HARVARD COLLEGE
INDEPENDENT STUDY FINAL PROJECT

Predicting startup success: A simple approach using Crunchbase data

Supervisor:
David Parkes

Author:
Rodrigo Chaname

2021

1 Motivation

Across the different types of alternative investments, venture capital has gained traction and importance in the last 20 years. The reason behind this is that despite the enormous amounts of risk that these investments carry, successful ones tend to have astronomically high returns (Zider, 1998).

The high risk that these investments carry is the direct consequence of the nature of the venture capital market. Venture capitalists mainly target early-stage companies. Many of these firms are pre-revenue. In other words, they have yet gone to market with their products and have yet to see any cash inflows. Some other firms are in a pre-profit stage. These firms, usually larger in size, sacrifice profits to achieve faster growth. Because of this, they do not have stable financial and accounting data that could be used to inform potential investment decisions. As a result, investors have relied on a decision-making process that evaluates firms on a qualitative basis.

Many funds of different sizes and across different sectors have started to adopt data analysis tools to improve the quality of their investment decisions. A great example is Hone Capital, a venture capital fund in Silicon Valley that is leveraging human and computer-produced insights to make investment decisions (Quarterly, 2017). Because of this, we have decided to explore the power of predictive analytics in venture capital investments in this paper. Our objective is to motivate more VC funds to start collecting internal data to leverage statistical techniques to improve their investment decision-making.

2 Qualitative decision-making model

We will start our analysis by exploring the traditional venture capital decision-making criteria. Understanding how investors decide will allow us to select the best parameters for our two models. Fried and Hirsch propose a qualitative model that outlines the investment process and identifies meaningful decision-making criteria. Such a model was made through a survey study of 18 venture capitalists from funds located in Silicon Valley, Cambridge, Boston, and the Southwest United States. The fund sizes ranged from $4mm to $400mm, and mainly invested in pre-seed, seed, series A, and Series B. Furthermore, the funds invested in a wide range of industries, including AI, software, microprocessors, education, food & beverage, and surgical products (Fried & Hisrich, 1994).

The survey results were used to build a preliminary decision-making model that was then reviewed by an industry panel of 5 newly selected VCs. Based on the feedback of the panel, the researchers made further modifications.

Then, Fried and Hirsch classified the criteria into three different groups. The first group is *concept*, which has the following sub-criteria. The first one is *high potential for earnings growth*. The researchers argued that high-earnings growth can come from rapidly growing markets, market-share acquisition, or significant cost-cutting. The second sub-criterion is having *product-market fit*. In other words, successful startups have business ideas that have already been validated by the market. The third sub-criterion is having *competitive advantage* which is achieved by entering a novel market or by having proprietary technology. The fourth one is having *reasonable capital requirements*. The researchers claim that high capital requirements can be detrimental to investors' future returns. Furthermore, a startup that has secured additional funding is more attractive to investors.

The second group is *management*. Investors prefer startups whose founders display *integrity* and *hard work*. They are expected to have an exceptional track record regarding previous job and education experience. Furthermore, founders associated with failed ventures are not penalized. In fact, investors prefer veteran founders with *previous early startup experiences* as long as they show the they learned from their previous failures. Finally, founders who are *realistic* and have a *clear vision* of the future of their company are also desirable for venture capital investors (Fried & Hisrich, 1994).

The last group is *returns*. Investors are interested in companies that provide a *clear exit opportunity*, either through an initial public offering, acquisition, or share buyback. Venture capitalists also prefer both *large absolute returns* and *high rates of return*. Investment opportunities that offer a high rate of return with a small absolute return are usually ignored (Fried & Hisrich, 1994).

While these criteria are very hard to measure quantitatively, they provide us with a good understanding of what drives venture capital investment decisions. We will be able to leverage this information during the feature selection process for our model.

3 Objective and assumptions

As stated in the motivation of the paper, our goal is to build machine learning models that can predict whether a particular company will be successful. We will draw inspiration from the work of Zbikowski who built three different models that predicted company success based on Crunchbase data (2021). Many of our decisions including feature selection, engineering, and hyper-parameter tuning are heavily influenced by his paper. The following are the assumptions we will make for our model.

Location

Unlike Zbikowski's models that predicted success regardless of location, we chose to only look at companies founded in California, US. We wanted to abstract away from the influence of location and focus on more important predictors as described by the qualitative model we introduced in the previous section. Furthermore, as California is the most preeminent startup hub in the country, we will have plenty of observations to feed our models.

