars: the force awakens Review Sentiment Analysis Started
 sicario Review Sentiment Analysis Started
 carol Review Sentiment Analysis Started
 the revenant Review Sentiment Analysis Started
 spotlight Review Sentiment Analysis Started
 bridge of spies Review Sentiment Analysis Started
 ex machina Review Sentiment Analysis Started
 the big short Review Sentiment Analysis Started
 creed Review Sentiment Analysis Started
 45 years Review Sentiment Analysis Started
 anomalisa Review Sentiment Analysis Started
 brooklyn Review Sentiment Analysis Started
 the diary of a teenage girl Review Sentiment Analysis Started
 tangerine Review Sentiment Analysis Started
 the hateful eight Review Sentiment Analysis Started
 straight outta compton Review Sentiment Analysis Started
 avengers: age of ultron Review Sentiment Analysis Started
 room Review Sentiment Analysis Started
 son of saul Review Sentiment Analysis Started
 jurassic world Review Sentiment Analysis Started
 the end of the tour Review Sentiment Analysis Started
 mission: impossible - rogue nation Review Sentiment Analysis Started
 mistress america Review Sentiment Analysis Started
 crimson peak Review Sentiment Analysis Started
 the gift Review Sentiment Analysis Started
 spectre Review Sentiment Analysis Started
 shaun the sheep movie Review Sentiment Analysis Started
 beasts of no nation Review Sentiment Analysis Started
 amy Review Sentiment Analysis Started
 furious 7 Review Sentiment Analysis Started
 slow west Review Sentiment Analysis Started
 black mass Review Sentiment Analysis Started
 trainwreck Review Sentiment Analysis Started
 the lobster Review Sentiment Analysis Started
 the assassin Review Sentiment Analysis Started
 the danish girl Review Sentiment Analysis Started
 joy Review Sentiment Analysis Started
 southpaw Review Sentiment Analysis Started
 taxi Review Sentiment Analysis Started
 me and earl and the dying girl Review Sentiment Analysis Started
 queen of earth Review Sentiment Analysis Started

tomorrowland Review Sentiment Analysis Started
the visit Review Sentiment Analysis Started
mississippi grind Review Sentiment Analysis Started
terminator genisys Review Sentiment Analysis Started
the good dinosaur Review Sentiment Analysis Started
minions Review Sentiment Analysis Started
dope Review Sentiment Analysis Started

2016 started ...

