AUGUST 2020

TEXAS
The University of Texas at Austin

ANALYZING MOVIE DATA WITH PYTHON

By: Luke Bravo, Matt Jiang, Ali Daanesh Sayyed

MASTERS OF SCIENCE BUSINESS ANALYTICS 2021
The University of Texas at Austin

Overview

- Description of Project Goals
- Exploratory Analysis
- Solution and Insights

Description - What We're Investigating

- Dataset on movies released between 1927 - 2016
- Includes information
  - movie's production
  - reviews, genres
  - actors involved
  - directors involved
  - post production value
  - variety of other variables

Steps We Took

- Initially we did an Exploratory Analysis
- Realized there was significant amount of data on production of movies
- Questions we asked about the data:
  - Can we predict movie rating?
  - Can we construct a reliable model to predict revenue & budget based on the variables we have in the data?
  - Which movies, actors and directors are most notable in the features we are using to analyze?
  - Can We Explore various predictive modeling methods to analyze our data?

Importance of the Problem

- Netflix, Hulu, and HBO Max want to understand what drives movies success
- Our analyses provide helpful insights for big media companies to create the most attractive content for their audience
- Potentially help increase profit margins for media companies

Importance of the Problem Continued

- Looking for:
  - trends within the industry that provide economic insights to how movies generate revenue
  - factors that contribute movie success
  - help media companies select the best cast
  - popular genres in order to maximize gross profit

Exploratory Analysis

- Adjusting prices to 2020 USD
  - Pulled data on monthly CPI and inflation
  - Used PANDAS to calculate annual averages and scale each movie based on release date
- Simple Explorations
- Regression Based Explorations

Longest & Shortest Movie in Dataset

```
df['duration'].max()
```

```
511.0
```

```
df['duration'].min()
```

```
7.0
```

How many unique movie titles have The in their name

```
df['movie_title'].dropna().drop_duplicates()\
.map(lambda s: s.split(' ')[0] == 'The').value_counts()
```

```
False    3934
True      982
Name: movie_title, dtype: int64
```

Top 10 directors in dataset

```
df['director_name'].value_counts()[:10]
```

```
Steven Spielberg     26
Woody Allen          22
Martin Scorsese      20
Clint Eastwood       20
Spike Lee            16
Ridley Scott         16
Steven Soderbergh    15
Renny Harlin         15
Tim Burton           14
Oliver Stone         14
Name: director_name, dtype: int64
```

Top 10 Genres

```
df['genres'].value_counts()[:10]
```

```
Drama                              233
Comedy                             205
Comedy|Drama                       189
Comedy|Drama|Romance               185
Comedy|Romance                     157
Drama|Romance                      150
Crime|Drama|Thriller                98
Horror                              67
Action|Crime|Drama|Thriller         65
Crime|Drama                         62
Name: genres, dtype: int64
```

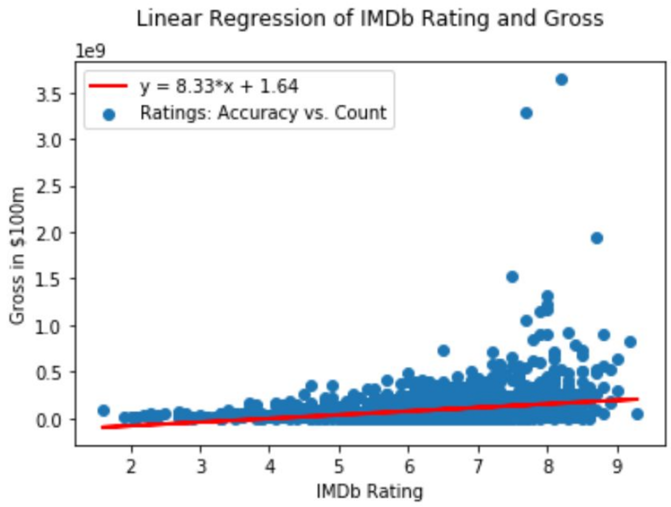
```
def splitem(s):
    return s.split('|')[0]

df['onegenre'] = df['genres'].map(splitem)
df['onegenre'].value_counts()
```

```
Comedy         1313
Action         1113
Drama           944
Adventure       439
Crime           340
Biography       250
Horror          221
Documentary      84
Animation        61
Fantasy          48
Mystery          32
Thriller         21
Sci-Fi           13
Western          12
Family           11
Romance           6
Musical           4
Film-Noir         1
Game-Show         1
Music             1
History           1
Name: onegenre, dtype: int64
```

