Lorraine Zhou, Sahil Natu, Bret Jaco, Jonathan Jan

MIS s381N - Project Report

Group 10

Description of Project Goals

1. Description

Investing your money in a company can be a very risky but rewarding endeavor. Our report analyzes a large dataset of Kickstarter campaigns to discover what factors contribute to successful campaigns and predict whether a given campaign will meet its target funding goal. Kickstarter is a global crowdfunding platform where entrepreneurs post their product ideas in the hopes of gathering investors. In return, the investors receive the first editions of the entrepreneur's product.

2. Importance of the Problem

Investors are constantly looking for extra information and trends to base their decision making off of. Often, Kickstarter campaigns fail to reach their goal, or are cancelled due to poor management. This wastes not only the investors' money, but also their time, with many campaigns promising to deliver products for months before shutting down. Our analysis attempts to simplify the process by providing valuable insight into the factors that contribute to a successful Kickstarter campaign. This insight would be very beneficial to the millions of investors and entrepreneurs currently participating in Kickstarter crowdfunding.

Exploratory Analysis

Upon sourcing our dataset on Kickstarter campaigns from Kaggle, we performed exploratory data analysis to unearth patterns and quirks in the data. The observations are further discussed below.

This dataset includes 378661 data records and 15 columns in total, which is composed of 5 numerical features, 9 categorical features and one target variable, which is the state of the funding campaign. Some important variables are Funding Goal, Data Launched, Number of Backers, and Pledged Amount, etc. Below are some insights we've gotten from running exploratory data analysis on this dataset:

Main Category *(See fig 2.a)*: The data shows quite a bit of variation in project fundraising success based on the Main Category. Dance and Theater have a high success rate of about 65% while Journalism and Technology have a low success rate of 25%. Mean success rate for the data is 40%.

Country *(See fig 2.b)*: Maximum number of projects are from the US, with Great Britain being a distant second. In fact, the US accounts for over 79% of total projects in the data.
Backers *(See fig 2.c)*: As the number of backers backing the project increases, the likelihood of a project succeeding is more than that of it failing. In this data, more projects succeed than fail as the number of backers increases beyond 21.
Launch Year *(See fig 2.d)*: The number of projects each year has been rising, both successful and failed. The dip after 2015 can be explained by the fact that many projects would still be on-going and thus not captured in the data taken in 2018.
Goal Amount (USD) *(See fig 2.e)*: Projects with a goal below USD 1000 have a more likelihood of success than failure. This trend reverses for projects with a goal above USD 1000. Thus, projects with lesser goal amount succeed more often.
Goal Amount vs Pledged Amount *(See fig 2.f)*:The scatterplot for Goal vs Pledged amounts in USD shows that many projects have secured funding way beyond their goal amount (y=x line). Moreover, there seems to be a threshold for Goal amount at USD 2 million, with all projects with goals higher than that seem to have failed in their Kickstarter campaign.

Solution and Insights

Given that this is a classification problem, we decided to use predictive models that do a good job on predicting categorical variables.

But, before we fitted the models, we first did some data cleaning work on the raw dataset to make it ready for model fitting.

Data cleaning process:

To evaluate our data, some columns are numeric, and others are categorical. During the cleaning process, we initially checked for missing and duplicated rows and removed them. Moreover, there were no nulls on the dataset. For the second step, we processed some values in the ‘state’ column. We dropped the index where the state was ‘undefined’, ‘suspended’, and ‘canceled’. For the third step, since the columns listed below will not be useful in our prediction models, they had to be removed from our dataset:

Category, Launched, Deadline: These are very specific and we decided to use the main category and launch year to make our predictions.
ID, Name: These columns serve no purpose in training our models.
Currency: It is partially represented in the country data, uniting European countries to Euro currency. Therefore, we removed the currency column.

USD Pledged: This column is removed since we are considering the USD Pledged Real column which gives the same information.
Pledged, Goal: These columns are removed since we are considering Pledged and Goal amounts converted to USD in Pledged USD Real and Goal USD Real columns.

Lastly, we had to do dummy variable processing to transform the categorical variables into numerical data for the prediction models. There were three variables conducted by one-hot encoding (Main Category, Country and USD Pledged Real), and state was conducted by label encoding that we changed from 'successful' to 1 and from 'failed' to 0.

Predictive Modeling:

Besides the data cleaning process, there are two more preparatory steps we did before we actually fitted the models: first, we first set up independent variables X and dependent/target variable y, with X being the combination of all columns except "state" and y being the binary column "state" - we wanted to use all other variables in our updated dataset to predict if a row (a crowdfunding campaign) is successful or not. The second preparatory step is to split the data into training and testing sets, with training being the 70% rows in the full dataset and testing being the 30% remaining rows.

