

Predicting Kickstarter Crowdfunding Success with Machine Learning Methods

Matias Pietruszka, Taeang Moon, Matthew Lehman, Dan Qi

March 18, 2022

1. Introduction

Kickstarter is a public crowdfunding platform tailored towards producing creative ideas. Through Kickstarter, “creators” are able to make a page and a campaign to share information, details, and fundraising goals for a project they have, and “backers” are able to support these projects and contribute funds to turn them into a reality. According to Kickstarter’s website (link), since the company’s launch on April 28, 2009, 21 million people have backed a Kickstarter campaign, \$6.4 billion has been pledged, and 216,733 ideas have been successfully funded, achieving Kickstarter’s stated mission “to help bring creative projects to life”.

Interestingly, in order for a Kickstarter project to be funded, the project must raise enough money to meet its stated funding goal; otherwise, if a campaign falls short of its target, the creator receives nothing. As such, for new creators with aspiring projects that are looking to raise money through Kickstarter, it is extremely important to set both a sufficient funding goal to create their ideas *and* an attainable funding goal for a successful campaign. Likewise, it is also crucial for creators to design campaigns and present their ideas in order to appeal to potential backers and solicit contributions (and big contributions!); by the same token, if there are distinctive underlying attributes of campaigns that affect the probability of their crowdfunding success, creators should take advantage of these characteristics and exploit them to their benefit.

But what actually makes a Kickstarter campaign successful? Our goal in this project is to use various machine learning methods to answer this question. With greater insight into the true nuances behind historically successful (and failed) Kickstarter projects – and, specifically, with an applicable machine learning classifier to predict the probability of a new Kickstarter campaign’s success – future creators can use these models and findings to construct their ideal campaigns, set optimal funding goals and objectives, maximize the contributions pledged towards their project, and bring their ideas to life.

2. Data

Our data come from a web scraping site called Web Robots (link), which runs a data crawl once a month to scrape Kickstarter projects and uploads their information on a page of their website titled “Kickstarter Datasets” (link). These Kickstarter datasets are large, comprehensive, and relatively common throughout other machine learning projects, studies, and competitions, so we are extremely confident that the data are accurate, robust, and representative of true Kickstarter cases.

Web Robots has been scraping Kickstarter data since April 2014, and their website easily contains data on hundreds of thousands of campaigns. For this project, we decided to focus on a slightly smaller (and more current) subset of their available data. More specifically, Web Robots uploads their data for each month as a large ZIP file of dozens of CSVs, often with upwards of 50 individual files containing thousands of observations. To create a feasible and representative sample of recent campaigns, we download each of Web Robots’ 12 monthly data ZIP files from 2019 – i.e. their last full year of scraped Kickstarter data prior to

the COVID-19 pandemic – we select the first CSV file within each ZIP file, and we merge these 12 CSVs together to create our master dataset. This results in a sample of 45,216 Kickstarter projects, which we go on to clean and evaluate below; however, it would be interesting to see how our findings may change with different data, more data, or data since the onset of COVID-19, which we think could be an important influence on how many campaigns (and what types of campaigns) have been funded over the past few years.

2.1. Data Processing and Preparation

These raw Web Robots data contain 45,216 observations of Kickstarter projects, each of which has 38 variables. Some of these features have valuable information about the project (e.g. the project’s creator, the project’s name and descriptive “blurb”, the project’s category), the structure of the campaign (e.g. the campaign’s creation date, the campaign’s start date, the campaign’s end date, the campaign’s funding goal), and the campaign’s results (e.g. the number of backers, the amount of contributions pledged, the “state” of the campaign as a success or a failure, etc.). Other features are less-useful project ID numbers, URLs, repetitive measurements, or variables with incomplete data. And while the data as a whole are relatively clean, some useful variables have messy formats and convoluted structures, all of which we address through our data processing.

First, we filter the dataset to only include our variables of interest. These include the aforementioned features related to the project’s information (specifically the project’s name, blurb, and category), the campaign’s information (specifically the campaign’s create date, start date, end date, and goal converted to \$USD), and the results (specifically the number of backers, the amount pledged in \$USD, and the campaign’s state as a success or as a failure), as well as additional variables including the project’s country of origin, the currency of crowdfunding, and whether the project was endorsed by the Kickstarter staff as a “staff pick”¹.

Since we aim to predict the probability of a campaign being a success, we filter out all non-completed campaigns, i.e. all campaigns with a state that is not labeled as “successful” or “failed”, which include canceled campaigns, suspended campaigns, and currently-active campaigns. As an important note – and as we reiterate in Section 3 – when building models to predict this campaign success probability, we also omit the other two result-based dependent variables in the dataset (number of backers and donation amount) as potential predictors, since both of these values are obviously unknown prior to the campaign, and both are directly reflective of the project’s outcome. Additionally, if there are multiple campaigns with the exact same name and blurb, we filter out the more recent of the two (i.e. the campaign with the latest create date), since many previously-failed Kickstarter projects are often reposted in a second attempt to solicit crowdfunding, so these duplicated campaigns are not independent of all other observations.

