# Predicting Kickstarter Success

MSDS 6372: Project 2

Authors: Travis Deason, Yejur Kunwar, and Vanessa Torres

**Abstract**

The success of Kickstarter projects may have some association with variables like goal, number of backers, amount of money pledged, and a few other key features. This study focuses on producing a model that could help determine those said features in order to help predict the success of a Kickstarter project. The methodology includes brainstorming, natural language processing, and random forest algorithms. The following models could be helpful in determining and improving the success rate of Kickstarter projects.

**Introduction**

The platform to spark ideas, design, share, and start a project can be very helpful when projects come to life. Kickstarter is a similar platform to bring projects to life through crowdfunding. With an "all-or-nothing" funding as a core part for a project, pledges can range anywhere from $100 to as high as a billion dollars.

This study is an attempt to design a model that can predict the success rate of a Kickstarter project by utilizing statistical measures. Leveraging the concepts of random forest algorithm to analyze the variables listed in data description alongside with some key features is the thought process for predicting outcomes in this study.

**Data description**

Extracted from Kickstarter, https://www.kaggle.com/kemical/kickstarter-projects/data the datasets in this study, are projects that started on the Kickstarter platform in 2016 and 2018. The 2016 file consists of 323750 observations while the 2018 file has 378661 observations. Both datasets have 13 common explanatory variables - 6 categorical and 7 continuous. The categorical variables that also have subtypes are main_category (15), country (21), currency (13), state (5), and category (>100) (see A.1).

Noticeably, when reading the 2016 .csv file, python reads 4 additional "unnamed" columns and in the 2018 file, there are 2 additional variables, "usd_pledged_real" and "usd_goal_real" (As shown in, Table 1).

Table 1: Variables and data description

| Variables | Type | Description |
|---|---|---|
| ID | numeric | internal kickstarter id |
| name | string | name of project |
| category | string | category |
| main_category | string | category of campaign |
| currency | string | currency used to support |
| deadline | date | crowdfunding deadline |
| goal | numeric | fundraising goal |
| launched | date | date launched |
| pledged | numeric | amount pledged by crowd |
| state | string | state pledged from |
| backers | numeric | number of backers |
| country | string | country pledged from |
| usd pledged | numeric | amount of money pledged |
| *usd_pledged_real | numeric | pledged amount in USD (conversion made by fixer.io.api) |
| *usd_goal_real | numeric | goal amount in USD |

*column only in the 2018 file

**Exploratory data analysis**

The quick look with example of rows and columns after adding some features extracted from NLP(natural language processing) algorithms are listed in images below. We were interested to find out how the noun, adjective, verbs, and other language structures used in titles would help us build most predictive models.

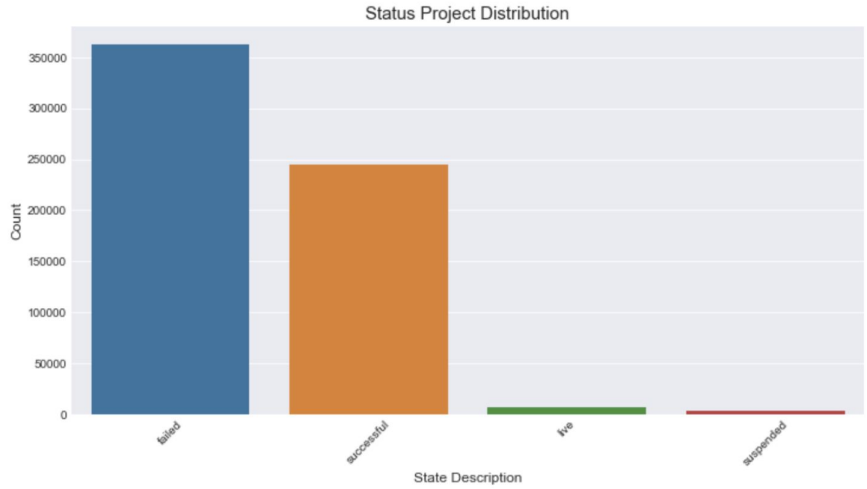
Table 2: Description of datasets with added features