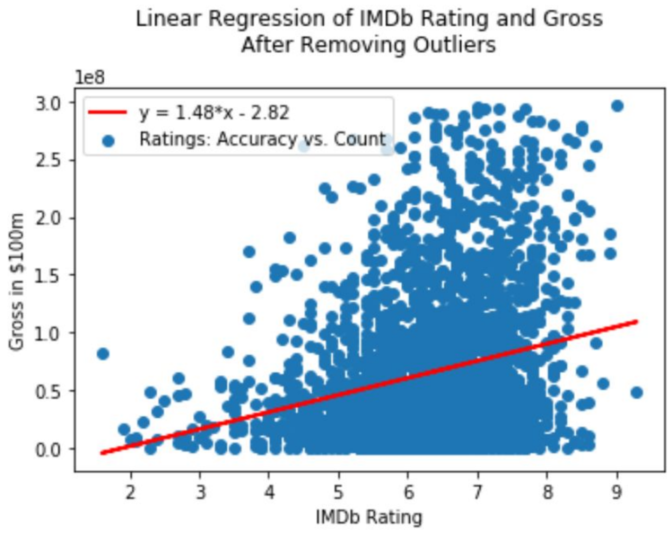
Top 10 Scoring Genres

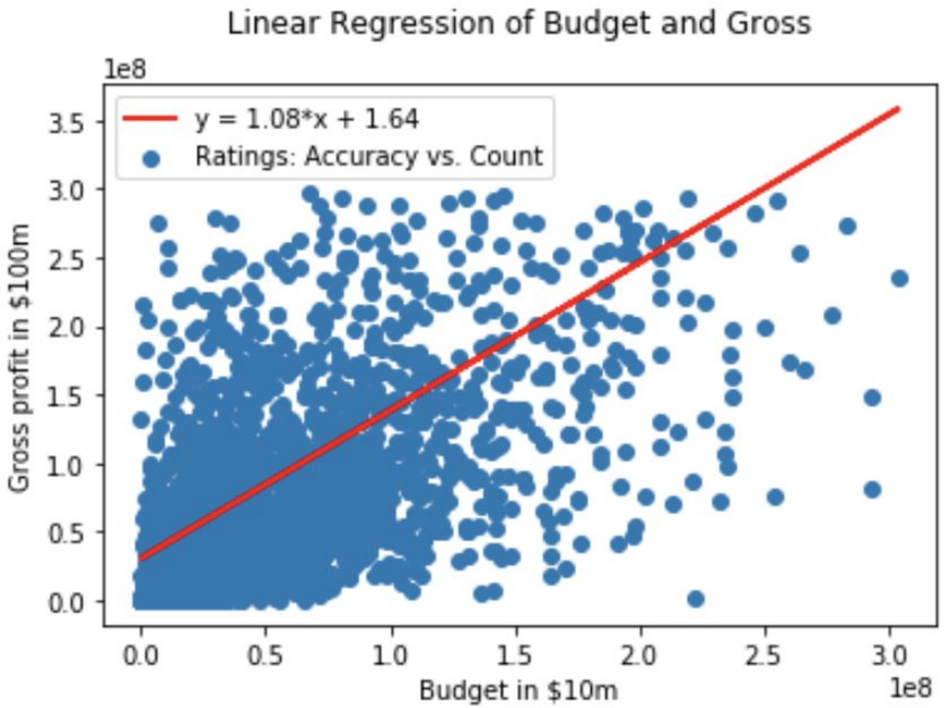
```
df.groupby('onegenre')['imdb_score'].mean()\
.sort_values(ascending = False)[:15]
```