After filtering out observations, we transform several variables with inconsistent factor levels, and we convert all date variables to date-time format using the `lubridate` package. Then, we create new features from

¹None of us were familiar with this “staff pick” feature beforehand, so we conducted comprehensive research about its meaning, its timing, and its validity as a potential predictor. As an aside, in 2016, Kickstarter relabeled these staff picks as “Projects We Love” (link), but boolean data for both types of Kickstarter endorsements are saved in the same staff pick variable within the dataset (which we continue to refer to as “staff picks” for simplicity). From the best that we could tell, it appears that Kickstarter typically labels a project as a staff pick prior to a campaign’s start date, but has also occasionally done so within a campaign’s first few days. Importantly, we could not find any compelling evidence of any projects that have been newly-labeled as a staff pick a substantial amount of time into their campaign. Additionally, in Kickstarter’s post with tips for getting featured as a staff pick (link), all of Kickstarter’s advice pertains to the format and content on a project’s page (e.g. creating a page with an intuitive structure, compelling images, “showing, not telling” the idea), while the post says nothing about the project’s results, which appear to be meaningless (or unknown) in Kickstarter’s staff pick decisions; similar third-party blogs tend to echo the same sentiments. As such, it appears to us that staff-picked projects are generally not reflections of an *ongoing* campaign’s success, but rather pre-campaign labels that may predict the *subsequent* success of projects. Additionally, although becoming a staff pick is more of an art than a science, there are clear benefits to being a staff-picked project. Staff-picked campaigns have prime placement on Kickstarter’s website, they appear in Kickstarter’s newsletter, and they are promoted to Kickstarter’s millions of followers through social media channels. Correspondingly, according to Wharton Professor Ethan Mollick (link), the chances of a project getting successfully funded jumps from 50% as a non-staff pick to 92% as a staff pick (and we find similar results ourselves!). All in all, staff pick does not seem like a dependent variable in the dataset (such as number of backers and donation amount); rather, it seems appropriate – and extremely important – to use staff pick as a predictor instead. And as further evidence and confirmation, the other aforementioned machine learning projects with the same Web Robots datasets all consider staff pick as a potential feature – as two examples, here are research posts by Riley Predum (link) and Laura Lewis (link) – so after conducting our own research, we thought it was best to follow their lead and do the same.

several pre-existing variables. These include nine variables for the year, month, and day of week for each campaign’s create date, start date, and end date, as well as two variables for the length of time between a campaign’s create date and start date (i.e. the “wait time”) and for the length of time between a campaign’s start date and end date, all of which we use to replace the create, start, and end dates themselves. We also create two distinct variables for the Kickstarter-labeled “category” and “subcategory” of each project, both of which are originally contained within the category variable. Additionally, we create proxy variables for the length of each project’s name and the length of each project’s blurb to use as potential predictors in our classifier instead of the actual name and blurb themselves. However, we also try experimenting with natural language processing techniques using this blurb text in an attempt to extract additional information that could relate to (or predict) a project’s success, which we describe in further detail in Section 5.3.

Lastly, we filter out the 2% of campaigns with the lowest crowdfunding goals (in \$USD), which corresponds with a goal threshold of \$100. Since many Kickstarter campaigns are simply posted for the purpose of raising a few dollars here and there, we make this decision so that our analyzed data are reflective of more substantial and expensive creative projects, which is where we believe our models and results can provide the most applied value. However, this 2% (and \$100) cutoff is relatively arbitrary, and while we did similar exploratory data analysis using different ranges of funding goals (and different cutoffs for goal outliers), it would be interesting to see how this decision might affect our final models.

All in all, our cleaned dataset contains 36,929 observations and 22 variables, which we summarize below.

Table 1: Variable Key

Variable	Description
state	result of campaign as a success or a failure – dependent variable
category	classified category of project on Kickstarter
subcategory	classified subcategory of project on Kickstarter
name_length	number of words in campaign name on Kickstarter page
blurb_length	number of words in campaign blurb (description) on Kickstarter page
country	country of project origin
currency	currency of campaign funding
staff_pick	was the campaign labeled by Kickstarter as a "staff pick"?
year_created	create year for campaign page
month_created	create month for campaign page
weekday_created	create day of week for campaign page
year_launch	start year for campaign crowdfunding
month_launch	start month for campaign crowdfunding
weekday_launch	start day of week for campaign crowdfunding
year_deadline	end year for campaign crowdfunding
month_deadline	end month for campaign crowdfunding
weekday_deadline	end day of week for campaign crowdfunding
campaign_wait	length of campaign wait in days (i.e. from create date to start date)
campaign_length	length of campaign duration in days (i.e. from start date to end date)
goal	campaign crowdfunding goal (in \$USD)
backers_count	*number of campaign backers
usd_pledged	*total funds pledged towards the campaign (in \$USD)

Note:

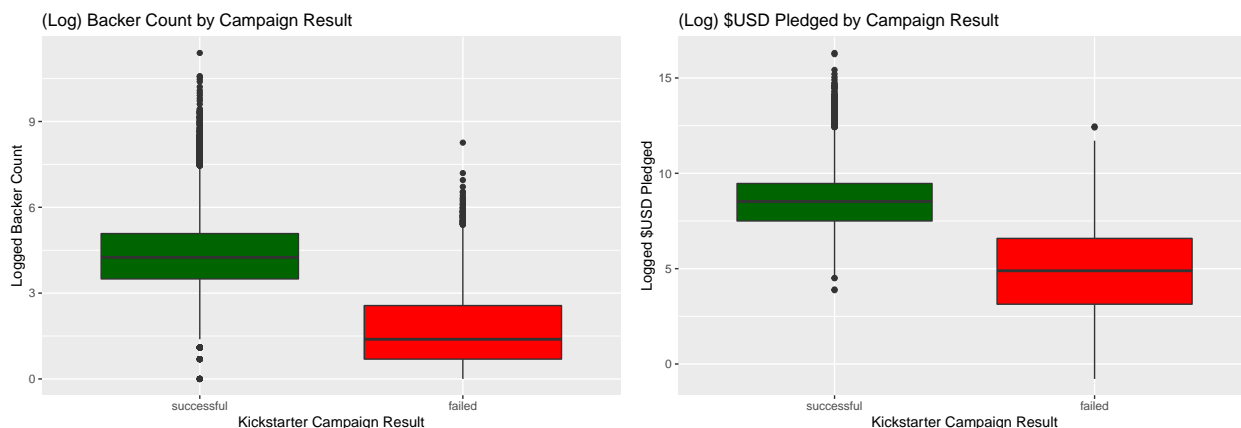
*"backers_count" and "usd_pledged" are both result-based dependent variables; as such, they are included in EDA, but they are omitted as potential predictors of "state" in models

For model fitting, we set aside 20% of the data (7,385 observations) as our test set, and we use the remaining 80% (29,544 observations) as our training set. We split the data such that each set has comparable proportions of successful and failed campaigns – although the classes are not particularly imbalanced as a whole (59.65% successful, 41.35% failed) – as well as comparable proportions across features with some rel-

atively sparse categories, specifically each project’s category, subcategory, country of origin, and currency of crowdfunding. We include some of our most important exploratory data findings from the training dataset, although these results are virtually analogous across different training sets and across the entire dataset.