After we're done with the above preprocessing steps, we began our model fitting process. The first model we chose to run is the Logistic Regression model. Logistic regression is a model that has both computation efficiency and low over-fitting risk. We fitted this model with X and Y in the training dataset, and got a training accuracy rate of 0.8913, which is already pretty good. Then, we used this model to predict the 30% untouched data in the testing set, and got an accuracy rate of 0.8895, which is a little lower than the training accuracy rate but did not get hurt too much (only around 1%).

The next model we tried to fit was the Decision Trees model. Decision trees are easy to understand, and it selects features automatically, but sometimes it tends to overfit. Just like what we did in the logistic regression model, we fit our training data to the decision trees model, and got a prediction accuracy rate of 0.9879, which is extremely high and might be a sign of overfitting. And then we predicted the labels of data rows in the testing set, and got an accuracy rate of 0.9000. This result is 8% worse than training accuracy rate, confirming that the tree model is overfitting the training data, but 90% is also quite solid as test accuracy rate, indicating that our model is doing really well, and also beating logistic regression model.

The third classifier we used is the Random Forest model with maximum features = 10. Random forest is an ensemble model and also a tree-based model, but it is more robust to outliers and less likely to overfit. The accuracy rate from the training set is 0.9834, a little lower than the training accuracy rate of the single tree model. The test accuracy rate of random forest with max_features = 10 is 0.9180, the best test rate we’ve gotten so far. We've also tried some parameter tuning on this model, such as setting the maximum features parameter to automatic and none, but none of them work better than the first one of this classifier.

The last classifier we chose to use is the K-Nearest Neighbors model. To find the best k value, we ran a 3 fold cross validation for k values ranging from 1 to 100. As shown in Fig 4.e, a graph of accuracy versus k values, we found that when between k = 15 and k = 20, we get the best prediction accuracy. Then we tried knn with k in the range of 15 to 20 on the training set, and got the best test accuracy rate which is 92.30% when k = 17. *(See Fig 4.f)* This is a significant improvement over the baseline accuracy, but it is also a computationally expensive method. We’ve also tried the Naive Bayes classifier, but the prediction accuracy rate is 53.91%, not so much improvement from the baseline.

Based on all model results, the most accurately-predicting classifier we got is the KNN model with K = 17. Although it is a little time-consuming to run the knn model, the accuracy still offsets this small disadvantage in this project.

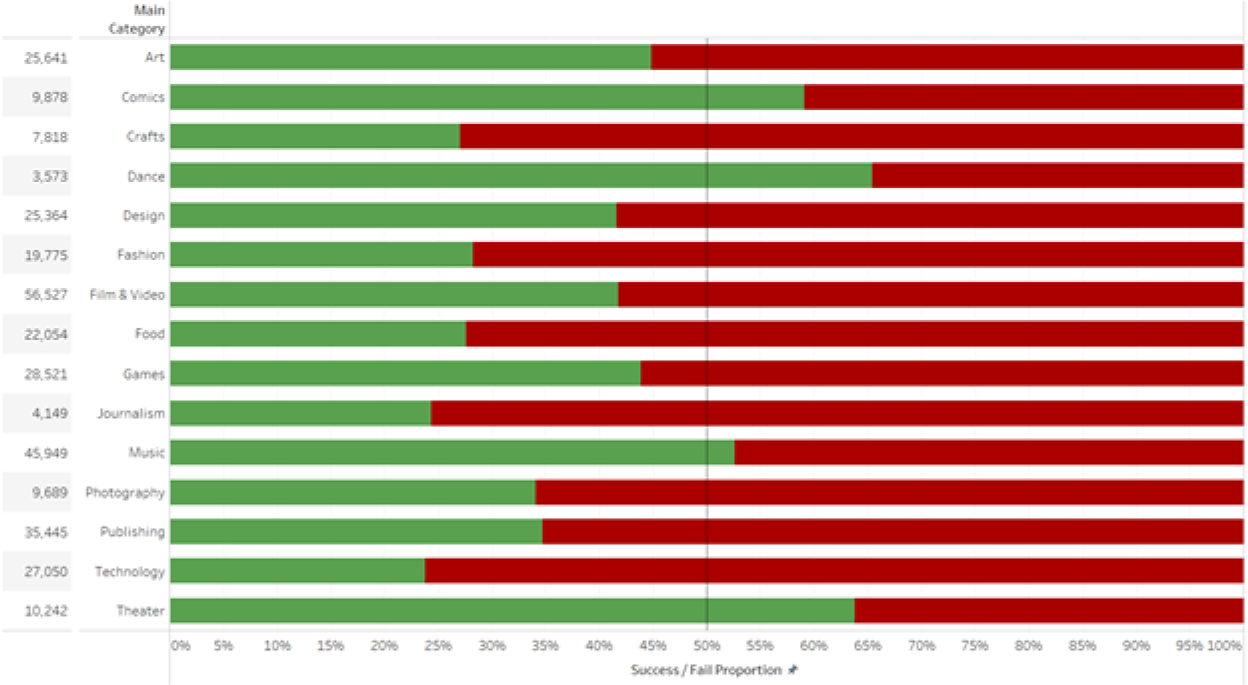
Besides the model fitting process, we’ve also looked into the importance of each feature for all classifiers except K-Nearest Neighbors. *(See Fig 4.g - Fig 4.i)* For logistic regression, the two most important variables are “backers” and “film and video category”. For decision tree model, the two most important variables are “backers” and “usd_goal_real”. And for the random forest model, the two most important variables are the same as those of the decision tree.

