Kickstarter Data Analysis Result

Authors:

Ryan H. W. C. Pi X. Dai

Ch.1 Introduction

Kickstarter is a famous public-benefit corporation based in New York City that helps entrepreneurs crowdfunding on creative projects. Since people can get a chance to start their business by backers' funding towards a successful Kickstarter project, this platform has grown tremendously in the past few years. Kickstarter has a "All or Nothing" rule that defines a project is successful if the pledged amount has reached above the original goal before the deadline, the creator then can collect the funds. Whereas, the hard truth of doing business is that not every entrepreneur can get enough funding to launch their projects in the end. In order to find what contributes to a fundraising project's success, a dataset of information been collected from launched projects on Kickstarter platform needs to be analysed and each attribute needs to be compared to their contribution and effect towards the project's success or failure.

To analyze the dataset, several feature visualization methods are being performed to facilitate the analysis of the dataset, and interesting findings are being discovered. The category which the Kickstarter project launched appears to be a key element towards a project's success. More discoveries will be discussed in details later in the experiments.

The project then starts with feature reduction. Feature importance computing is being performed to inspect any inapplicable attributes that should be dropped from the dataset. Feature reduction is performed to reduce the data dimension to get rid of highly correlated attributes. Since the dataset is generated by web robots, many numerical data should be transformed to categorical data for easy interpretation, feature encoding is implemented accordingly, and new features are being generated with several redundant attributes. Missing values are being replaced in the dataset. Hence, feature reduction turns to an essential task in data preprocessing in this project.

To make the prediction of each project's success or failure, two different methods are chosen to examine the preprocessed dataset. Classification and regression models' training and prediction are performed based on different target values. Analysis are being made for each model's performance and hence giving a conclusion of the "winning" solution. From a close inspection and the latter models are proved to have a better performance from their evaluation metrics.

Ch.2 System Design & Implementation details

2.1 Algorithms

The algorithms being used in this project for generating prediction models are classification models and regression models. In order to predict the final outcome of a project, among 54 attributes two attributes are picked as the target fields applied for fitting the two different models.

The first target field is attribute status which has five different labels including successful, failed, live, canceled and suspended describing each projects current situation. Because the labels are not numerical, a classification problem is formed in order to predict these labels. From the preprocessed training dataset, support vector machine is a good fit for this high dimensional dataset. Quick model generation produced from its relatively memory efficiency is another reason that SVM is used for classification. Decision tree is also picked to test its performance for comparison. Since the Kickstarter dataset has the high dimensional attributes, decision tree is tend to give accurate prediction based on the training dataset but exhibits overfitting.

The second target field is the amount of pledge in USD. Since the amount being pledged by backers decides a project's final outcome, to estimate the amount a project can raise from the backers is another metric to evaluate a project's potential to success. After dropping all the dependent attributes, regression problem is formed build on the requirement of real value prediction. Therefore, regression prediction is employed with KNN-Regression model for its simplicity and easy interpretability. KNN-Regression models uses the same simple KNN to calculate the average distance of the K nearest neighbors. To have a better evaluation of the regression models, a second model is needed in order to make comparisons. With multiple hidden layer between input and output, multi-layer perceptron regression model is picked for comparison with KNN-Regression.

2.2 Technologies & Tools used

The applied tools includes Pandas, Numpy, Scikit-learn, Seaborn, Matplotlib and Word-cloud. Each package's usage is explained in detail at the following paragraph.

Pandas is used to perform data analysis of the dataset. The original dataset is read from a .csv file. Feature extraction, scaling, transformation and encoding are all performed by Pandas. Pandas is the most essential data preprocessing tool used in this project.

Numpy is used for arrays and matrices building and calculation. It is used for preprocessing and essential to attributes selection.

Scikit-learn is another critical package that provides the dataset learning models. The package is very useful for splitting the dataset, and applying the classification models and regression models in this project.

Seaborn package is used in this project to plot this correlation map with each pair of attributes' covariance printed on the map. Due to the amount of attributes (54 in this project), data reduction has to be performed to reduce data redundancy. Before carrying out the data reduction process, a correlation map should be produced to identify highly related attributes.The plotted correlation map is then used to determine which attributes to keep and which attribute to discard based on their correlation coefficient. The plot is shown in chapter 3.2.

Matplotlib is another package used to visualize the dataset and predicted results. Many interesting findings are found from plotted dataset. The plotted graphs are presented in chapter 3.2 with elucidations.