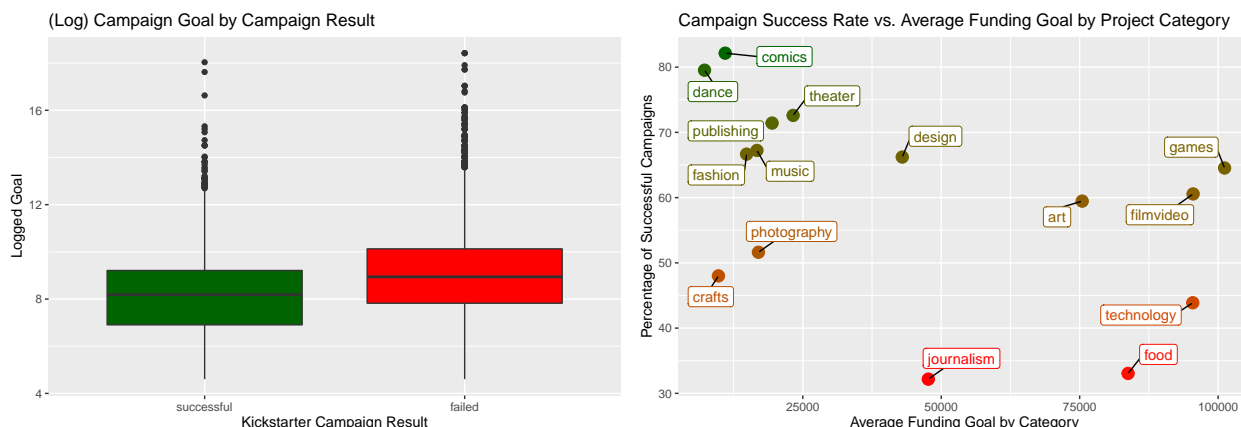
2.2. Exploratory Data Analysis

Overall, 17,624 of the 29,544 Kickstarter campaigns in the training dataset were successful (59.65%), while the other 11,920 campaigns failed (40.35%). Unsurprisingly, the success of a campaign is *extremely* highly related to the number of project backers and the level of project contributions, the two other potential dependent variables mentioned above; successful campaigns averaged 240 backers (median of 70 backers) and \$22,481 in contributions (median of \$5,025), while failed campaigns averaged just 13 backers (median of 3 backers) and just \$1,090 in contributions (median of \$60). These vast discrepancies can be seen in the two plots below (which exclude the 2,332 failed campaigns with zero backers and no pledged contributions).



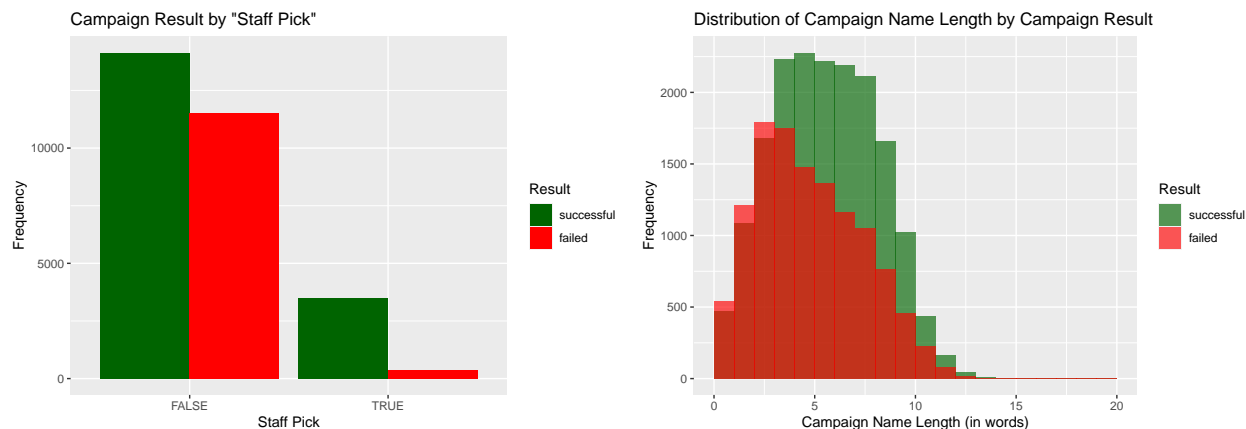
Interestingly – and perhaps unsurprisingly – there are also notable differences in the funding goals between successful and failed campaigns: successful projects had much lower goals (average of \$19,534, median of \$3,600) than failed projects (average of \$103,576, median of \$7,630), as seen in the left plot below. This finding makes logical sense, because a project is only deemed successful if its funding goal is met.

Since a campaign’s goal is presumably related to a project’s cost, we would expect different categories of projects to have different average goals – with more expensive projects generally asking for greater funding – so we were curious if this inverse relationship between goals and successes holds across projects in different categories as well. From the second plot below, it appears it might (to some extent): categories exhibit a wide range of average funding goals (ranging from \$7,258 for dance projects to \$101,170 for games projects) and a wide range of success rates (ranging from 32.18% for journalism projects to 82.14% for comics projects), but average goals and success rates by category exhibit a moderate negative correlation (with $R^2 = 0.203$).



