slide_deck_template

August 2, 2019

1 Kickstarter Campaigns and their Attributes

1.1 by Michael Mosin

1.2 Investigation Overview

In this investigation, I wanted to look at the various attributes available about Kickstarter Project Campaigns, and explore their relationship to campaign outcomes - success, and various degrees of failure. The main focus was on the number of backers, the proportion of the funding goal that was achieved, whether the project was a Staff Pick, and average donation per backer. Project categories and their relationships with these variables were also explored.

1.3 Dataset Overview

The original data consisted of approximately 3786 projects and 37 variables. It was reduced to 3669 projects with the removal of projects that were still 'live' and therefore yet to have a defined outcome. The proportion-of-goal-funded variable was contructed from the variables 'goal' and 'pledged,' while project categories and subcategories were extracted from an originally untidy 'category' variable. There were also variables related to the length of time the project existed.

```python
[1]: # import all packages and set plots to be embedded inline
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb

%matplotlib inline

# suppress warnings from final output
import warnings
warnings.simplefilter("ignore")
```

```python
[2]: # load in the dataset into a pandas dataframe
campaigns = pd.read_csv('final_dataset.csv')
```

1.4 Distribution of Project Campaign States

First and foremost, it would be appropriate to become familiar with the distribution of campaign outcomes. Doing so we learned that a majority of campaigns are indeed successful.

```
[3]: base_color = sb.color_palette()[0]
     ax = sb.countplot(data = campaigns, x = 'state', color = base_color);
     plt.suptitle('Counts of Campaign Outcomes')
     plt.xlabel('Final Campaign State')
     plt.ylabel('Count')
     for p in ax.patches:
         height = p.get_height()
         ax.text(p.get_x()+p.get_width()/2.,
                 height + 20,
                 '{:.0f}'.format(height),
                 ha = 'center')
     plt.show();
```

Counts of Campaign Outcomes
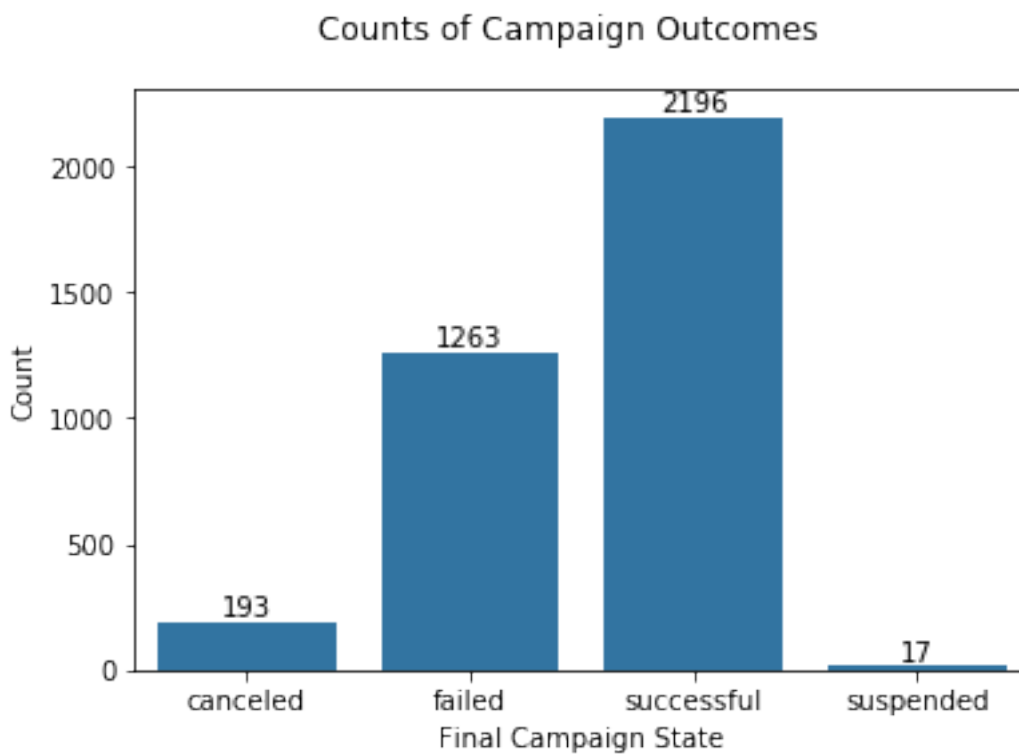
1.5 Number of Backers per Campaign

Now, the question on my mind was, "How many people does it take to successfully or unsuccessfully fund a campaign?"

