CS 5010 Final Project Write-up

Group _predictUs_()

Charu Rawat (cr4zy)
Karan Gadiya(khg8mh)
Ning Han(nh4mq)
Charisma Ravoori(cr2st)

August 8th , 2018

Introduction

Kickstarter is a crowdfunding platform founded in 2009, which has helped successfully fund more than 148,000 projects and generate more than $3.4 billion in the US. Out of the 191 crowdfunding platforms in the US, Kickstarter has grown into one of the most popular platforms enabling people to bring to life their creative ideas. To create a project or launch a campaign on Kickstarter, creators simply add a Project Title along with certain attributes to it such as - description of the project, fundraising goal, deadline of project and an assortment of backer rewards on Kickstarter website which serve as an incentive to the backers to donate money to the project. Kickstarter has a "ALL OR NOTHING" model, which means that if the project doesn't meet its funding goal in time before the deadline, the creators don't get any money donated by the backers. And instead, Kickstarter charges 5% of the pledged amount as its fee.

Out of the thousands of projects that get launched on Kickstarter every month, only a few succeed. According to their official website, they state that the average success rate on Kickstarter is ~36%. This small number is why it becomes very important to understand the dynamics of crowdfunding and how that plays into determining the outcome of a project. Our project envisages to predict this - That given certain attributes about a project can we predict the success or failure of a Kickstarter campaign? And what really distinguishes a successful campaign from a failed one? We hope to leverage data to gain some insight into the trends and transform it into something actionable. Our analysis would be useful for creators who want to check how their project or campaign can perform on Kickstarter before they invest time and energy in launching it on the website. And to also be able to adjust project specs and goal which could help the maximize their chances of success. We also would expect it to be useful to people or "backers" who want to determine the outcome of project based on its current state before they commit to donating money.

The Data

The Kickstarter data set that we have used for our analysis has been sourced from Kaggle.com. The dataset called Kickstarter Projects, created by a crowdfunding enthusiast Mickael Mouille, while the raw data itself has been collected from the original Kickstarter Platform. Our reason for choosing this dataset is driven by the fact that crowdfunding is an industry that is rapidly expanding across the world. The industry is projected to grow over $300 billion the next 7 years by 2025! And while its generating millions of dollars, it is bringing to life ideas in various sectors and helping people fulfil their entrepreneurial dreams. As people move towards such non-traditional methods of obtaining investments, there is a need for better decision making for the creators as well as donators before time, money and energy is invested in it. Using data here can help us answer questions in a multitude of dimensions – from the binary outcome of a project to giving us insight into social behavior of humans, understanding what really drives human interest and how that collective interest or "crowd behavior" leads to someone's goal getting achieved. The dynamics of this interplay was fascinating and given the data we have we wanted to explore this.

For our analysis, we focused only on projects launched in US and those with a binary outcome of either success or fail. For each Kickstarter project in our dataset, it has a status of 5 levels: failed, successful, live, suspended, and canceled. Since we were only interested in projects with status of either successful or failed, we removed the projects with status of other three levels. We also dropped some columns that we felt we wouldn't need for our project analysis. At the end, we only kept nine columns from the dataset, which are ID, category, main category, state, deadline, launched, goal, pledged and backers. We also added 5 columns based on the existing columns such as calculating the funded percentage as a ratio of pledged money to goal, calculating days or duration of project using the launch and deadline date columns. We also converted some variables like category, state into categorical type and formatted the date columns appropriately.

The original Kickstarter data contained roughly 378,000 projects or rows and our final data after pre-processing contained 261,360 rows that we used for our analysis.

Beyond the Original Specifications

Advanced Query: Predictive Modelling-

Due to Kickstarter's "All or Nothing" model, what forms the core of this analysis is the end result of a project – success or failure. Keeping that goal in mind, we built a couple of predictive models to classify a project as a success or failure based on certain attributes. In this case, we used the following attributes to determine the end status of the project based on the relationships discovered between in the exploratory data analysis. The following attributes were used as our independent variables in predictive modelling - 'main category', 'category', 'launched month', 'deadline month', 'goal', 'duration ' ; all that are set by the user on initiating a project on Kickstarter.