Conclusion

Overall, we believe our project provides a reasonable analysis on crowdfunding campaigns and the factors that influence them. It is a little surprising that some features are actually more important, such as number of backers, and we found ourselves a little off the mark in our initial intuitions about variable importance. Our final models suggest that besides number of backers, the funding goal and campaign category also play a big role in the success of crowdfunding. Thus, the exploratory data analysis coupled with the output from our classification models provides a reasonable estimation of whether a project would succeed in its crowdfunding goals, and also highlights some important factors for people to consider before starting a crowdfunding campaign.

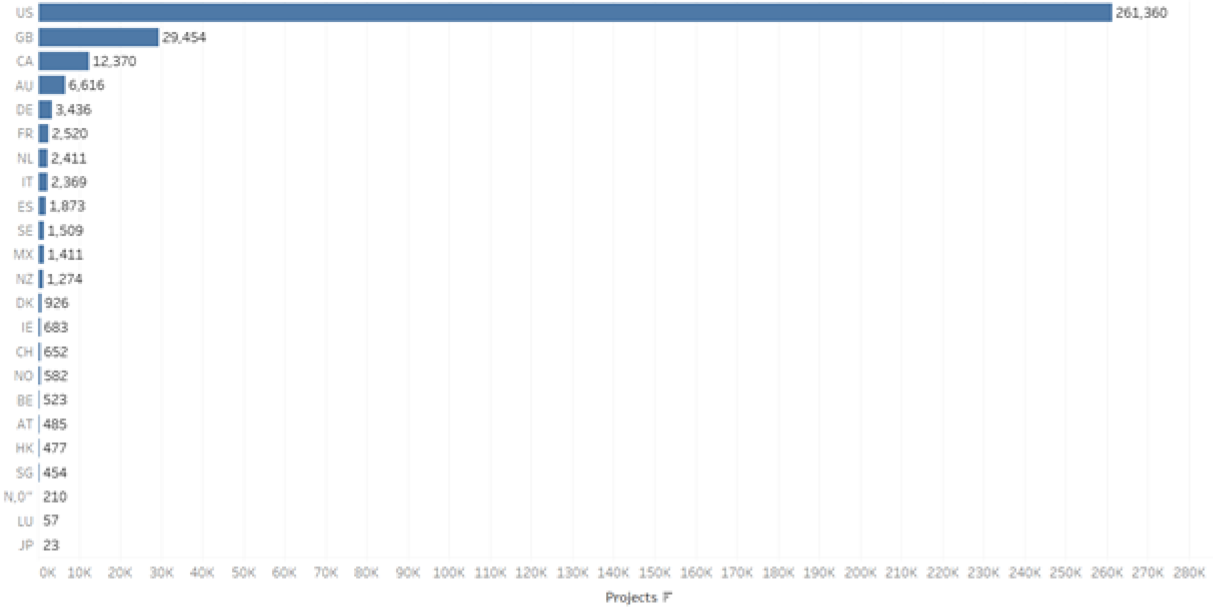
Appendix

Fig 2.a Main Project Category



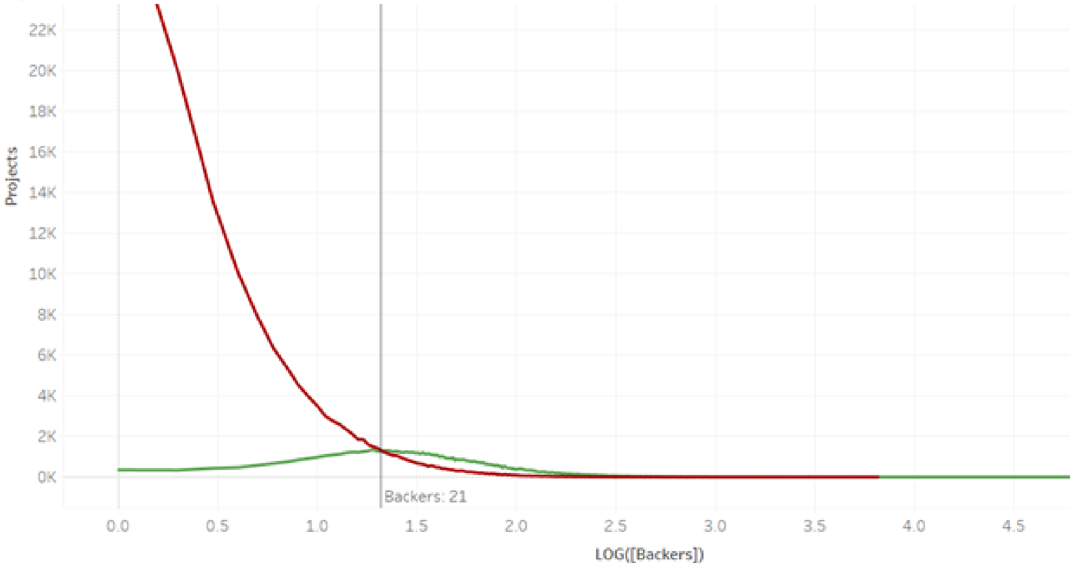
(First column shows absolute number of projects by Main Category. Green and Red bars represent success and failure rates of projects by Main category)

Fig 2.b Country



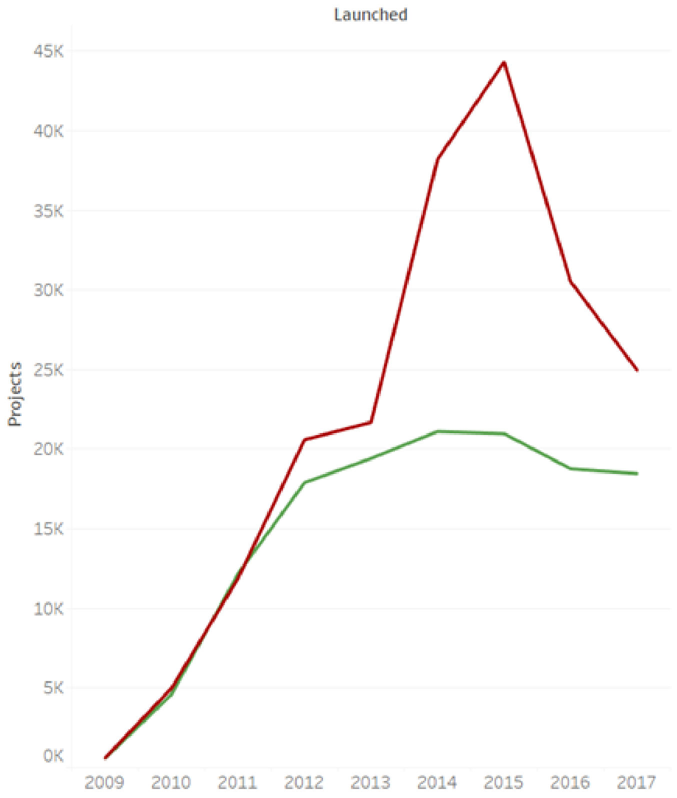
(Bars represent number of projects by Country.)