Originally, the data displayed a VERY wide exponential distribution of backers skewed by many projects having zero or few backers, and by a few outlier projects with tens of thousands of backers.

This led to conducting a log transformation of the x-scale, which demonstrated a bimodal distribution - around 0 to 1 backers for the unsuccessful campaigns, and around 40 to 60 backers for the successful campaigns.

```
[4]: df_successful = campaigns[campaigns['successful']==True]
     df_notsuccess = campaigns[campaigns['successful']==False]

     log_binsize = 0.1
     bins = 10 ** np.arange(0, np.log10(df_successful['backers_count'].
      →max())+log_binsize, log_binsize)

     plt.figure(figsize=[8, 5])
     plt.hist(data = df_successful, x = 'backers_count', bins = bins, alpha=0.5,␣
      →label='Successful')
     plt.hist(data = df_notsuccess, x = 'backers_count', bins = bins, alpha=0.5,␣
      →label='Not Successful')
     plt.xscale('log')
     plt.xticks([1, 3, 10, 30, 100, 300, 1e3, 3e3, 1e4, 3e4], [1, 3, 10, 30, 100,␣
      →300, '1k', '3k', '10k', '30k'])
     plt.xlabel('Number of Backers')
     plt.ylabel('Number of Projects')
     plt.suptitle('Log Transformed Histogram of Number of Backers\n(exclusive of␣
      →zero)', fontsize=16)
     plt.legend(loc='upper right')
     plt.show();
```


Log Transformed Histogram of Number of Backers
(exclusive of zero)
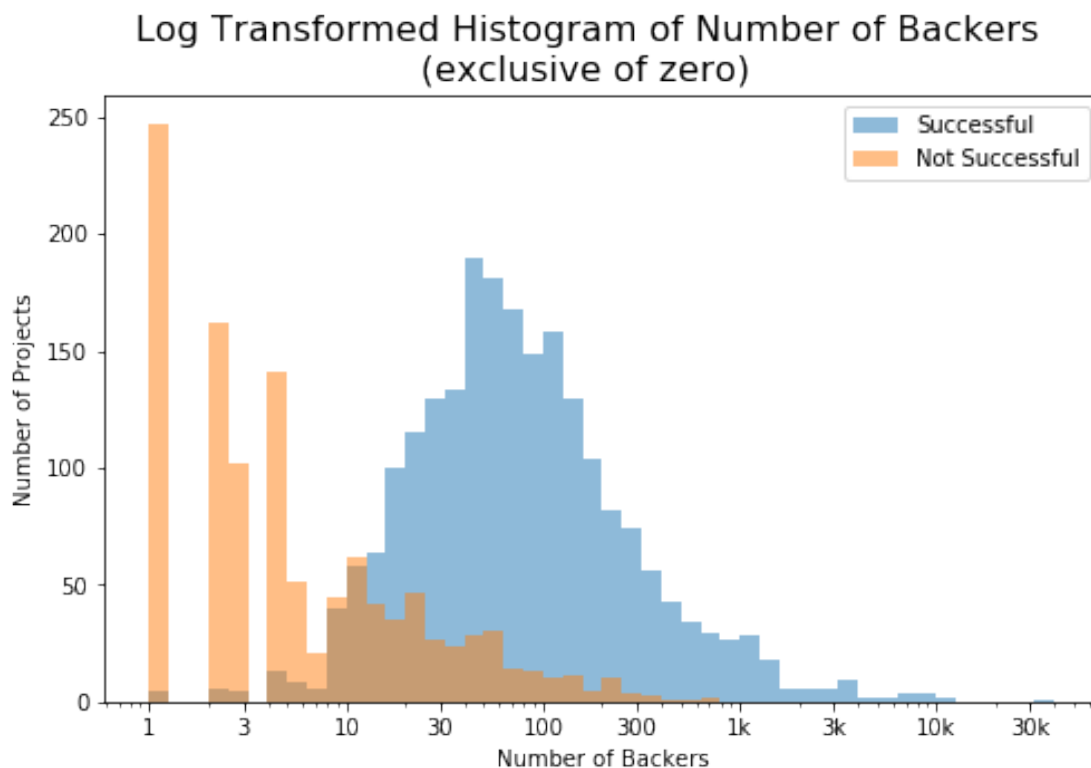
1.6 Distribution of Final Funding as Proportion of Goal

Next: What do the distributions of funding proportions look like for successful and unsuccessful campaigns?