Word-cloud is a data analysis package for natural language processing. The Kickstarter dataset contains several attributes are mainly string with names and blurb words. Tag clouds are built to visualize the frequency used in these projects, and interesting findings are presented when these clouds are built by all successful projects and failed projects.

2.3 System Design Diagram

Below is the system diagram showing the entire project's workflow and dataflow (Fig.1). Starting from the left, the first step is visualizing the dataset from different perspectives. After gaining a general view of the dataset, data preprocessing is performed to prepare the dataset ready for training. In this project, feature reduction is a major task of data preprocessing since the large dimension of attributes. After data preprocessing, two training datasets are generated for two kinds of training: classification and regression. With the training models being built, the two kinds of models are evaluated by confusion matrix and RMSE, respectively.

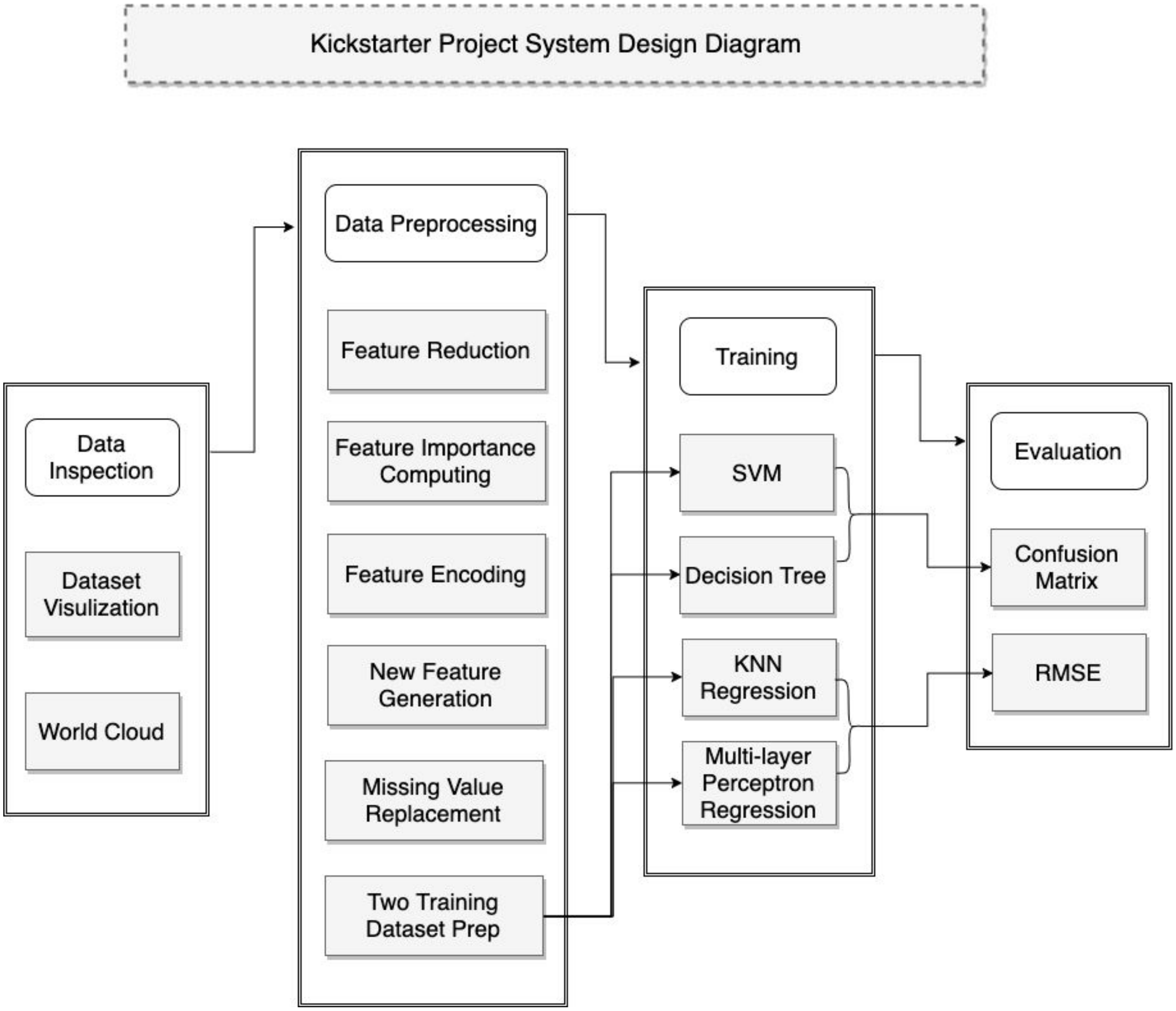


Fig.1 System Design Diagram

Ch.3 Experiments / Proof of concept evaluation

3.1 Dataset Description and Data Preprocessing.

The dataset used in this project is from a Kaggle project with data attrived from Kickstarter platform from 2009 to 2017 generated by web robots. The dataset is 56 MB that has 99,035 records. There are a total of 54 attributes for each records that representing the projects' origin location, name, blurb word description, amount of goal and pledged, launch duration etc.,

Since the dataset was generated by web robots, many of the attributes are not easy to be interpreted. A through attributes investigation have been done on each attributes to compare against Kickstarter's project page to figure out the meaning of the attributs. Some baffling examples are: prct_goal which later being discovered meaning the ratio between the pledged amount, and spotLight which means a special recommendation with the project's story on the front page. The whole list of attributes description is listed in Appendix.