| Features | Description |
|---|---|
| country | country of project (string, categorical) |
| currency | currency by country (string, categorical) |
| deadline_month | deadline for crowdfunding (string, categorical) |
| goal | goal for fundraising (int) |
| launched_month | project launch month (string, categorical) |
| length | length from start to launch (int, number of days) |
| main_category | main categories of project (string, categorical) |
| name$adj_count | NLP- count of adjectives in project name (float) |
| name$determinator_count | NLP- count of determinator in project name (float) |
| name$noun_count | NLP- count of noun in project name (float) |
| name$possesive_count | NLP- count of possessive words in project name (float) |
| name$preposition_count | NLP- count of preposition in project name (float) |
| name$punc_count | NLP- count of punctuation in project name (float) |
| name$verb_count | NLP- count of verb in project name (float) |
| name$word_count | NLP- count of total words in project name (int) |
| success | success rate (Boolean: True, False) |

A clean up of dropping the 4 “unnamed” columns from the 2016 file, combining both files, replacing the “NAs” to 0 for the “usd pledged” variable, and removing all of the “NAs” from the entire file was done before getting started with the analysis. In doing so, the state variable showed that more than 55% of Kickstarter projects had failed.

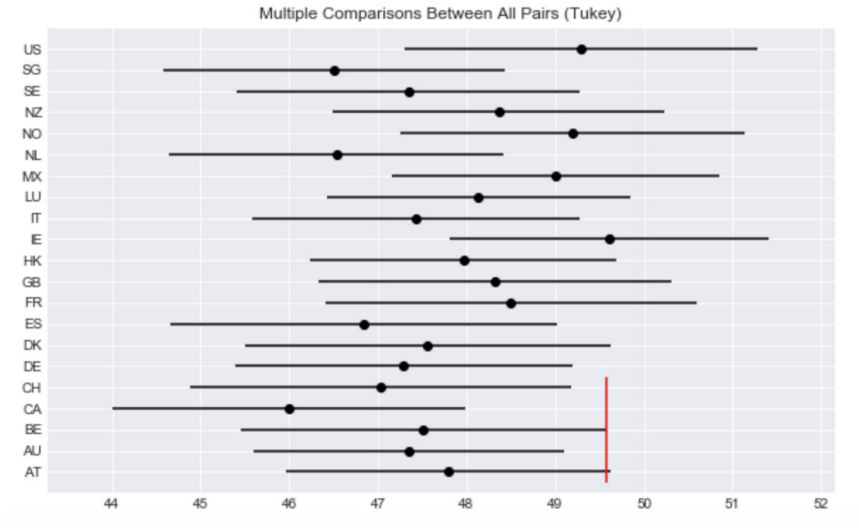
Fig 1: Count of successful, failed, and other project status

```
State Percentual in %:
failed        58.69
successful    39.62
live           1.16
suspended      0.53
Name: state, dtype: float64
```




Based on the initial glance, it was also determined that a significant difference (p-value=0.89) between countries and their pledges existed therefore could not be concluded. Though, when a second ANOVA was ran for US pledges only, a p-value of 0.04 was found. Furthermore, as a final look, a Tukey multiple comparison between all pairs was performed and found that US and GB were the only countries with a mean difference.