Final Funding as Proportion of Goal = pledged amount / goal amount.

Since a successful campaign is defined as one which met its funding goal, then it comes as no surprise that the split occurs at the proportion of "1": all campaigns inclusive and higher are successful, all below - not so much.

What is interesting is the exponential nature of both segments. I did conduct a log transform but it still left a curving exponential distribution due to the large range of successful proportions and the large density of campaigns with proportion values of 0 and 1.

```
[5]: binsize = 0.1
bins = np.arange(0, 5+binsize, binsize)

plt.figure(figsize=[8, 5])
plt.hist(data = df_successful, x = 'funded_prop', bins = bins, alpha=0.8,␣
 ↪label='Successful')
plt.hist(data = df_notsuccess, x = 'funded_prop', bins = bins, alpha=0.8,␣
 ↪label='Not Successful')
plt.xlabel('Final Funding as Proportion of Goal')
plt.ylabel('Number of Projects')
plt.suptitle('Histograms for Final Funding as Proportion of Goal', fontsize=16)
plt.legend(loc='upper right')
plt.show()
```
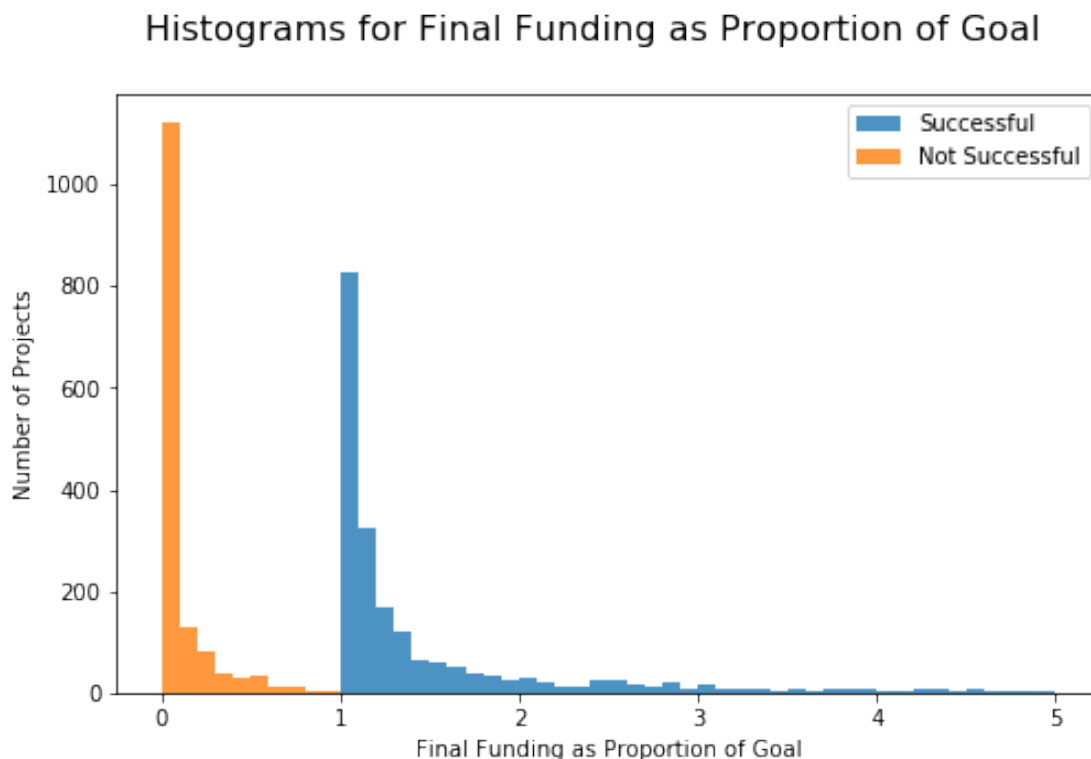
1.7 Staff Pick versus State of Campaign

Staff Pick is a means of playing favorites, I presume, so perhaps there is an effect on success should a project be deemed worthy of the designation.