Other potential predictors appear to have strong relationships with successful Kickstarter projects, too. We find some of these relationships to be extremely logical. Most notably, out of the 3,871 campaigns in the training dataset that Kickstarter labels as a staff pick (13.10% of the training data), 89.98% are successful, which is *significantly* greater than the 55.08% success rate among non-staff-picked campaigns (left plot below) and the 59.65% success rate across the entire training data. This mirrors the aforementioned results from Ethan Mollick, and it makes intuitive sense that the campaigns which Kickstarter chooses to endorse and promote have a much higher success rate! Evidently, staff pick appears to be a very important predictor.

Some of these relationships are unexpected to us, too. For example, successful campaigns have a noticeably longer average name length than failed campaigns (6.04 words vs. 5.27 words – a large discrepancy given that there are well over 10,000 observations from both classes). This is even clearer when looking at the disparate distribution shapes and peaks of name length by campaign state (right plot below).



Luckily, we can turn to our machine learning tools to discover more unsurprising and surprising relationships in the data, determine the most significant features, and deploy them in our models to predict campaign success. Future creators should definitely care about (and consider) these findings!

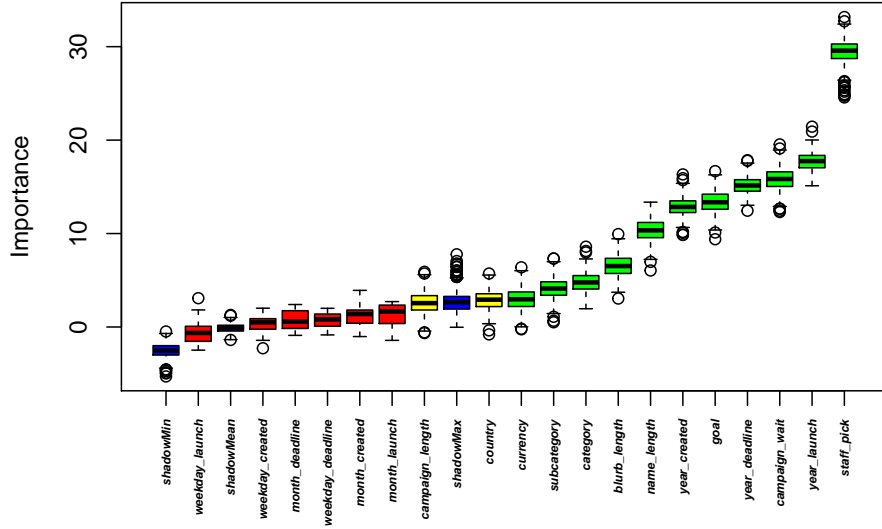
3. Model Methodology

With a better understanding of our (cleaned) data, we turn to modeling the relationships between our data's features and a project's success. As mentioned above, we now exclude all variables that are also measurements of a campaign's result as potential predictors in the training dataset, namely the number of project backers and the campaign's donation amount. Our goal is to predict the probability of a new project's success, so other dependent variables (which are unknown at the start of a project) are not of interest; obviously, campaigns with more backers and more donations are more likely to be successful, but that is not useful information for our models! And once again, we decide that it is best and most reasonable to include staff pick as an independent variable, which – corresponding with our findings throughout Section 2 – we find to be an extremely important predictor in our subsequent variable selection.

3.1. Variable Selection

We start our variable selection process by using the Boruta feature selection algorithm with random forest models, which has inherent randomness built in by creating shuffled copies of all variables. We subset 2% of the training data (592 observations) for this variable selection, and we use the remaining 98% (28,952 observations) for our model cross-validation, feature refinement, and parameter tuning with these Boruta-selected predictors. Using these 592 projects, Boruta confirms 11 attributes as important for predicting a campaign's success (shown below), all of which we use as potential predictors for our subsequent models. Note that we also experiment with larger feature-selection datasets and different feature-selection methods for robustness; their results and selected predictors largely mirror the same Boruta attributes found below.

Boruta Feature Selection – Variable Importance



As seen in the Boruta importance scores, the 11 most important predictors include whether the project was a staff pick, the year (but not the month or the day) of each project’s creation, launch, and deadline, the time between the campaign’s creation and the campaign’s launch (but not the time between the campaign’s launch and the campaign’s deadline), the funding goal of the project, the length of the project’s name and blurb, the project’s category and subcategory, and the currency of the project’s crowdfunding.

3.2. Model Fitting

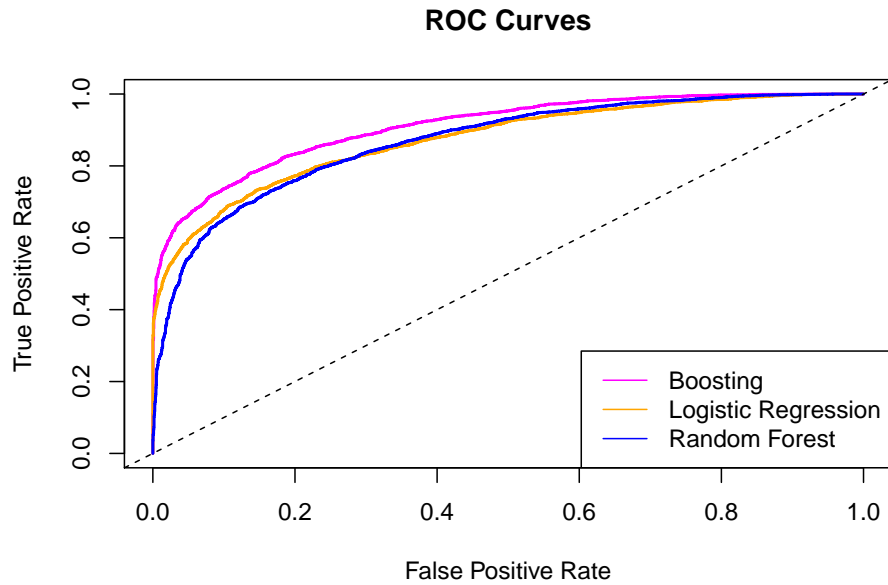
With these Boruta-selected features as our potential predictors, we first fit a XGBoost model (with the 98% of the training data *not* used for feature selection) using cross-validation over a hypergrid of possible tuning parameters. One of these tuning parameters is the number of predictors, as ranked by their importance in the Boruta plot above; since Boruta did not find all 11 of these variables to be equally important, cross-validating over different model parameters with different numbers of predictors (i.e. different Boruta-importance thresholds) should theoretically lead to a more accurate final model than a boosting model which blindly includes all Boruta-selected predictors. However, ultimately, the cross-validation run that minimizes the out-of-sample logarithmic loss includes all 11 predictors, so we use all Boruta-selected variables (and the corresponding optimal tuning parameters) to fit our final boosting model with the complete training data.

