Predicting Business Success with Machine Learning

Western Governors University

Mohammad Iqbal

Research Question

In this paper we will discuss business success and what are factors associated with a successful business using the Crunchbase dataset. According to the Small Business Administration, half of all businesses fail within the first 5 years (sba.gov, 2020).This is a huge amount of wasted time and effort which can potentially be put towards more productive aims. This study will use Logistic Regression and Random Forest to attempt to make a predictive model in order to assess the viability of using these methods to accurately predict business success.

This project will use Logistic Regression and Random Forest classifiers to determine what factors are associated with successful businesses. Feature importance of the predictor variables can be used to determine what factors are most associated with the target variable. (Brownlee, 2020).

The null hypothesis is that there is no statistical difference between successful and unsuccessful companies given our dataset. The alternate hypothesis is that there is a statistical difference between unsuccessful and successful companies.

The dataset to be used is the publicly available dataset provided on the crunchbase.com website through their export csv functionality. There are a few limitations of the dataset. It does quantify relevant metrics to company success, however profitability and growth, 2 key metrics of any successful business, are absent (Investopedia, 2020). Also notably missing and impossible to quantify are personal characteristics of the founders that are important to success such as passion and determination (Gerber, 2013). With all that said, we do still have a significant amount of

information about the companies and it will give us a good idea of what characteristics are associated with successful companies.

A similar analysis on the same dataset was done by Data Analyst Allison Glazer (Glazer, 2019). We can use her analysis as a starting point to see how we would approach this problem.

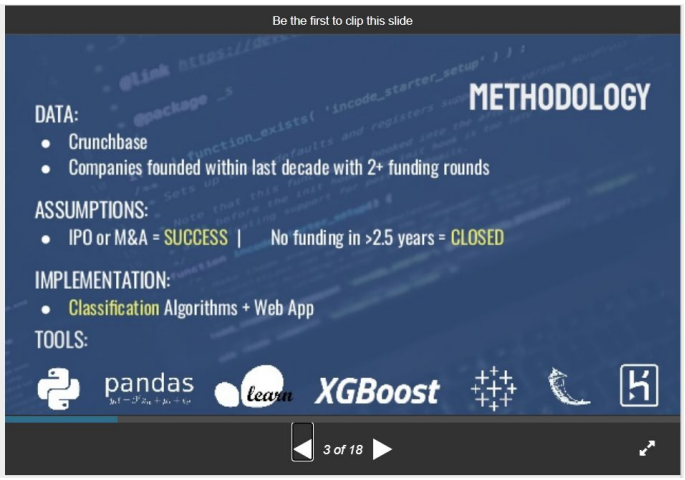
I feel it is important to discuss this article and this project since it has gained considerable popularity and is one of the top Google search results for its search terms. It will be seen by novices in the field and the general public as a sound analysis, which I would argue it is not.

To start, she too was studying what factors were associated with business success using the same dataset. Her conditions for success for the target variable were that the company raised 2 rounds of funds in the last 2.5 years and was either IPO or in M&A. There are a couple of problems with this, the first being that 2 rounds of funding is completely arbitrary, and there is no justification given for formulating this assumption. It is also not clear what “2 rounds” is referring to and if she is counting seed as a round or not. For example based on her criteria a company that raised a pre-seed and a seed round in the last 2 years would be a success. A company that raises a pre-seed and a seed on average has a funding of approximately 6 million dollars (Loizos, 2019). On the other hand maybe a company skips both pre-seed and seed, and simply goes for a Series A as their first investment, and doesn’t choose to raise funding for the next 3 years. A Series A company can expect to have an average funding of approziamtely 16 million dollars (Loizos, 2019), but this company would be a failure based on her analysis. As you can see, this is not a good way to define the target variable since a

company with more funding can be seen as a failure and a company with less funding can be seen as a success.

She also claimed that if a company did not receive funding in the last 2.5 years, it was closed. This is not only incorrect but simply does not even make sense. Businesses do not simply shut down because they have not received equity funding for 2 1/2 years. They can sustain themselves on their own cash indefinitely, provided they cover their expenses.

Another problem with her analysis was that a company in M&A is a success. She stated that if a company is acquired that it was a success and this is true. An acquired company usually receives cash or stock options from the acquiring company that can be traded for cash and this is generally considered to be a successful startup (Hayes, 2020). However, in this particular dataset and in general, M&A funding status refers both to companies that have been acquired, have acquired other companies or both (Hayes, 2020). She did not seem to realize this in her analysis. This will most likely lead to an inaccurate model since this particular “M&A” value in the Funding Status feature included both companies that were acquired and companies that were acquiring other companies.



Source:

https://www.slideshare.net/AlisonGlazer/metis-project-3-predicting-startup-success

She is not completely wrong however Funding is a good way to judge startup success. She used the Funding Status column as her target variable, when Last Equity Funding type is a better option since it can describe the Funding in more detail.

According to Y-combinator, a startup accelerator which has invested in over 2000 companies, raising a seed or angel round is usually done pre-market fit, which means before there is a valid business model for the company, and is more about founder and product potential. Series A round in contrast is to grow your business after having found a valid business model (Seibel, 2018). This means that when a company raises a Series A they have a valid business model and therefore a valid business. We can use this as our criteria of success and definition of our target variable. We can say a company is successful if it has at least raised a Series A funding round. Our Last Funding Type variable only shows the last funding round which may or not be Series A,

but all later funding rounds after Series A, such as Series B and Series C, will have to be included since they will have by definition at least raised a Series A. The variable will be called “Has Reached Series A” and will be a binary of True if the company reached Series A and False if the company did not. The code to do this in Python is provided below.

```
#Create Target Variable
targetValues = ["Series A", "Series B", "Series C", "Series D", "Series E", "Series F",
                "Series G",  "Series H", "Series I", "Series J", ]

dfMain["Has Reached Series A"]=dfMain["Last Equity Funding Type"].isin(targetValues)
```

Data Collection

We will use the dataset provided by Crunchbase. It is one of the largest datasets on businesses that exists.