Timeframe

In a similar fashion to Zbikowski's work, we decided to include companies founded between 1995 and 2015. The reason is that the startup ecosystem in the US has been driven by advances in technology and science that date back to the 90s. Companies founded after 1995 have leveraged internet, software, and technology to disrupt industries. Secondly, we can see that startups founded after 2015 are too young. Zbikowski performed an analysis on the webpages of companies. He found that younger companies, with less than 5 years since founded, have the highest percentage of inactive webpages while having an operating status in the Crunchbase dataset. Because of this, it is hard to assign a good truth value to them (2021).

Target variable

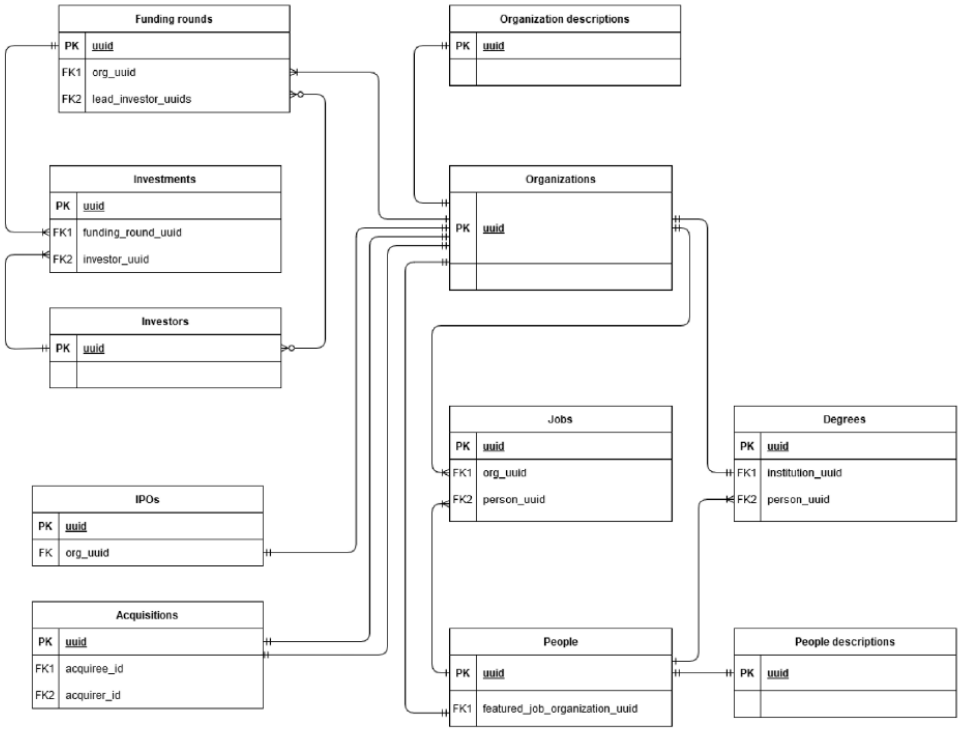
We also adopted the definition of success that Zbikowski proposed (2021). We say that any successful companies were either acquired, IPOed, or raised series B and are currently operating. We can express this mathematically as follows.

$$\text{target} = \begin{cases} 1, \text{ if} & (\text{series\_b} = 1 \wedge \text{status} = \text{'operating'}) \\ & \vee \ \text{status} = \text{'acquired'} \ \vee \ \text{status} = \text{'ipo'} \\ 0, \text{else} & \end{cases}$$

4 Data wrangling

We sourced our data from Crunchbase (www.crunchbase.com). According to its website, Crunchbase is a platform for finding information about public and private companies. We applied and ultimately gained access to its research offerings. We downloaded the data on October 28th, 2021.

These tables form a relational database, as the tables can be queried together using SQL-like language. The following is a simplified ERM diagram of the database from Zbikowski's work (2021).



The tables we extracted from Crunchbase were relatively clean and well organized. However, not all of them will contain information that would help us predict startup success. Because of this, we performed an exploratory data analysis to accomplish the following three goals. The first one was to determine which tables contain information relevant. The second one was to clean each selected table. Lastly, the third one was to merge the cleaned tables into a final dataset that can be fed to our machine learning models.

4.1 Table selection

We briefly explored each of our 19 tables using Python's Pandas library. The results and the rationale behind our decisions can be found in the table below.

