Final Project

Group 2 (Siemens)

By

Nolan Beck, Matthew Costello, Spencer Daugherty, Brian Freiss, Jake Gadaleta

Executive Summary:

We have spent this semester examining crowdfunding environments, focusing mainly on KickStarter, in order to find techniques which maximize success on the platform. Our initial objective for this project was to use our analysis techniques to determine what classification strategies will lead to large quantities of funds received and large quantities of individual donors/donations. This information may be particularly useful for those looking for some start-up financial capital for projects, enabling them to ensure the reception of the necessary money to get those projects off of the ground. However, it may also be used by those looking to donate to certain KickStarter projects, as donors may want to be sure that their money will be used in a successful project. This analysis may assist in their decision making as they choose which projects to donate to.

We found that most KickStarter projects tend to fail, which leads to us to question what variables tend to impact these fail states. After running through a litany of models, our decision tree was able to find that the most impactful variables had to do with how much money was asked for, how much money was provided, and the amount of people giving to the project.

Project Background:

The initial inspiration behind this project stemmed from the brother of one of the researchers, Jake Gadaleta. That particular brother was known to be considering using KickStarter to get a project off of the ground, and so Jake decided to look into the formation of a successful KickStarter project. The hope was that by doing this analysis, Jake's brother may be able to utilize the strategies identified herein to raise the financial capital necessary to complete his venture.

Data Description:

This data is from 2016-2018 and contains more than 350,000 records, each with 16 variables. For more information on the variables, please see the attached data dictionary. The data was originally released on Kaggle for a data science project. We used Python to expand the data with two new variables ("success_factor" and "success"), both of which were used as targets for the models.

Data Preparation and Processing:

Exploring the Data

We first attempted to sift through the data with SAS Enterprise Miner in order to locate any outliers. With this information, we were able to gauge whether or not they should be removed from the data set. We accomplished this by using Cluster nodes. We found several data points that were fairly egregious, such as a project that had a “success_factor” of 200.00 (this means that it garnered two hundred times the amount of funds requested), and some that had a “success_factor” of 0.000001.

Preparing the Data

For our initial “rough” round of analysis, we elected to filter out many of the KickStarter projects that were substantially more successful than the others in the dataset (such as the one mentioned above with a “success_factor” of 200.00). This was because these projects led to our first cluster analysis providing a heavily skewed view of the data. Following this filter, we were much more successful in developing a succinct data spread which could be used for further modeling.

After the clustering, we decided to attempt some regressions in order to find the optimum model for predicting monetary gain. However, we ran into an issue nearly immediately due to the size of the data. The initial thought was that by swapping out Regression for HP Regression we could handle the higher amount of data (that being the purpose of the HP Regression node), but it still did not work. This resulted in us randomizing the rows in the dataset and taking only the first 300,000, as we were unable to successfully regress the entirety of the dataset.

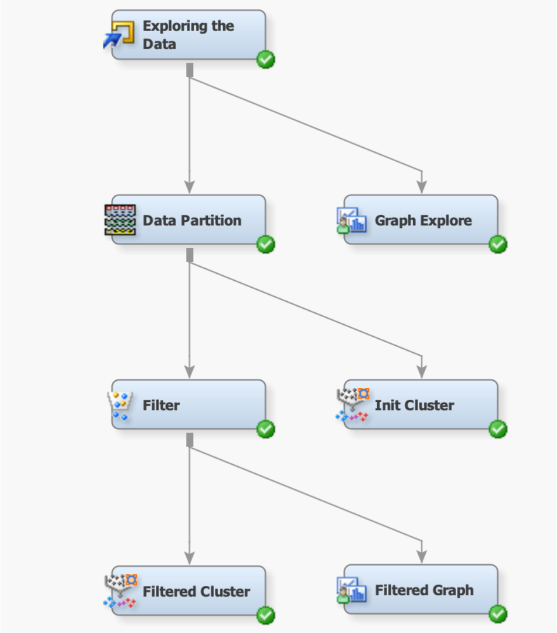

Diagram 1

Regressions

After limiting the data, we were able to disregard the HP nodes and resort back to the nodes that we have been using all semester. We decided to use the variable “success_factor” as our target variable. After filtering, the initial regression used all variables and obtained an Adjusted R-Squared value of 0.6225, finding “backers” and “usd_pledged” as the greatest indicators. We then followed up the initial regression with one including only these top indicators. This made our model take a massive hit, as the new Adjusted R-Squared became 0.0000, making this model effectively defunct. Because our model didn’t hold up, we moved forward and attempted to implement Principle Component Analysis instead.

