Akshay Shah

Brown University

12/2/20

https://github.com/akshay7424/Data1030_Project_Akshay.git

Predicting Startup Success

I. Introduction

This project will explore the potential success of startup companies by examining key features that economically and financially impact these companies. The target variable is categorical, the exit status of the company, and the only two options are acquired or closed. For simplicity, this model considers IPO and acquired the same category of 'successful' exit. This is a classification problem with the goal of predicting discrete success or failure of startup companies.

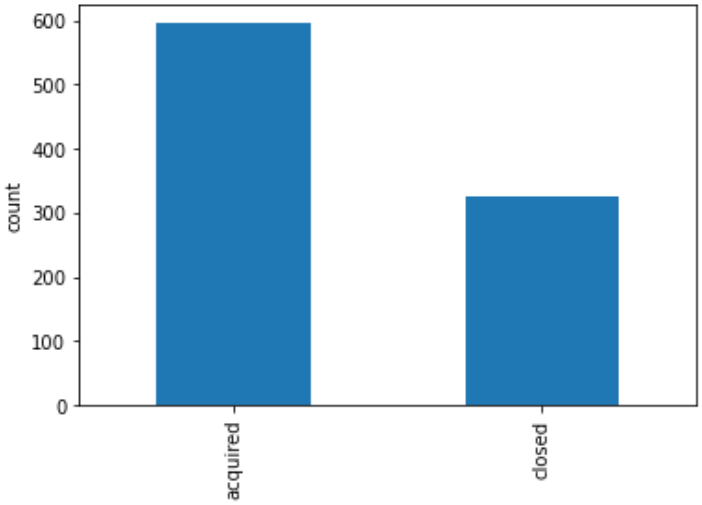
Looking at forecasts of startup companies using machine learning is a relevant yet largely untouched field. Currently, venture capital firms use intuition, financial projections, and personal connections to determine which companies receive seed money. Understandably, there is high risk and uncertainty associated with venture capital where only a handful of companies become unicorns (> $1bn valuation) with successful exits. The goal of this machine learning project is to shed some light into what variables contribute to a company's success, outside of financials, to reduce the randomness and personal bias in venture capital investment.

The dataset has 49 total columns however, four of the columns are unique id columns and one is a duplicate. After ignoring these five, the remaining features are made up of categorical features, continuous, and dates to be transformed into continuous features. The dataset is from Kaggle however, there is no prior publications on the dataset. A few of the feature names are ambiguous: relationships refer to larger corporate partnerships and milestones are qualitative progress indicators. The dataset contains 922 data points.

II. Exploratory Data Analysis

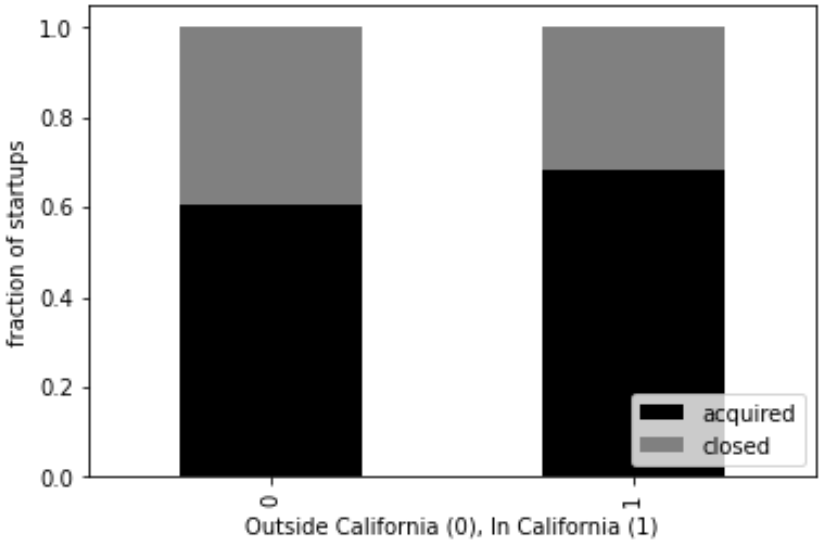
Exploring each of the features revealed interesting relationships within the dataset. Figure 1, regarding the target variable, shows that the overall number of startups acquired nearly doubles the number closed. Given that most startups fail, this dataset shows skew towards more successful startups.

