KICKSTARTER

OMIS 3490
ML Final Paper

Ntandokayise Chuma | Prasenjit Choudhury

Santa Clara
University

Contents

Introduction

Kickstarter is an American corporation based in Brooklyn, New York, that maintains a global crowdfunding platform focused on creativity. The company's stated mission is to "help bring creative projects to life". As of December 2019, Kickstarter has received more than $4.6 billion in pledges from 17.2 million backers to fund 445,000 projects, such as films, music, stage shows, comics, journalism, video games, technology, publishing, and food-related projects.

Kickstarter helps artists, musicians, filmmakers, designers, and other creators find the resources and support they need to make their ideas a reality.

People who back Kickstarter projects are offered tangible rewards or experiences in exchange for their pledges. This model traces its roots to the subscription model of arts patronage, where artists would go directly to their audiences to fund their work.

To date, tens of thousands of creative projects — big and small — have come to life with the support of the Kickstarter community.

However, just like all things related to under-development projects, some Kickstarter projects may fail while some may succeed.

Now the question arises, with so many Kickstarter projects online, is there a way to predict or guess in advance whether a Kickstarter project would be a success or failure in reaching the funding target?

This is an important question to answer as the outcome of a project affects both the people who're putting up the funds to back the project, and also the Kickstarter company itself as it's revenue generation depends on the success of a project.

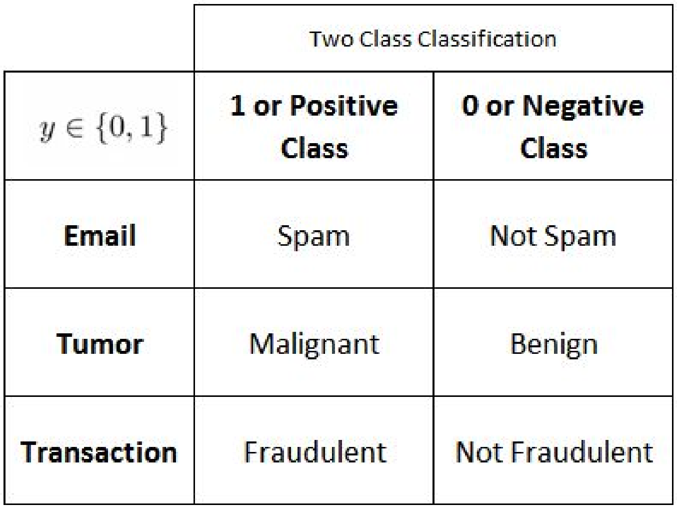
As a backer the number one concern is ***"will I get my copy of the completed project or item that I have put my money towards? Or will my money be refunded after a few days/months when I could have put that money to some other better use?"***

As Kickstarter company the number one concern is ***"will this project meet its funding goal so that we get to generate revenue?".*** Kickstarter charges 5% flat fee of the complete project funding goal + $0.20/backer transaction fee to generate revenues. But Kickstarter can get the fee only if the project meets funding goals. Otherwise in the event of project cancellation or failure, money is returned to backers.

We plan to use data collected from past Kickstarter projects to build a model which can predict whether a project will be successful or not. This information would be beneficial to both backers and Kickstarter to decide which projects to invest in.

Problem Categorization and Algorithm Selection

So, the problem that we're trying to solve here is a classification problem where the dependent variable (in simpler terms, the outcome or result) is in a categorical form such as Yes/No, True/False, Dead/Alive and so on, while the independent variables (in simpler terms the causes for the outcome) are in continuous form.



	Two Class Classification	
$y \in \{0, 1\}$	**1 or Positive Class**	**0 or Negative Class**
Email	Spam	Not Spam
Tumor	Malignant	Benign
Transaction	Fraudulent	Not Fraudulent

There are many classification algorithms available which can be used to solve classification problems, such as:

- Logistic Regression
- Naive Bayes Classifier
- Nearest Neighbor
- Support Vector Machines
- Decision Trees
- Boosted Trees
- Random Forest

- Neural Networks

We decided to use the Logistic Regression algorithm to solve our classification problem.

A little bit about Logistic Regression:

This algorithm applies a logistic function to a linear combination of features to predict the outcome of a categorical dependent variable based on independent predictor variables.

The odds or probabilities that describe the outcome of a single trial are modelled as a function of explanatory variables. Logistic regression algorithms help estimate the probability of falling into a specific level of the categorical dependent variable based on the given predictor variables.

Advantages of Using Logistic Regression

- Easier to inspect and less complex.
- Robust algorithm as the independent variables need not have equal variance or normal distribution.
- These algorithms do not assume a linear relationship between the dependent and independent variables and hence can also handle non-linear effects.
- Controls confounding and tests interaction.

Drawbacks of Using Logistic Regression

- When the training data is sparse and high dimensional, in such situations a logistic model may overfit the training data.
- Logistic regression algorithms cannot predict continuous outcomes. For instance, logistic regression cannot be applied when the goal is to determine how heavily it will rain because the scale of measuring rainfall is continuous. Data scientists can predict heavy or low rainfall but this would make some compromises with the precision of the dataset.
- Logistic regression algorithms require more data to achieve stability and meaningful results. These algorithms require a minimum of 50 data points per predictor to achieve stable outcomes.
- It predicts outcomes depending on a group of independent variables and if a data scientist or a machine learning expert goes wrong in identifying the independent variables then the developed model will have minimal or no predictive value.

- It is not robust to outliers and missing values.