The Principle Component Analysis became a larger disaster than the Regressions. The main issue that we encountered is that “success_factor” is essentially too small of a variable to be accurately defined by any model. We attempted to adjust the Eigen Value to all possibilities that SAS would allow, but none generated an effective regression afterwards.

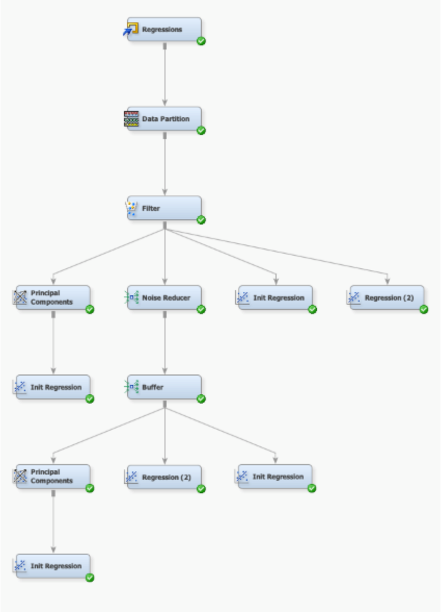

Diagram 2

Auto Neural Networks

We attempted to run the Auto Neural Network using a variety of settings that SAS had provided to us. We wanted to predict “success_factor” to start, but the network was unable to accomplish this goal. We had initially thought that perhaps the algorithm was failing due to the close range of the “success_factor” variables, reaching from 0.01 - 2.00. We attempted to correct this by manipulating the data and increasing the “success_factor.” This was achieved by multiplying it by a value of 100. However, this did not solve our Neural Network issue. We believe that the variables were largely incapable of predicting the exact value within due to something outside of the collected data. This could be something related to marketability power, social media support, or some other confounding variable. At this point we opted to switch gears, looking to see if we could predict the binary variable success.

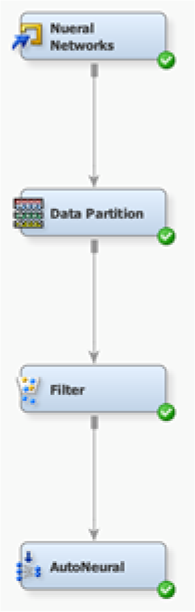

Diagram 3

Decision Trees

The variable's success can hold one of two values: “True” (denoting that it raised as much or more money than it requested) or “False” (the project failed to reach its goal). Because we were aiming to predict a binary variable, we decided to use a Decision Tree. The Decision Tree was largely given the opportunity to use any variable (though we removed “state,” as it contained mirrored values), and we allowed SAS to make the majority of decisions through that methodology. Because this initial Decision Tree proved to be very useful, we then decided to take the next step and create a secondary Decision Tree aimed this time at the “state” variable. One thing that we noticed in these first two decision trees was that two sets of variables contained similar information (this being the amount of money pledged): “usd_pledged,” and “usd_pledged_real.” By only keeping one of these variables, we managed to reduce the steps that Decision Trees took for both the “state” and “success” variables.

The stated Decision Trees are slightly less accurate due to some overlap in the senary values, such as “CANCELLED”, “FAILED”, and “SUSPENDED,” which all denote that the project failed. We should note that due to this overlap, we believe that it is acceptable that there are multiple values in the Tree’s terminal nodes. Having multiple values in the same terminal node, as long as they overlap, does not lessen the model to any degree.