Fig 2: Multiple comparison between countries and pledges



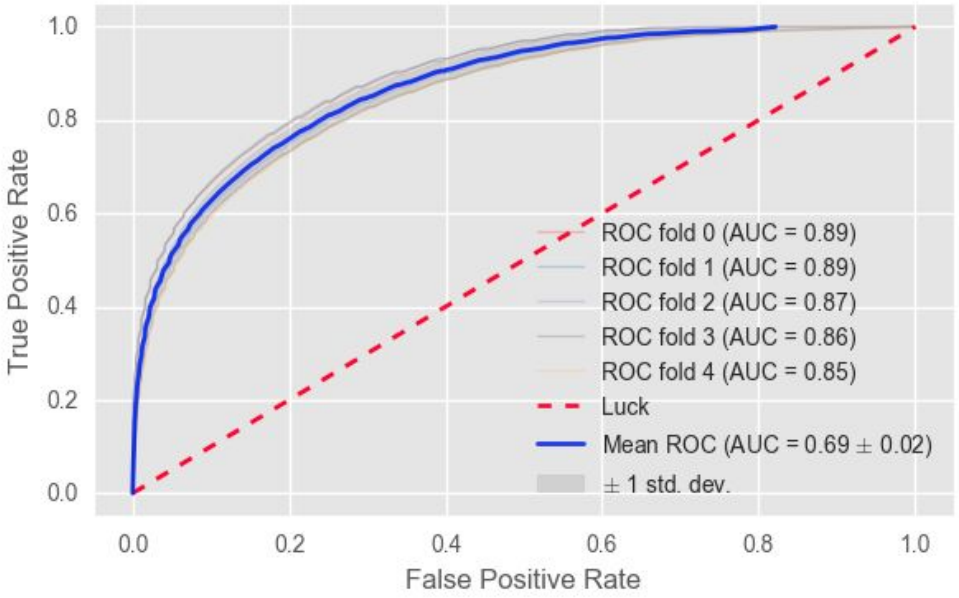
## Restate problem

The team started with a simple question which focused around predicting success rate of Kickstarter projects. The dataset suggested over 50% of Kickstarter projects failed or do not reach the successful status. In this study, the primary focus is on utilizing available variables in the dataset as well as finding some more feature variables to better predict outcomes.

## Models

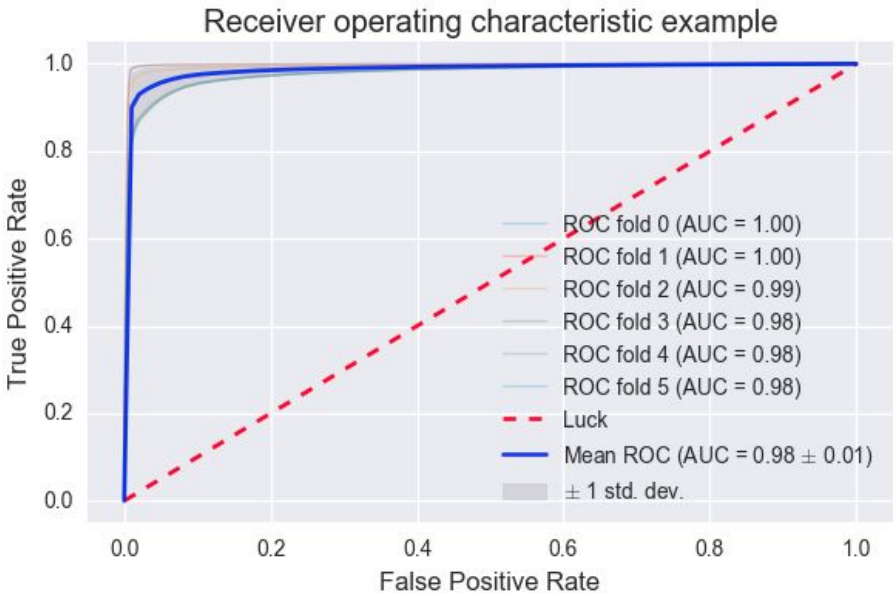
The model used for classification was a Random Forest Decision tree classifier, by utilizing the Sklearn python package to implement this algorithm. The first model did not incorporate the values derived from the name variable, the ROC plot for that model is shown below.