The data itself comes to Crunchbase through 4 different ways. The first being their extensive network of 3,500 global investment firms that submit portfolio updates to Crunchbase every month. The investment firms are presumably incentivized to make their portfolio updates public in order to attract more investors to their funds. Next is community contributions to the dataset. Crunchbase has a large community of entrepreneurs and executives so this lends credibility to the community contributions. Next is the AI and Machine learning algorithms which they claim scans the data for accuracy and detects anomalies. Finally is the in house data science team which builds and maintains the algorithms (Crunchbase | Knowledge Center, 2020).

The data can be found after signing up to the Crunchbase website and using their “Export CSV” functionality. Unfortunately this functionality limits us to 1000 rows of data at a time. This is not a problem. We will simply run this export functionality 130 times to get a substantial amount of data that we will need for our analysis. This will produce 130 different csv files. To work with this dataset we can simply combine all these separate files into one as shown in the code below.

Extract Data

```
2]: path = "C:/Users/iqbal/Desktop/Code/crunchbase test/Crunchbase Data"
    all_files = glob.glob(os.path.join(path, "*.csv"))

    data = pd.concat([pd.read_csv(f) for f in all_files ])

    data.to_csv("combined_csv.csv", index=False)
```

We can also combine it into one csv file for future use. Now our combined dataset is available in the **data** variable.

Our Dataset contains 153256 rows and 81 columns. The number of rows will surely be reduced in our data cleaning step as we remove rows with missing data. The dataset is 83 Mb in size.

We can see basic info about the dataset by running the command **data.info()**, which will give us all the column names, the number of non-missing values and the data type. The output is given on the following pages.

```
]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 153256 entries, 0 to 999
Data columns (total 81 columns):
 #   Column                                Non-Null Count   Dtype
---  ------                                --------------   -----
 0   Organization Name                     153255 non-null  object
 1   Organization Name URL                 153256 non-null  object
 2   Industries                            150875 non-null  object
 3   Headquarters Location                 140502 non-null  object
 4   CB Rank (Company)                     153256 non-null  object
 5   SEMrush - Monthly Visits              59505 non-null   object
 6   SEMrush - Average Visits (6 months)   52198 non-null   object
 7   SEMrush - Monthly Visits Growth       55562 non-null   object
 8   SEMrush - Visit Duration              59505 non-null   object
 9   SEMrush - Page Views / Visit          59505 non-null   float64
 10  Operating Status                      153256 non-null  object
 11  Number of Articles                    76246 non-null   object
 12  Headquarters Regions                  119810 non-null  object
 13  Estimated Revenue Range               76194 non-null   object
 14  Founded Date                          144468 non-null  object
 15  Founded Date Precision                144468 non-null  object
 16  Exit Date                             21397 non-null   object
 17  Exit Date Precision                   21397 non-null   object
 18  Closed Date                           12575 non-null   object
 19  Closed Date Precision                 35860 non-null   object
 20  Company Type                          151016 non-null  object
```

21	Investment Stage	779 non-null	object
22	Industry Groups	150872 non-null	object
23	Number of Founders	95572 non-null	float64
24	Founders	95572 non-null	object
25	Number of Employees	140746 non-null	object
26	Number of Funding Rounds	115715 non-null	float64
27	Funding Status	88785 non-null	object
28	Last Funding Date	115710 non-null	object
29	Last Funding Amount	100986 non-null	float64
30	Last Funding Amount Currency	100992 non-null	object
31	Last Funding Amount Currency (in USD)	100986 non-null	float64
32	Last Funding Type	115710 non-null	object
33	Last Equity Funding Amount	92736 non-null	float64
34	Last Equity Funding Amount Currency	92741 non-null	object
35	Last Equity Funding Amount Currency (in USD)	92736 non-null	float64
36	Last Equity Funding Type	107005 non-null	object
37	Total Equity Funding Amount	100129 non-null	float64
38	Total Equity Funding Amount Currency	100132 non-null	object
39	Total Equity Funding Amount Currency (in USD)	100128 non-null	float64
40	Total Funding Amount	109766 non-null	float64
41	Total Funding Amount Currency	109770 non-null	object
42	Total Funding Amount Currency (in USD)	109765 non-null	float64
43	Top 5 Investors	88252 non-null	object
44	Number of Lead Investors	56448 non-null	float64
45	Number of Investors	88254 non-null	float64
46	Number of Acquisitions	13179 non-null	float64
47	Acquisition Status	24247 non-null	object
48	IPO Status	153256 non-null	object

49	IPO Date	8171 non-null	object
50	Delisted Date	850 non-null	object
51	Delisted Date Precision	1175 non-null	object
52	Money Raised at IPO	2348 non-null	float64
53	Money Raised at IPO Currency	2349 non-null	object
54	Money Raised at IPO Currency (in USD)	2348 non-null	float64
55	Valuation at IPO	1420 non-null	float64
56	Valuation at IPO Currency	1420 non-null	object
57	Valuation at IPO Currency (in USD)	1420 non-null	float64
58	Stock Symbol	8164 non-null	object
59	Stock Symbol URL	8172 non-null	object
60	Stock Exchange	8162 non-null	object
61	IPqwery - Patents Granted	47243 non-null	object
62	IPqwery - Trademarks Registered	47243 non-null	object
63	IPqwery - Most Popular Patent Class	21930 non-null	object
64	IPqwery - Most Popular Trademark Class	40962 non-null	object
65	Investor Type	1378 non-null	object
66	Accelerator Program Type	65 non-null	object
67	Accelerator Duration (in weeks)	51 non-null	object
68	Acquisition Type	13242 non-null	object
69	SEMrush - Visit Duration Growth	43969 non-null	object
70	SEMrush - Page Views / Visit Growth	55562 non-null	object
71	SEMrush - Bounce Rate	59505 non-null	object
72	SEMrush - Bounce Rate Growth	50848 non-null	object

```
 73  SEMrush - Global Traffic Rank                59505 non-null   object
 74  SEMrush - Monthly Rank Change (#)            55562 non-null   object
 75  SEMrush - Monthly Rank Growth                55562 non-null   object
 76  BuiltWith - Active Tech Count                138310 non-null  float64
 77  G2 Stack - Total Products Active             76114 non-null   object
 78  Aberdeen - IT Spend                          21107 non-null   float64
 79  Aberdeen - IT Spend Currency                 21692 non-null   object
 80  Aberdeen - IT Spend Currency (in USD)        21107 non-null   float64
dtypes: float64(21), object(60)
memory usage: 95.9+ MB
```

Removing Unnecessary Features

The first thing we will need to do is remove unnecessary columns. In this early in the data processing stage we will be careful about which ones we remove. To ensure this, we will only remove columns that are obviously not necessary for data analysis. Later on we will incorporate more advanced dimensionality reduction techniques, but for now we will simply use basic data analysis knowledge to manually remove the columns. Certain columns are grouped together to avoid repetitive explanations. Basic explanation of why these features were removed is given.

