Data Mining and Predictive Analytics: Final Project

Topic: Predicting the possibilities of success for projects listed on Kickstarter

Group Number: 5

Group Members: Abhijit Haridas, Nishant Jadhav, Pratik Kunjir, Rakshit Sinha and Vaibhav Balasubramanian

---

Executive Summary: **2**

Data Processing: **2**

Data Cleaning 2

Feature Engineering 2

Text Mining 6

Tokenizing Training Blurb Description 6

Creating Generalized Vocabulary 6

Predicting Numerical probabilities of Success using a L1 Regularized Logistic Regression 6

Model Improvement 7

Model Selection and Evaluation **7**

Model Specification Comparison 7

Evaluation of Various Models 7

Hyperparameter Tuning of Catboost 8

Learning Curves: 8

Key Takeaways: **9**

Challenges: **9**

Things we could have done differently: 9

Things we could have done if we had more time: 9

Business takeaways: 10

New Concepts Learnt: **10**

Executive Summary:

This report describes the steps taken in order to analyze and predict the possibility of projects listed on Kickstarter.com being successful in securing funding from investors. A project is considered successful if it earns more funding than the specified goal amount. If the project is successful, the fundraiser gets to keep all of the money raised, else they do not get any of the money.

The variable selected for analysis and prediction is success. The reason for selecting this variable is it allows for easy understanding and interpretation of how a project would fare given the criteria for success provided by Kickstarter.

By knowing the possibility of success based on detailed analysis of past data, future project creators would be able to understand what it takes to ensure their project receives the required funding. Project creators would also get a view on the factors that influence the possibility of project success, which they can use to curate their listing on Kickstarter to their advantage.

Potential investors can also benefit from the analysis performed as they can view which categories of projects have been successful, whether the funding requested for a particular project is appropriate and who are the top creators. They can use this to make investment decisions - which categories are hot to invest in and whom to back in order to get the best returns on investment.

Finally, this report encompasses predictive models used to make predictions. Companies with data analysts can implement some of the models described below to create their own predictions which they can use to drive actionable insights.

Data Processing:

This section covers the steps undertaken to process the dataset to create a model for predicting success possibilities for the projects listed.

Data Cleaning

Missing values in multiple columns were imputed. Below are a few highlights:

1. Upon inspection we found that tag length has the highest correlation with numfaces_project. Missing values for tag length were imputed by the median of tag length per numfaces_project category.
2. For the missing values in color background and color foreground, the backfill method was used to impute missing values. Backfill was used as there was no significant correlation with any other column.
3. The number of missing values in rewards_steps was 989 which was imputed by the median value of that column. This was done to avoid any issues caused by outliers(if in case a mean was used).
4. Many of the categorical columns were also filled using the backfill method as no significant correlations could be established.

Feature Engineering

The columns in the Kickstarter dataset were inspected and new features created. The below table describes the features created and the benefits received:

Feature Created	Benefit
1. Top 10 creators based on number of projects listed and average goal requested	Selected based on intuition, but no significant benefits were received
2. Target duration	This variable was created by computing the difference between launch date and created date as well as deadline date and launch date, hence it was an

	important variable for our prediction model.
3. Funding Duration	This variable was created by computing the difference between launch date and created date as well as deadline date and launch date, hence it was an important variable for our prediction model.
4. City column from Location Slug	location slug was split into city and state and city was not found to be important variable for our prediction
5. State column from Location Slug	location slug was split into city and state and state was found to be important variable for our prediction
6. Location type - Grouping less frequently occurring location types	Selected based on intuition to see if any location type was important, but no significant benefits were received
7. Top Categories within the Category Parent	We were able to analyze some of the categories in top parent and they were important for our prediction
8. Female Creator > Male Creator and	Selected based on intuition to see if having more male participation in the project was an important factor in achieving success but no significant benefits were received
9. Female Project > Male Project	Selected based on intuition to see if having more female participation in the project was an important factor in achieving success but no significant benefits were received
10. Affinity difference - The difference between positive and negative affinity	selected to find out if there is any numerical relationship between this variable and success, and it was of significant importance for our prediction
11. Goal amount Quantile	the goal amounts were divided into intervals to see if any particular range of goal amounts had the maximum number of success possibilities, but no significant benefits were observed
12. min age for the age of creators	The creator age was divided into min age and max age to see if any of the intervals had an impact on our prediction,it was found that min age was significant for our prediction
13. max age for the age of creators	The creator age was divided into min age and max age to see if any of the intervals had an impact on our prediction,it was found that max age of creators was not significant for our prediction
14. Range of age of creators	The creator age was divided into a range of ages to see if any of the intervals had an impact on our prediction,it was found that range age of creators was not significant for our prediction
15. Min age for age of projects	The Project age was divided into min age and max age to see if any of the intervals had an impact on our prediction,it was found that min age of projects was not significant for our prediction
16. Max age for age of projects	The Project age was divided into min age and max age to see if any of the intervals had an impact on our