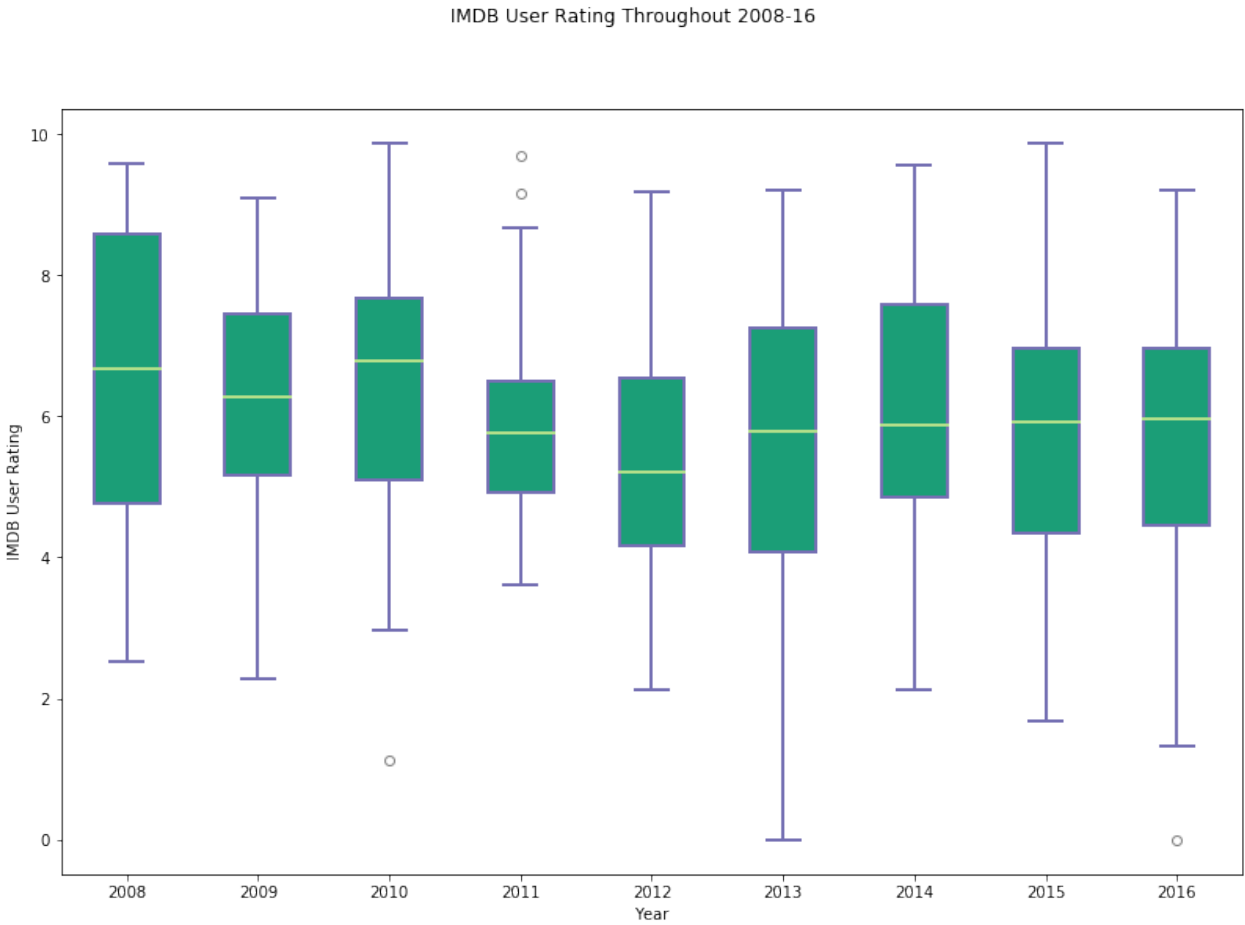
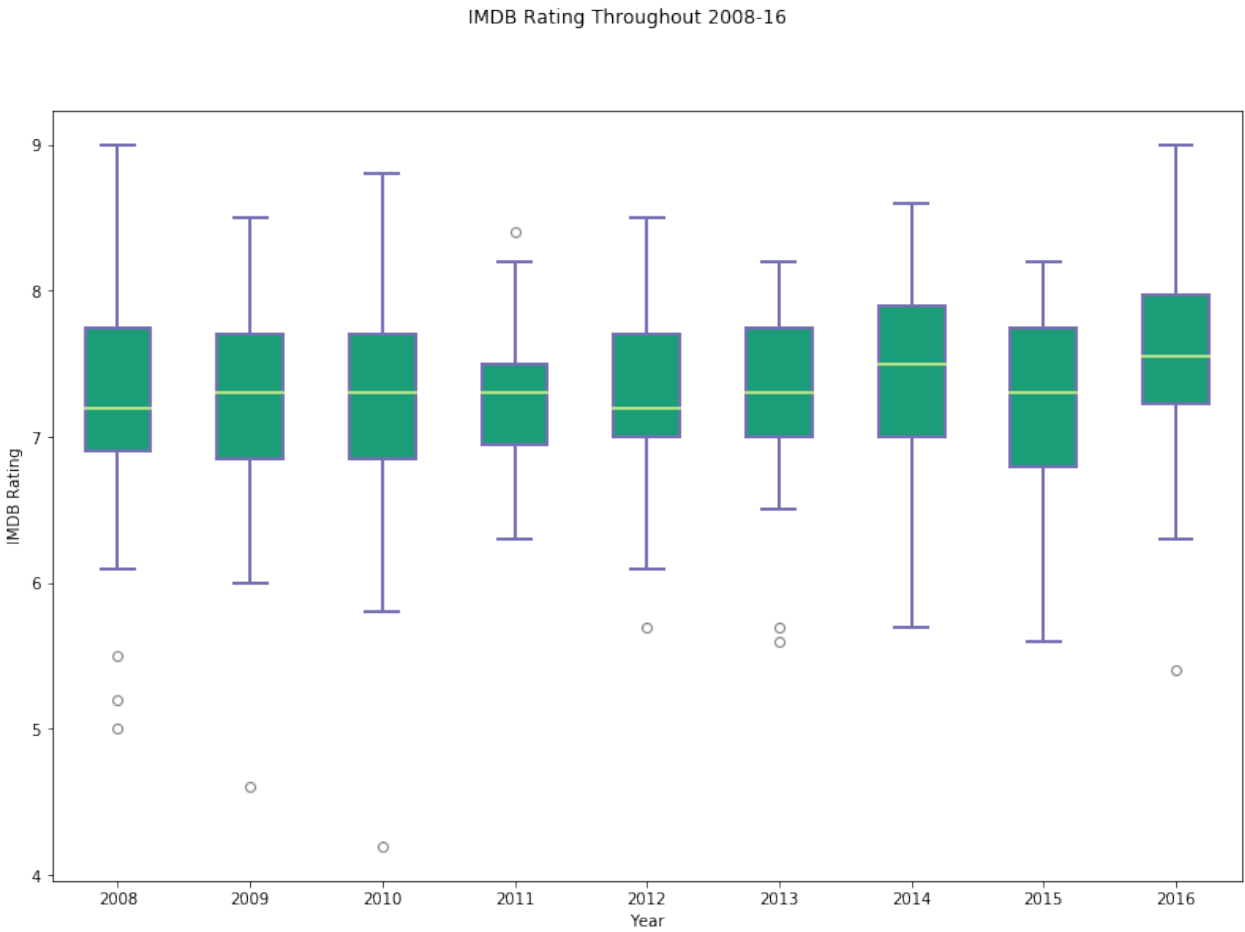
moonlight Review Sentiment Analysis Started
la la land Review Sentiment Analysis Started
manchester by the sea Review Sentiment Analysis Started
arrival Review Sentiment Analysis Started
hell or high water Review Sentiment Analysis Started
deadpool Review Sentiment Analysis Started
zootopia Review Sentiment Analysis Started
jackie Review Sentiment Analysis Started
moana Review Sentiment Analysis Started
rogue one: a star wars story Review Sentiment Analysis Started
elle Review Sentiment Analysis Started
toni erdmann Review Sentiment Analysis Started
weiner Review Sentiment Analysis Started
captain america: civil war Review Sentiment Analysis Started
kubo and the two strings Review Sentiment Analysis Started
paterson Review Sentiment Analysis Started
the handmaiden Review Sentiment Analysis Started
silence Review Sentiment Analysis Started
sing street Review Sentiment Analysis Started
love Review Sentiment Analysis Started
the nice guys Review Sentiment Analysis Started
hunt for the wilderpeople Review Sentiment Analysis Started
american honey Review Sentiment Analysis Started
everybody wants some!! Review Sentiment Analysis Started
the jungle book Review Sentiment Analysis Started
doctor strange Review Sentiment Analysis Started
hail, caesar! Review Sentiment Analysis Started
fences Review Sentiment Analysis Started
nocturnal animals Review Sentiment Analysis Started
finding dory Review Sentiment Analysis Started
o.j.: made in america Review Sentiment Analysis Started
10 cloverfield lane Review Sentiment Analysis Started
loving Review Sentiment Analysis Started
ghostbusters Review Sentiment Analysis Started
hacksaw ridge Review Sentiment Analysis Started
things to come Review Sentiment Analysis Started
the edge of seventeen Review Sentiment Analysis Started
hidden figures Review Sentiment Analysis Started
sully Review Sentiment Analysis Started
suicide squad Review Sentiment Analysis Started
20th century women Review Sentiment Analysis Started
13th Review Sentiment Analysis Started
batman v superman: dawn of justice Review Sentiment Analysis Started
midnight special Review Sentiment Analysis Started
lion Review Sentiment Analysis Started
sausage party Review Sentiment Analysis Started
fantastic beasts and where to find them Review Sentiment Analysis Started
certain women Review Sentiment Analysis Started