Organization Name, Origination URL - This data is completely irrelevant for our analysis.

Industries, headquarters - We will instead use Headquarters Regions and Industry Groups and remove these to avoid redundancy.

CB Rank - This is a primary key crunchbase uses to structure its data, but is not useful for our analysis.

Founded Date Precision, Exited Date, Exited Date Precision, Closed Date, Closed date precision - These variables are missing far too many rows and thus are not suitable for out analysis

Company type - Data exploration has shown that only 756 companies are non profit, while over 150,000 are for profit. Such an imbalanced class is likely to cause problems with our model and this data is also not relevant for what we are researching.

Investment Stage - only 779 populated rows. This is far too few.

Founders - Like the company name, the names of the individual founders is completely irrelevant for our analysis.

Funding Status - We will use Last Equity Funding type instead of this, so to avoid redundancy we will remove this variable.

Last Funding Amount, Last Funding Amount Currency, Last Equity Funding Amount, Last Equity Funding Amount Currency, Total Equity Funding Amount, Total Equity Funding Amount Currency, Total Funding Amount, Total Funding Amount Currency - We will use the USD currency version of each funding amount variable and remove these to avoid redundancy.

Top 5 Investors - This seems like it would be a good variable to include in our model, but basic data exploration has shown that there are over 66,000 unique values. This is far too many categories.

Number of Acquisitions, Acquisition Status - Too many missing rows

IPO Status - This variable is not relevant to what we are researching

IPO Date, Delisted Date, Delisted Date Precision, Money Raised at IPO, Money Raised at IPO Currency, Money Raised at IPO Currency (in USD), Valuation at IPO, Valuation at IPO Currency, Valuation at IPO Currency (in USD), Stock Symbol, Stock Symbol URL, Stock Exchange, - All these variables are missing too many rows, and are irrelevant pieces of data.

IPqwery - Patents Granted, IPqwery - Trademarks Registered, IPqwery - Most Popular Patent Class, IPqwery - Most Popular Trademark Class - Too many missing rows

Investor Type, Accelerator Program Type, Accelerator Duration (in weeks), Acquisition Type - Too many missing rows

G2 Stack - Total Products Active - This is very similar to the “Built With - Active Tech Count” variable and will be removed to avoid redundancy.

SEMrush - Visit Duration Growth, SEMrush - Page Views / Visit Growth, SEMrush - Bounce Rate, SEMrush - Bounce Rate Growth, SEMrush - Global Traffic Rank, SEMrush - Monthly Rank Change (#), SEMrush - Monthly Rank Growth - These are missing a lot of rows and are also overly specific to web traffic.

Aberdeen - IT Spend, Aberdeen - IT Spend Currency, Aberdeen - IT Spend Currency (in USD) - Too many missing rows

We can remove all unnecessary variables by simply including the ones we want in a Python Pandas Dataframe. The code to do this is given below.

```
dfMain = data[['Number of Articles', 'Headquarters Regions',
               'Estimated Revenue Range', 'Industry Groups', 'Number of Founders',
               'Number of Employees', 'Number of Funding Rounds',
               'Last Funding Amount Currency (in USD)',
               'Last Equity Funding Amount Currency (in USD)',
               'Last Equity Funding Type',
               'Total Equity Funding Amount Currency (in USD)',
               'Total Funding Amount Currency (in USD)', 'Number of Lead Investors',
               'Number of Investors', 'BuiltWith - Active Tech Count', "SEMrush - Monthly Visits",
               'SEMrush - Average Visits (6 months)', "SEMrush - Visit Duration", "SEMrush - Page Views / Visit",
              ]]
```

Pandas optionally allows dropping of unnecessary columns like so:

```
]: dfMain = data.drop(columns=["Organization Name", 'Last Funding Type', "Industries", "Operating Status", "Headquarters Location",
                               "Founded Date", "Organization Name URL", "CB Rank (Company)", "Founded Date Precision",
                               "Exit Date", "Exit Date Precision", "Closed Date", "Closed Date Precision", "Company Type",
                               "Investment Stage", "Founders", "Top 5 Investors", "Last Funding Amount Currency","Last Funding Date"
                              "Total Equity Funding Amount Currency", "Total Equity Funding Amount", "Total Funding Amount",
                               "Total Funding Amount Currency", "Number of Acquisitions", "Acquisition Status",
                               "IPO Status", "IPO Date", "Delisted Date", "Delisted Date Precision",
                               "Money Raised at IPO", "Money Raised at IPO Currency",
                               "Money Raised at IPO Currency (in USD)", "Valuation at IPO", "Valuation at IPO Currency",
                               "Valuation at IPO Currency (in USD)", "Stock Symbol", "Stock Symbol URL", "Stock Exchange",
                               "IPqwery - Patents Granted", "IPqwery - Trademarks Registered",
                               "IPqwery - Most Popular Patent Class", "IPqwery - Most Popular Trademark Class",
                               "Investor Type", "Accelerator Program Type", "Accelerator Duration (in weeks)" ,
                               "Acquisition Type", "G2 Stack - Total Products Active", "Aberdeen - IT Spend",
                               "Aberdeen - IT Spend Currency", "Aberdeen - IT Spend Currency (in USD)",
                               "SEMrush - Monthly Visits", "SEMrush - Average Visits (6 months)", "SEMrush - Monthly Visits Growth",
                               "SEMrush - Visit Duration", "SEMrush - Page Views / Visit", "SEMrush - Visit Duration Growth",
                               "SEMrush - Page Views / Visit Growth", "SEMrush - Bounce Rate", "SEMrush - Bounce Rate Growth",
                               "SEMrush - Global Traffic Rank", "SEMrush - Monthly Rank Change (#)", "SEMrush - Monthly Rank Growth"
                              ])
```

After removing these variables and dropping missing rows we get the following variables for use in our analysis.