Tables	Status	Selection rationale
acquisitions.csv	Not selected	Information included in organizations.csv
category_groups.csv	Not selected	Information included in organizations.csv
checksum.csv	Not selected	Irrelevant information
degrees.csv	Selected	Necessary information about people
events.csv	Not selected	Irrelevant information
event_appearances.csv	Not selected	Irrelevant information
funding_rounds.csv	Selected	Necessary information about investment rounds
funds.csv	Not selected	Irrelevant information
investment_partners.csv	Not selected	Irrelevant information
investments.csv	Not selected	Information included in funding_rounds.csv
investors.csv	Not selected	Information included in funding_rounds.csv
ipos.csv	Not selected	Information included in other organizations.csv
jobs.csv	Selected	Necessary information about people
organizations.csv	Selected	Necessary information about companies
org_parents.csv	Not selected	Irrelevant information
organization_descriptions.csv	Not selected	Text data
people_descriptions.csv	Not selected	Text data
people.csv	Selected	Necessary information about people

4.2 Table cleaning

Organizations table

One of our most important tables is organizations.csv. This table compiled all company information within Crunchbase, including startups, mature enterprises, and investment funds. First, we dropped all columns containing irrelevant information, from social media URLs to the Crunchbase rank. Then, we filtered out all organizations and kept only startups founded between 1995 and 2015 in California. Lastly, we exported this table. Its final columns can be found in the table below.

Final columns of organizations table	Description
uuid	Unique organization identifier
status	Company status (IPO, acquired, etc.)
founded_on	Date the company was founded

Jobs table

Jobs.csv is another important table in our dataset. It contains information about current and previous employment for each person registered in the database. Again, we started by dropping all columns containing irrelevant information, from social media URLs to the Crunchbase rank. From the qualitative model we introduced before, we know that venture capital investors pay considerable attention to a startup's founding team and current CEO when making investment decisions. Therefore, we decided to filter out all positions except those of CEO, founder, and co-founder.

Degrees table

The degrees.csv table allow us to see the attempted and earned degrees of each person registered in our database. We started by dropping all the irrelevant columns. From our qualitative model, we know investors value founders who have strong professional and academic backgrounds. Advanced degrees like MBAs and PhDs are usually signals of the academic capability of a person. Because of this, we then proceeded to collapse our table to see how many degrees each person had attempted and attained. We also kept the respective degree completion dates, if any.

People table

The table people.csv is another one of the pivotal tables as it allows us to connect people with their past and current jobs, the organizations where they have worked, and the degrees they hold. We began by dropping all the irrelevant columns. Then, we decided to *left-join* people.csv with degrees.csv and jobs.csv on each person's unique identifier. Mergin the tables allowed us to bring information together and We leveraged Pandas to add new columns. Some of these are whether a person was a veteran founder, the date of their first founded venture, etc. We exported this table. Its final columns can be found below.

Final columns of people table	**Description**
person_uuid	Unique person identifier
org_uuid	Unique organization identifier
is_founder	True if person is the one of the company's founder
is_current_ceo	True if person is current company's CEO
number_founded	Number of companies founded by person
first_venture_on	Date of the first venture founded by person
is_veteran_founder	True if person is a veteran founder
num_degs_attempted	Number of degrees attempted by person
num_degs_finished	Number of degrees finished by person
first_deg_completed_date	Date of completion of first finished degree

Funding rounds table

The funding_rounds.csv table includes information regarding the funding that each organization in the database received, from angel to series F investments. Just like with the other tables, we proceeded to drop all the columns with irrelevant information. Then, we filtered out all the investments that happened beyond series B, which as we claimed before, is one of our success targets. We exported this table. Its final columns can be found in the table below.

Final columns of rounds table	**Description**
org_uuid	Unique organization identifier
average_investor_count	Average number of investors across funding rounds
investment_type_angel	True if received angel investment
investment_type_seed	True if received seed investment
investment_type_series_a	True if received Series A investment
investment_type_series_b	True if received Series B investment

4.2.1 Final dataset

In the previous section, we exported three cleaned tables. Now, we will use them to construct our final dataset. We used Pandas to merge them on the unique organization identifier *org_uuid* . We made sure to do a *outter left-join* so to avoid dropping the observations that have missing values accross our tables.

Categorical data binning

There were originally 47 category groups a company could belong to in the *category groups* predictor. If we were to hot-encode this column, we would end up with too many dummy variables. This large number of predictors would be undesirable for our study, as most of them will have very little predictive power. Because of this, we followed Zibkowski's approach in his Crunchbase data analysis (2021). We decided to label the categories with below-median observations as *other*. The median was 2804 observations, and we ended with a total of 25 categories.

Data imputation

The following is a table with all our features, the percentage of missing observations.

