

Predicting the Opening Weekend Box Office Results

Kandice Marcacci
Metis DS - Project Luther



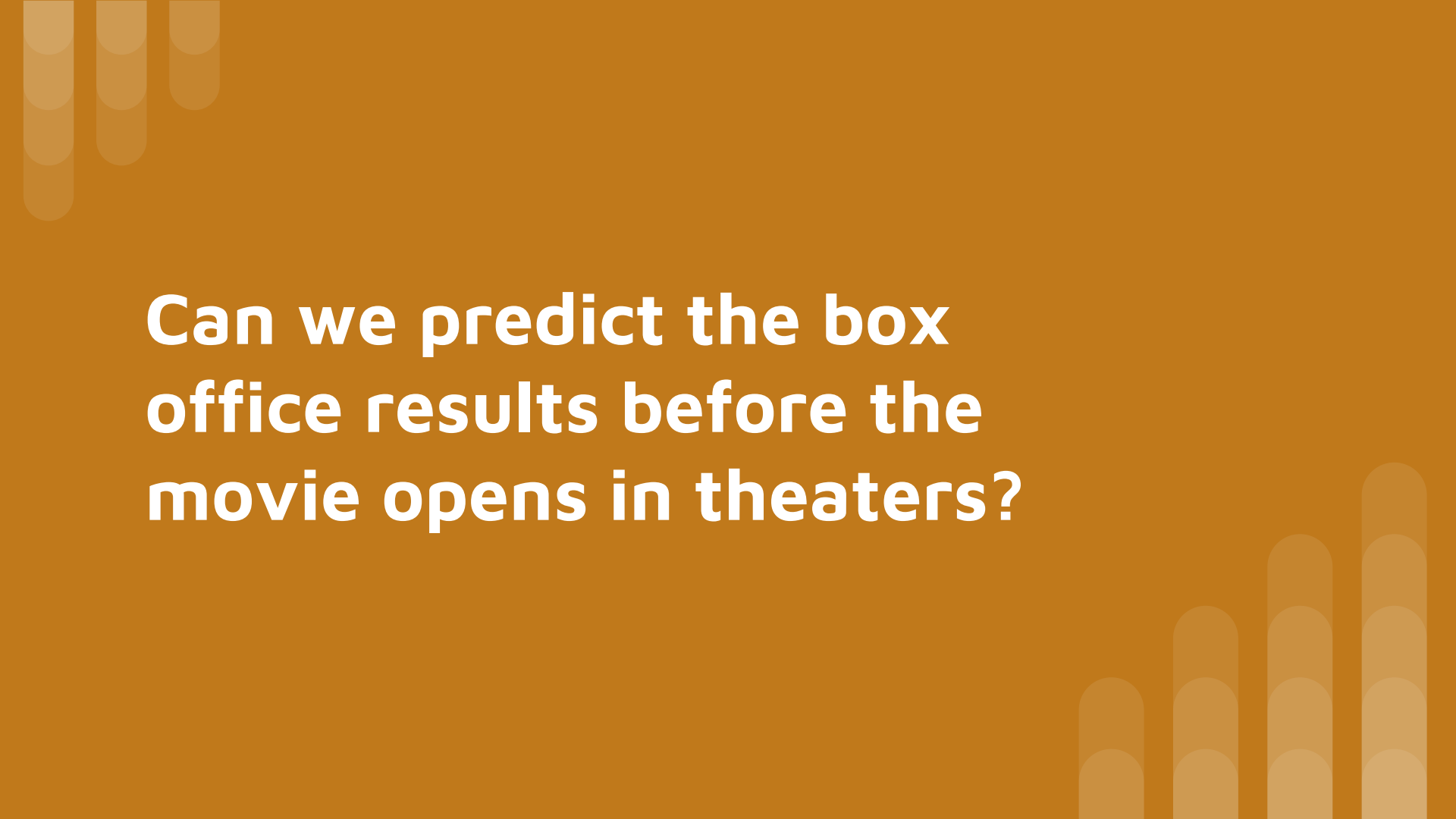


Why movies?