Prior to implementing the classification techniques on our data, we encoded categorical variables such as main category and sub category and divided our data into train and test using a 77-33 split. The classification models that we implemented were - Logistic regression, Random forest and kNN.

Amongst the 3 models, we found Random Forest to be the most suitable fit for our data because of a relatively higher accuracy rate and a higher area under the ROC curve. The Random Forest method (with 100 estimators) took comparatively less time to run than kNN.

Classification Models

	Accuracy	Area under ROC
Random Forest	63.15%	0.615
Logistic	59.83%	0.538
kNN	59.83%	0.606

Table 1 Predication Accuracies of Classification Models

Web Interaction

For the web interaction portion of our project we made use of an interactive visual library in python called Bokeh. Bokeh provides a simple way to create elegant and professional looking displays with an interface where users can modify inputs, scale and intents of the charts being produced.

In our webpage (local), we created 3 tabs, each with its own functionality. The first tab has a histogram that graphs frequency of projects over the goal. On this tab we included a sliding mechanism that allows you to manipulate the x-axis scale. So if originally the scale goes up to $40,000, it can be changed to only show the projects with a goal upto $20,000. The second tab contains a graph showing the distribution of projects over two factors: main categories and number of backers. This tab contains a drop down mechanism that allows the user to switch back and forth between the two. The third tab shows the distribution of projects over the months in a year. This tab has a checkbox widget that allows the user to choose one of the 15 main categories as a filter for the graph. So for example, if the user chooses the 'Food' box, the graph will display the distribution of 'Food' related projects over the months. Also, all three tabs allowed a user to get more information about each point on the graph by hovering over it.

The code to create these graphs is relatively similar to most other plotting libraries in Python. We have to identify the x-axis and y-axis and use the relevant plotting function to display the graph. However to add the functionalities we had to incorporate an extra function called the 'Call-back function.' This function took in 3 attributes: attr, old, and new. This function performed the necessary tasks to enable the functionality.

	FAIL	SUCCESS	TOTAL

In the end however, were only able to create the graphs and add the 'hover' function. Unfortunately, we could not get the widgets to work.

Web-crawling

We downloaded our dataset from Kaggle website, but we web-crawled a test data from the Kickstarter website to test our predictive model. Our web-scraping code is based on selenium library, because the Kickstarter website used a lot of javascripts. We used chrome webdriver to get the main website content and created a list of project urls with for loop. We then opened each project page to get their goal, their category information, and their deadline.

For launch date information, it is only shown after clicking a button "Updates" in each project webpage. We using the click() function in the library and generated another for loop to get the launch date information. After getting all the information we need, we formatted the launch date and deadline date to the date formats we used in our data frame. We also formatted the goal to integer type to fit in our predictive model.

Finally, we saved the result dataframe into a csv file. Our final result contains 12 observation (we only got the projects on the first page of the kickstarter website), and 4 variables, as our test data. We firstly modified the test data to the variables we used in our predication model: launch month, deadline month, goal and days(duration between launch date and deadline date). However, when we conducted the predication on test data, the results showed that all projects as failed. This was not because the model was poorly built, but because we were not able to match the encoding for 'category' variable to those encodings in our train dataset variable 'category'. We were not able to find the way to solve the issue here with our current modeling and coding skills, but in the future, we would like to think more predictively when we build model and test more to avoid this kind of problems from happening.

	category	launched_m	deadline_m	goal	days	predictions
0	5	1	0	5000	35	fail
1	5	1	1	1110000	32	fail
2	2	1	0	11618	34	fail
3	7	1	0	50000	22	fail
4	0	1	0	12500	30	fail
5	9	2	1	8000	30	fail
6	1	1	0	5000	24	fail
7	6	2	0	15000	28	fail
8	3	1	1	500000	59	fail
9	8	0	0	40000	58	fail
10	9	1	0	42069	32	fail
11	4	1	0	30000	30	fail