Predictors	Count	% of missing values
category_groups_list	67469	0.01
success	67469	0.00
years_since_found	67469	0.00
num_degs_attempted_by_curr_ceo	67469	0.93
num_degs_finished_by_curr_ceo	67469	0.93
max_number_founded_by_one_founder	67469	0.82
avg_num_degs_attempted_by_founders	67469	0.82
avg_num_degs_finished_by_founders	67469	0.82
at_least_one_veteran_founder	67469	0.82
years_between_degree_founding	67469	0.88
years_between_first_curr_founding	67469	0.84

It is not surprising to see that many of our variables have large numbers of missing observations. Based on how we built our final dataset and how we believe Crunchbase collects data, we made several imputation assumptions. When we initially picked our predictors, we did it with the intention of using them as proxies for the criteria that venture capitalists look for when making an investment decision. These predictors can be seen as boxes to be checked. For imputation purposes, we are assuming that *NaN* observations signal that a particular company does not check the box that specific predictor represents. The following is a table detailing all our imputation decisions and the specific values we used.

Predictors	Imputation assumption	Imputation value
num_degs_attempted_by_curr_ceo	*NaN* means no degrees attempted	0
num_degs_finished_by_curr_ceo	*NaN* means no degrees finished	0
max_number_founded_by_one_founder	*NaN* means no previously-founded ventures	0
avg_num_degs_attempted_by_founders	*NaN* means no degrees attempted	0
avg_num_degs_finished_by_founders	*NaN* means no degrees finished	0
at_least_one_veteran_founder	*NaN* means no veteran founders	0
years_between_degree_founding	*NaN* means no degree finished	0
years_between_first_curr_founding	*NaN* means no previously-founded ventures	0

4.3 Feature engineering

The next step in our process is to select the features we will input into our model. To do so, we will take a look at different plots to choose our predictors. Before that, we have to make sure that all our variables are correctly encoded. Our dataset is mainly composed of continuous and categorical variables. In the following table, we can see the type of each predictor and the encoding we used, if any.

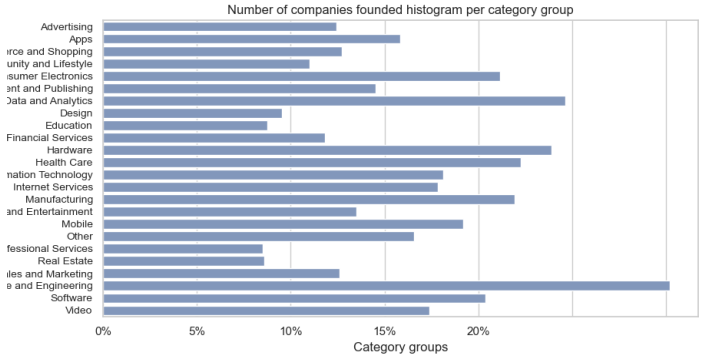
Predictors	Variable type	Encoding type
category_groups_list	Categorical nominal	One-hot
success	Categorical nominal	One-hot
years_since_found	Discrete	None
num_degs_attempted_by_curr_ceo	Discrete	None
num_degs_finished_by_curr_ceo	Discrete	None
max_number_founded_by_one_founder	Discrete	None
avg_num_degs_attempted_by_founders	Continuous	None
avg_num_degs_finished_by_founders	Continuous	None
at_least_one_veteran_founder	Categorical nominal	One-hot
years_between_degree_founding	Discrete	None
years_between_first_curr_founding	Discrete	None

4.4 Feature selection

In this section, we will rely on our earlier qualitative decision-making model and the results of our data analysis to choose the most important predictors.

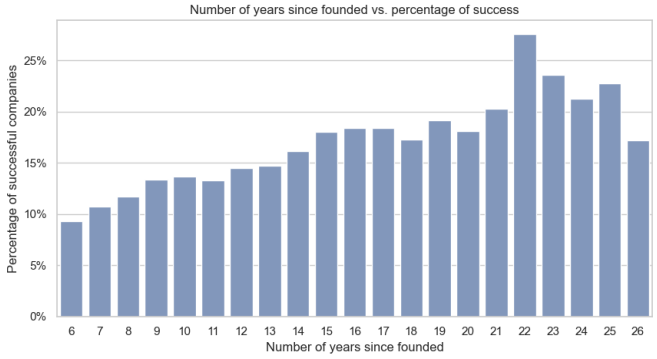
Category groups

In the plot below, we can see the percentage of success as a function of the company's industry vertical. We notice that there are categories where the success is higher such as *Science and Engineering*, *Hardware*, and *Analytics*. Similarly, we have categories with smaller success rates including *Education*, *Design*, and *Professional Services*. These results are consistent with our qualitative model from the first section. Venture capitalists want to invest in industries with high potential for growth which are often related to technology and science. Because of this, we will include this variable in our final dataset.