The 54 different attributes are not tailor-made for any machine learning algorithm. Many redundant attributes with repeated and related information presented in the dataset can be easily noticed by simple glimpse. To fully understand the correlation of each pair of the attributes, a correlation map of the attributes is being generated, and it can be seen that most attributes can be dropped due to the fact that they are highly correlated. With the inapplicable attribute removed, some new features are generated from these attributes to fill the missing information after dropping them. Examples are the length of the project's blur slogan and their names are possibly related to the project's success; therefore, the ratio between the length of words and length of characters in these two attributes are being generated as two new attributes. The result will be tested based on the performance of these attributes, but later it may be removed if bad results are produced caused by them.

The final attributes that has being preprocessing ready for next step are country, city, goal_USD, status, staff pick, backers count, categories, spotLight, deadlineY, pledge_per_person, length of kick, city pop, latitude, longitude, county, state, days spent making campaign, all_time_backers_city, mean_pledge_city, len_name, word_name, word name radio, len_blurb, word_blurb, word blurb radio.

3.2 Visualization of Dataset

Some interesting findings are discovered from graph visualization of some attributes in the dataset. To form a general view of the dataset, the percentage of success and failure is being investigated. A pie chart is shown below (Fig.2), and the percentage of success and failure can be seen is around the same. Interestingly, many projects are canceled eventually before their fundraising deadline. To fully investigate what kinds of projects have succeeded and failed most. An interesting plot is generated on the right below (Fig.3). Music and publishing are two categories that both succeeded and failed the most in all categories. Technology which is ranked third in the total number of categories succeeded less and failed more. From this plot shows that the number of being successful and failed is divided very even for most categories.

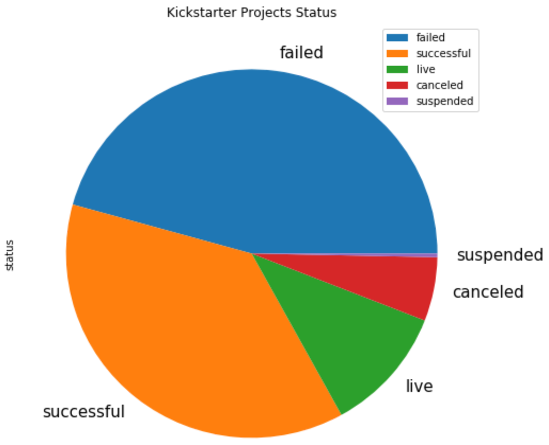


Fig. 2 Status Distribution

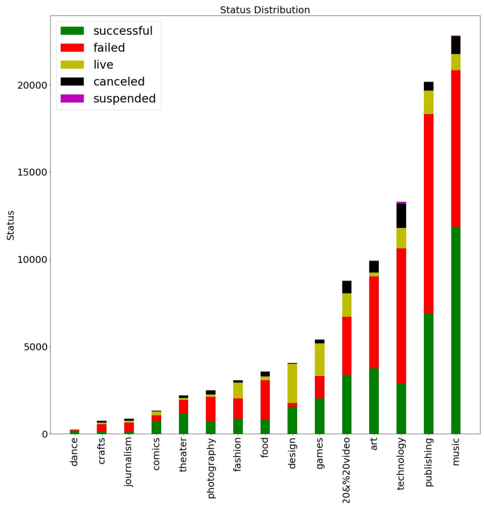


Fig.3 Category Distribution

The time span from 2009 to 2017 with each month's number of launched projects and their final result is shown below (Fig.4). It can be seen that time is irrelevant to a project's success since the amount of successful project are around the same throughout the entire time span.