Field	**Data**
Headquarters Regions	Categorical
Industry	Categorical
Estimated Revenue Range	Categorical
Number of Employees	Categorical
Number of Articles	Nominal
Number of Founders	Nominal
Number of Funding Rounds	Nominal
Last Funding Amount Currency (in USD)	Continuous
Last Equity Funding Amount Currency (in USD)	Continuous
Total Equity Funding Amount Currency (in USD)	Continuous
Total Funding Amount Currency (in USD)	Continuous
Number of Lead Investors	Nominal

Number of Investors	Nominal
BuiltWith - Active Tech Count	Nominal
SEMrush - Monthly Visits	Nominal
SEMrush - Average Visits (6 months)	Continuous
SEMrush - Visit Duration	Nominal
SEMrush - Page Views / Visit	Continuous

Data Extraction and Preparation

Now that we have done basic data collection we can now begin to extract and prepare for our analysis.

We can now start exploring some of our data. We can do this by first using the .value_counts() function to see the values for our “Estimated Revenue Range” variable.

```
dfMain["Estimated Revenue Range"].value_counts()
```

```
$1M to $10M       5084
$10M to $50M      2302
Less than $1M     2204
$100M to $500M     663
$50M to $100M      568
$1B to $10B        142
$500M to $1B       135
$10B+               33
Name: Estimated Revenue Range, dtype: int64
```

We see that 33 companies have an estimated revenue range of over 10 billion dollars. We can then use the following code to take a closer look at who these companies are.

```
BigCompanies = data.loc[data["Estimated Revenue Range"] == "$10B+"]
```

```
BigCompanies
```

Not surprisingly we see giant conglomerates which are household names such as Facebook and the Alibaba group.

			Northeaster...		Goods	
4	Facebook	290,690	San Francisco Bay Area, Silicon Valley, West C...	$10B+	Apps, Commerce and Shopping, Community and Lif...	6.0
5	Alibaba Group	20,189	Asia-Pacific (APAC)	$10B+	Commerce and Shopping, Information Technology,...	18.0
24	Xiaomi	19,081	Asia-Pacific (APAC)	$10B+	Consumer Electronics, Hardware, Internet Servi...	8.0
27	Tencent Holdings	5,417	Asia-Pacific (APAC)	$10B+	Advertising, Gaming, Internet Services, Mobile...	5.0

These companies are far too large and will add a lot of noise to our smaller companies which have much smaller metrics. To address this we can drop these companies with the following code.

```
]:
#Remove Outliers
dfMain = dfMain[dfMain["Estimated Revenue Range"] != "$10B"]
```

After removing the outliers we can now look at some summary statistics for our data.

:

	Number of Founders	Number of Funding Rounds	Last Funding Amount Currency (in USD)	Last Equity Funding Amount Currency (in USD)	Total Equity Funding Amount Currency (in USD)	Total Funding Amount Currency (in USD)	Number of Lead Investors	Number of Investors	BuiltWith - Active Tech Count
count	11131.000000	11131.000000	1.113100e+04	1.113100e+04	1.113100e+04	1.113100e+04	11131.000000	11131.000000	11131.000000
mean	2.220016	3.990567	4.330174e+07	4.240990e+07	9.179507e+07	1.083454e+08	2.358638	7.193064	47.799569
std	1.176216	2.629161	2.118566e+08	2.332552e+08	4.633101e+08	6.200764e+08	1.665147	6.460872	24.959352
min	1.000000	1.000000	1.000000e+03	1.000000e+03	1.500000e+03	1.500000e+03	1.000000	1.000000	0.000000
25%	1.000000	2.000000	3.100000e+06	3.343300e+06	6.000000e+06	6.440978e+06	1.000000	3.000000	30.000000
50%	2.000000	3.000000	1.000000e+07	1.000000e+07	1.960000e+07	2.043342e+07	2.000000	5.000000	45.000000
75%	3.000000	5.000000	2.930160e+07	2.900000e+07	6.008766e+07	6.488955e+07	3.000000	10.000000	62.000000
max	18.000000	41.000000	1.378712e+10	1.378712e+10	1.860000e+10	2.521245e+10	23.000000	108.000000	234.000000

We can see that the mean number of founders for a company is around 2. Companies have had an average of almost 4 funding rounds and with an average of approximately 42 million dollars in their last funding round and about 92 million dollars in total equity funding. Total funding however is around 108 million dollars which is total equity funding plus the rest coming from non-equity sources such as loans and debt. On average companies have around 2 lead investors with about 7 total investors. Companies on average use 47 different technologies.

Now we can look at the other categorical values. We can start with "Headquarters Regions". This variable is of course describing the regional location of the company headquarters.

```
: dfMain["Headquarters Regions"].nunique()
: 46
```

We do see that there are 46 unique headquarters regions. We also notice something else. There are a few regions that have only a few values

```
East Coast, Northeastern US                                    21
Great Lakes, East Coast, Northeastern US                       18
Greater Philadelphia Area, East Coast, Southern US              9
New England, Northeastern US                                    6
Greater Los Angeles Area, Inland Empire, West Coast             2
Greater Philadelphia Area, East Coast, Northeastern US          1
Central America, Latin America                                  1
Name: Headquarters Regions, dtype: int64
```

We can simply group values that have less than 20 observations in a value called "Other".

```
#Handle Headerquarters regions with low values
dfMain["Count"]=dfMain.groupby("Headquarters Regions").transform('count')
dfMain["Headquarters Regions"].loc[dfMain["Count"] < 20 ] = "Other"
```