Fig 2.c Number of Backers



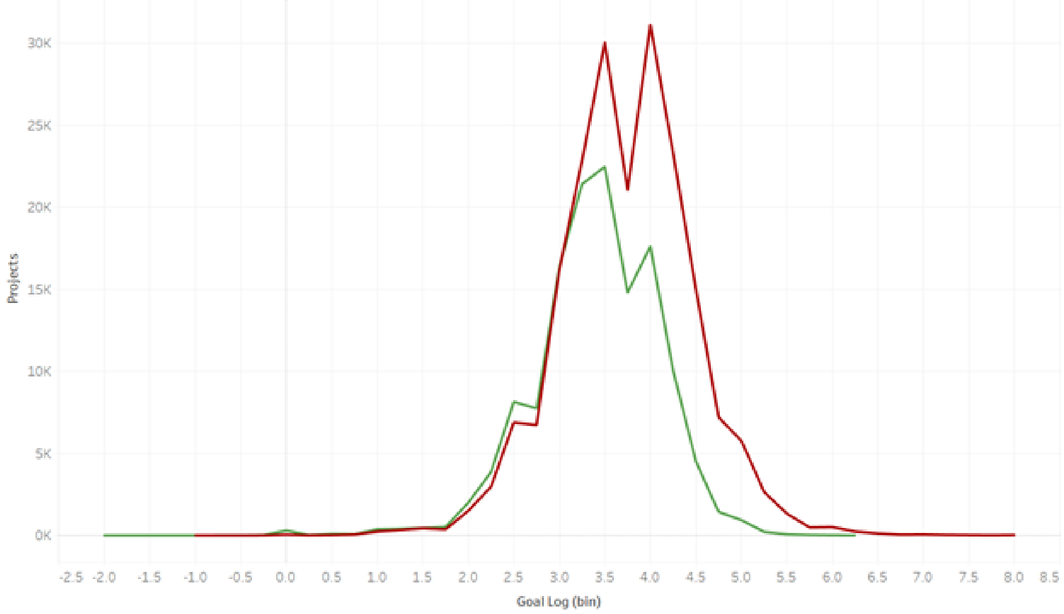
(Red and Green Line represent number of projects that have failed and succeeded respectively by number of backers (log). Reference line represents number of minimum number of backers past which more projects succeed rather than fail)

Fig 2.d Launch Year



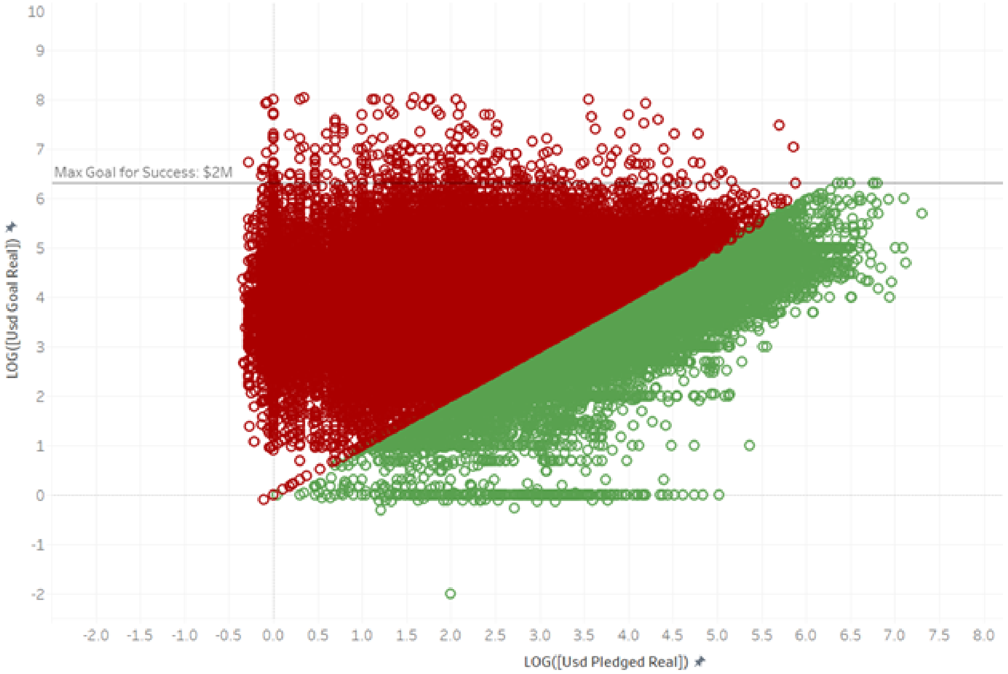
(Red and Green lines represent the number of projects that have failed and succeeded respectively by Launch Year)

Fig 2.e Goal Amount in USD



(Red and Green lines represent the number of projects that have failed and succeeded respectively by Goal Amount in USD (log))

Fig 2.f Goal Amount USD vs. Pledged Amount USD

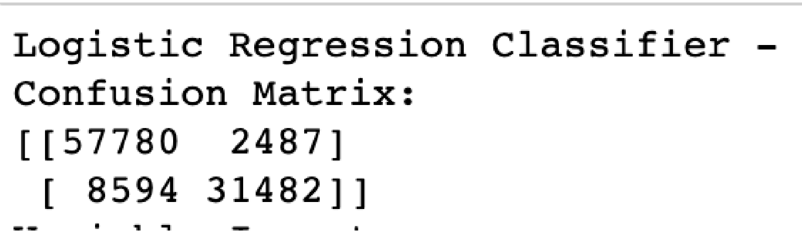


(Red and Green dots represent each project that has failed and succeeded respectively in a scatter plot of Goal Amount USD vs Pledged Amount USD. The reference line represents the maximum Goal Amount USD for which any project in the dataset has succeeded)