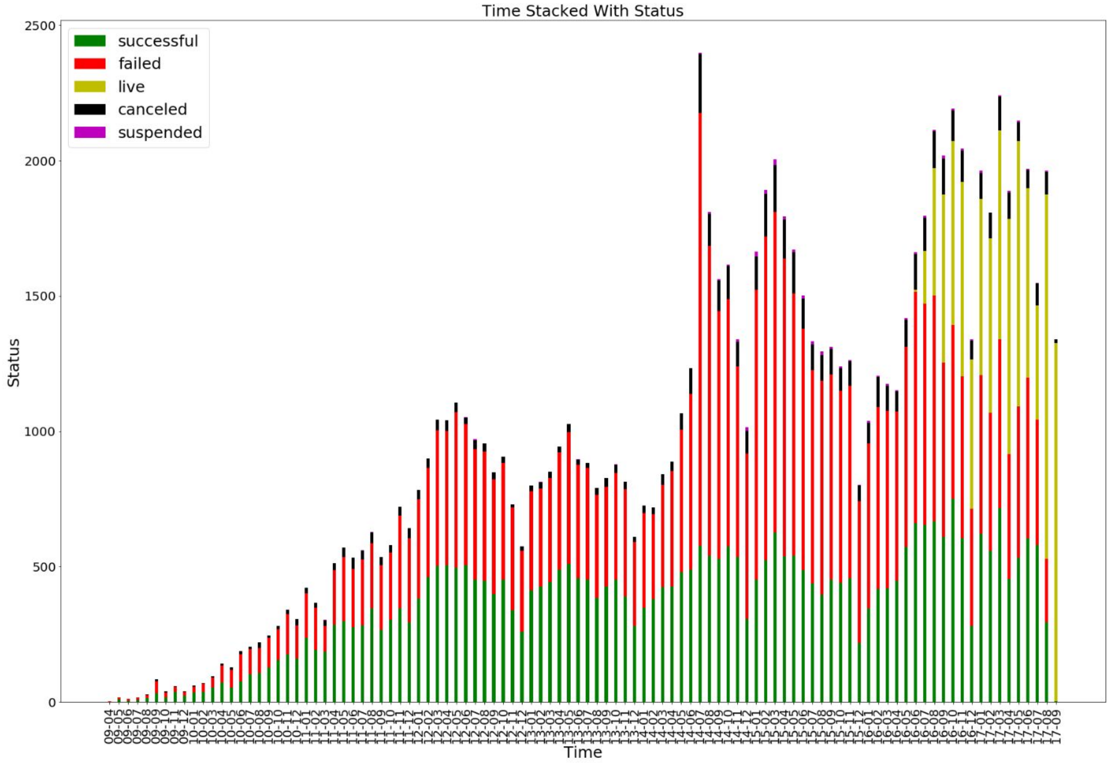


Fig.4 Status - Time Distribution

Both the average amount pledged (Fig.5) and average amount pledged per backer (Fig.6) for each category is shown below. The design sector is the most funded on average by category wise, and the technology sector is the most funded by individual backers.