We can now look at “Industry Groups”, which is the industries the company belongs to. We notice that each company has multiple industry groups.

```
dfMain["Industry Groups"].value_counts()
```

```
Biotechnology, Health Care, Science and Engineering                                                                  426
Health Care                                                                                                          215
Financial Services                                                                                                   108
Education, Software                                                                                                   94
Financial Services, Lending and Investments                                                                           89
                                                                                                                    ...
Advertising, Design, Sales and Marketing, Software                                                                     1
Artificial Intelligence, Commerce and Shopping, Data and Analytics, Mobile, Platforms, Sales and Marketing, Software  1
Data and Analytics, Design, Financial Services, Information Technology                                                 1
Data and Analytics, Education, Health Care, Sales and Marketing, Software                                              1
Financial Services, Gaming, Other, Payments, Software, Sports                                                          1
Name: Industry Groups, Length: 4922, dtype: int64
```

This is a problem since there are 4922 unique group combinations. A very high number of categorical values will cause major problems in our model when we try to do One Hot Encoding, since each value for a categorical variable will be a feature and this will give us almost 5000 features, which will cause problems in our model. We can fix this by only using the first value of the group and assigning it to a new column named “Industry”. Our code is as follows.

```
# Get First value of Industry Group
dfMain["Industry"] = dfMain["Industry Groups"].str.split(',')
dfMain["Industry"] = dfMain["Industry"].apply(lambda x: x[0])
```

If we check our “Industry” column now we see we have a much more reasonable 42 values.

```
|: dfMain["Industry"].nunique()

|: 42
```

Now for our last categorical variable “Number of Employees”

```
]: dfMain["Number of Employees"].value_counts()

]: 11-50         4168
   101-250       2008
   51-100        1848
   1-10          1006
   251-500        910
   501-1000       565
   1001-5000      395
   10001+         134
   5001-10000      97
   Name: Number of Employees, dtype: int64
```

These values look reasonable and do not need to be cleaned.

We can now perform One Hot Encoding of the categorical variables. One Hot Encoding turns each value in a categorical feature into its own column, with a 1 indicating this row has that value and a 0 indicating it does not (Brownlee, 2017). We can use the following code to implement One Hot Encoding in Python.

```
#Apply One hot encode
ohe = OneHotEncoder()
oheMatrix = ohe.fit_transform(cat_vars.astype(str))
feat_names = ohe.get_feature_names()
```

For example here are some column names generated by One Hot Encoding. .

```
'x0_Greater San Diego Area, West Coast, Western US',
'x0_Greater Seattle Area, West Coast, Western US',
'x0_Gulf Cooperation Council (GCC)', 'x0_Latin America',
'x0_Midwestern US', 'x0_Nordic Countries, Scandinavia', 'x0_Other',
'x0_Research Triangle, East Coast, Southern US',
'x0_San Francisco Bay Area, Silicon Valley, West Coast',
```

So a row with a Headquarters Region of San Francisco Bay Area will have a value of 1 in this column called ”x0_San Francisco Bay Area, West Coast, Western US” and a value of 0 for all the other Headquarters regions which are all prefixed with “x0”. “Estimated

Revenue Range" and "Number of Employees" are prefixed with "X2" and "X3" respectively.

```
'x0_West Coast, Western US', 'x0_Western US', 'x1_nan',
'x2_$100M to $500M', 'x2_$10B+', 'x2_$10M to $50M',
'x2_$1B to $10B', 'x2_$1M to $10M', 'x2_$500M to $1B',
'x2_$50M to $100M', 'x2_Less than $1M', 'x3_1-10', 'x3_10001+',
'x3_1001-5000', 'x3_101-250', 'x3_11-50', 'x3_251-500',
'x3_5001-10000', 'x3_501-1000', 'x3_51-100'], dtype=object)
```

One Hot Encoding is required for both our Logistic Regression and Random Forest algorithms in Python, since they are incapable of directly working with strings. This is a limitation of the implementation of the algorithms in Python rather than a theoretical limitation of the Logistic Regression and Random Forest algorithms. (Brownlee, 2017).

Our Categorical Variables are now prepped and ready for the Machine Learning algorithms, we can now focus on our numerical variables.

First we will turn our Founded Date variable into a format that can be understood by the algorithm since dates are not a usable format for processing. What would be more helpful is having a "Years Active" variable that will hold the number of years the company has been in operation. Doing this in Python is simple; we only need the following code.

```
# Create Years Founded Variable
dfMain['Years Active'] = pd.to_datetime(dfMain["Founded Date"], errors="coerce")
dfMain["Years Active"] = pd.DatetimeIndex(dfMain["Years Active"]).year
dfMain["Years Active"] = 2020  - dfMain["Years Active"]
```

Some of our numerical variables are in Integer format but are required to be in Float format to work with the Python Implementation of the algorithms. To do so is simple and can be done with the following code.

```
#Remove commas from numbers
locale.setlocale(locale.LC_NUMERIC, '')
num_vars = num_vars.astype(str)
num_vars = num_vars.applymap(atof)
```

We also have to scale our numerical values which means turning all the numerical values into a smaller range of numbers such as 1-15 or 0-1. Currently our numerical values range from 1 to 25 billion. This is too big of a range. Large differences in the magnitudes of values may lead to inaccurate results in our model (Saini, 2019). To solve this we can scale our numerical variables with Python's StandardScaler() class in sklearn package. Our code is simple and is as follows.

```
#scale numerical variables
scaled_num_vars = preprocessing.StandardScaler().fit_transform(num_vars)
```

Now we see that our values are within a less drastic range of values.

```
: scaled_num_vars

: array([[ 0.02718835,  1.51338159,  0.38395454, ..., -0.02370621,
           0.2239484 , -0.32919923],
         [-0.0327764 ,  1.51338159,  0.38395454, ..., -0.02396372,
          -0.06442185,  0.14610818],
         [ 0.06498432, -0.18706261, -0.37677846, ..., -0.02104169,
           0.32843037,  0.03730287],
         ...,
```

Now with both our numerical and categorical variables prepared we can move on to our analysis.

