# Predict Kickstarter Campaign Success

**Author:** 

**GORDON MACMILLAN** 

### **Problem Statement**

#### Dataset of Kickstarter Campaign

- 400,000+ Past Sales
- Campaign data from 2009-2017
- o fundraising goals between \$.01 \$100 million
- 31 initial model features

#### GOAL

- Predict Future Kickstarter Campaign Success
- Predict Future Staff Picks

# Data Analysis Files

Divided the analysis into 3 notebooks

- data\_exploration.ipynb
- modeling\_success.ipynb
- modeling\_staff\_picks.ipynb

And one python file

helper\_functions.py

## EDA: Goals / Dates

#### Key Findings:

 The hour the campaigns are launched at and created at seem to make a difference





# **EDA: Configuration**

#### Key Transformed Features:

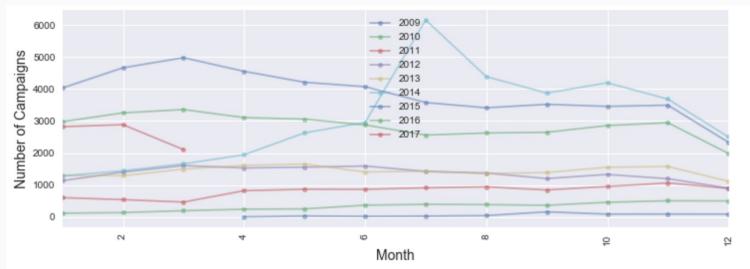
- Conversion of Unix time values to pandas datetime entries
- Conversion of target variable to True/False (1.0/0.0)
- Name and blurb text information cleaning of stopwords and excess whitespace

#### Key Engineered Features

- Time delta between creation and launch measured in days
- Time delta between launch and deadline
- Time components from datetime entries
- Conversion of time components of datetimes to one-hot encoded booleans
- Conversion of category to one-hot encoded booleans

# **EDA: Interesting Realization**

Large spike in the number of Kickstarters launched in July 2014



Not sure why the legends in the center of page. Will fix later

## **EDA: Dropping Columns**

Given enough time would go into full detail as to why.

#### Dropped columns:

- id
- backers\_count
- pledged
- usd\_pledged
- etc...

## **Model Selection**

Logistic Regression (I2 regularization)

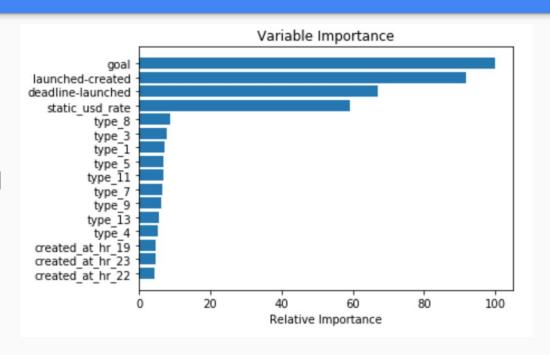
Random Forest

**Gradient Boosted Trees (best)** 

## Feature Importance

#### Standouts:

- Goal amount
- Time diff launched-created
- Time diff deadline-launched
- exchange rate



## Given more time would have tried

**XGBoost** 

SVC

**Neural Network** 

## **Future Work**

Spend more time working on EDA and Feature engineering

Enhance use of Cross-validation and hyper parameter tuning

Fit similarly valued goals into bins