I chose to predict box office results because:

- I wanted my project focus to be on learning the tools and increasing my python proficiency over hunting for what might be more interesting insights.
- By choosing movies, I was able to get started working right away with the data rather than spending any time toying around with other ideas to carry me away from my focus.
- Since I actually really like movies and intuitively felt there should be some predictability, I was interested in exploring it.



**Can we predict the box
office results before the
movie opens in theaters?**

Phase 1

Gathering the data





Beautiful Soup

- I focused on this one data scraping tool because I found it to be easy to understand and flexible enough for what I was trying to achieve.





Beautiful Soup

- It allowed me to exercise my programming skills by utilizing for-loops, list comprehensions, and user-defined functions.
- I was able to extract data for every movie available on the website by scraping for href tags and then iterating through them to request the movie pages.

Secret menu

Box Office **Mojo**



Search Site

Search...

Features

[News](#)

[Release Sched.](#)

[Showtimes](#)

at 

Box Office

[Daily](#)

[Weekend](#)

[Weekly](#)

[Monthly](#)

[Quarterly](#)

> MOVIES

ALPHABETICAL INDEX

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Numbers

Title (click to view box office)	Studio	Total Gross / Theaters		Opening / Theaters		Open
#Horror	IFC	n/a	n/a	n/a	n/a	11/20/2015
\$9.99	Reg.	\$52,384	4	\$478	1	12/12/2008
\$upercapitalist	Truly	\$15,919	1	\$8,372	1	8/10/2012
(500) Days of Summer	FoxS	\$32,391,374	1,048	\$834,501	27	7/17/2009
(Untitled)	Gold.	\$230,600	25	\$18,002	3	10/23/2009

Predicting Opening Weekend - Data selection

Option 1 - Selectively collect features that could be meaningful.

Option 2 - Pull everything I can and then see what I have.



Challenges

- Due to the timing of Opening Weekend in the movie making pipeline, I couldn't use several of the numeric amounts available on the website. For example, Domestic Totals.
- Many of the available movie features are categorical requiring more feature engineering before they can be used in a linear regression model.
- Despite having over 17K movies, less than $\frac{1}{5}$ had successfully scraped budget data.



Plan of attack: Roll with the punches

- Turn some of the categorical features into indicators
- Do another scrape of some other tables on the site to pulling in number of theaters, franchise indicators, distributors.
- Supplement the BoxOfficeMojo data with the-numbers data to join more budget data to the movies lists.

Phase 2

Cleaning the data





Challenges

- Despite some pre-cleaning during the scraping process, there were still several hiccups in the process of getting the scraped data into a manageable DataFrame object.
- Perhaps the biggest obstacle in my way was that I am still learning pandas.
- Oh no! How am I going to join the two data sources?



Plan of attack

- Get pandas skills under control by keeping at it.



- Utilize fuzzywuzzy to match the titles in the two sources.

```
!pip install fuzzywuzzy
```

```
from fuzzywuzzy import fuzz  
from fuzzywuzzy import process
```



DataFrame

Predicting (y): Opening Weekend Box Office

Features (X):

Budget, # Opening Theaters, Runtime, Top 20 Distributor Indicator, Genre (grouped), Rating (grouped), Franchise Indicator, Season of Release Date