Following from this predictor refinement (or lack thereof), we use the same 11 predictors to experiment with a variety of other classifiers to predict a campaign’s success. We include our results for our best-performing logistic regression model and our best-performing random forest model – which also follows from cross-validation tuning with the non-feature-selection training data – in Section 4 below, but we omit many of our slower (and worse-performing) models from our analysis for the sake of time, simplicity, and brevity.

4. Results

Below, we present the ROC curves, the AUCs, the misclassification rates, the confusion matrices, and summarization boxplots for our selected boosting model, logistic regression model, and random forest model on the 7,385 test set observations.

ROCs:



AUCs and Misclassification Rates:

Table 2: Model Performance Summary

Model	AUC	Misclassification Rate
Boosting Model (XGBoost)	0.90848	0.18172
Logistic Regression Model	0.87282	0.21530
Random Forest Model	0.86727	0.21842

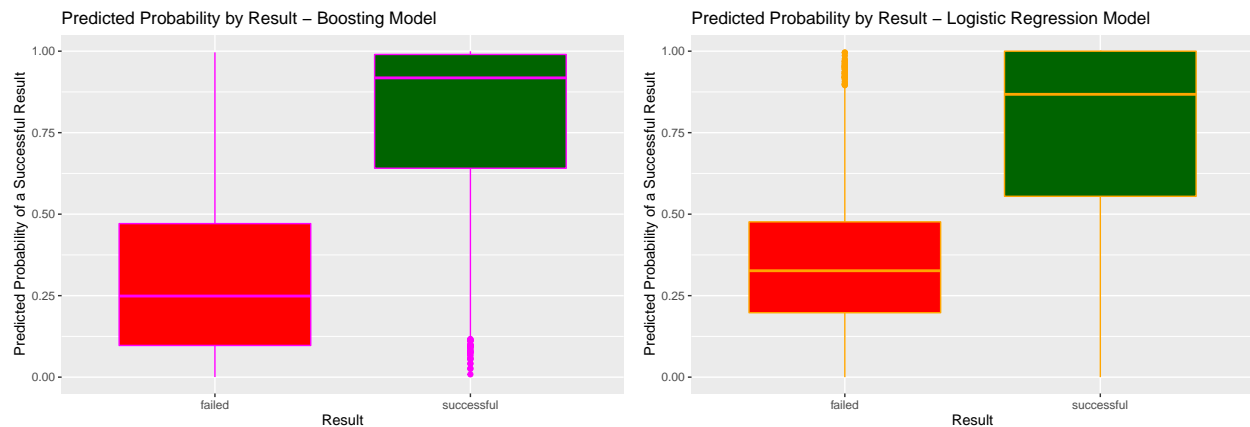
Confusion Matrices: (F is a predicted/actual failed project, S is a predicted/actual successful project)

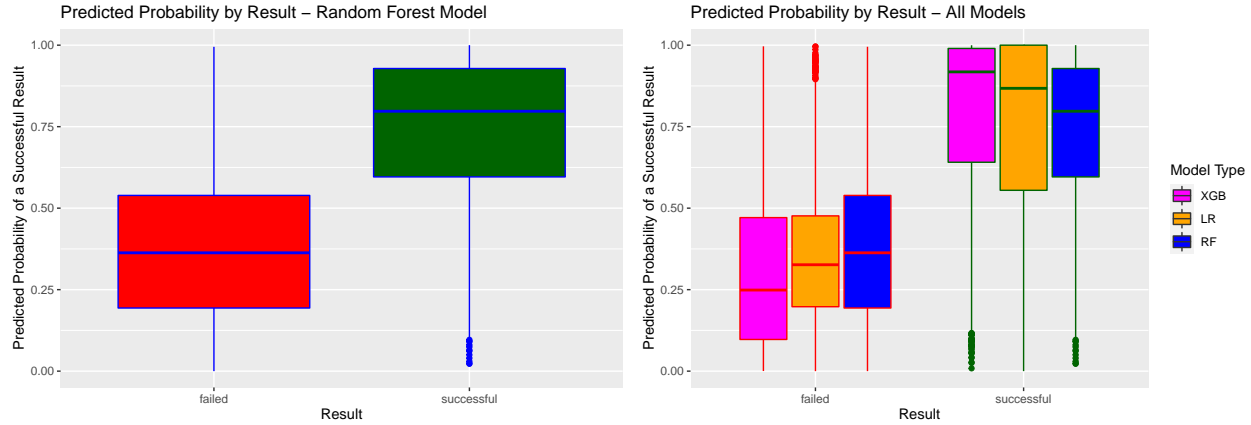
	F	S
<i>XGB: F</i>	2322	684
<i>XGB: S</i>	658	3721

	F	S
<i>LR: F</i>	2318	928
<i>LR: S</i>	662	3477

	F	S
<i>RF: F</i>	2085	718
<i>RF: S</i>	895	3687

Boxplots:





ROC curves are probability curves that plot each model’s true positive rate (i.e. the rate of correctly predicting that a successful test set Kickstarter campaign will be a success) against its false positive rate (i.e. the rate of incorrectly predicting that a failed test set Kickstarter campaign will be a success) as a model evaluation metric; the best-performing models will achieve higher true positive rates at lower false positive rates. As such, from the ROC curves, the boosting model appears to be the best-performing model on the test dataset, followed by the logistic regression model, and lastly by the random forest model. The boosting model noticeably has the highest curve across all false positive rates, whereas the ROCs for the logistic regression model and the random forest model are more similar to one another. Interestingly, the logistic regression model has very high true positive rates at low false positive rates (and similar rates to the boosting model, especially relative to the random forest model), but as the logistic regression predicts on more data, its performance becomes much more comparable to – and perhaps slightly worse than – the performance of the random forest. Also, all three models appear to predict the test data extraordinarily well, and all three ROCs represent exactly what we would expect to see from highly accurate and predictive fits.