Table 2 Prediction results based on the test data from web-crawling

Results

Summary Stats Between Successful and Failed Projects

We wanted to explore the factors that differentiated successful and failed projects. We analyzed a couple of summary statistics for the whole dataset. As shown in Table 1, our dataset contains more failed projects (58%) than successful projects (42%). The average number of backers (271) of the successful projects is significantly higher than that (17) of the failed projects. In generally, successful projects asked less and achieved more than failed projects.

Number of Projects	152061	109299	261360
Proportion	58%	42%	100%
Number of Backers(avg)	16.77	270.18	122.74
Number of Backers(std)	73.43	1593.15	1039.32
Duration(days avg)	35.48	32.39	34.19
Duration(days std)	13.51	12.03	13
Goal(avg)	$ 60,664.24	$ 9,695.67	$ 39,349.53
Goal(std)	$ 1,356,864.41	$ 28,790.07	$ 1,035,436.64
Pledged(avg)	$ 1,331.17	$ 23,212.89	$ 10,481.96
Pledged(std)	$ 6,999.57	$ 160,709.99	$ 104,622.74

Table 2 Summaries for successful and failed projects

Impact of Project Category type

We used the "main_category" and the "sub_category" column in our data set to group the data and observe dynamics that the categorization of projects introduce. We compared the difference between the success rates across main categories. The results are given in Figure 2.1, 2.2 and 2.3. As the Figure 2.1 shows, the top 3 most popular main categories of Kickstarter projects were Film& Video, Music and Publishing, while Dance, journalism and crafts had the least project counts in the total 15 main categories with less than 10k projects for each category. When we looked at the popularities of the subcategories in these Kickstarter projects in Figure 2.2, product design which was one of the subcategories of design was the most popular subcategory in our dataset, with almost 14k project counts.

However, the most popular categories do not turn out to be the most successful ones. The top 3 main categories with highest success rate were Dance, Theater, and Comics as showed in Figure 2.3. Projects in Technology, Journalism and Crafts categories had the highest failure rates.