Phase 3

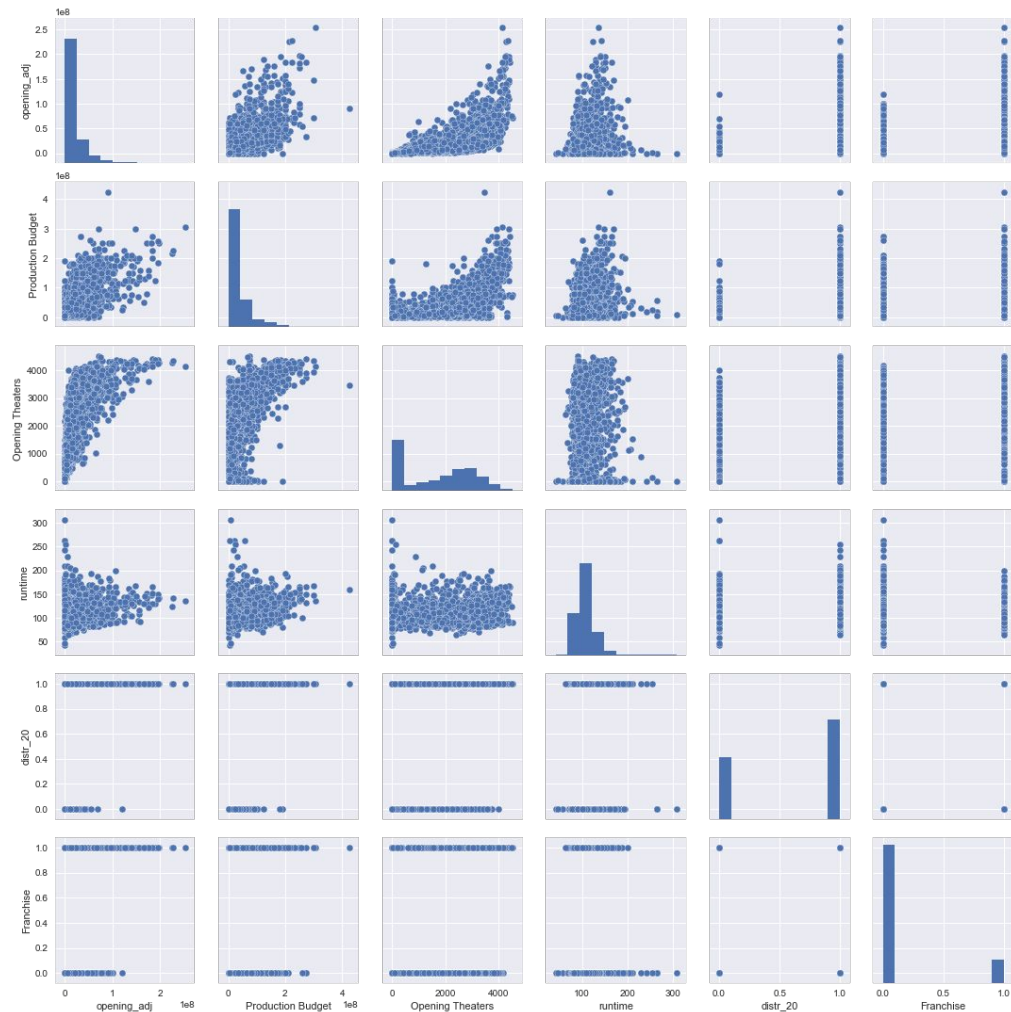
Model the data



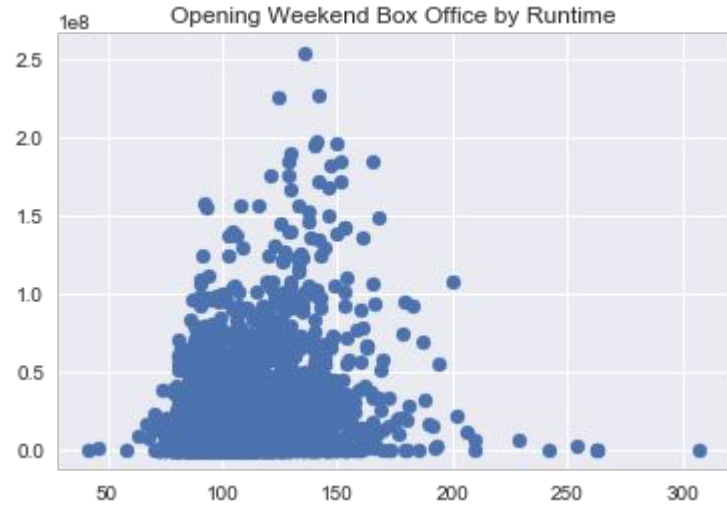
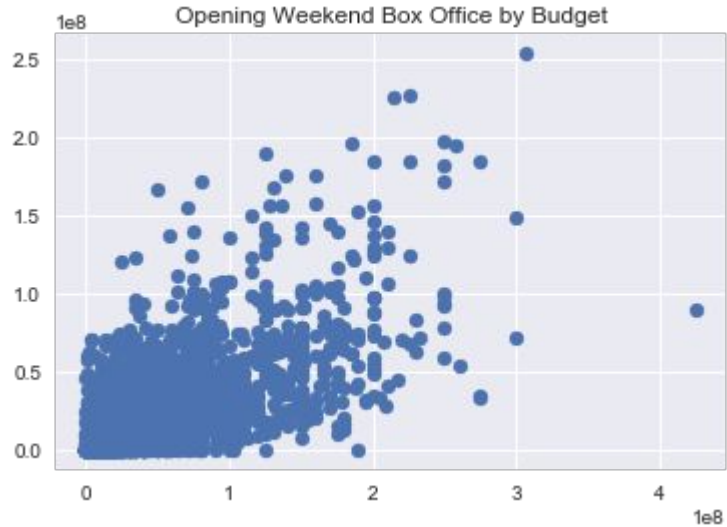