Fig 3: Model without NLP features




Overall model performance as measured by area under the curve of a 5 fold cross validation was found to be 69%; which is only 19% better than random chance. An attempt to improve on this performance, features based on the Kickstarter name (such as amount of capital letters, total word count, and parts of speech count) were introduced. The final model used the gini impurity criteria to determine tree splits, utilized bootstrapping samples equivalent to the full sample size randomly selected with a seed of 42, and each tree was modeled using the square root of the total 85 features (which is equivalent to 9 features per tree). Each tree was allowed to continue splitting until a single datapoint exists on the terminal leaf. The final model had 30 trees. The final model had an area under the curve of .98 as measured by a 5 fold cross validation. The ROC curve for the final model is shown below.

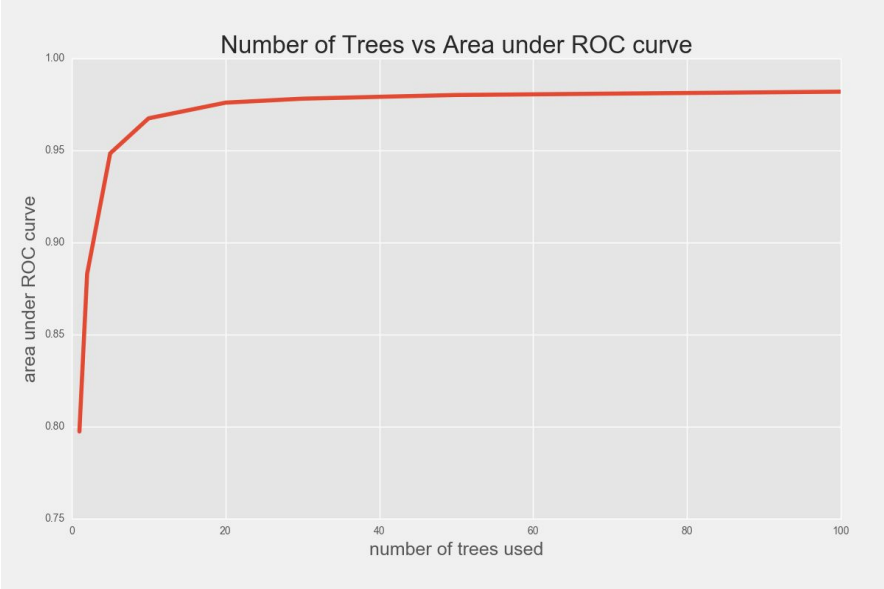
Fig 4: Model with NLP features




The improvement in model performance between the model which incorporates features derived from the kickstarter name and the model which does not is strictly due to the additional features. All other criteria between the two models are identical.

The determination to use 30 trees in our final model was based on running the model on a 2 fold cross validation using 1, 2, 5, 10, 20, 30, 50 and 100 trees. Using the figure below, it was identified that the model does not improve significantly for tree counts above 30.

Fig 5: Plot of Number of Trees vs Model Performance
(Area Under Receiver Operator Characteristic curve)