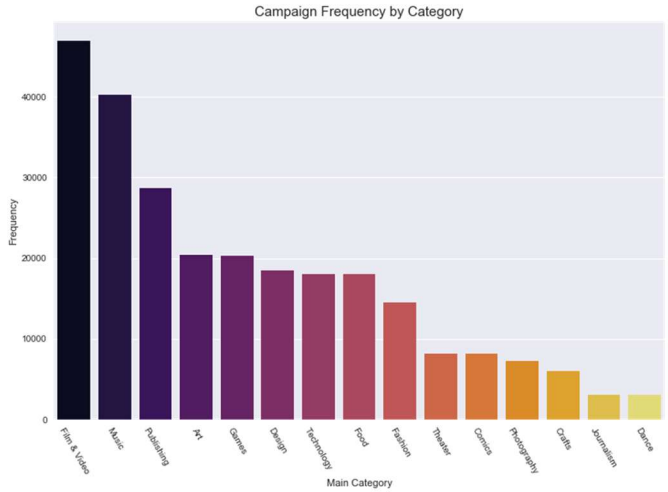


Figure 2.1 Project frequencies by main category

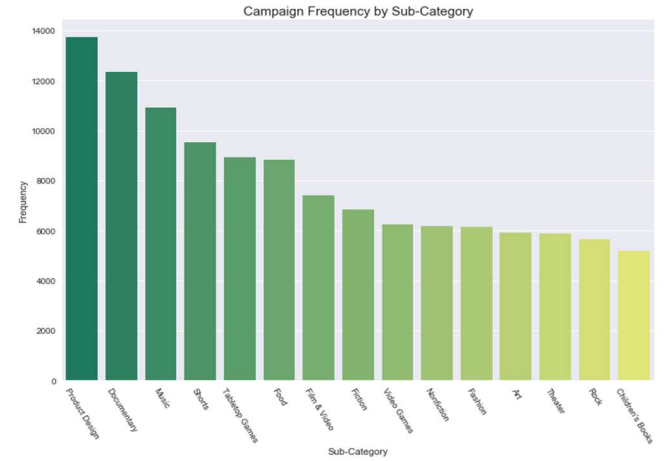


Figure 2.2 Project frequencies by subcategory

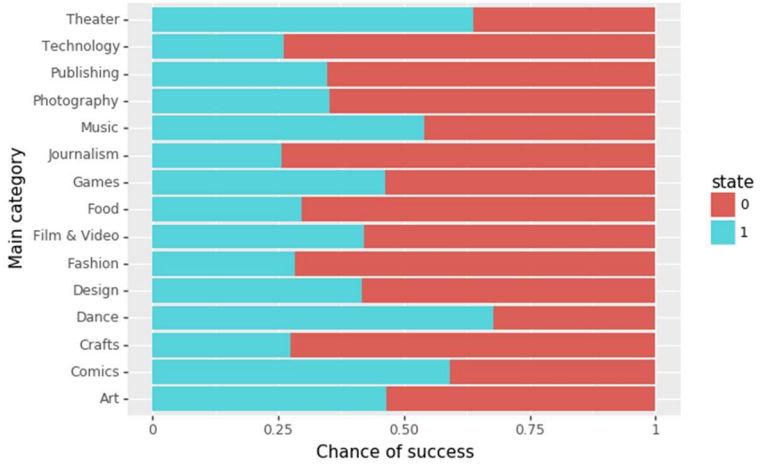
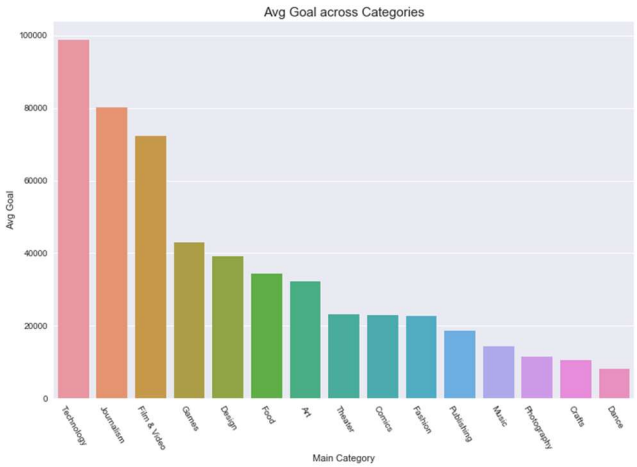


Figure 2.3 Project success and fail rates by main categories

Project Attributes across Categories

We further analyzed the dynamics of project categorization with other project attributes such as average goal (Figure 3.1), average pledged amount per individual (Figure 3.2), average backers and average days (Figure 3.3). The average amount of goal setted by the requesters varies siginificantly across 15 categories. Technology had the highest average goal, appromiaxiately 100k, while the lowest aveage goal for main category Dance was less than 10k. It's fair to conclude that projects with modest goals tends to have a higher chance of success. The Figure 3.2 shows that the amount of money pledged per individual by categories. Technology also had the highest average pledged per individual, with more than $ 120 per backer. Backers in most other categories funded similar average amount per person, while people who financed comics projects gave the least amount per person, which is around $50. Looking at the average number of people who donate to a project i.e. backers and the average amount spent by each backer, it seems like there are more people who are motivated to donate money in the space of Games, Product Design and Technology. In fact, Tech projects get some of the most generous individual pledges, on average $130 per backer. Despite that, these categories have relatively high failure rates.

The last chart showed that successful projects took shorter time to be completed than failed projects in the same category. In the 30-38 day period from when a project launches, if a project doesn't attain at least 100 backers, the project will likely fail. We can also see that projects in the Dance category are likely to achieve



success faster than the other categories as long as it meets the average backer threshold number.