Table 4.a Models and Their Test Accuracy Rates Comparison

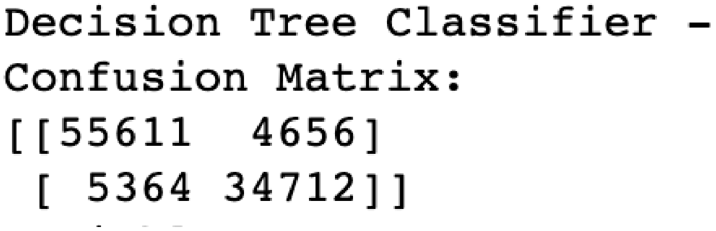
Classifier	**Test Accuracy Rate**
Logistic Regression	88.96%
Decision Tree	90.01%
Random Forest	91.81%
K-Nearest Neighbors (K = 17)	92.30%
Naive Bayes	53.91%

Fig 4.a Logistic Regression Confusion Matrix



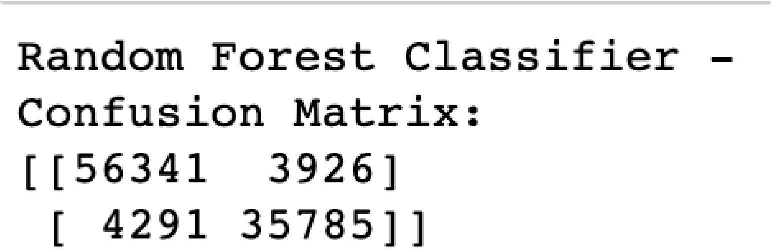
```
Logistic Regression Classifier -
Confusion Matrix:
[[57780  2487]
 [ 8594 31482]]
```

Fig 4.b Decision Tree Confusion Matrix



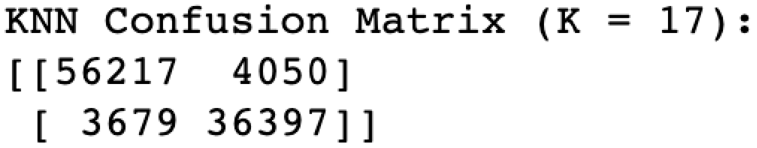
```
Decision Tree Classifier -
Confusion Matrix:
[[55611  4656]
 [ 5364 34712]]
```

Fig 4.c Random Forest Confusion Matrix



```
Random Forest Classifier -
Confusion Matrix:
[[56341  3926]
 [ 4291 35785]]
```

Fig 4.d KNN Confusion Matrix (with k = 17):



```
KNN Confusion Matrix (K = 17):
[[56217  4050]
 [ 3679 36397]]
```

Fig 4.e knn - Optimal k Value Selection (1 - 100)

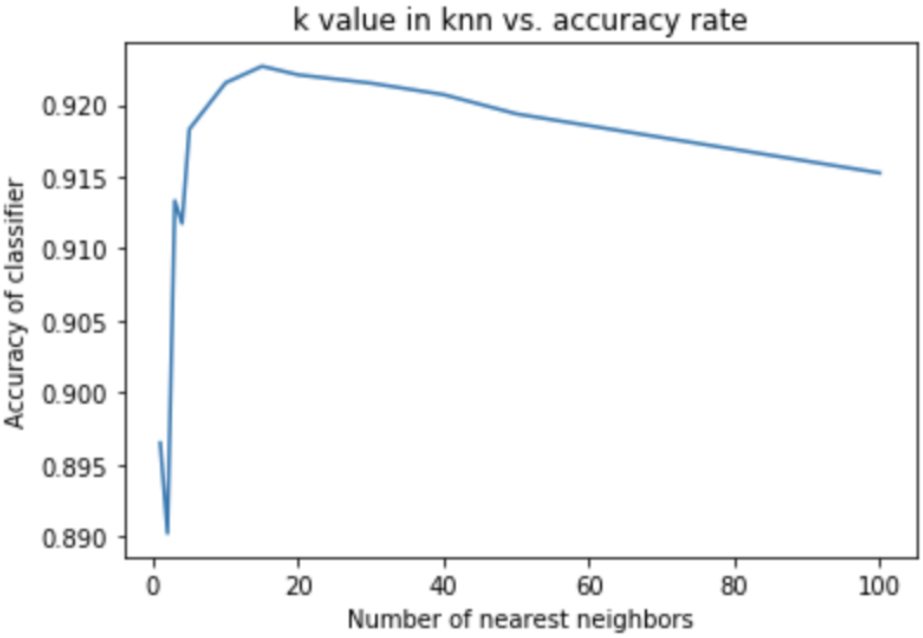


Fig 4.f knn - Optimal k Value Selection (14 - 20)