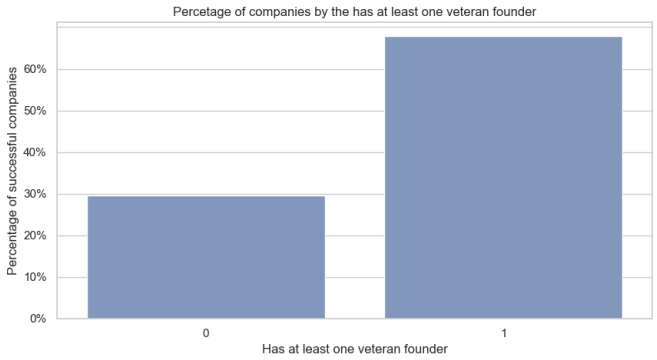
Number of years since founded

Below, we can see the percentage of success as a function of the company's years since its founding. This plot shows us that younger companies are much more likely to fail. This can be explained by the fact that relatively older companies have built stronger teams, found clients, and have product-market fit. We can confidently use this predictor as a proxy for company experience and product market-fit, which are two criteria that venture capitalists use when making investment decisions. Because of that, we will include this information in our model.



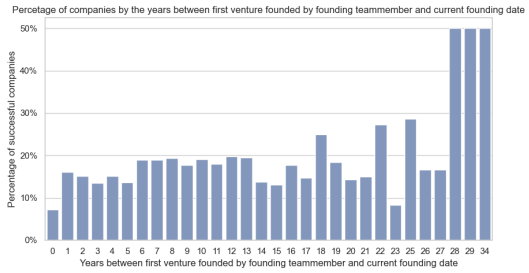
At least one veteran founder

In this plot, we see the impact on success of having at least one veteran co-founder. As we described in our qualitative model, veteran founders are desirable for investors regardless of the success of their previous ventures. Furthermore, previous founder experience is synonym with managerial experience, good character, and company vision. Because of this, we have decided to include this predictor in our final dataset.

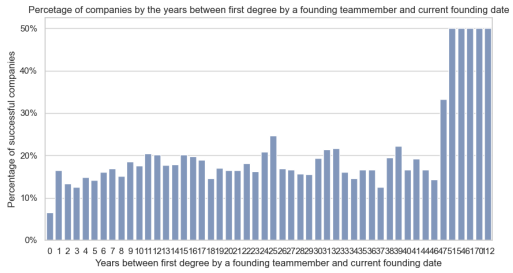


Years between degree or first founding, and current company founding

The leftmost plot displays startup success as a function of the years between the earliest team degree completion date and the founding date. The rightmost plot startup company success as a function of the years between the earliest company founding by a team-member and the current founding. In both, we see positive correlation between the number of years and company success. The more years, the more time the founders have spent honing their skills. From our qualitative model, we know investors prefer founding teams with strong managerial experience. Because of this, we will include both predictors in our dataset as the former signals overall team experience and the latter signals the managerial experience of a veteran founder.



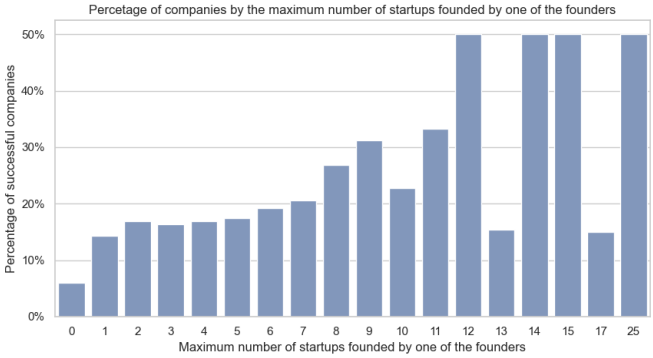
(a)



(b)

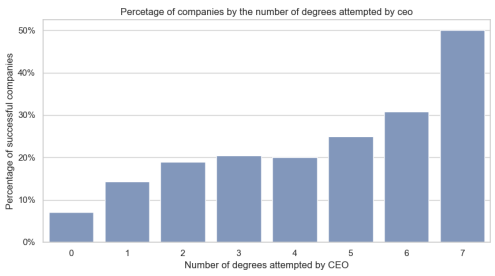
Maximum number of companies founded by one of the founders

This is a plot of success as a function of the maximum number of previous startups founded by a founding team member. We see a positive correlation between current company success and the number of previous ventures founded. We know that investors prefer veteran founders regardless of the success of their previous ventures. The more startups founded can also be a proxy of the experience of a veteran founder, and can complement some of the previous predictors. Because of this, we included this variable in our final dataset.