figure 1: value counts for exit status



Further plots were created to examine how location affects startup success. The focus of this was to compare the entrepreneurship and venture capital hub, California, with the rest of the country. Figure 2a compares startups in California with the target variable exit status. Supporting figure, 2b, examines California startups versus funding. The resulting observations fell within expectation that Californian startups would succeed and receive funding at a higher rate than out-of-state counterparts.

figure 2a: exit status vs. in CA



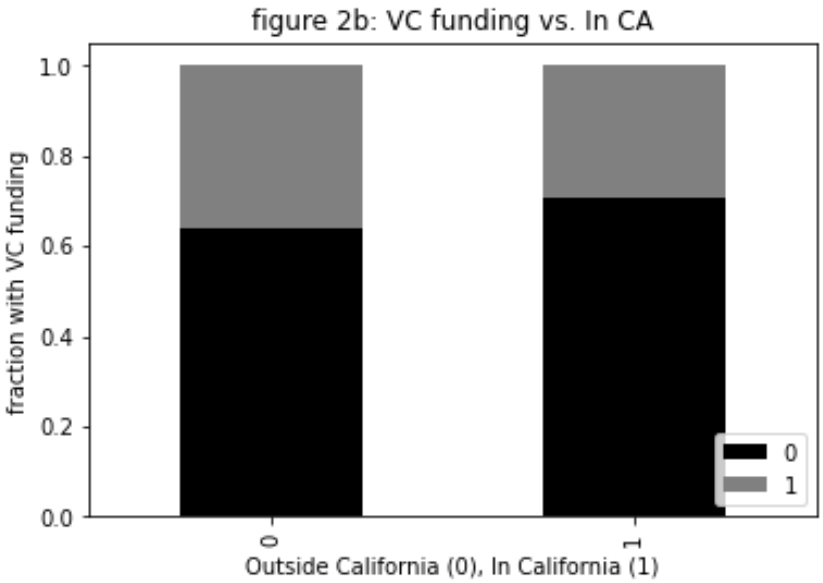
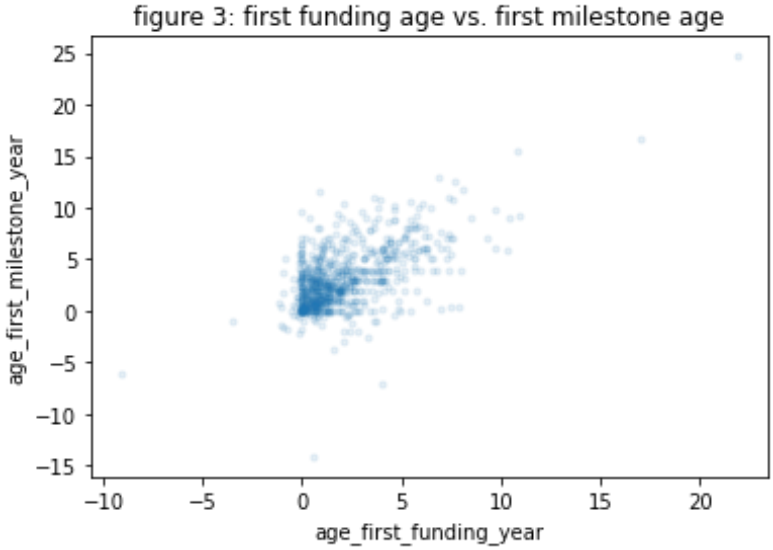


Figure 3 displays the relationship between the age a startup was when it first received funding relative to when it reached its first milestone. The scatter plot shows a high concentration near age zero meaning most companies received funding as they were incorporated. There are a few outliers that received funding much later or earlier in their lives. The outliers also showed that they reached their first milestone shortly after receiving funding. Ages that are represented as negative numbers refer to companies before they are officially formed (i.e. funding an idea in an incubator).



III. Methods

A 60/20/20 train, validation, test split was initially considered since the dataset is IID but relatively small with less than 1,000 data points. The original splitting strategy discussed above was a basic 60/20/20 train, validation, and test split. However, the initial idea was replaced with a k-fold cross validation method of splitting. Each model was trained across ten random states and each split contained four splits.