Figure 3.1 Average goal by main category

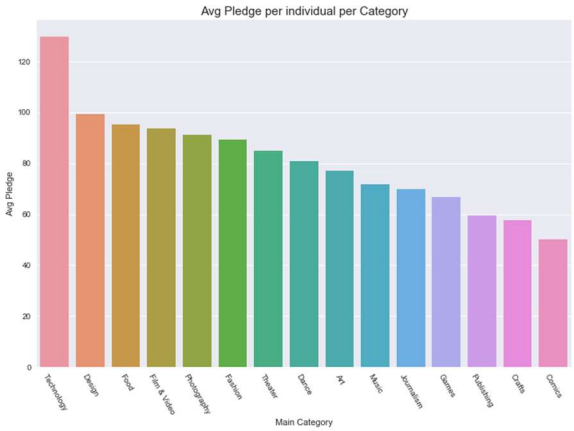


Figure 3.2 Average pledged amount per individual by main category

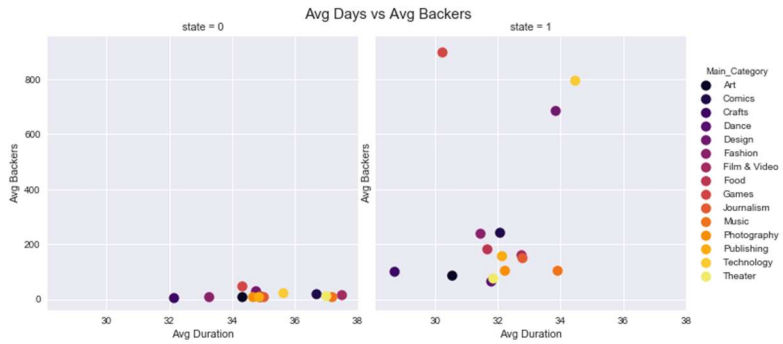


Figure 3.3 Average days vs average backers by successful and failed projects by main category

The Impact of Goal Amount on the Outcome of a Project

Does a higher goal decrease your project's success rate on Kickstarter? The following figure shows that the higher the goal was in our dataset; the less proportion of its goal would be funded. Most projects with a goal that under 50k achieved their goal and a lot of them received more than they expected, however, projects with a goal higher than 100k mostly failed to achieve their goal.

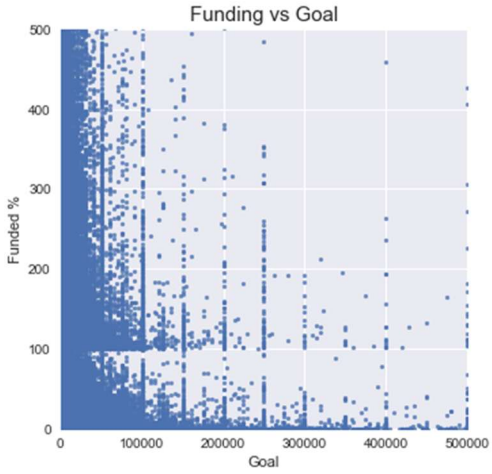


Figure 3.3 Funded percentage by goal

Impact of Time, Project Launches through the Years

We then wanted to know whether time has an impact on the success rate of the project in our dataset. We have two time related variables in our dataset: launch date and duration(days). Based on launch date, we analyzed the project success rate over years (Figure 4.1) and project distributions over months (Figure 4.2). According to the Figure 4.1, the number of projects increased since 2009 and reached the highest point at around 45k, in 2015, however, the success rate in each year followed an opposite trend, which decreased over years and reached its lowest point at 2015. There were less projects launched during recent two years, however, the projects launched in 2016 and 2017 were more likely to be successful compared with 2015. As for the month factor, projects posted in December were least likely to be successful. Figure 4.3 is our analysis on

project distribution by duration. Majority of the Kickstarter projects have an average lifetime spanning between 30-38 days. 60% of the most successful projects have had a lifetime of 32-36 days.