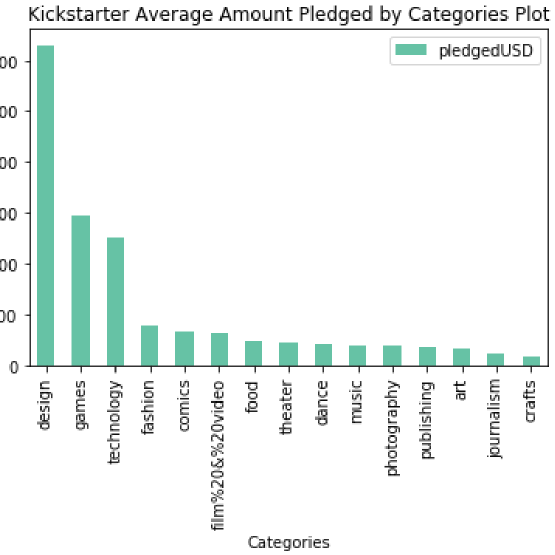


Fig.5 Category - Pledge

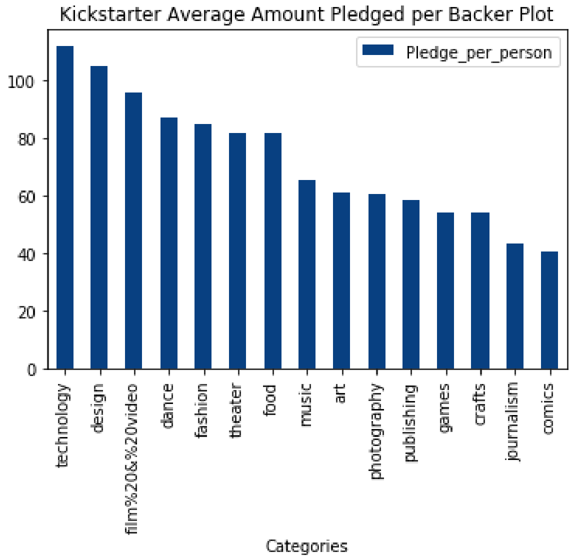
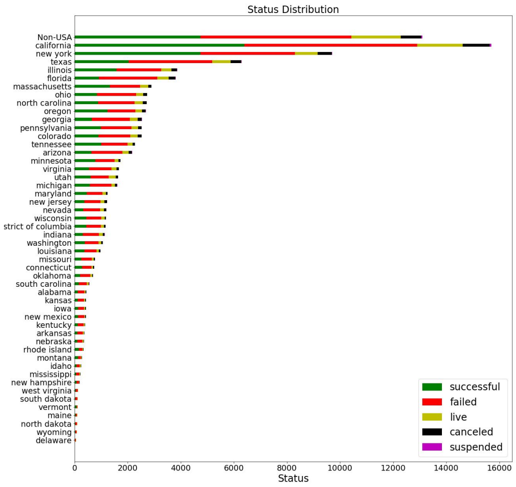


Fig.6 Category - Pledge/person

From the country perspective, most Kickstarter projects are launched from the U.S. (Fig.7), which accounts for about 86% of total projects. Interestingly, around 16% of total projects are launched in the state of California. State of New York and Texas followed as second and third. Lastly, the correlation map (Fig.8) used in attribute reduction is shown on the right.



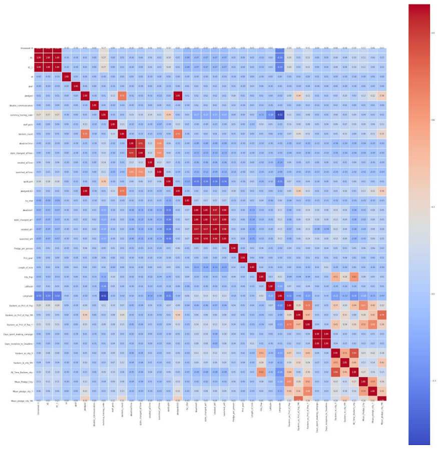


Fig.7 Country Distribution

Fig.8 Correlation Map

3.3 Methodology followed

The dataset is then being standardized by taking the z-score on all the numerical columns. Training and testing dataset is being split by using Scikit-learn model's train test split with random state set to 50 and test set accounts for 25% of the total dataset. The training set has a size of 61620 tuples, and the testing set has a size of 20540 tuples. Two different kinds of training datasets are being prepared with different target fields for classification models and regression models.

The validation is being performed by using trained model to predict testing set, and the result is produced by applying a confusion matrix for classification, and RMSE measurement for regression. Precision and recall are reported on the notebook, and RMSE is printed for reference. Later, the results are compared and analyzed for their performance.

3.4 Analysis of Results

For the classification models, SVM gives an accuracy of 0.5094 and decision tree gives an accuracy of 0.9828. For the regression models, KNN-Regression has a final RMSE at 0.4923, and multi-layer perceptron regression has a final RMSE at 0.49945.

The prediction accuracies of classification models did not perform as expected, the accuracy with SVM has only 0.51 which is essentially random guessing. The decision tree gives a very high accuracy rate which appears to be a good result, but it may have overfitted on the training dataset due to the high dimension from the attributes. On the other hand, regression models are performing comparative well with a relative desirable RMSE rate. Both KNN-Regression and MLPR gives satisfactory prediction of the final pledged amount. For the classification models, there may be many reasons for the inferior results, including the models flaw with dealing high dimensional dataset and the complexity of preprocessed dataset. These reasons can have a significant impact on the final predictions. Dataset preprocessing is here proved to be a crucial part in data mining projects which affects the final accuracy of the models.