AUCs measure the area under the ROC curve, so an ideal model with good separability would have an AUC near one. Corresponding with the ROCs, the boosting model has the highest AUC (approximately 0.91), followed by the logistic regression model and the random forest model (both approximately 0.87).

As seen in the same table, with a probability threshold of 0.5 – meaning that if the model predicts that a project has a probability of being a success equal to or greater than 50%, then the model classifies the project as a predicted success – the boosting model also has the lowest test set misclassification rate (approximately 18%), followed by the logistic regression and random forest model (both approximately 22%). But once again, all three models appear to perform extremely well on the test data, and all three models are clearly capturing similar strong predictive signals from the training data.

Furthermore, the confusion matrices for the three models depict the types of correct and incorrect predictions that each classifier makes. With the same 0.5 probability threshold, the XGBoost model leads to the most true positives and true negatives (i.e. correct predictions) *and* the fewest false positives and false negatives (i.e. incorrect predictions). Interestingly, the logistic regression leads to noticeably more negative predictions on the test set (i.e. predicting that campaigns will fail) than the other two models, while the random forest leads to noticeably more positive predictions on the test set (i.e. predicting that campaigns will be successful) than the other two models; as such, the boosting model appears to strike a proper and more-accurate balance.

Lastly, the boxplots show the distribution of each model’s predicted probabilities of success for all 7,385 test set projects, separated by the project’s actual result. Evidently, the predicted success probabilities for successful projects are significantly higher than the predicted success probabilities for failed projects across all three models, which is exactly what we would hope to see. Nevertheless, the boosting model appears to predict lower probabilities for failed projects than the other two models *and* higher probabilities for successful projects than the other two models; in other words, the boosting model makes better predictions, and the boosting model is more confident in its better predictions. This serves as further confirmation that while all three models are great, the boosting model is the best.

In conclusion, from the ROCs, AUCs, misclassification rates, confusion matrices, and boxplots, we believe

that this XGBoost model will be the most accurate model for predicting the probability of a Kickstarter project’s success (across both included and omitted models). As such, we select this boosting model as our model of focus, and we further interpret its results, its predictions, and its implications in Section 5 below.

5. Implications & Applications

As mentioned in Section 1, a predictive machine learning model for the probability of a Kickstarter campaign’s success would have numerous applications for future creators and projects, while also enabling greater insight into the features and factors that make a Kickstarter campaign successful (or unsuccessful). This XGBoost appears to be the exact predictive model we had set out and aimed to develop. As such, we want to investigate some of the boosting model’s results, applications, potential improvements, and implications, which we do below through a couple of different methods and approaches based on our findings from above.

5.1 Implication: Staff Picks

First, we want to dive deeper into the true effect of being labeled by Kickstarter as a staff pick. From the aforementioned previous research and from our exploratory data analysis, staff-picked projects have a *significantly* greater success rate than non-staff-picked projects; correspondingly, the dataset’s staff pick variable was found to be the most important predictor in our variable selection by a wide margin.

But what is the true effect of being labeled as a Kickstarter staff pick, and how does it differ between different types of projects and campaigns? We attempt to quantify these effects using our boosting model to calculate the “staff pick uplift” of the 7,385 test set projects. More specifically, for each test set observation, we predict the probability of the campaign’s success given that the campaign *was* labeled as a staff pick and the probability of the campaign’s success given that the campaign *was not* labeled as a staff pick, both while holding constant all other campaign attributes. Then, we subtract the latter probability from the former in order to estimate the effect of a staff pick label on each individual project’s predicted success. These uplifts range from a high of 0.771 (i.e. one specific campaign is a whopping 77.1% more likely to be successful as a staff pick than as a non-staff pick) to a low of 0.001 (i.e. another specific campaign is just 0.1% more likely to be successful as a staff pick than as a non-staff pick), as seen in the plots below.



Of note, many test set projects have a predicted uplift near zero; nevertheless, the predicted uplift from *each* test set project is *greater* than zero. In other words, for each of the 7,385 test set projects, the boosting model predicts that the project is more likely to succeed as a staff pick than as a non-staff pick, although for many projects, this predicted staff pick effect is extremely minimal (and over 1,000 projects have an effect of less than 1%). However, for campaigns with predicted uplifts greater than about 5% – which is still the vast majority of test set campaigns – the uplifts appear to follow a relatively standard distribution. Across all 7,385 projects, the overall average staff pick uplift is 0.29 (i.e. a 29% increase in the probability of success), and the overall median staff pick uplift is 0.32 (i.e. a 32% increase in the probability of success).

These results make intuitive sense, especially given the strong staff pick effects discovered earlier. Additionally, the boosting model’s predicted success probabilities for projects with small predicted uplifts are generally either very low (i.e. very bad campaigns) or very high (i.e. very good campaigns); as such, it also makes sense that the effect of a staff pick label would have little bearing on the outcome of these more-extreme campaigns and ideas. Interestingly, unlike the boosting model’s uplift predictions, the (omitted) uplift predictions that we calculate with several of our other models result in a few test set campaigns having negative predicted uplifts. Nevertheless, it is overwhelmingly clear from our best-performing boosting model that all projects have a greater chance of success – and most projects have a *much* greater chance of success – as a staff-picked project than as a non-staff-picked project, so creators should definitely focus on getting picked by Kickstarter in order to significantly boost their probability of reaching their funding goal.