	prediction,it was found that max age of projects was not significant for our prediction
17. Range of age of Projects	The project age was divided into a range of ages to see if any of the intervals had an impact on our prediction,it was found that range age of project was not significant for our prediction
18. Created Month	dates were split into created month and created year, but created month was not of any significant importance for our prediction
19. Created year	dates were split into created month and created year, but created year was not of any significant importance for our prediction
20. Launch Month	dates were split into launch month and launch year, but launch month was not of any significant importance for our prediction
21. launch year	dates were split into launch month and launch year, but launch year was not of any significant importance for our prediction
22. Deadline month	dates were split into deadline month and deadline year, but deadline month was not of any significant importance for our prediction
23. Deadline year	The deadline dates were split into months and years, and the deadline year was found to be important.
24. Top 5 States	selected based on intuition to see what were the top 5 states of the listed projects but this was not of significant importance for our prediction
25. Reward amount processing	rewards were split based on range and number of rewards offered and then we found that the number of rewards and average reward amount offered were significant for prediction
26. Top tags	was selected based on intuition to see whether it had any significant importance for our prediction but was found that it was not of any significant importance for our prediction
27. Count top tag	Number of times the tag appeared in the entire dataset
28. Is Top Tag	Top tag appeared more than 50 times in the entire training data . Count top tag>50
29. Average Goal per day	Average goal was computed by comparing the goal requested and duration for funding, this variable was found to be significant for prediction
30. pred_ridge_blurb	Ridge model numerical predictions on blurb
31. 2009_per_capita_incom	Per capita household income for 2009. Was not significant for the model
32. 2010_per_capita_incom	Per capita household income for 2010. Was not significant for the model

33. 2011_per_capita_incom	Per capita household income for 2011. Was not significant for the model
34. 2012_per_capita_incom	Per capita household income for 2012. Was not significant for the model
35. 2013_per_capita_incom	Per capita household income for 2013. Was not significant for the model
36. 2014_per_capita_incom	Per capita household income for 2014. Was not significant for the model

The above features were the optimal number of features engineered. If we tried to exceed this number, then a loss in accuracy was observed indicating towards a problem of '*curse of dimensionality*'.

For instance, we tried to add the number of verbs, adverbs and adjectives for both blurb and reward description. However, this led to decrease in overall validation accuracy for our final model. Hence, we removed this feature.

Categorical variables were converted to dummies to see if any specific types of categories were significant in predicting success. The categories for which dummies were created were as follows:

Category	Important Dummy Variables
goal_bin	No important variables
maxage_creator_bin	No important variables
minage_project_bin	No important variables
category_parent	Music, theater, games, dance and fashion categories were found to be important variables
top_tag	No important variables
istoptag	No important variables
category_name	No important variables
region	No important variables
State	The state of Florida was found to be an important variable
location_type	No important variables
Top_5_State	No important variables
deadline_year	A deadline year of 2014 was found to be an important variable
launch_month	No important variables
created_month	No important variables
deadline_month	No important variables
launch_year	No important variables
created_year	No important variables

isbwImg1	No important variables
isTextPic	No important variables
isLogoPic	No important variables
isCalendarPic	No important variables
isDiagramPic	No important variables
isShapePic	No important variables
contains_youtube	No important variables
top_creator_quartile	No important variables
top10_creators_by_num_projects	No important variables
top10_creators_by_average_goal	No important variables
category_in_top_10	No important variables
fem_creators_greater_than_male_creators	No important variables
fem_projects_greater_than_male_projects	No important variables
afinn_positive_higher	No important variables

Text Mining

Apart from feature engineering, text featurization was also performed on the short project description("blurb")

Tokenizing Training Blurb Description

The training dataset blurbs are tokenized and a vocabulary is created. Below were the pruning steps for tokenization:

1. Stop words removed
2. punctuations removed
3. numbers removed

Note: *Stemming was not performed as important words lost logical meaning.*

Creating Generalized Vocabulary

The following steps were performed to create a generalized vocabulary :

1. Each world appears at least once in each category_parent,
2. Min count of words across the corpus = 50.

A Document Term Matrix(DTM) was created and term frequencies were normalized using TF-IDF. The corpus finally had 116 words.