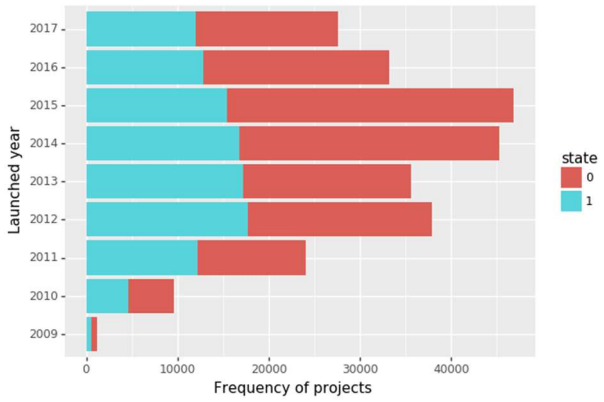


Figure 4.1 Project launched by year

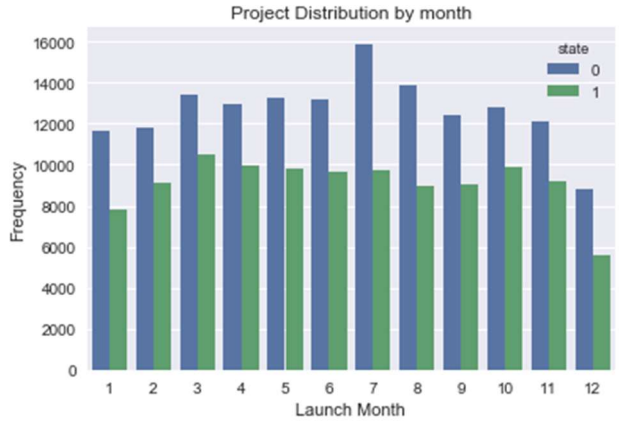


Figure 4.2 Average funded percentage across month

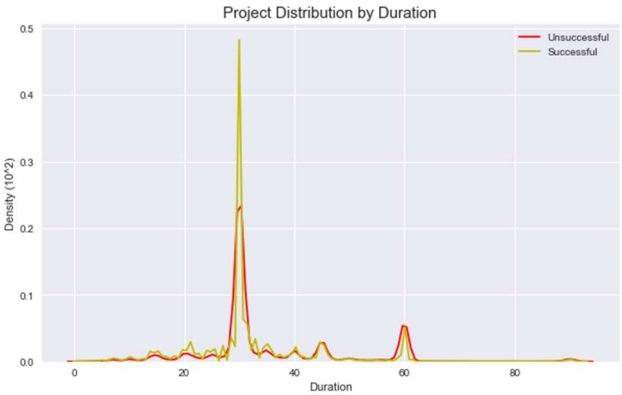


Figure 4.3 Project distribution by duration

Testing

To start the testing, we imported the unittest package in python and created a class with methods that tested each of the functionalities in the code. The first method, test_for_file(), tests whether the .csv file containing our Kickstart data was loaded into the assigned variable correctly. It does this in two ways. First it extracts all the column headers in the dataset and uses the assertEqual() statement to check it against the expected result. Then it uses the assertTrue() method to check if the length of the loaded set is the same as that of the original file. The second method, test_for_filter(), checks to see if the filtering has been done correctly. In the main code, we have removed all the unnecessary columns and rows. The test function implements this and checks to see if the filtered headers match the column headers for the filtered dataset. In the main code we have then removed un-used categories from the Kickstart['state'] column and replace 'successful' and 'unsuccessful' with (0, 1) respectively. To see if this has executed correctly, we have added the third test method, test_for_rename(). In this it calls the function we used to perform this part of the pre-processing and uses the assertEqual() method to check if the extracted values are the same as the expected. The final method in the testing class tests whether the pre-processed data gets written back into a .csv file correctly. It does this by checking calling a function in the assertTrue() method to check if the file exists in the system or not.