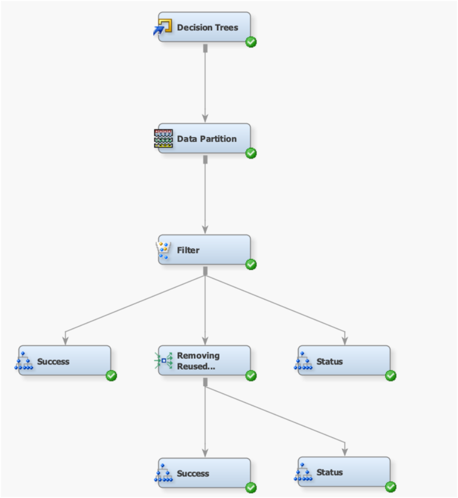

Diagram 4

Our Conclusions:

Final Diagram Used

As a result of our analysis methodology, we had a separate diagram for each of the various analysis techniques. Thus, see *Diagram 2* in "Regressions," *Diagram 3* in "Auto Neural Networks," and *Diagram 4* in "Decision Trees" for the final analysis diagrams.

Findings

In the end, the most valuable models that we produced were our Decision Trees, and specifically those trees that filtered out the repeated data. These models tell us that the project's goal, the pledged amount, and the number of backers were the three most important factors in this analysis. We also learned that there is no mixture of the given variables that accurately determines the "success_factor" of any given project.

One hypothesis that we entered this data set with was that the primary determining factor to receiving the funding required would be some level of category. However, that was really not the case here, as we instead found that the number of backers ("backers") tends to supersede all other variables. The Tree found that if the project maintained 17 or more backers, it had a significantly higher chance of succeeding (71.50% true as compared to 28.50% false). After this decision, the Tree alternated between using "usd_pledged_real" and "usd_goal_real" with smaller and smaller ranges in order to procure a set of optimal ranges.

All of the terminal nodes managed to predict the correct values with less than 5% error. The exception was a singular terminal node containing 12 values with a near even split between true and false.

Managerial Implications

The most important conclusion to be drawn from this analysis is that what creators decide to make has significantly less impact on their chance of raising funds than we originally anticipated. What a creator should do in order to make sure that their project has the highest possible chance of success is to go into the idea with friends already willing to donate. Even a little money that gets the initial ball rolling might incentivize donations from unrelated parties as well. Additionally, taking care to ensure that the project is not overvalued is critical. While it may be difficult to pick a reasonable target price, as it will vary from project to project, we found that it should not exceed 14.92% more than the minimum necessary. We made this determination by taking the ranges given by our Decision Tree and finding the percentage of the average difference.

Data Dictionary:

Variables Explicitly Referenced as Targets in this Report

- “success_factor”: a variable that denotes whether or not an individual project met its goal. A value of 1 means that the goal was met, a value of greater than 1 means that the goal was surpassed, and a value of less than 1 means that the goal was not met
- “success”: a boolean denoting if the project had more money pledged to it then it set as its goal containing the values [“TRUE”, “FALSE”]. This data can be represented by “success_factor,” but was purposefully parsed out in order to run decision trees
- "state": a senary variable denoting the current state of the project. “State” can be defined as “CANCELLED,” “FAILED,” “LIVE,” “SUCCESSFUL,” “SUSPENDED,” or “UNDEFINED”

Variables Not Explicitly Referenced as Targets in this Report, but Included in Dataset

- “Index”: an unused variable which indicates the location of the object prior to the data being shuffled and aspects being cut off
- “ID”: KickStarter internal database reference number
- “Name”: the name of the project; commas were removed because the data was saved as a comma separated values file
- “Category”: a user-defined large scale description of the project. In the data set there are 159 unique categories
- “Main_category”: KickStarter defined description of the project. There are 15 unique values: “ART,” “COMICS,” “CRAFTS,” “DANCE,” “DESIGN,” “FASHION,” “FILM & VIDEO,” “FOOD,” “GAMES,” “JOURNALISM,” “MUSIC,” “PHOTOGRAPHY,” “PUBLISHING,” “TECHNOLOGY,” “THEATER”

- “Currency”: the currency requested for the project. There is a high correlation between currency and country
- “Deadline”: the last day of fundraising; this date can shift on the decision of creator
- “Goal”: the money requested
- “Launched”: the start date
- “Pledged”: amount of money pledged, recorded in the currency of the project; does not include tax information
- “Backers”: an integer denoting the amount of people who gave to this project
- “Country”: country of origin of the project
- “Usd_pledged”: the amount pledged converted to USD, does not include taxes
- “Usd_pledged_real”: the amount pledged after taxes
- “Usd_goal_real”: the amount of the goal in USD