Predicting Numerical probabilities of Success using a L1 Regularized Logistic Regression

The DTM was binded to the target variable and logistic regression was performed on this dataset. The numerical predictions(probabilities between 0 and 1) were stored in a vector and later binded to the original Kickstarter training dataset as a new column "pred_ridge_blurb".

Model Improvement

This feature ranked 9th in the top features for our final model and improved the accuracy by a minor 0.6%

Model Selection and Evaluation

The next step after feature engineering was to train a model to learn and perform predictions. The dataset was split in a 70:30 ratio for training and validation. A number of models were tried in order to find the model that provided the highest accuracy. The models tried were:

1. Regularized logistic regression using Ridge and Lasso models
2. Gradient Boosting Machine (GBM)
3. Random Forest
4. XGBoost
5. Catboost
6. Light GBM

From the models used, Catboost provided the highest validation accuracy of **76.57%**. Catboost is a model that makes use of gradient boosted decision trees. The number of trees is determined by the starting parameters provided. Catboost comes with an overfitting detector that stops creating trees when overfitting is detected. More details on Catboost can be found on the below links:

https://catboost.ai/en/docs/concepts/algorithm-main-stages

https://en.wikipedia.org/wiki/Catboost

Model Specification Comparison

A variety of models were fitted on the engineered dataset to assess effectiveness and accuracy of predictions on the validation data. The following steps were undertaken to evaluate, compare and arrive at the final model for the dataset.

1. Cross Validation: The validation accuracy was computed using a default **five- fold** cross validation to generalize the results on the entire dataset and avoid reporting inflated accuracies due to overfitting.
2. Hyperparameter Tuning: Parameters of Ensemble based models such as learning rate, number of trees/estimators/max depth were tuned using an extensive “Grid Search” based iterative loop to determine the best combination of parameters for the model.

Evaluation of Various Models

Model	Cross Validation Accuracy	Training Time
Logistic Regression	0.6483	15.8 minutes
XgBoost	0.7389	4 minutes
GBM	0.7485	10.7 minutes
Random Forest	0.7398	13 minutes
Light GBM	0.7413	7 minutes
Catboost	0.7657	5.4 minutes

Final Model Selected: **Catboost**

Parameters:

learning_rate=0.03,
depth=6,
use_best_model=True,
iterations=2200,
boosting_type='Ordered',
eval_metric='Accuracy'

Hyperparameter Tuning of Catboost

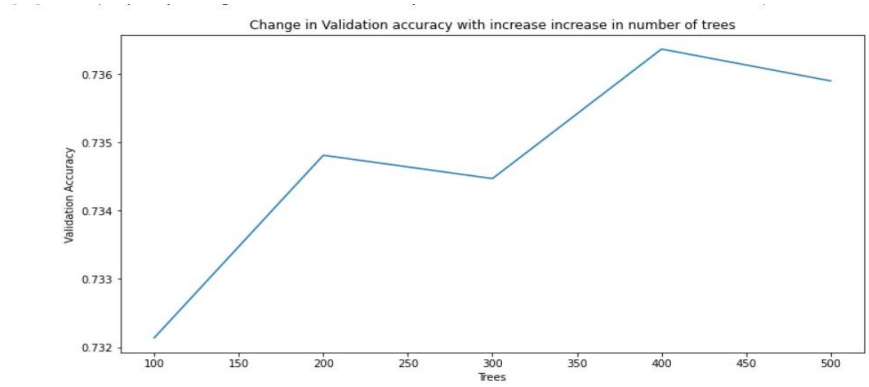
Gridsearch was employed with closely chosen step size for each vector of the below three hyper - parameters.