Some code snippets:

```
class PTestCases(unittest.TestCase):

    def test_for_file(self):
        #read data into csv
        csv = pd.read_csv('C:/Users/Dell/Desktop/UVA DSI notes/CS/Project/ks-pro
jects-201801.csv', dtype = {'category': 'category', 'main_category': 'categor
y', 'state': 'category'}, index_col = 0)
        #get column headers
        head = list(islice(csv,14))
        #see if headers match
        self.assertEqual(head, ['name', 'category', 'main_category', 'currenc
y', 'deadline', 'goal', 'launched', 'pledged', 'state', 'backers', 'country', 'u
sd pledged', 'usd_pledged_real', 'usd_goal_real'], 'not working!')
        #see if length matches
        self.assertTrue(len(csv) == 378661, 'Not working')
```

```
    def test_for_rename_categories(self):
        csv = pd.read_csv('C:/Users/Dell/Desktop/UVA DSI notes/CS/Project/ks-projects-201801.csv', dty
pe = {'category': 'category', 'main_category': 'category', 'state': 'category'}, index_col = 0)
        data = csv.query('country == "US"').loc[:, ['category', 'main_category', 'deadline', 'goal',
'launched', 'pledged', 'state', 'backers']].query('state == "successful" or state == "failed"')
        #print(type(data['state']))
        #data['state'].astype('category')
        data['state'] = data['state'].cat.remove_unused_categories().cat.rename_categories([0, 1])
        #print(data['state'])
        #print(data['state'].cat.categories)
        self.assertEqual(data['state'].cat.categories.tolist(), [0, 1], 'Error!')
```

Conclusions

We have been able to leverage this data to gain insight into Kickstarter project outcomes and understand that there are many nuances to the outcome or the success of a project and it's not just about size of a project or the general popularity of the domain under which it falls.
Some of the main and specific takeaways from our analysis are :

- Projects that achieve success generally have a duration of 32-38 days.
- Projects that have a goal set to < $ 50k have better chances of being funded and even being overfunded up to 5 times it's goal.
- Projects in the artistic space such as Dance, Music, Theatre have better odds of succeeding owing to the shorter duration and modest goals set.
- Projects in categories such as Tech, Design are harder to fund despite having a larger base of backers owing to high goals set for such projects. That being said, owing to the same characteristics , such projects that have a modest goal will have an excellent chance to succeed.
- The data does have predictive power. Our model using random forecast classification yielded an accuracy rate of 64% in predicting the outcome of projects.

We would like to do the following to enhance our analysis and improve our accuracy in predicting the outcome of projects on Kickstarter-

- Source data that contains more attributes which can help predict better such as project origin city, social media tags, user history, project description which based on our research correlate strongly to project outcomes.
- Leveraging advanced feature engineering and techniques of classification.
- Robust validation in determining model performance.
- Handle outlier detection in a sophisticated way.
- Further functionality -> Predict the time that a project can take to achieve it's goal. To also predict the number of backers a project would require to meet its funding requirement.

Our analysis would be useful to -

- People or Creators who want to check how their project or campaign can perform on Kickstarter before they invest time and energy in launching it on the website. And to be able to accordingly adjust project specs and goal which could help the maximize their chances of success.
- People or Backers who want to determine the outcome of project based on it's current state before they commit to donating money.

Resources

Dataset:

https://www.kaggle.com/kemical/kickstarter-projects

Bokeh doc:

https://bokeh.pydata.org/en/latest/

Web-crawling:

https://www.kickstarter.com/

Analysis Ideas:

https://www.cbc.ca/news/canada/montreal/what-10-000-kickstarter-projects-reveal-about-canada-s-entrepreneurs-1.4084372

https://towardsdatascience.com/predicting-the-success-of-kickstarter-campaigns-3f4a976419b9

Plotting (seaborn documentation):

https://seaborn.pydata.org/