Based on the following bar charts, it does appear that projects which are selected as Staff Picks are more likely to succeed than those that aren't. In fact, almost a fifth of successful campaigns are designated as Staff Pick. Of course, there were a few Staff Picks that failed or were suspended.

```python
table=pd.crosstab(campaigns.staff_pick,campaigns.state)
table = table.div(table.sum(1).astype(float), axis=0)

fig, axarr = plt.subplots(1, 2, figsize=(12, 6))
plt.suptitle('Final State of Campaign and Staff Pick', y =1.0, fontsize=16)

table.plot(kind='bar', stacked=True, ax=axarr[0])
axarr[0].set_title('Stacked Bar Chart of Staff Pick vs State of Campaign')
axarr[0].set_xlabel('Staff Pick')
axarr[0].set_ylabel('Proportion of Campaigns');

axarr[1] = sb.countplot(data = campaigns, x = 'state', hue = 'staff_pick',
 ↪order = campaigns['state'].value_counts().index)
n_points = campaigns['state'].value_counts()
i=0
for p in axarr[1].patches[0:7]:
    if i >= 4:
        i = 0
        height = p.get_height()
        axarr[1].text(p.get_x()+p.get_width()/2., height + 5, '{:.2f}'.
 ↪format(height/n_points[i]), ha = 'center')
        i += 1
    else:
        height = p.get_height()
        axarr[1].text(p.get_x()+p.get_width()/2., height + 5, '{:.2f}'.
 ↪format(height/n_points[i]), ha = 'center')
        i +=1
plt.title('Final Project State split by Staff Pick\n(intra-categorical
 ↪proportions over bars)')
plt.xlabel('Final Project State')
plt.ylabel('Count')
```
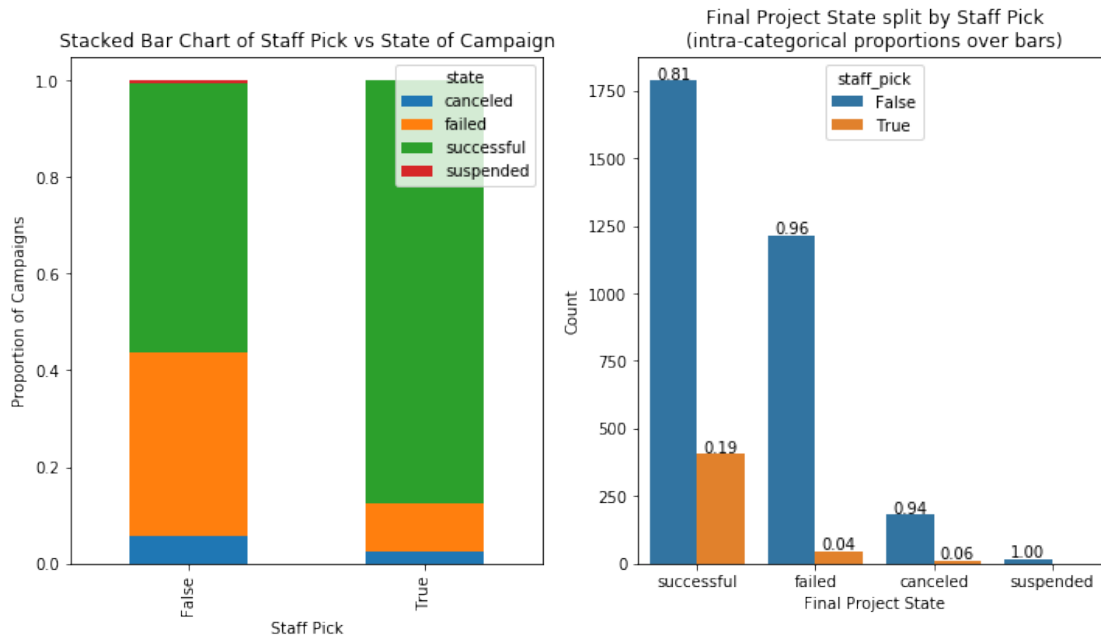
[6]: Text(0, 0.5, 'Count')

Final State of Campaign and Staff Pick

1.8 Differences in Success Rates Among Main Project Categories

Okay, so we have looked at backers, funding, and Staff Pick. My consequent interest was in seeing how projects in different categories differ between each other in terms of successfully acheiving funding goals.