5.2 Application: “Moustache Men” Case Study

Next, we want to see how our model could work in action. More specifically, we want to test how a creator could use our boosting model *before* making a Kickstarter page in order to optimize some of the campaign parameters within their control – namely the campaign’s goal, the name length, the blurb length, and the time between the campaign’s creation and launch – and to maximize their probability of success. As such, we select a failed project from our test set as our case study – and, particularly, a project that we would have loved to see succeed – and we try and use our model to hypothetically improve its chance of being funded.

We decide on a campaign called “Moustache Men” (link), a 2014 graphic design project out of the United Kingdom that simply appears to be cartoons of men with large mustaches; as an example, a picture of one of these mustached men is featured on the campaign page in a hot air balloon with a dog and bird. In addition to being cute and hilarious, other “Moustache Men” campaign information and results are described below; note that from the campaign’s attributes, our boosting model predicts “Moustache Men” to have a 36.8% chance of success (so with the same 50% threshold from before, the model’s prediction is a true negative).

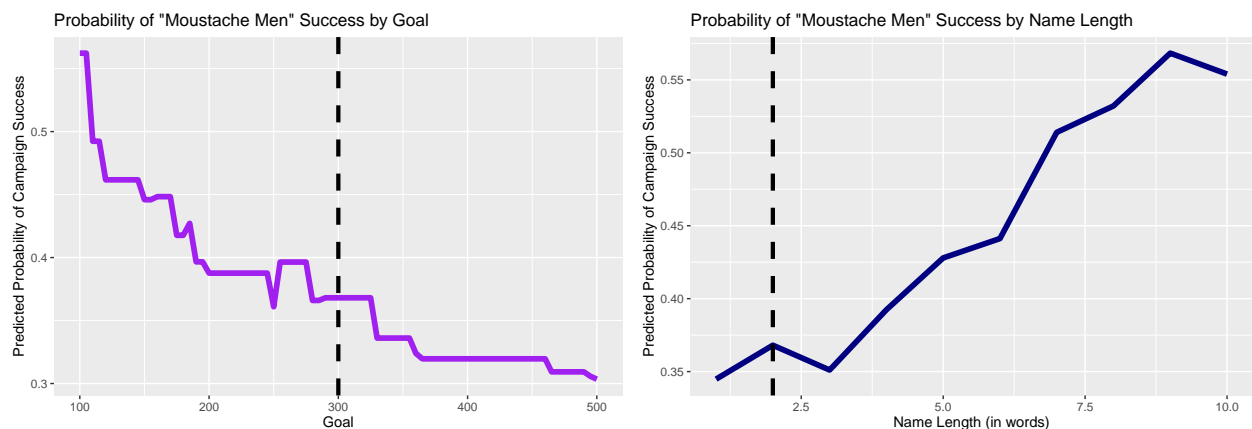
Table 3: Actual “Moustache Men” Parameters and Campaign Summary

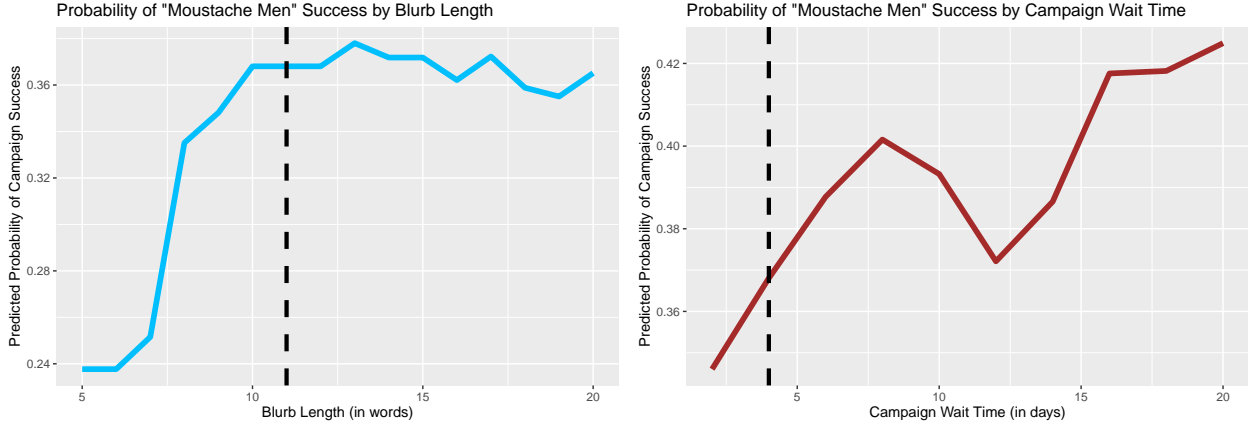
Goal	Name	Blurb	Campaign Wait	Staff Pick	Predicted Prob.	Backers	Pledged	Currency
300	2	11	4	FALSE	0.36806	8	150	GBP

Note:

“Moustache Men” Blurb: “My first exhibition seeking to bring a smile and learn more” (11 words)

We want to see if “Moustache Men” could have been made into a reality. As such, we analyze the effects of changing the campaign’s goal (ranging from 100 to 500 GBP, with all financial amounts in the campaign’s original currency), the campaign’s name length (ranging from one to 10 words), the campaign’s blurb length (ranging from five to 20 words), and the campaign’s wait time (ranging from 2 to 20 days), and we plot our boosting model’s predicted probability of success for each hypothetical campaign below.





Unsurprisingly, “Moustache Men”’s goal is inversely related to its probability of success: the higher funding request, the lower probability of reaching that funding request, which is the metric for a successful campaign. According to our model, by lowering the campaign goal to 100 GBP, “Moustache Men” would have increased their probability of success by nearly 20%, from 36.8% to 56.2%. Interestingly, “Moustache Men” could have increased their predicted probability of success by *even more* (to 56.8%) by simply increasing the length of their name to nine words; “Moustache Men” is short and catchy, but evidently, a longer or more-descriptive title would have been better. And while blurb length and campaign wait time seem to be less influential on the campaign’s results, it appears that longer descriptions and longer wait times – perhaps giving the campaign more time to become a staff pick – generally increase “Moustache Men”’s predicted success.