cameraperson Review Sentiment Analysis Started
i, daniel blake Review Sentiment Analysis Started

```
In [20]: #Join year Wise Dataset
final_google = pd.concat([dic_google[year] for year in yearlist(2008,2016)
])
```

```
In [21]: #IMDB User Rating VS Year
year_score, year_user_score = [], []
years = yearlist(2008,2016)
for year in years:
    obj = dic_google[year]
    year_score.append(list(obj['score']))
    year_user_score.append(list(obj['IMDB_review_rating']))
indexes = np.arange(len(years))
fig = plt.figure(1, figsize=(13.5, 9))
ax = fig.add_subplot(111)
bp = ax.boxplot(year_score, patch_artist=True)
boxplot_color(bp)
plt.suptitle("IMDB Rating Throughout 2008-16")
ax.set_xticklabels(years)
plt.ylabel('IMDB Rating')
plt.xlabel('Year')
plt.show()

fig = plt.figure(2, figsize=(13.5, 9))
ax = fig.add_subplot(111)
bp = ax.boxplot(year_user_score, patch_artist=True)
boxplot_color(bp)
plt.suptitle("IMDB User Rating Throughout 2008-16")
ax.set_xticklabels(years)
plt.ylabel('IMDB User Rating')
plt.xlabel('Year')
plt.show()
```

```
In [22]: #Preprocess Certificates
certificate = pd.get_dummies(final_google['rating'], prefix = "certificate", dummy_na=False)
final_google = pd.concat([final_google, certificate], axis=1, join='inner')
```

```
In [23]: #Getting Ready For Regression Analysis
interesting_columns = [u'title', u'IMDB_review_rating', u'score', u'IMDB_review_helpful_metric', u'IMDB_review_neutral_metric_neg', u'IMDB_review_sentiment_pos', u'budget', u'faces', u'fb_likes', u'overview_neg_score', u'overview_pos_score', u'revenue', u'runtime', u'Action', u'Adventure', u'Animation', u'Comedy', u'Crime', u'Documentary', u'Drama', u'Family', u'Fantasy', u'History', u'Horror', u'Music', u'Mystery', u'Romance', u'Science Fiction', u'Thriller', u'War', u'Western', u'certificate_12', u'certificate_13', u'certificate_16', u'certificate_A.G.', u'certificate_G', u'certificate_NC-17', u'certificate_Not Rated', u'certificate_PG', u'certificate_PG-13', u'certificate_R', u'certificate_TV-14', u'certificate_TV-MA', u'certificate_TV-PG', u'certificate_Unrated']
regression = final_google[interesting_columns]
final_google[interesting_columns].describe()
```