The dataset is neither a group structure nor is it time series. The goal of the machine learning question in this project is to predict the outcome of startup companies so the categorical feature status was chosen as a target variable. Furthermore, label encoder is used to preprocess the target variable since it is categorical.

The preprocessing was split between categorical and continuous variables which were transformed using one hot encoder and standard scaler, respectively. One hot encoder was used on the following features: state code, zip code, city, industry (category code), average participants, has VC, has angel, funding rounds (A, B, C, D), is top 500, and milestones.

Standard scaler was chosen for all continuous features as most do not have a specific range where the min/max scaler would be ideal. The following features were transformed using standard scaler: latitude/longitude, age first/last year funding, age first/last year milestone, relationships, funding rounds, average participants, and total funding. Additionally, four columns with dates as entries were converted to continuous features using a method called epoch time. This looks at the time elapsed since January 1, 1970 so dates can be easily processed. The four features were date founded, closed, first funding, last funding.

The original dataset had four columns containing missing values, shown in figure 4. The missing values in column, Unnamed: 6, were replaced with the string 'missing' since it is a categorical feature. The date closed at column shows as object since it had not yet been transformed into a continuous feature. Sklearn's simple imputer with mean imputation was used on the three continuous columns: closed at, age first milestone, and age last milestone. Although simple imputation is often detrimental to the model, it is employed for this project since two of the columns with missing values have relatively low fractions of missing values.

Figure 4: Missing values percentages and data types

```
fraction of missing values in features:
Unnamed: 6                  0.533623
closed_at                   0.636659
age_first_milestone_year    0.164859
age_last_milestone_year     0.164859
dtype: float64
data types of the features with missing values:
Unnamed: 6                   object
closed_at                    object
age_first_milestone_year    float64
age_last_milestone_year     float64
dtype: object
```

When determining which models may be the most effective classification algorithms, I considered support vector, k-neighbors, random forest, logistic regression, and xgboost classifiers. Ultimately, support vector, k-neighbors, and logistic regression classifiers were selected to train the data. The ML pipeline kept track of test scores and best models for each of the ten random states that were iterated over. GridSearchCV was used for hyperparameter tuning and since the problem was classification, the scoring parameter was accuracy score. The return statement provides the grid, best hyperparameter combination (best model) and the test scores.

An additional consideration that went into the ML pipeline making sure random states were set within the split and k-fold for each iteration in the loop. The parameters of the ML_pipeline method was set to take X, y, preprocessor, ML algorithm, and parameter grid to simplify applying the pipeline to different algorithms. The label encoder was applied on the target variable within the pipeline method.

IV. Results

The baseline accuracy of the model was 64.6%, since roughly a third of the startups closed and two thirds succeeded.

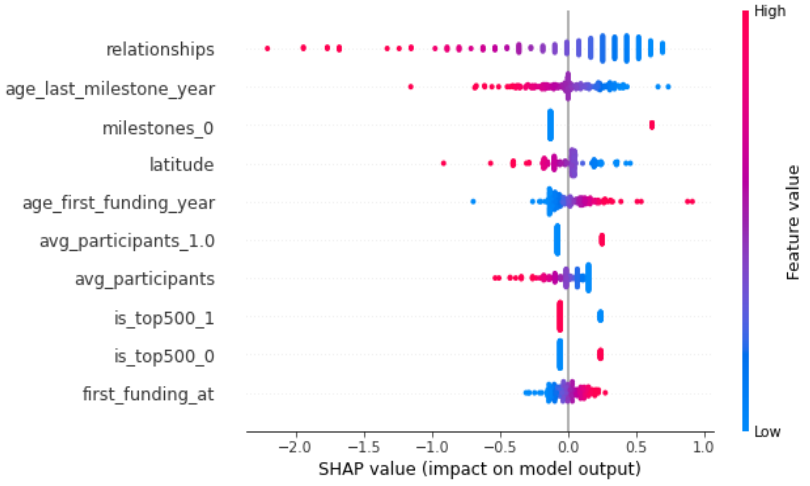
Support vector classifier yielded an average test score of 84.6% with a standard deviation of 2.2%. The optimal parameter combination with C as 10 and gamma as 0.01 resulted in a test score of 88.6%. The best test score was 10.7 standard deviations above the baseline accuracy score.