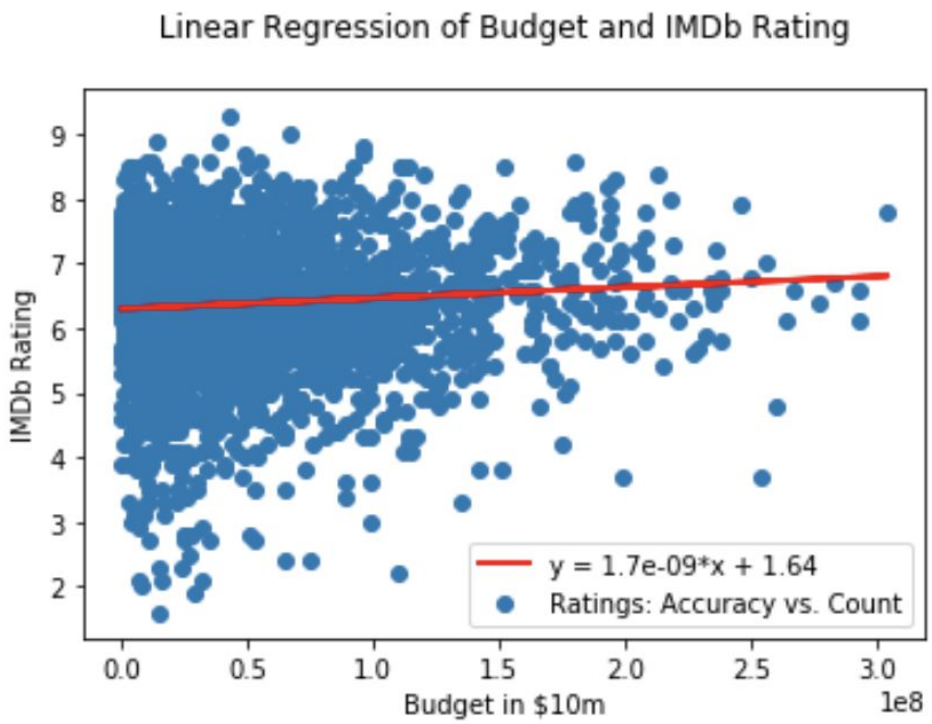
```
onegenre
Film-Noir      7.600000
History        7.500000
Music          7.200000
Documentary    7.167857
Biography      7.157600
Crime          6.902941
Drama          6.767161
Animation      6.631148
Western        6.583333
Mystery        6.534375
Adventure      6.530068
Fantasy        6.381250
Action         6.231626
Comedy         6.194136
Musical        6.000000
Name: imdb_score, dtype: float64
```



Remove outliers

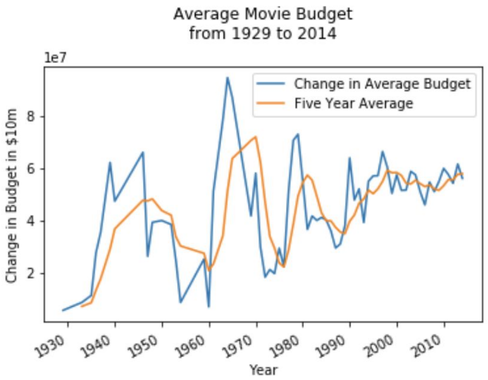


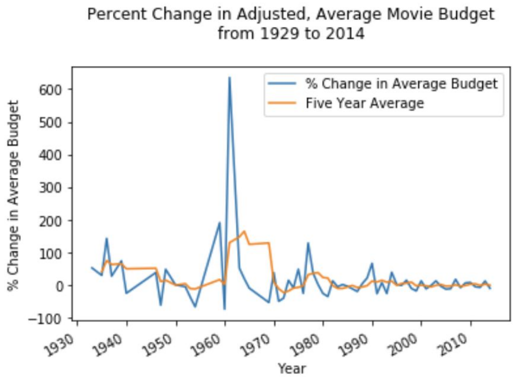

Linear Regression of Budget and Gross
1e8
y = 1.08*x + 1.64
Ratings: Accuracy vs. Count
Gross profit in $100m
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Budget in $10m
1e8