Some Applications of Logistic Regression

- Logistic regression algorithm is applied in the field of epidemiology to identify risk factors for diseases and plan accordingly for preventive measures.
- Used to predict whether a candidate will win or lose a political election or to predict whether a voter will vote for a particular candidate.
- Used to classify a set of words as nouns, pronouns, verbs, adjectives.
- Used in weather forecasting to predict the probability of rain.
- Used in credit scoring systems for risk management to predict the defaulting of an account.

Data Cleansing and Preparation

We used a publicly-available dataset containing information on 378k+ past projects on Kickstarter including their fund-raising goal status, whether they failed or succeeded.

The dataset can be found here: Kickstarter Projects

The dataset contains 378661 row items with 15 columns (features):

ID	internal kickstarter id
name	name of project
category	category of the project - ex: comics, music, board games etc.
main_category	larger categories: books, music, games etc.
country	country where the project is being executed
currency type	currency type ex. USD, CAD, GBP, INR etc.
launched	date on which the project was launched
deadline	date by which the project needs to finish fund-raising
goal (home currency)	total funding needed in native currency
goal (converted to USD)	total funding needed converted into USD
backers	number of people who have invested money in the project

pledged (home currency)	total funding raised in native currency
pledged (converted to USD)	total funding raised converted into USD
state	state of the project - success, failure etc.
usd pledged	redundant field

In order to use the data to solve the classification problem, we had to drop a few fields and also normalize some of the data to bring it into a single common currency denomination.

Step 1: Standardize targets into a single currency (USD)

Using commonly available fee-to-use API: https://fixer.io, we were able to convert the currency fields:

- Convert values in the field “goal (home currency)” and place in the field “goal (converted to USD)”
- Convert values in the field “pledged (home currency)” and place in the field “pledged (converted to USD)”

Step 2: Identify and drop non-significant features/independent variables

- Id
- name
- category
- launched (date)
- deadline (date)
- goal (home currency)
- pledged (home currency)
- currency type

Step 3: Create dummy variables for string value data

A logistic regression algorithm can only accept numeric values and not string values in independent variables, we created dummy variables for:

- Main Category
- Country
- State

Model/Solution

- **Define the model**

  Per our discussion above, the model we used is a Logistics Regression model, a supervised, classification model that basically is used to predict the probability of a certain class or event happening such as pass/fail, win/lose, alive/dead or healthy/sick. In this particular case, we are predicting the probability of a success/fail outcome from the Kickstarter dataset.

  The model, largely, uses the: ‘linear_model.LogisticRegression()’, a scikit-learn library specifically for Logistics Regression, as the name suggests.

- **How did you improve the model?**

  The first attempt on the code resulted in a model that was overfitting, which meant trying several diagnostic approaches to identify the issue before we could move on to improving the model. After substituting the variables and trying different combinations, we discovered that the variables 'pledged (converted to USD)' and ‘goal (converted to USD)’ were highly correlated, and therefore could not be used together in one model. We then created three different scenarios that produced different accuracy outcomes.

  The first model that needed to be improved on had a near 64% accuracy. From the 44 independent variables we identified as the starting point for running the model after cleaning the data, this model was missing the 'pledged (converted to USD)' and ‘backers’ variables. The second model needing improvement resulted in an 82% accuracy and the only missing variable was the ‘goal (converted to USD)’. The variables 'pledged (converted to USD)' and ‘backers’ were part of the model in this case. The most accurate model with a 91% accuracy was missing only the 'pledged (converted to USD)' with the rest of the variables (43 total) included in it.

  The conclusion was that the variable ‘backers’ carries a lot of weight in the entire model. Intuitively, it makes sense. A project that gains support from a high number of backers means that it gained the confidence of many, meaning that the chances of it being a success are high.

- **What models did you try?**

  Using the model that had a 91% accuracy, we tried using another scikit-learn library: ‘LogisticRegression()’ and the accuracy of the model dropped to about 75%. Because of the lower accuracy, we abandoned the library without trying other models.

Final Result

- **Did you properly evaluate your experiments/models?**

  We are confident that we properly evaluated our models. We used two different methods utilizing the scikit-learn’s metrics and confusion matrix libraries. The metrics library evaluates the models using several scores including the accuracy score, classification report, precision score, and the recall score. While all scores are important and present different ways of evaluating the model, the accuracy score, which shows the fraction of samples predicted correctly, is what we mostly used in this analysis.

  The confusion matrix, on the other hand, shows the model’s True Positives,True Negatives, False Positives and False Negatives. The matrix basically quantifies the number of positive or negative predictions, based on whether or not the predictions were true or false.

  Furthermore, we used the ‘logmodel.predict’ library to predict what the test dataset will be after running the models and the predictions were highly accurate, particularly for the model that had a 91% accuracy.

- **Do your conclusions match your results?**

  Our conclusion and results do match. We did expect the different variables we discussed above to impact the model differently - we just did not know what the magnitude of the impact would look like.

  The fact that the variable ‘backers’ impacted the accuracy of the model the most was a bit of a surprise. From an assumption stand point, we thought the amount of money that a project receives is the most significant factor and therefore should have the most impact on the model.

The assumption turned out to be inaccurate. In retrospect, now it looks obvious that since Kickstarter projects are crowd-funded, having a larger number of backers is a more significant factor in success as the backers can bring in other backers and can chip in the funds to reach the fund-raising goal.

All in all, there was an alignment between our conclusions and the results.

- **Can your solution be used in real-life?**

  Based on the evaluation methods we used to assess the model, we have a solution that we are confident can be used in real life. For both the company, Kickstarter, and the project backers, having this kind of information would be beneficial in making a scientifically informed decision in a very efficient manner.