Out[23]:

	IMDB_review_rating	score	IMDB_review_helpful_metric	IMDB_review_neutral_n
count	457.000000	457.000000	457.000000	457.000000
mean	5.981328	7.294967	0.677632	0.515748
std	1.898766	0.644338	0.094402	0.185119
min	0.000000	4.200000	0.000000	0.000000
25%	4.583333	7.000000	0.624548	0.409820
50%	5.960000	7.300000	0.668078	0.523272
75%	7.476190	7.800000	0.725631	0.621879
max	9.880000	9.000000	0.939548	2.000000

8 rows x 45 columns

```
In [24]: pd.DataFrame({ 'Final_DataSet_Column_Names' : list(final_google.columns) })
```

Out[24]:

	Final_DataSet_Column_Names
0	title_g
1	title
2	overview
3	production
4	release
5	revenue

6	budget
7	runtime
8	tagline
9	imdb
10	genres
11	poster
12	year
13	Animation
14	Family
15	Drama
16	Action
17	Crime
18	Thriller
19	Romance
20	Science Fiction
21	Adventure
22	Fantasy
23	Mystery
24	Horror
25	Comedy
26	Documentary
27	History
28	War
29	Music
30	Western
31	overview_pos_score
32	overview_neg_score
33	faces
34	fb_likes
35	score
36	rating
37	director
38	cast