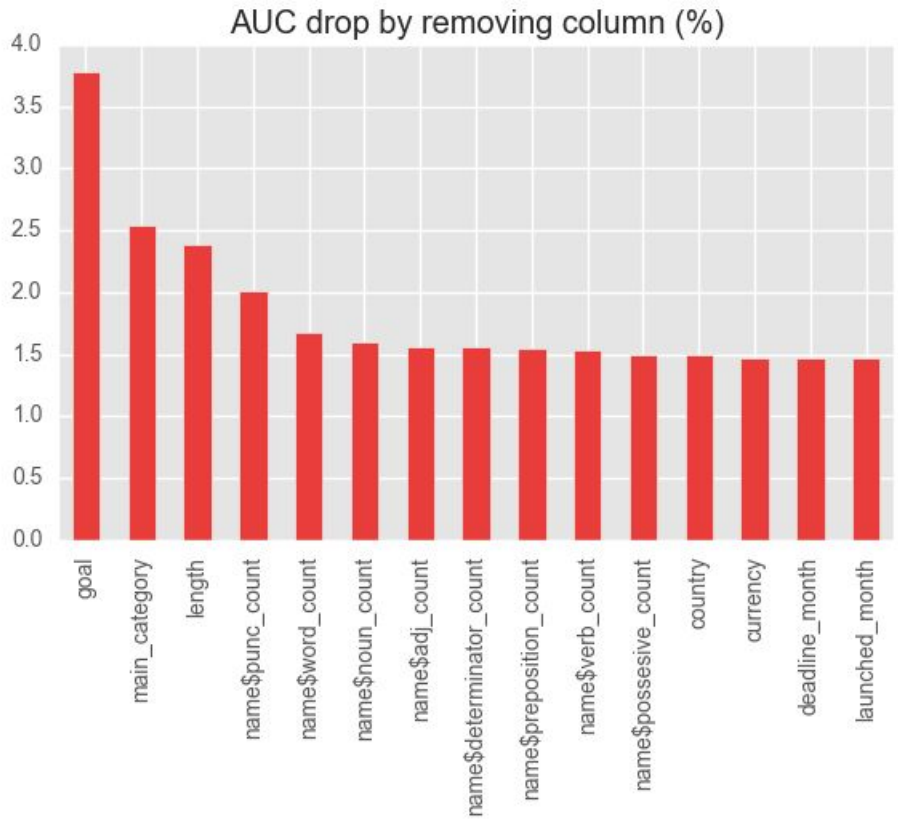


**Interpretation**

The probable uses for this model could include determining the odds of an investment in a kickstarter eventually resulting in a variable product. Because we utilized a random forest classifier, this model cannot be used to determine individual contributions of specific features, but the model could be useful in determining future investment.

To try and get a better understanding of the model, we counted the number of splits in the decision tree for each category. This should not be considered an accurate means of determining feature importance, but this measure identified the ratio of punctuation characters and the initial kickstarter goal as the most important features in the model. To emphasise, this method of interpretation does not mean these variables were actually important to model prediction. In fact, the final model was run one additional time with punctuation characters ratio removed, and there was no measurable impact on the area under Receiver Operator Characteristic (ROC) curve performance. In fact, the model proved highly resilient against the removal of any one feature. The figure below shows thea area under the ROC curve impact by removing any one feature.

Fig 6: ROC drop by removing each single feature



Only by removing the kickstarter goal from the analysis, does this drop the AUC by more then 3%, the majority of features are clustered around dropping the AUC metric by 1.5 percent.

## Conclusion

The Kaggle dataset we collected contained a small amount of overall features, but a very large and varied amount of kickstarters. We were able to extract additional meaningful data by measuring features in the actual kickstarter names; while this is a highly simplified version of
Natural Language Processing (NLP), it does allow us to improve our ability to predict if a kickstarter will be successful. This study has shown it is possible to dramatically improve a model's performance by using simplified NLP methods. This exercise also demonstrates it is possible to predict the likelihood of a kickstarter's success using a relatively small subset of data points.

**APPENDIX**

**A.** The Dataset for this report was extracted from https://www.kaggle.com/kemical/kickstarter-projects/data

**A.1.**

| Variable | Subtype |
|---|---|
| main_category | Publishing, Film & Video, Music, Food, Crafts, Games, Design, Comics, Fashion, Theater, Art, Photography, Technology, Dance, Journalism |
| country | AT, AU, BE, CA, CH, DE, DK, ES, FR, GB, HK, IE, IT, LU, MX, NL, NO, NZ, SE, SG, US |
| currency | BP, USD, CAD, NDK, AUD, EUR, MXN, SEK, NZD, CHF, DKK, HKD, SGD |
| state | failed, successful, live, undefined, suspended |
| category | > 100 |

**B.** All code and data sources are available at https://github.com/tdeason416/DS6372_grp2.

**C.** Embedded source Code:

- python script file clean_data.py
- python script file extract_features.py
- python script file functions.py
- python script file analysis.py

**D.** Jupyter notebook code demonstration:

- Feature Extraction
  - https://github.com/tdeason416/DS6372_grp2/blob/master/simple_eda.ipynb
- Model Analysis and optimization
  - https://github.com/tdeason416/DS6372_grp2/blob/master/Modeling.ipynb