Pair Plots

- What does it all mean?
- Looks kinda crazy to me!
- Could probably log to address the skewness.
- Time?

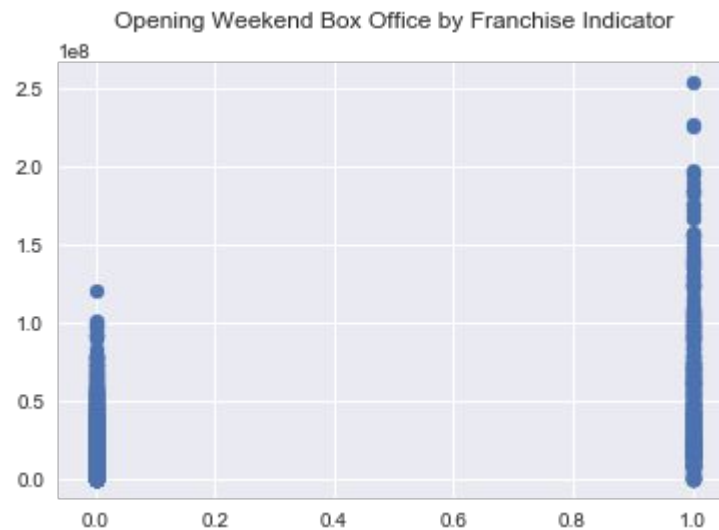
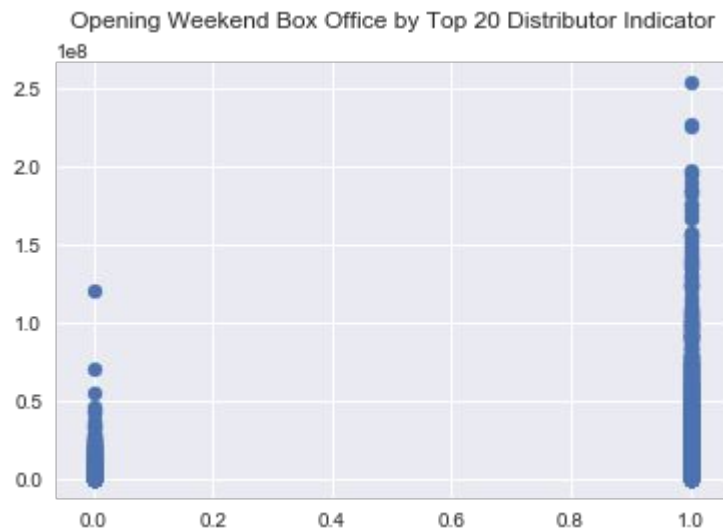


A few scatter plots...





A few scatter plots...