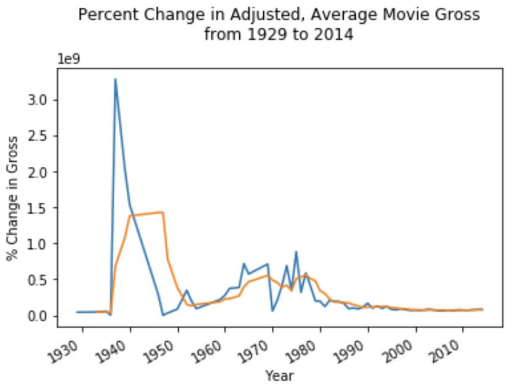

Linear Regression of Budget and IMDb Rating
IMDb Rating
Budget in $10m
1e8
y = 1.7e-09*x + 1.64
Ratings: Accuracy vs. Count
9
8
7
6
5
4
3
2
0.0
0.5
1.0
1.5
2.0
2.5
3.0

Solution and Insights

- OLS regressions showed us our data on its own cannot solve the problems on its own;
- Low R^2 suggests there are external variables that attribute to the patterns revealed by OLS.
- In light of this: used time series to get big picture of external variables' effect

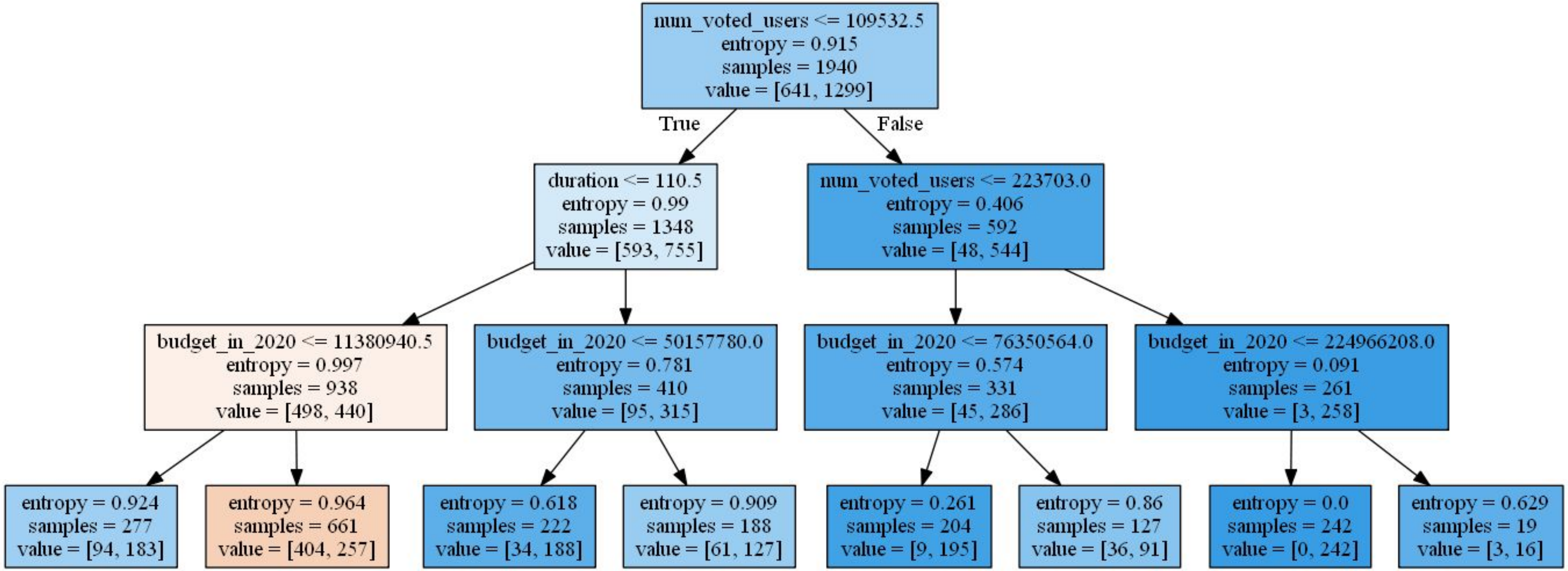






Solution and Insights

- Decision trees that has no limit on max depth and number of variables overfits the data
- Stratified K-fold cross validation determines an optimal depth and limit variables complexity
- Decision trees model is not without flaws and has limitation in accuracy prediction
- Multivariate regressions to predict adjusted budget do not provide reliable data


num_voted_users <= 109532.5
entropy = 0.915
samples = 1940
value = [641, 1299]
True
False
duration <= 110.5
entropy = 0.99
samples = 1348
value = [593, 755]
num_voted_users <= 223703.0
entropy = 0.406
samples = 592
value = [48, 544]
budget_in_2020 <= 11380940.5
entropy = 0.997
samples = 938
value = [498, 440]
budget_in_2020 <= 50157780.0
entropy = 0.781
samples = 410
value = [95, 315]
budget_in_2020 <= 76350564.0
entropy = 0.574
samples = 331
value = [45, 286]
budget_in_2020 <= 224966208.0
entropy = 0.091
samples = 261
value = [3, 258]
entropy = 0.924
samples = 277
value = [94, 183]
entropy = 0.964
samples = 661
value = [404, 257]
entropy = 0.618
samples = 222
value = [34, 188]
entropy = 0.909
samples = 188
value = [61, 127]
entropy = 0.261
samples = 204
value = [9, 195]
entropy = 0.86
samples = 127
value = [36, 91]
entropy = 0.0
samples = 242
value = [0, 242]
entropy = 0.629
samples = 19
value = [3, 16]

```
                            OLS Regression Results
==============================================================================
Dep. Variable:         budget_in_2020   R-squared:                       0.165
Model:                            OLS   Adj. R-squared:                  0.163