Analysis

We can start by looking at our target variable. We see the ratio of false observations to true observations at a ratio of roughly 4:6. This is considered an acceptable level of imbalance and is considered a “slight imbalance” as opposed to a “severe imbalance” (Brownlee, 2019). The analysis can proceed as usual, there is no need to adjust or compensate for the imbalance.

```
24]: dfMain["Has Reached Series A"].value_counts()

24]: True     6490
     False    4608
     Name: Has Reached Series A, dtype: int64
```

We can now look at the correlations of numerical variables and our target variables.

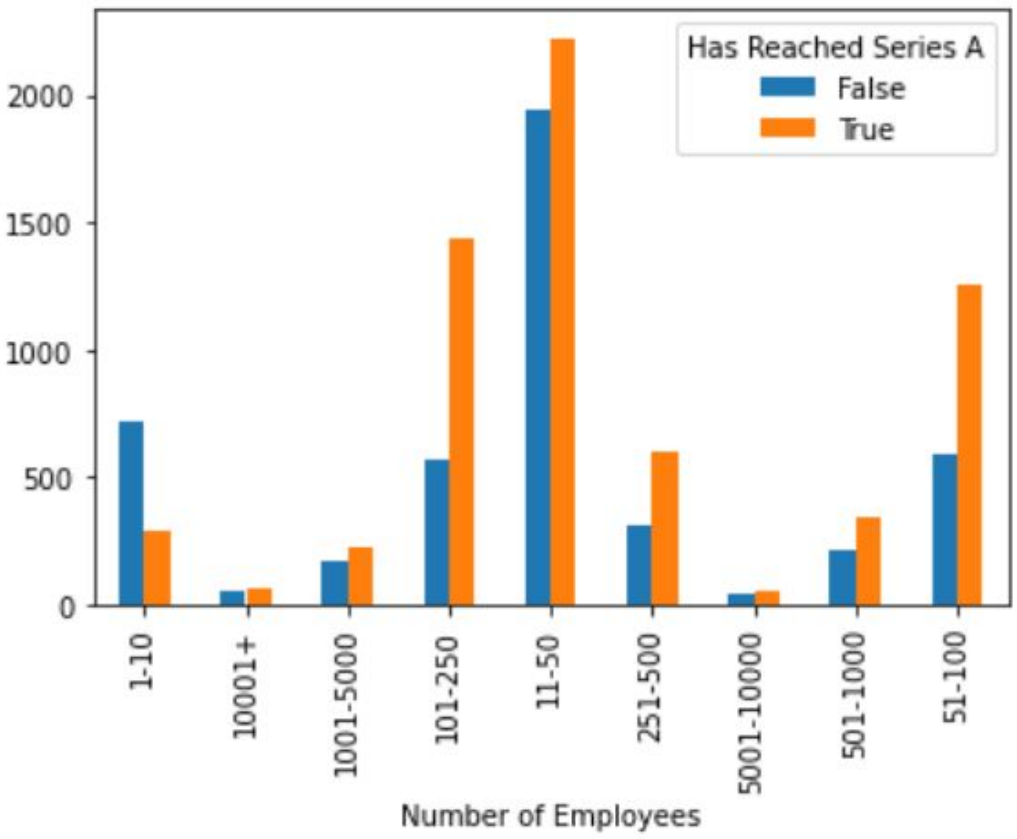
```
]: num_vars.corrwith(dfMain["Has Reached Series A"])

]: Number of Articles                              -0.007323
   Number of Founders                               0.077417
   Number of Funding Rounds                         0.114631
   Years Active                                    -0.049818
   Last Funding Amount Currency (in USD)            0.048748
   Last Equity Funding Amount Currency (in USD)     0.032204
   Total Equity Funding Amount Currency (in USD)    0.019229
   Total Funding Amount Currency (in USD)           0.018138
   Number of Lead Investors                         0.215619
   Number of Investors                              0.222137
   BuiltWith - Active Tech Count                    0.081618
   SEMrush - Monthly Visits                         0.010303
   SEMrush - Average Visits (6 months)              0.010361
   SEMrush - Visit Duration                         0.035443
   SEMrush - Page Views / Visit                     0.026032
   dtype: float64
```

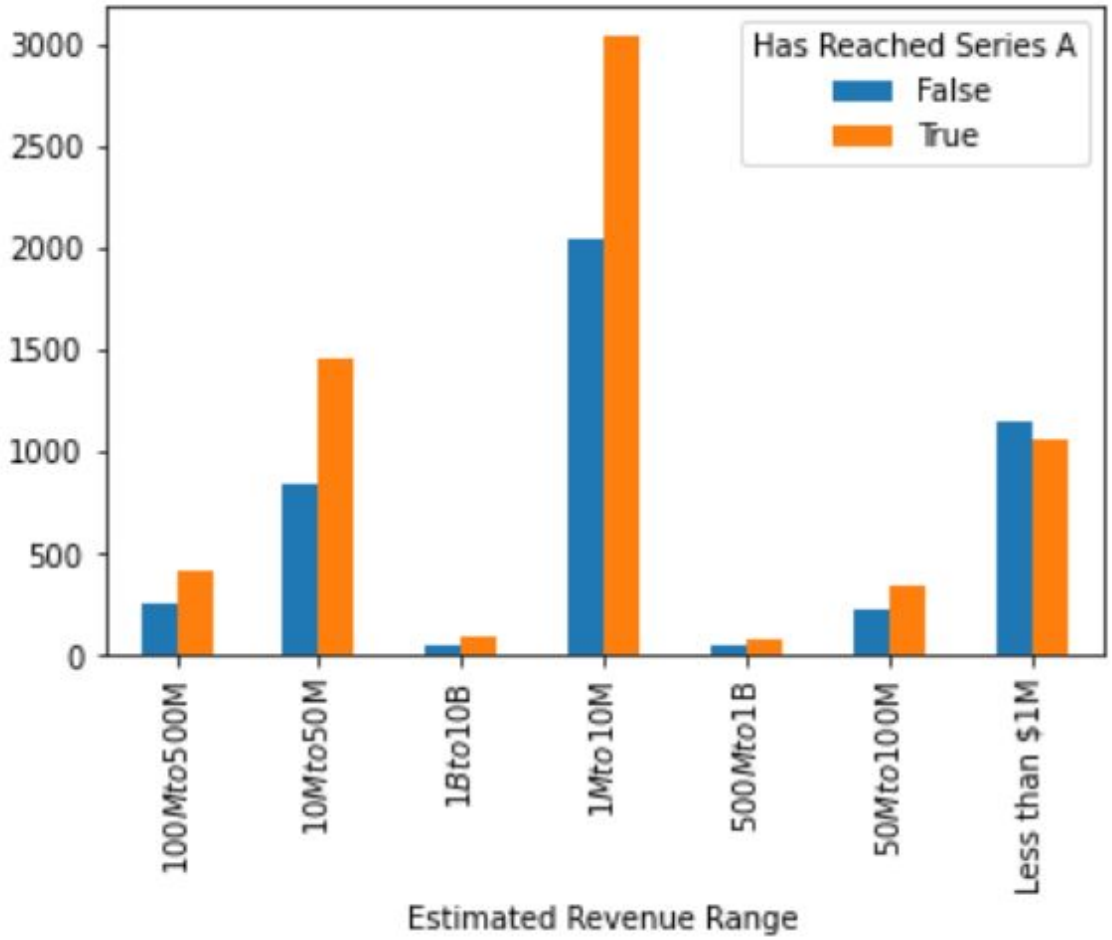
It appears the funding round has a .11 correlation with the target variable. This makes sense since the Series A is itself a funding round. Another 2 standout