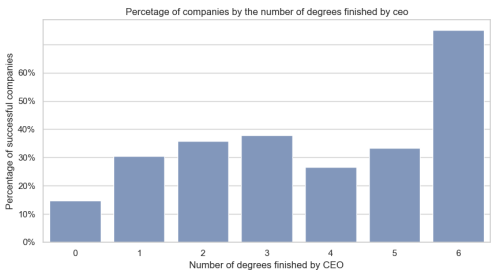


Number of degrees attempted vs. finished by current CEO

The leftmost plot displays startup success as a function of the average number of degrees attempted by the founding team of the company. The rightmost one displays startup success as a function of the average number of degrees finished by the founding team. As described by the qualitative model, investors prefer companies whose team members are experienced and have an ambitious vision. Both the number of degrees attempted and finished are excellent proxies for the experience and capabilities of the founding team. However, we believe that attempted degrees send a weaker signal compared to finished degrees. Because of this, we will pick the latter metric.



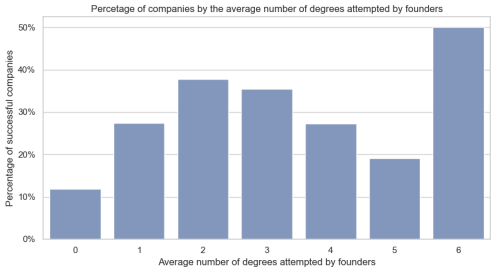
(a)



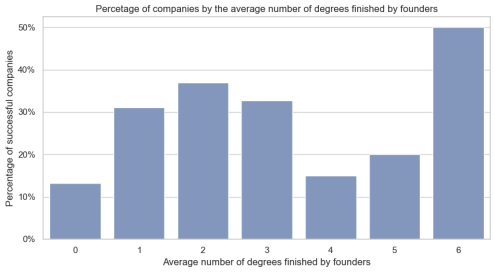
(b)

Average number of degrees attempted vs. finished by founders

The leftmost plot displays success as a function of the degrees attempted by the current CEO. The rightmost one displays success as a function of the degrees finished by the current CEO. As described by the qualitative model, investors prefer to invest in companies with an experienced chief executive. Both the number of degrees attempted and finished are proxies for the capabilities of a CEO. It is plausible to believe that those with advanced degrees are more experienced than those with 1 or 0 degrees. This is what we see in the two plots. We believe that attempted degrees sends weaker signal compared to finished degrees. Because of this, we will pick the latter metric.



(a)



(b)

4.5 Final dataset

Our finalized dataset has a total of 67334 observations and 31 predictors, including our multiple dummy variables. Out of the total number of observations, 12.54% correspond to successful companies while 87.46% correspond to failed companies. We noticed a major class imbalance which is a feature of the question we are answering. Seeing a successful company is a rare event, as most of the companies will inevitably fail (Fried & Hisrich, 1994). Despite the class imbalance, we chose not to take any steps to fix it and we relied on accuracy, precision, and F1 scores to assess the performance of our models (Zbikowski & Antosiuk, 2021).

4.6 Train-test split

Following convention, we decided to do an 80-20 train-test split. Furthermore, we decided to sample our observations in a stratified fashion to ensure we have enough success observations in both our training and testing dataset. We will also use the training dataset as both training and validation sets through the use of cross-validation process.

4.7 Models

We decided to use three different models. These are logistic regression, XGBoost, and a naive majority classifier. Gradient boosting algorithms that use decision trees as base estimators have gained popularity in the last few years, being the preferred algorithm in data science competitions worldwide. Logistic regression has been consistently used in previous work involving Crunchbase data (Zbikowski & Antosiuk, 2021). Finally, we will use the naive majority classifier as a baseline to measure the performance of our models.

4.8 Data transformation

We decided to scale our data for our logistic regression model. The reason behind this decision is that our model will use regularization, and thus, should have scaled data. In this particular opportunity, we decided to standardize our data due to the presence of outliers shown previous section plots. For our majority classifier and XGboost model, we did not scale the data. These two models are perfectly capable of handling data regardless of the units used. Our decisions regarding scaling are also very similar to those made by Zbikowski (2021).

4.9 Initial results

Following Zbikowski's work, the results shown below were obtained by running a 10-fold cross-validation on the full training set and averaging each score. The logistic regression model was initialized with no regularization penalty, with a *saga* solver, and with 1000000 iterations. The rest of the parameters were the default ones from the Sci-kit Learn package. In a similar fashion, the XGboost model was initialized with an *error* test metric. All the other parameters were the default ones from the XGboost package.