Method:                 Least Squares   F-statistic:                     77.81
Date:                Fri, 07 Aug 2020   Prob (F-statistic):          2.62e-103
Time:                        21:23:05   Log-Likelihood:                -52967.
No. Observations:                2772   AIC:                         1.059e+05
Df Residuals:                    2764   BIC:                         1.060e+05
Df Model:                           7
Covariance Type:            nonrobust
===========================================================================================
                              coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------------
Intercept               -7.389e+07   8.16e+06     -9.053      0.000   -8.99e+07   -5.79e+07
color[T.Color]           1.541e+07   5.34e+06      2.884      0.004    4.93e+06    2.59e+07
duration                 7.136e+05   4.24e+04     16.814      0.000     6.3e+05    7.97e+05
director_facebook_likes   292.2598    275.245      1.062      0.288    -247.447     831.967
actor_3_facebook_likes   2806.1843    576.105      4.871      0.000    1676.545    3935.824
actor_1_facebook_likes    128.6488     57.619      2.233      0.026      15.667     241.630
actor_2_facebook_likes    789.3986    238.723      3.307      0.001     321.305    1257.492
aspect_ratio             1.399e+07   2.44e+06      5.723      0.000     9.2e+06    1.88e+07
==============================================================================
Omnibus:                      793.880   Durbin-Watson:                   0.555
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2204.359
Skew:                           1.503   Prob(JB):                         0.00
Kurtosis:                       6.171   Cond. No.                     1.89e+05
==============================================================================
```

Recap

- Description of Project Goals
  - Movie Dataset; How can we help grow companies like Netflix
- Exploratory Analysis
  - Simple Analyses
  - Regression Analyses
- Solution and Insights
  - OLS Regressions
  - Decision Tree Models