Ch.4 Discussion & Conclusions

4.1 Decisions Made

- Features related to location, population, goal, categories are selected to represent the data
- Numerical attributes like the city population needs to be standardized before training
- Robust models including SVM, decision tree, MLPR and KNN-Regression are used for comparison

4.2 Difficulties Faced

- Attributes selection/reduction
- Complex data visualization
- Correlation map generating
- Categorical data encoding

4.3 Things That Worked

- Visualizing dataset gives a general picture of the attributes
- Selected models gives satisfactory prediction
- Cross validation with proper model parameters

4.4 Things that Didn’t Work Well

- Evaluation on real project's success/failure prediction
- Evaluation of feature selection and transformation
- Training model selection

4.5 Conclusion

The project starts with analyzing the Kickstarter dataset by visualizing different attributes to discover the hidden knowledge in the dataset. Interesting findings are being observed from the graph representations of the data. It can be seen that real-world data is “dirty” with redundancy and complexity that requires numerous preprocessing work. Attribute selection therefore becomes the major task in this project. By applying correlation analysis between each pair of features, highly correlated attributes are being removed since they carry redundant information. Decisions are being made to create new attributes by using categorical attributes that cannot be fed into the classification and regression models used in this project. The effectiveness of the preprocessed dataset as well as the learning models’ choice contribute to the final accuracy of the prediction results. Between the two kinds of prediction models that have been chosen, regression triumph classification models with the preprocessed dataset. Whereas, the effects of the chosen attributes on the performance is still unclear. Further investigation on the attributes selection should be conducted to achieve a better prediction result.

Ch.5 Project Plan / Task Distribution

Task	Member
Literature research	Chao Pi
Baseline implementation	Xunan Dai, Ryan Haoyuan Wang
Attributes investigation	Xunan Dai, Chao Pi
Data visualization	Chao Pi, Ryan Haoyuan Wang
Data preprocessing	All
Model implementation	Xunan Dai
Report writing	Ryan Haoyuan Wang

Tab.1 Task Distribution

Appendix

Attribute description:

1) country: country the project initiated
2) city: city the project initiated
3) id: Kickstarter project ID (drop)
4) name: project's name (drop)
5) blurb: a short description of the project (drop)
6) goal: fundraising goal of the project(drop)
7) pledged: actual amount being raised by supporters (drop)
8) status: project status, most finished projects are labeled as successful or failed
9) slug: a hyphen-delimited version of the name feature (drop)
10) disable_communica: if the creator disabled the communication channel (drop)
11) currency: currency used for pledge (drop)
12) currency_symbol: currency symbol (drop)
13) currency_trailing_code: if the currency has trailing (drop)
14) deadline: date of the final pledge date (drop)
15) state_changed_at: change state at (drop)
16) created_at: project created at(drop)
17) launched_at: project launched at (drop)
18) staff_pick: staff pick as special recommended to backers
19) backers_count: number of backers
20) deadlineTime(drop)
21) state_changed_atTime(drop)
22) created_atTime(drop)
23) launched_atTime(drop)
24) categories: project's category
25) spotLight: special recommendation with a project's story on the front page
26) pledgedUSD: pledged amount in USD
27) ex_USd: exchange rate with USD (drop)
28) deadlineYM(drop)
29) state_changed_atYM(drop)
30) created_atYM(drop)
31) launched_atYM(drop)
32) deadlineY
33) state_changed_atY(drop)
34) created_atY(drop)
35) launched_atY(drop)
36) pledge_per_person: average of pledge per person
37) prct_goal: ration between goal and pledged (dropped in classification model)
38) length_of_kick: day of the project on Kickstarter
39) city_Pop: Project's city population
40) latitude
41) longitude

42) county: county the project initiated
43) state: state the project initiated
44) backers_as_Prct_of_Pop(drop)
45) backers_as_Prct_of_Pop_YM(drop)
46) backers_as_Prct_of_Pop_Y(drop)
47) days_spent_making_campign
48) days_inception_to_Deadline(drop)
49) backers_in_city_Y(drop)
50) backers_in_city_YM(drop)
51) all_Time_Backers_city
52) mean_Pledge_City(drop)
53) mean_pledge_city_Y(drop)
54) mean_pledge_city_YM(drop)