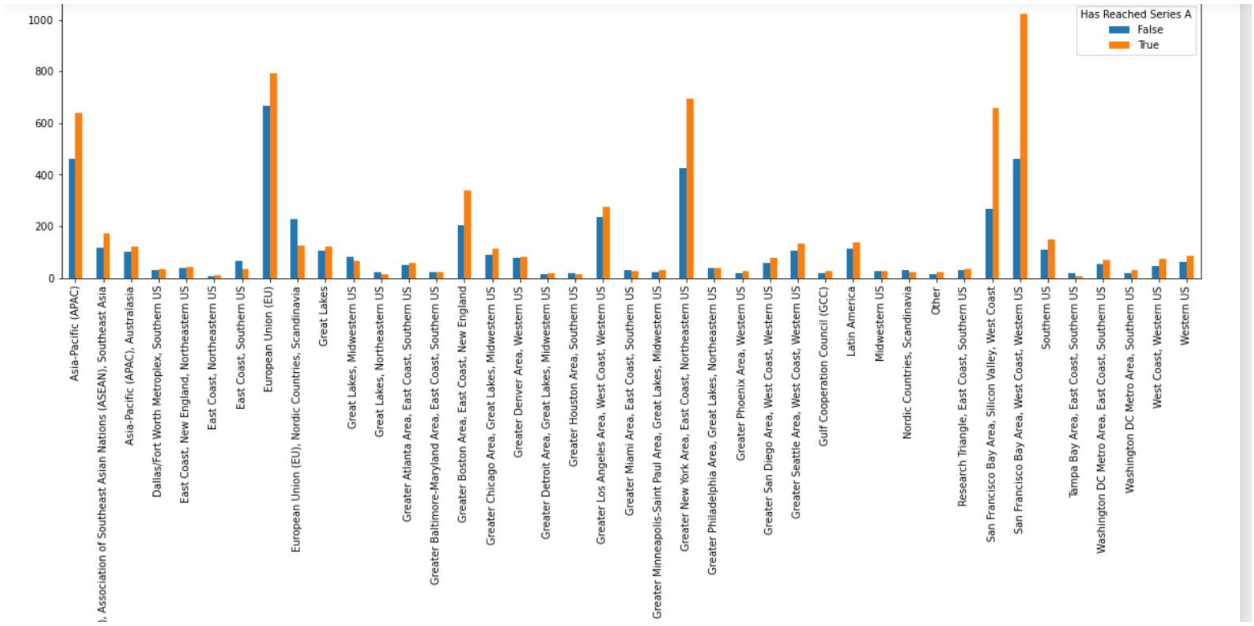
correlations are Number of Lead Investors and Number of Investors with .21 and .22 respectively. This also makes sense since Series A companies tend to have a large amount of investors compared to non-series A companies.

We can now look at some visualizatoins that will help us understand our data better, we can start with the categorical values. It appears that the number of employees is not a strong predictor of a company having reached Series A. We do see that a lot of companies in the 1-10 employees range haven't reached Series A and this makes sense since smaller companies are still usually at seed funding rounds. 11-50 number of employees seem to be even and this is interesting since this is the stage that companies usually go to Series A. 101-250 and 51-100 ranges have an imbalanced number of observations for True which also makes sense since larger companies are usually at Series A.



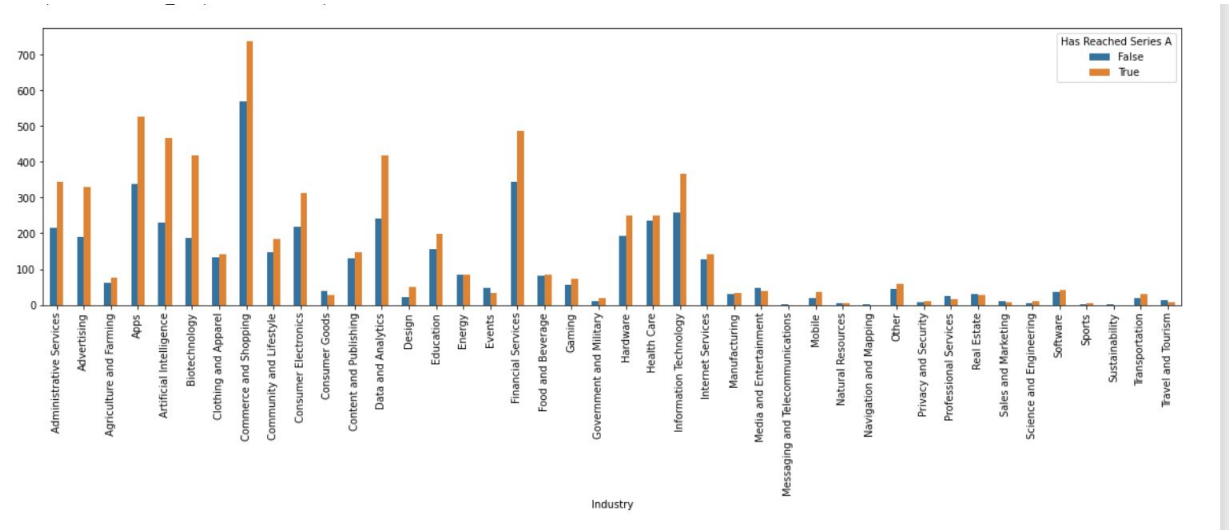
We can now look at values for the Estimated Revenue Range variable. We surprisingly notice not too many standouts for the values between Series A and non Series A. 10Mto50M and 1Mto10M seem to have slightly more observations for Series A companies. We can't say for sure why this might be since the other classes are relatively balanced.