Classification scores	**Majority classifier**	**Logistic cegression**	**XGBoost**
Accuracy	0.8746	0.8738	0.8722
Precision	0	0.3554	0.3721
Recall	0	0.0078	0.0275
F1	0	0.0153	0.0512

We can see that our logistic regression and XGboost models achieved a classification accuracy of roughly 87%. The bench-marking majority classifier obtained the same *accuracy*. The high *accuracy* of our models is a consequence of the class imbalance of our dataset. To understand how well our models performed, we have to take a look at other classification metrics. We notice that our two models have very low *precision* scores. The logistic regression *precision* was slightly less than that of the XGboost model. While these scores are clearly better than the bench-mark, they are still very low. We see similar results when we look at the *recall* metric. The XGboost model outperformed the logistic regression one, but the two scores are extremely low. This is very concerning as this means that our models incorrectly classify successful companies as unsuccessful.

When we compare the performance of our models against those of Zbikowski, we can see major differences. Those models achieve *precision* scores in the low 80% and *recall* scores in the low 30%. We have two hypotheses that explain the underperformance of our models against Zbikowski's. The former is that we have chosen unideal hyper-parameters that diminish the predictive power of our models. If this were true, we should see major improvements if we tuned our model. The latter is that these results are a consequence of our initial assumptions. Zbikowski chose to include geographical location as a predictor while we chose to only look at companies in California. In fact, the two most important features in their work are *country code* and *region* (2021).

4.10 Hyper-parameter tunning

We will now tune our hyper-parameters to improve the performance of our model. We did a 5-fold cross-validation grid-search to find the best parameter for our logistic regression and XGboost models using the *GridSearchCV* function from *Sci-kit Learn*.

Logistic regression

Following Zbikowski's work, we did a grid-search on the following parameters. We will refit our *GridSearchCV* with the *F1* metric. Furthermore, we kept using the *saga* solver, and 1000000 maximum iterations.

Parameters	**Alternatives**
Regularization penalty	L1, L2
Inverse of reg. strength (C)	0.01, 0.1, 0.5, 10, 50, 100

The *GridSearchCV* algorithm found the best parameters: $C = 10, \text{Penalty} = \text{L1}$.

XGboost

Unlike the previous example, we manually tuned the hyper-parameters of our XGBoost classifier. We performed a 10-fold cross-validation for each of the parameter combinations described below. We will pick the combination that obtained the best cross-validation *F1* score.

Gamma	**Learning rate**	**Max depth**	**Min child weight**	**N estimators**	**Cross-val F1 score**
0	1	10	1	100	0.1343
0	1	10	1	100	0.1343
0.5	1.5	15	1	100	0.1336
0	1	10	1	200	0.1336
0.5	1.5	15	1	200	0.1336
0.	1	10	1	300	0.1362
0.5	1.5	15	1	300	0.1336
0	1	10	1	400	0.1397
0.5	1.5	15	1	400	0.1336
0	1	10	1	500	0.1373
0.5	1.5	15	1	500	0.1336

The resulting best parameters are: Gamma $= 0$, learning rate $= 1$, max depth $= 10$, min child weight $= 1$, n estimators $= 400$.

4.11 Optimized results

Our hyper-parameter optimization obtained the following scores.

Classification scores	**Majority classifier**	**Logistic regression**	**XGBoost**
Accuracy	0.8746	0.8736	0.8532
Precision	0	0.3725	0.2500
Recall	0	0.0112	0.0853
F1	0	0.0218	0.1271

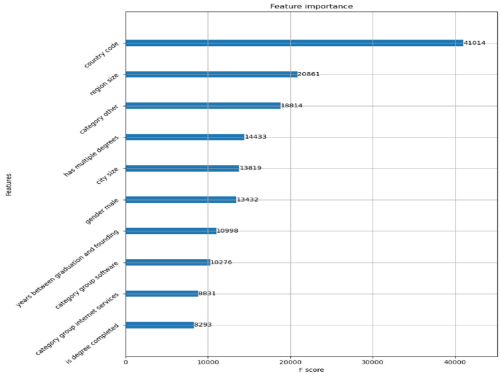
We can see that there were improvements in our two models across all classification metrics. For our logistic regression, the *accuracy* of the model remained roughly the same at 87%. We also saw an increase in the *precision* metric, which went up 2 percent units to 37%. Similarly, there is an increase in our *recall* metric to 1.1%. Lastly, our *F1* metric doubled to 2%. For our XGboost model, the accuracy metric did not change and remains at 86%. We saw a substantial 10% decrease in the *precision*, which is now 25%. However, we saw a larger increase in our *recall* metric, which is now 8%. Lastly, the *F1* metric substantially increased to 12%.