When tuning all four parameters together, we find that with a goal of 100 GBP, a name length of nine words, a blurb length of eight words, and a campaign wait time of 20 days, “Moustache Men” could have increased their predicted probability of success to 78.5%. Note that these are the minimum campaign goals and maximum campaign wait times that we examine, so perhaps this probability could be further increased over a wider range of possibilities. However, by keeping *all* campaign parameters the same and simply being labeled as a staff pick, “Moustache Men” would have increased their predicted probability of success to 85.3%! As such – and following from our previous findings – our model shows that while “Moustache Men”’s creators could (and should) take many steps to optimize their campaign, if Kickstarter picked the project, it would have increased the odds of seeing “Moustache Men” in the real world more than anything else.

Table 4: Optimal "Moustache Men" Parameters vs. Actual "Moustache Men" Parameters as a Staff Pick

Goal (GBP)	Name Length	Blurb Length	Campaign Wait	Staff Pick	Predicted Success Prob.
100	9	8	20	FALSE	0.78537
300	2	11	4	TRUE	0.85294

5.3 Extension: Natural Language Processing

In addition to the above analysis, we also try our hand at examining the data using natural language processing (NLP). In particular, we analyze the blurb text using bidirectional encoder representations from transformers (BERT) to predict whether campaigns in the training data were successful. BERT is a relatively new and innovative neural network development in NLP that uses transformer encoders and masked language modeling to pre-train a language model that one can use for a variety of tasks. In this case, we use a pre-trained BERT model, specifically a distilBERT model for its smaller computational cost, and we fine-tune it for a classification task. For each row of the training dataset, we assign a successful campaign a classification factor of 1 and a failed campaign a classification factor of 0, which we try and predict using just the blurb text data. To do so, we first separate the training dataset into a (smaller) training dataset and a validation set. Then, we input each short blurb of the new training set into the pre-trained BERT model, and we feed its BERT output into a classification layer to ultimately classify the blurb as a 1 or 0. The BERT

and classification layer model are trained and fine-tuned through multiple runs of the neural network and stochastic gradient descent with the Adam optimizer, and they are subsequently tested using the validation set. We then select the model that performed best on the validation set to ultimately use for predictions on the test set. This model is able to achieve around a 72.2% accuracy in classifying campaign success simply from the blurb descriptions. Although these are far from definitive predictions – especially given the underlying 59.65% campaign success rate – we are pleasantly surprised by the accuracy of our results; we would imagine that humans would have a very hard time determining a campaign’s success from just a short description about its objective, but it seems that our model is able to determine the correct state of a majority of test set campaigns. Given more time (and greater NLP knowledge), we would love to investigate these predictions more, as well as to potentially use these results to enhance our ultimate boosting model; likewise, we would certainly recommend greater NLP exploration as a primary focus for future related research.

6. Conclusion

Our goal in this paper is to better understand what makes Kickstarter campaigns successful by building machine learning models to predict the probability of a Kickstarter campaign’s success, then investigating our results to reveal applied insights for future creators and their ideas. In our analysis, we find that feature selection using Boruta variable selection paired with a hyperparameter-tuned boosting model performs best on our test set across a few different evaluation metrics. We then use this resultant model to analyze the underlying features that are most predictive of successful campaigns – specifically focusing on Kickstarter’s staff picks, the most important predictor – and we demonstrate the model’s potential real-world use through an applied case study to help make “Moustache Men” a reality. In addition, our use of the BERT model trained on short text descriptions of campaigns produces promising results for the future use of NLP in predicting campaign success, which could help lead to even better models and findings in further research.

We discover many predictive aspects of Web Robots’ Kickstarter campaign data in our analysis, but there are definitely other facets to fundraising campaigns that are worth investigating. For example, Web Robots has no data on campaign engagement, so with more information, we could see how many clicks or impressions each campaign generates and predict project success with those additional variables. Additionally, social media could influence a campaign’s success, so it would be interesting to scrape social platforms to see how appearances and popularity might help a campaign’s funding; perhaps Kickstarter staff picks have a significantly greater social media presence! Furthermore, applying machine learning models to other online fundraising platforms such as GoFundMe, IndieGoGo, and Patreon may be informative to analyze the differences in optimal strategies across sites, or to create a generalized model for predicting online fundraising campaign success. Lastly, while we are primarily interested in the probability of a Kickstarter project’s success – the ultimate goal for Kickstarter creators – it might be revealing to analyze determinants for other related campaign-result metrics, namely each project’s number of backers and level of donations; although these dependent variables are not the direct objectives of a Kickstarter campaign, the consequent models could definitely have useful applications and insights across other marketing and crowdsourcing scenarios.

In terms of the NLP portion of the analysis, in the future, we would like to analyze the name, the text, and the story of the campaigns to predict their success. Since our blurb results end up producing much higher accuracy than we had initially expected, we are very interested in learning how the name and the story of campaigns can affect and influence their overall result. Campaign stories would be especially intriguing to investigate, as they include much more text – and, correspondingly, context – for the models to train on, and thus may end up yielding much better results. Finally, in the future, it would also be interesting to examine which words and phrases in campaign blurbs, titles, and stories are most impactful in predicting the success or failure of a campaign. Such information could help future creators to increase engagement with their campaign, whether through general impressions and clicks or through enhanced fundraising backing.

In conclusion, throughout this process, we have accomplished our ultimate objective of building and analyzing an accurate machine learning model to predict Kickstarter project success. While there is certainly more work that can and should be done, future creators should be able to apply this model to construct their optimal campaigns and – along with Kickstarter – help bring their creative projects to life.