Parameter	Number of Steps	Range of Vector
Learning Rate	13	0.03-0.9
Number of iterations	16	0.03-0.9
Depth	5	0.03-0.9

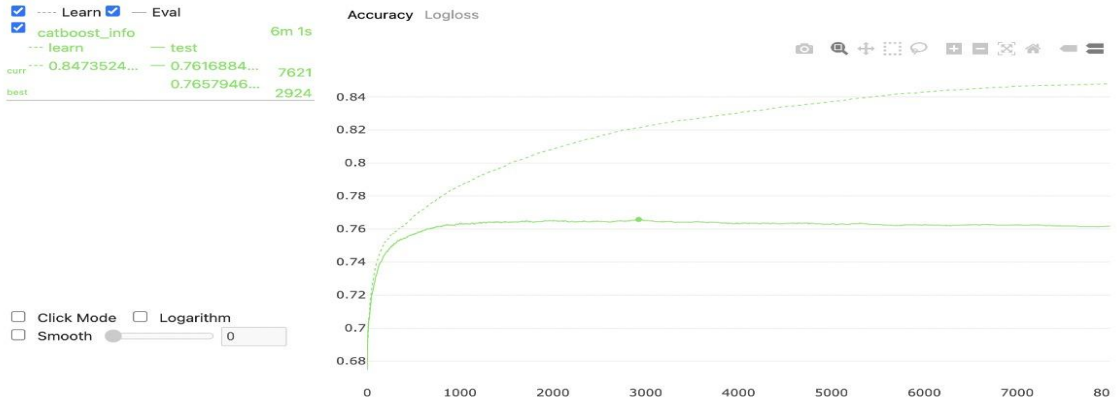
Total number of models assessed for finding best-tuned model: 13*16*5 = **1040**

Learning Curves:
Logistic Regression

Random Forest



Final Model Learning Curves:



Key Takeaways:

After training, validating and checking on the test set, we found that our accuracy was the highest on the testing set (way more than the training set), which is usually not the case (train accuracy > valid accuracy > test accuracy is the standard norm). This could be a red flag as it points towards the “sampling bias issues”. Every model was highly sensitive towards numerical features (having large numerical values). It “could” be the case that the test set was not properly sampled (we can see that all the samples in the test set were from only the year 2014, because of which the engineered “year column” did not have any variance). Thus, if the sampling would have been more “randomized” we could see more pragmatic results.

What went well?

After trying multiple models, the Catboost model provided us with the highest accuracy. When run on the test dataset, an accuracy of 82.21% was achieved which brought us third place among 30 teams performing predictions.

Challenges:

- Lot of text based columns which had missing values
- Lack of correlation between columns containing missing values leading to difficulty in data imputation
- Large number of categorical columns
- After changing categorical columns to numeric categories, there were a large number of dummy columns which increased the dimensions resulting in the issue of ‘*curse of dimensionality*’.
- Since a lot of categorical columns were string based, counts needed to be used to appropriately allocate the category.
- In order to find the number of topics present in blurb and rewards description columns, LDA needed to be implemented which was a challenge as it required domain knowledge.
- Large amount of time is required for models to be trained and tuned.

Things we could have done differently:

- In order to impute values more efficiently, other machine learning models could have been used. Eg. Regression models to impute reward steps.
- Statistical tests such as Mutual Information Gain and Chi-squared test could have been used to figure out the most important features with respect to the target variable(success).
- More than 2-3 models could have been combined as a pipeline to build a better hybrid model.

Things we could have done if we had more time:

- Tuning different hyperparameters with different ranges of values.
- Use more external data for modeling and inference purposes.

- Experiment with different class weights for modeling purposes.

Business takeaways:

- Goal is extremely important to predict the probability of success for a project on Kickstarter.
- The total number of rewards offered plays a significant role in the chances of project success.
- Date of creation and launch of projects also plays a crucial role in predicting project success.
- Ensemble based algorithms(Random Forest, Boosting etc.) are usually preferred for modeling purposes on such datasets.

New Concepts Learnt:

- Implemented Principle Component Analysis (PCA) in order to reduce the number of dimensions, however the decomposition did not turn out well as it led to a decrease in accuracy by 3%. The reason being the decomposition of various dimensions failed to capture the exact importance of some features (goal, reward steps etc.).
- Latent Dirichlet Allocation(LDA) was also used to capture/cluster the entire context from blurb and reward descriptions into 10 different topics. This led to an increase of 0.2% in the overall accuracy. It is to be noted that the different topic numbers were treated as numeric quantities because of the importance associated with each topic.