As it turns out, projects in the Music, Technology, Publishing, and Theater categories are popular among both creators and backers.

However, even some less common categories have relatively similar success rates, if not better: Comics, Fashion, Film & Video, Art, Photography.

A few that are less successful are Food, Crafts, Dance, Games, Design, Journalism.

```
[7]: plt.figure(figsize=[15, 5])
     ax = sb.countplot(data = campaigns, x = 'main_cat', hue = 'successful', order=
      ↪campaigns['main_cat'].value_counts().index);
     loc, labels = plt.xticks()
     ax.set_xticklabels(labels, rotation=45);
     handles, labels = ax.get_legend_handles_labels()
     ax.legend(handles[::-1], labels[::-1], title='Line', loc='upper right')
     n_points = campaigns['main_cat'].value_counts()
     i=0
     for p in ax.patches:
         if i >= 15:
             i = 0
             height = p.get_height()
             ax.text(p.get_x()+p.get_width()/2., height + 5, '{:.2f}'.format(height/
      ↪n_points[i]), ha = 'center')
```

```
        i += 1
    else:
        height = p.get_height()
        ax.text(p.get_x()+p.get_width()/2., height + 5, '{:.2f}'.format(height/
    →n_points[i]), ha = 'center')
        i +=1
plt.suptitle('Main Project Categories Split by Success\n(intra-categorical␣
    →proportions over bars)', fontsize=16)
plt.xlabel('Main Project Categories')
plt.ylabel('Count')
plt.show();
```
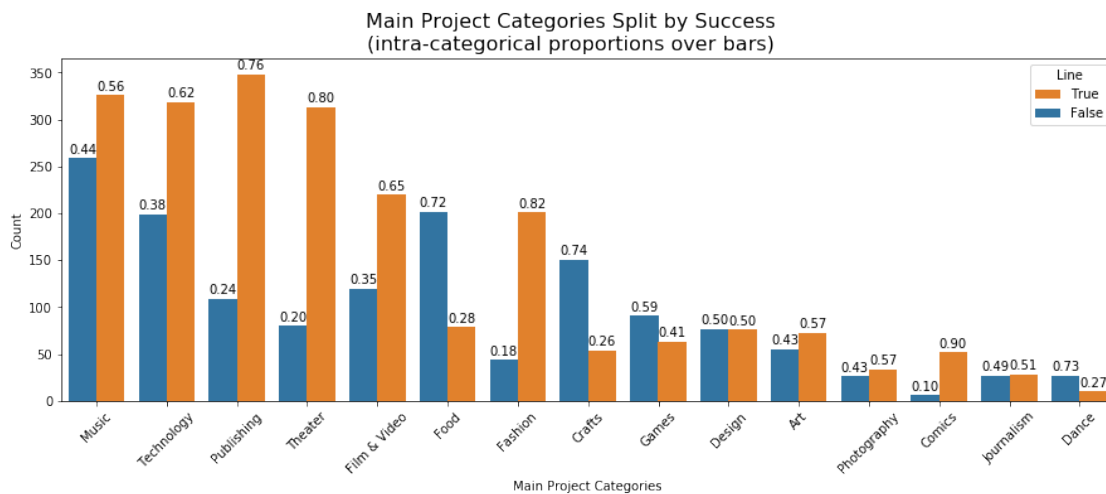

1.9 Distribution of Main Project Categories relative to Funded Ratio

We saw the categories split by rates of success. Now, we can check the spread of where they fall in terms of meeting or exceeding their funding goal.