K-neighbors classifier performed slightly worse across the ten random states with a mean test score of 76.8% and a standard deviation of 2.9%. Tuning the n_neighbors parameter showed that n_neighbors of five produced the highest test score, 82.2%, which was 6.1 standard deviations above the baseline accuracy.

Finally, logistic regression fared the worst with a mean test score at 74.9% and standard deviation of 2.4%. Logistic regression was run with l2 regularization and a max iteration of 10,000. After tuning the C parameter, the best test score was 79.5% with C at 0.1. This was 6.2 standard deviations above the baseline.

Local feature importance was observed using a linear explainer for the logistic regression model, shown in figure 5. Interestingly, relationships took the top spot for feature importance. This notion seems to support the original theory positing that the venture capital industry relies heavily on personal connections in absence of financial information. Furthermore, two features related to milestones were in the top three most important features. This is natural since a

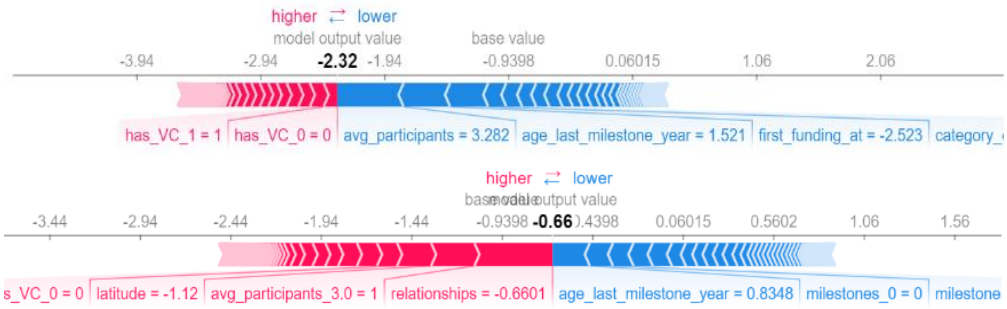
Figure 5: Shap value summary using linear explainer



startup hitting milestones would generally indicate some form of business successes. Interestingly, latitude was fourth on the list while longitude was missing.

Figure 6a and 6b show force plots using the linear explainer for a successful and failed startup respectively. The successful startup shows that average participants and age at last milestone had the heaviest impact on the output, while the failed startup relied on relationships and average number of participants. Additionally, it is worth noting that the base value and model output value are very close for the failed startup.

Figure 6a/b: Force plots for a successful and failed startup



Global feature importance was evaluated using feature perturbation. The top two features were last funding date and age at first funding. Intuitively, funding plays a large role in the success of a startup whether the funding is venture capital or angel investor, a startup needs capital to upkeep its operations.

V. Outlook

Location was surprisingly absent during feature importance analysis, given the stress on highly influential startups in California, New York, Massachusetts, and Texas during EDA. Aside from latitude in appearing in figure 5 and minorly in figure 6b, there are no relevant mentions of location in global or local feature importance. Perhaps feature engineering with converting zip code to median incomes per zip code would improve the test score and relevance of location in the model. Furthermore, qualitative features such as an ordinal CEO likeability measure could be implemented to improve the model.

A potential oversimplification of this project is the binary target variable of acquired/closed. While these seem like enough on the surface an IPO vs. acquisition carry vastly different implications for the future of a company. Generally, an acquisition comes with far more security and less volatility than an IPO. Using multilabel classification may shed some further light on the nuances between different ways a startup can succeed.

The intent of this project was to model and predict the success of startup companies using minimal financial information. The three models, support vector, k-neighbors, and logistic regression yielded surprisingly high test scores ranging from six to ten standard deviations above the baseline accuracy score indicating predictive power.

VI. References

Startup Success Prediction. (2020, September 16). Retrieved October 13, 2020, from https://www.kaggle.com/manishkc06/startup-success-prediction

https://github.com/akshay7424/Data1030_Project_Akshay.git