Add in some dummy indicators

Using patsy

- Genre (6 dummies)
- Rating (5 dummies)
- Season (3 dummies)

Manually

- Franchise (On/Off)
- Distributor/Studio (Top 20 Grossing, On/Off)



Correlations with Opening Weekend Box Office

- Some are expected and some obvious:
 - Budget
 - Theaters
 - Franchise

budget	0.697158
theaters	0.671584
franchise	0.537762
distr_20	0.465077
runtime	0.245156
PG_13	0.208601
Sci_Fi	0.126987
Summer	0.089881
PG	0.077506
Winter	0.031396
Horror	0.013873
Other	-0.012558
Thriller	-0.025890
NC_17	-0.026536
Comedy	-0.058566
Spring	-0.073305
Drama	-0.177272
R	-0.269121



First run

OLS Regression

- Adj. R-square = 0.647
- Distr_20: $p > |t| = 0.443$

OLS Regression Results

Dep. Variable:	opening	R-squared:	0.647
Model:	OLS	Adj. R-squared:	0.647
Method:	Least Squares	F-statistic:	1611.
Date:	Fri, 06 Oct 2017	Prob (F-statistic):	0.00
Time:	05:29:41	Log-Likelihood:	-78853.
No. Observations:	4397	AIC:	1.577e+05
Df Residuals:	4391	BIC:	1.578e+05
Df Model:	5		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
Intercept	-1.303e+07	1.35e+06	-9.633	0.000	-1.57e+07	-1.04e+07
budget	0.2202	0.007	30.085	0.000	0.206	0.235
theaters	5668.8714	280.737	20.193	0.000	5118.485	6219.258
runtime	9.326e+04	1.26e+04	7.407	0.000	6.86e+04	1.18e+05
distr_20	4.936e+05	6.44e+05	0.767	0.443	-7.68e+05	1.76e+06
franchise	2.098e+07	6.95e+05	30.186	0.000	1.96e+07	2.23e+07

Cut to...





Final

OLS Regression

- Adj. R-squared = 0.649
- Features:
 - Budget
 - Theaters (Opening)
 - Runtime
 - Franchise
 - Summer
 - Winter

OLS Regression Results

Dep. Variable:	opening	R-squared:	0.649
Model:	OLS	Adj. R-squared:	0.649
Method:	Least Squares	F-statistic:	1355.
Date:	Fri, 06 Oct 2017	Prob (F-statistic):	0.00
Time:	07:14:11	Log-Likelihood:	-78840.
No. Observations:	4397	AIC:	1.577e+05
Df Residuals:	4390	BIC:	1.577e+05
Df Model:	6		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
Intercept	-1.463e+07	1.38e+06	-10.572	0.000	-1.73e+07	-1.19e+07
budget	0.2170	0.007	29.652	0.000	0.203	0.231
theaters	5827.5131	224.873	25.915	0.000	5386.648	6268.378
runtime	9.857e+04	1.25e+04	7.864	0.000	7.4e+04	1.23e+05
franchise	2.079e+07	6.93e+05	29.991	0.000	1.94e+07	2.22e+07
Summer	2.447e+06	5.4e+05	4.535	0.000	1.39e+06	3.51e+06
Winter	2.204e+06	5.56e+05	3.967	0.000	1.11e+06	3.29e+06



Residuals





Train - Test - Split (70/30)

- Ran 10 times
- Test R^2 range: 0.62 - 0.66
- Features:
 - Budget
 - Theaters (Opening)
 - Runtime
 - Franchise
 - Summer/Winter

```
test r2: 0.65, train r2: 0.65
test r2: 0.65, train r2: 0.65
test r2: 0.62, train r2: 0.66
test r2: 0.63, train r2: 0.66
test r2: 0.64, train r2: 0.65
test r2: 0.66, train r2: 0.64
test r2: 0.66, train r2: 0.65
test r2: 0.63, train r2: 0.65
test r2: 0.64, train r2: 0.66
test r2: 0.64, train r2: 0.65
```

Possible Enhancements

- Use scrapped scraped data: actors, writers, directors, producers.
- Try different ways to cut up categorical values.
 - Top 50, Top 10%, etc
- Revisit Genre and Rating dummy indicators
- Explore over and under predictions
- Log long tailed variables
- Fit with Ridge, Lasso, or ElasticNet

My Takeaways:

- **Confident scraping with Beautiful Soup**
- **More confident in Pandas**
- **Love pickling!**
- **Experience with fuzzywuzzy matching**
- **Access to many detailed references for my future modeling needs**

Thank you for your time!