In the associated graph, looking at the median point for the boxplots relative to the "1" point on the y-axis, we can see the proportion of projects within each Main Category which met their goal or didn't. Since meeting one's funding goal (getting at least "1" for the funding proportion) defines a Kickstarter campaign success, we can see whether a majority of campaigns succeeded or failed.

```
[8]: plt.figure(figsize=[15, 7])
ax = sb.boxplot(x='main_cat', y='funded_prop', data=campaigns,␣
    →color=base_color, order = campaigns['main_cat'].value_counts().index)
loc, labels = plt.xticks()
ax.set_xticklabels(labels, rotation=45);
plt.suptitle('Distributions of Main Project Categories relative to their␣
    →Funding Goal Ratio', y = .93, fontsize=16)
plt.xlabel('Main Project Categories')
plt.ylabel('Final Funding as Proportion of Goal')
```

```
plt.ylim(-0.25,7);
```


Distributions of Main Project Categories relative to their Funding Goal Ratio
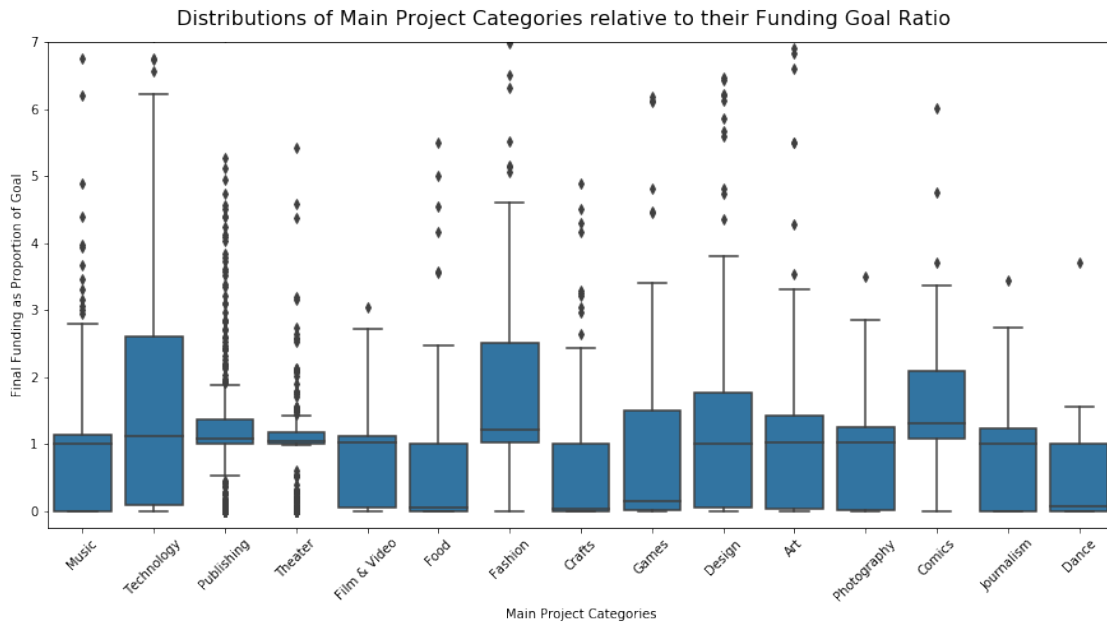
1.10 Spread of projects by their Final State, relative to Proportion of Goal Funded and Number of Backers:

This visualization displays what is to be expected at this point: campaigns that met or exceeded their goal (proportion funded >= 1) are successful. Those that were not able to do so were not successful.

```
[9]: sb.pairplot(x_vars=['funded_prop'], y_vars=['backers_count'], data=campaigns,␣
     ↪hue = 'state', height = 6);
     plt.xlim([0,4]);
     plt.ylim([0,1800]);
     plt.suptitle('Spread of Projects relative to Funding Proportion\n and Number of␣
     ↪Backers split by Final State',y= 1.05, fontsize=16)
     plt.ylabel('Number of Backers')
     plt.xlabel('Final Funding as Proportion of Goal');
```