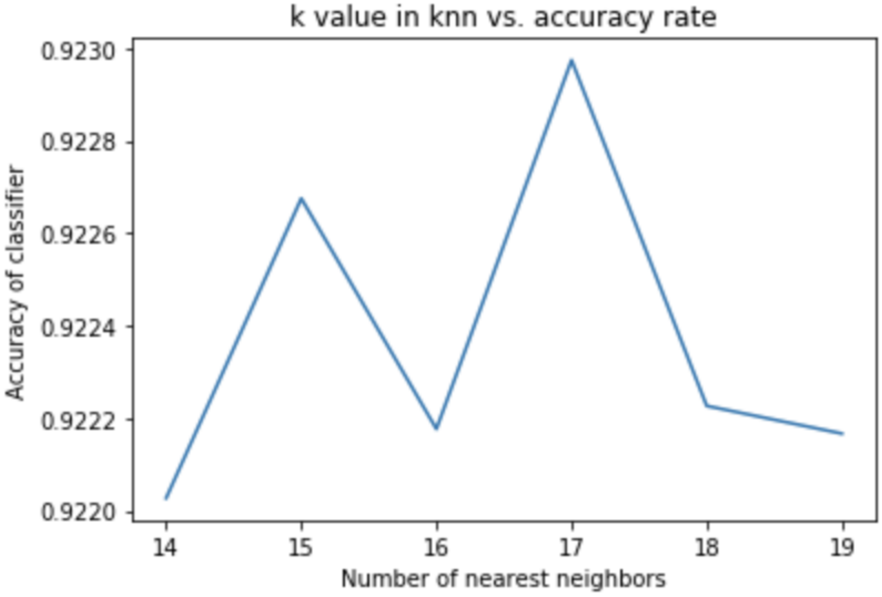
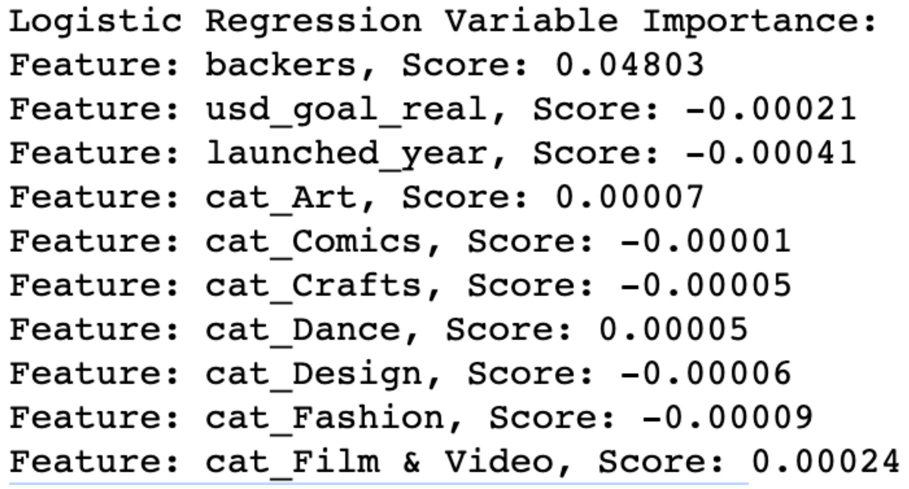
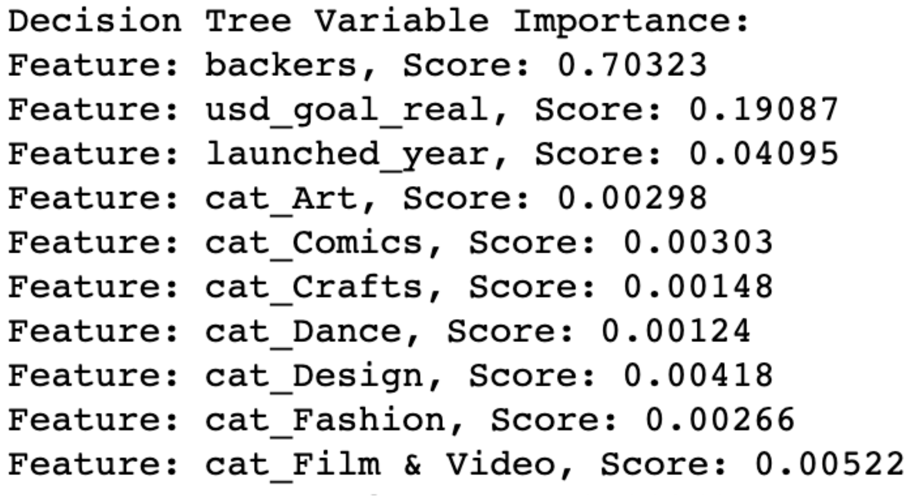