While we see some increases, our two models still do a poor job at classifying success. We can notice this when compare our models metrics to those of Zbikowski's models, shown below. In the last section, we proposed two hypothesis to explain these results. The former one was that performing hyper-parameter tuning would increase the performance, making them par with Zbikowski's models. The latter hypothesis is that the features we chose not to include, *country code* and *region*, are the ones with most predictive power. We disproved the former one and we tested the later in the following section.

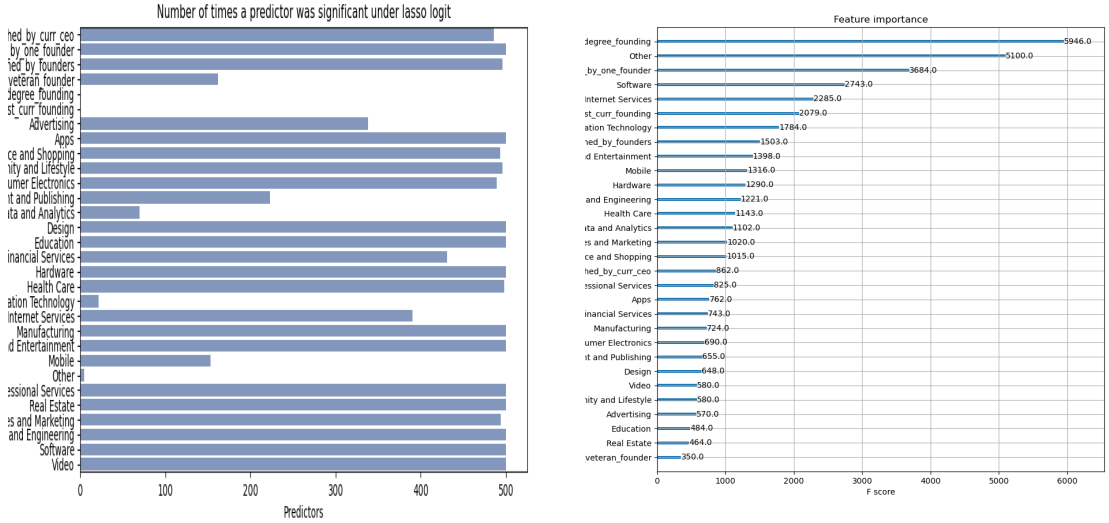
Classification scores	**Logistic regression**	**XGBoost**
Accuracy	0.86	0.85
Precision	0.67	0.57
Recall	0.21	0.34
F1	0.32	0.43

4.12 Feature importance

The following is a feature importance plot for Zbikowski's XGboost model (2021). As we hypothesized, the features with the most predictive power are *country code* and *region size*. Since we chose a simpler approach of predicting California-based startups, our models lacked the most important features. This partly explains the poor performance of our model.



The following is a feature importance plot for our two models. For the logistic regression one, we ran 500 bootstrapped lasso regressions and counted which features where the most predictive in each. For the XGBoost model, we leveraged the python package to obtain the most important features.



(a) Logistic regression

(b) XGBoost

5 Conclusion

In this paper, we see the predictive power of very simple machine learning models using Crunchbase data. While we hoped that our models had very strong performances across several classification metrics, previous work has shown that more complex models have more potential to predict startup success. For example, Yuxian and Yuan built a model that aims to predict whether an investor is likely to invest in a company based on their social relationship with them. This approach, which models social relationships through networks, is consistent with our initial qualitative model and would be an interesting feature to include in more complex prediction approaches (2013). Additionally, Dixon and Chong took a different approach that leveraged macroeconomic factors to make predictions (2014). We hope that this paper encourages venture funds across the world to start building data science and analytics teams, so that ultimately they can leverage the insights produced by machine learning models to better inform their investment decisions better.

References

Dixon, M., & Chong, J. (2014). A bayesian approach to ranking private companies based on predictive indicators. *Ai communications*, *27*(2), 173–188.

Fried, V. H., & Hisrich, R. D. (1994). Toward a model of venture capital investment decision making. *Financial management*, *23*(3), 28–37.

Quarterly, M. (2017, June). *A machine-learning approach to venture capital.* Author. Retrieved from `https://hbr.org/1998/11/how-venture-capital-works`

Yuxian, E. L., & Yuan, S.-T. D. (2013). Investors are social animals: Predicting investor behavior using social network features via supervised learning approach.

Zbikowski, K., & Antosiuk, P. (2021). A machine learning, bias-free approach for predicting business success using crunchbase data. *Information processing management*, *58*(4), 102555.

Zider, B. (1998, Aug). *How venture capital works.* Harvard Business Review. Retrieved from `https://hbr.org/1998/11/how-venture-capital-works`