Spread of Projects relative to Funding Proportion and Number of Backers split by Final State
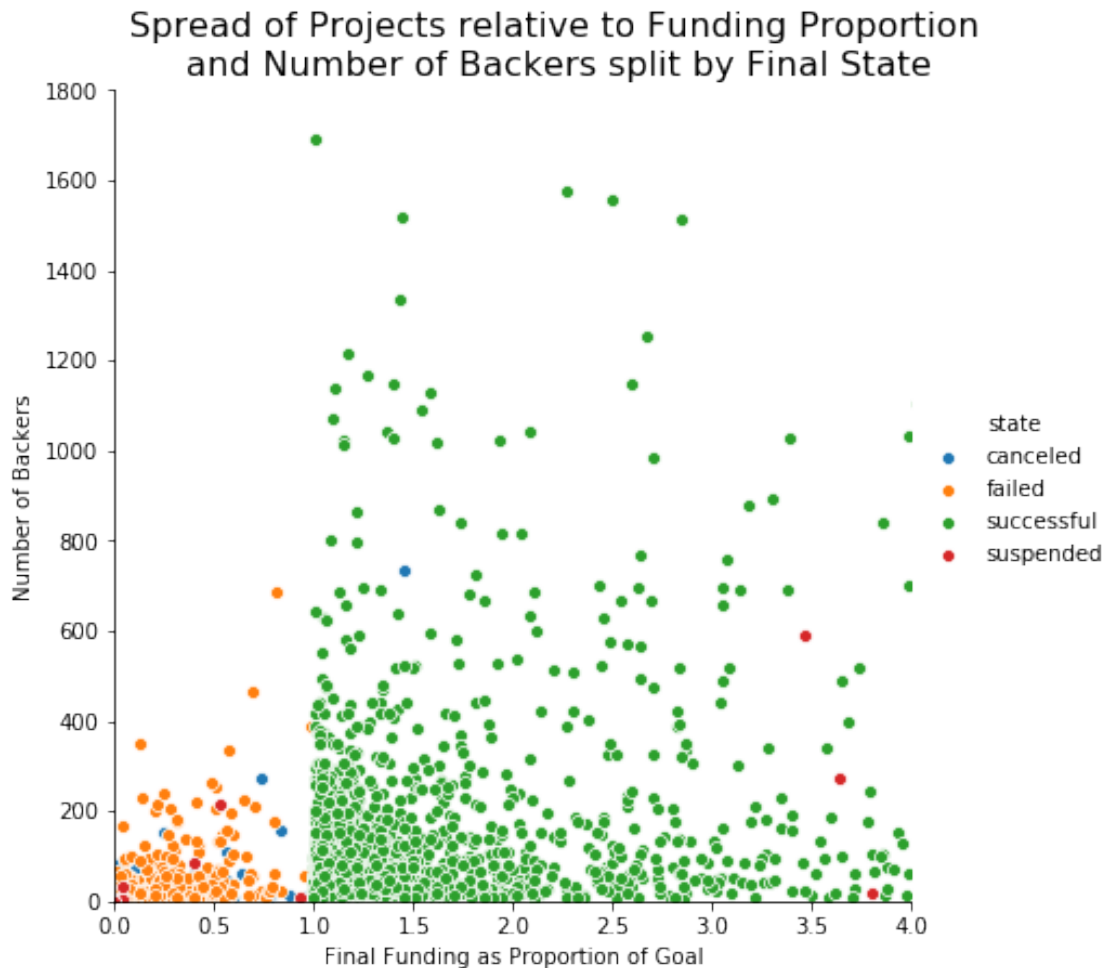
1.11 Comparing spread of projects by their Success relative to Average Donation and Number of Backers - Technology vs Games:

I selected these two categories because they were relative opposites in terms of proportion of successful projects.

In both cases, the data is very spread out, even after zooming in upon a decently dense area. There is a slight impression that the fewer backers a project has, the less likely it will be successful, even with high average donations.

```
[10]: #g = sb.FacetGrid(data = pokemon_sub, hue = 'type', size = 5, palette =␣
      ↪type_colors)
      #g.map(plt.scatter, 'weight','height')

      ax1 = sb.FacetGrid(data=campaigns[campaigns['main_cat'].isin(['Technology',␣
      ↪'Games'])],
                         col='main_cat', hue = 'successful', height = 6);
      ax1 = ax1.map(plt.scatter,
```

```
                'avg_donation',
                'backers_count').set_axis_labels("Average Donation ($)",
                                                  "Number of Backers").
  ↪set_titles('{col_name}')
plt.xlim([0,500])
plt.ylim([0,1200])
plt.suptitle('Success Relative to Average Donation vs Number of Backers', y= 1.
  ↪05, fontsize = 16)
plt.legend(['No','Yes'], title = 'Successful');
```


Success Relative to Average Donation vs Number of Backers