39	producer
40	IMDB_review_sentiment_pos
41	IMDB_review_sentiment_neg
42	IMDB_review_helpful_metric
43	IMDB_review_rating
44	IMDB_review_neutral_metric
45	certificate_12
46	certificate_13
47	certificate_16
48	certificate_A.G.
49	certificate_G
50	certificate_NC-17
51	certificate_Not Rated
52	certificate_PG
53	certificate_PG-13
54	certificate_R
55	certificate_TV-14
56	certificate_TV-MA
57	certificate_TV-PG
58	certificate_Unrated

```
In [25]: non_zero_regression = regression[(regression['budget']!=0) & (regression['fb_likes']!=0) & (regression['revenue']!=0) & (regression['runtime']!=0) & (regression['score']!=0) & (regression['IMDB_review_rating']!=0)]
pd.DataFrame({ 'Regression_DataSet_Column_Names' : list(non_zero_regression.columns) })
```

Out[25]:

	Regression_DataSet_Column_Names
0	title
1	IMDB_review_rating
2	score
3	IMDB_review_helpful_metric
4	IMDB_review_neutral_metric
5	IMDB_review_sentiment_neg
6	IMDB_review_sentiment_pos
7	budget

8	faces
9	fb_likes
10	overview_neg_score
11	overview_pos_score
12	revenue
13	runtime
14	Action
15	Adventure
16	Animation
17	Comedy
18	Crime
19	Documentary
20	Drama
21	Family
22	Fantasy
23	History
24	Horror
25	Music
26	Mystery
27	Romance
28	Science Fiction
29	Thriller
30	War
31	Western
32	certificate_12
33	certificate_13
34	certificate_16
35	certificate_A.G.
36	certificate_G
37	certificate_NC-17
38	certificate_Not Rated
39	certificate_PG
40	certificate_PG-13
41	certificate_R

42	certificate_TV-14
43	certificate_TV-MA
44	certificate_TV-PG
45	certificate_Unrated