Fig 4.g Logistic Regression Important Variables Snapshot



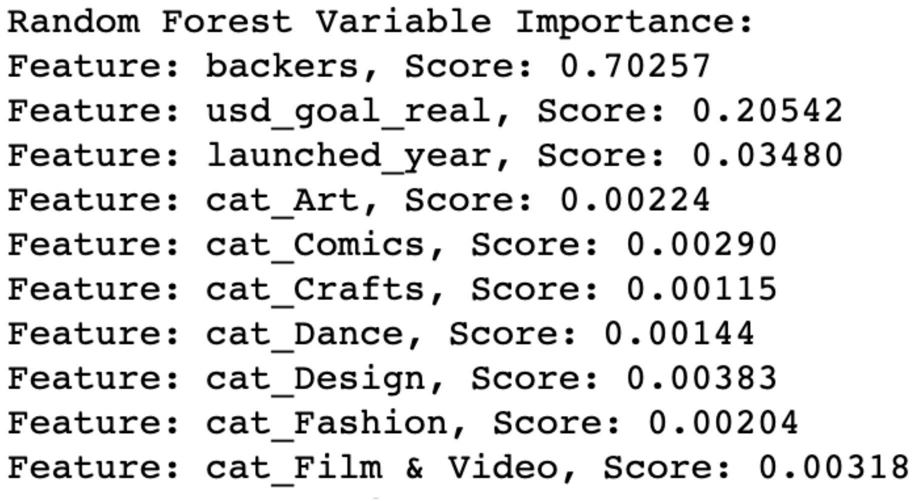
```
Logistic Regression Variable Importance:
Feature: backers, Score: 0.04803
Feature: usd_goal_real, Score: -0.00021
Feature: launched_year, Score: -0.00041
Feature: cat_Art, Score: 0.00007
Feature: cat_Comics, Score: -0.00001
Feature: cat_Crafts, Score: -0.00005
Feature: cat_Dance, Score: 0.00005
Feature: cat_Design, Score: -0.00006
Feature: cat_Fashion, Score: -0.00009
Feature: cat_Film & Video, Score: 0.00024
```

Fig 4.h Decision Tree Important Variables Snapshot



```
Decision Tree Variable Importance:
Feature: backers, Score: 0.70323
Feature: usd_goal_real, Score: 0.19087
Feature: launched_year, Score: 0.04095
Feature: cat_Art, Score: 0.00298
Feature: cat_Comics, Score: 0.00303
Feature: cat_Crafts, Score: 0.00148
Feature: cat_Dance, Score: 0.00124
Feature: cat_Design, Score: 0.00418
Feature: cat_Fashion, Score: 0.00266
Feature: cat_Film & Video, Score: 0.00522
```

Fig 4.i Random Forest Important Variables Snapshot



```
Random Forest Variable Importance:
Feature: backers, Score: 0.70257
Feature: usd_goal_real, Score: 0.20542
Feature: launched_year, Score: 0.03480
Feature: cat_Art, Score: 0.00224
Feature: cat_Comics, Score: 0.00290
Feature: cat_Crafts, Score: 0.00115
Feature: cat_Dance, Score: 0.00144
Feature: cat_Design, Score: 0.00383
Feature: cat_Fashion, Score: 0.00204
Feature: cat_Film & Video, Score: 0.00318
```

Citations

Dataset

Mouillé, M. (2018, February 8). *Kickstarter projects*. Kaggle. https://www.kaggle.com/kemical/kickstarter-projects?select=ks-projects-201801.csv.

Dataset License

Creative commons license deed. Creative Commons - Attribution-NonCommercial-ShareAlike 4.0 International - CC BY-NC-SA 4.0. (n.d.). https://creativecommons.org/licenses/by-nc-sa/4.0/.