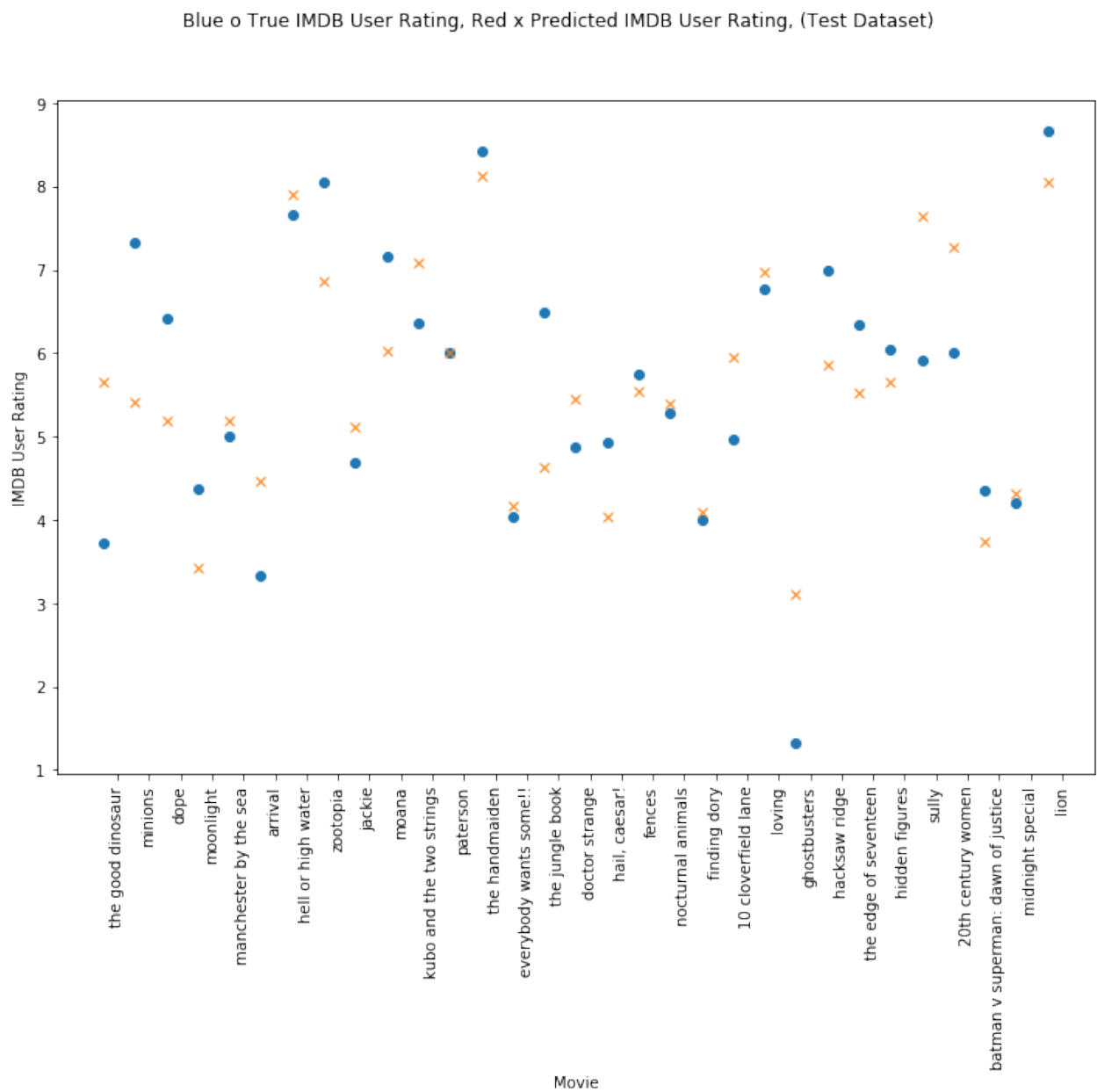
```
In [39]: #Regression Analysis Of Movies IMDB_User_Rating
reg_l = linear_model.ElasticNet(alpha = 0.0075, l1_ratio = 1, normalize=False, fit_intercept=True, copy_X=True)
reg_r = linear_model.Ridge(alpha = .01, normalize=False, fit_intercept=True, copy_X=True)
data = np.asarray(non_zero_regression)
x, y = data[:, 2:], data[:, 1]
x = normalize(x, norm='max', axis=0, copy=True, return_norm=False)
x_train, y_train = x[:300], y[:300]
this_scores = cross_val_score(reg_r, x_train, y_train, cv=10, n_jobs=1)
print "10-Fold Validation Scores", this_scores
reg_l.fit(x_train, y_train)
reg_r.fit(x_train, y_train)
print ""
print "Elastic Net R2", r2_score(y[300:], reg_l.predict(x[300:]))
print "Ridge R2", r2_score(y[300:], reg_r.predict(x[300:]))
```

```
10-Fold Validation Scores [ 0.4748902  0.62635043  0.47086847  0.59302983
 0.48807275  0.48071515
 0.39169209  0.48408271  0.28385876  0.38188356]
```

```
Elastic Net R2 0.651814548508
```

```
Ridge R2 0.607988492886
```

```
In [48]: #Plot Of Predicted Values
plotx = []
width = .9
y_true, y_pred = [], []
plotx = list(non_zero_regression['title'][300:])
y_true = list(y[300:])
y_pred = list(reg_r.predict(x[300:]))
indexes = np.arange(len(plotx))
plt.plot(list(xrange(len(plotx))), y_true, "o")
plt.plot(list(xrange(len(plotx))), y_pred, "x")
plt.xticks(indexes + width * 0.5, plotx, rotation='vertical')
plt.suptitle("Blue o True IMDB User Rating, Red x Predicted IMDB User Rating, (Test Dataset)")
plt.ylabel('IMDB User Rating')
plt.xlabel('Movie')
plt.show()
```



```
In [49]: non_zero_regression.to_csv("regression_dataset.csv")
         final_google.to_csv("final_complete_dataset.csv")
```