Has Reached Series A
False
True
3000
2500
2000
1500
1000
500
0
100Mto500M
10Mto50M
1Bto10B
1Mto10M
500Mto1B
50Mto100M
Less than $1M
Estimated Revenue Range

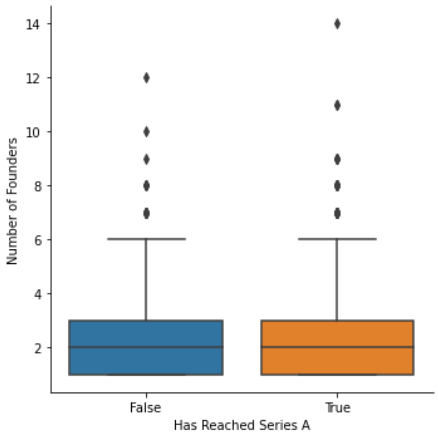


Now we see our variable of “Headquarters Regions”. We see that 2 values have standout values for Series A companies. Companies with Headquarters in San Francisco and New York have a very high ratio of Series A to non Series A companies. This makes sense since these 2 cities are well known as major startup hubs.

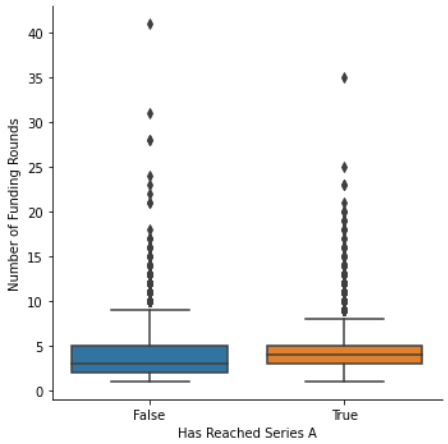
Now for our final categorical variable “Industry”. There are 3 values that standout. It appears that companies that are in Apps, Biotechnology, and Artificial Intelligence have the highest ratio of Series A to non Series A companies.



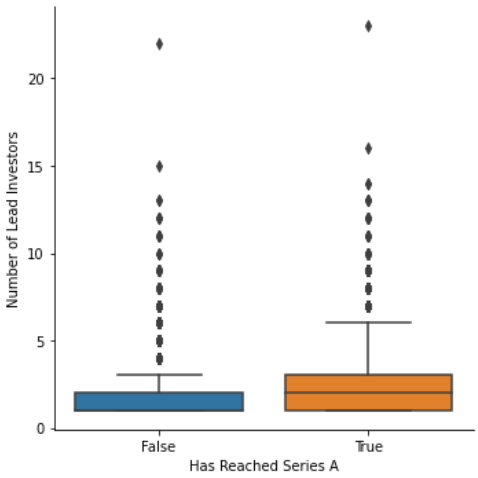
We can now look at our numerical variables. Starting off with “Number of Founders”. We don’t see too big of a difference in the number of founders between Series A and non Series A companies. We can say that the number of founders of a company is not associated with whether it reaches Series A or Not.



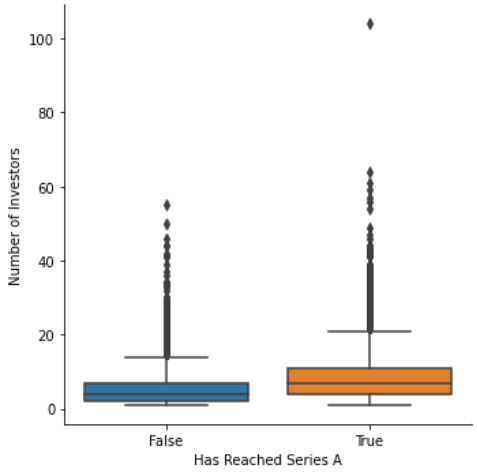
We see a similar distribution of observations for “Number of Funding Rounds”



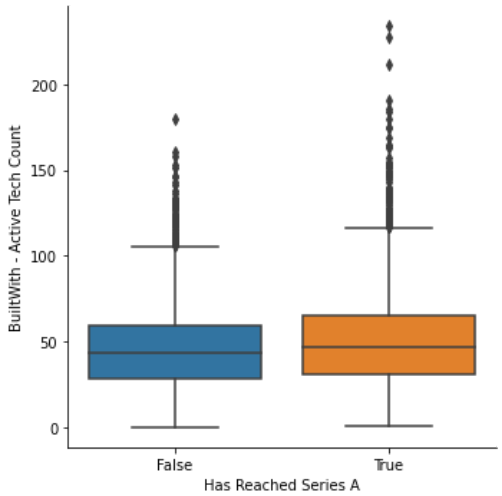
We see that companies that have reached Series A have on average a larger number of Lead Investors.



Companies that have reached Series A also seem to have on average more total investors as well.



Companies that have reached Series A also tend to be built with slightly more technologies.



We can now move to our main analysis and set up the predictive analysis. Implementation of the algorithms in Python is relatively straightforward with the sklearn library. Before we insert our dataset into the algorithms we need to split our data into training and test sets for honest assessment. We can do so as shown below.

```
y = dfMain["Has Reached Series A"]
x_train, x_test, y_train, y_test = train_test_split(x, y)
```

Our machine learning algorithms then be set up as follows:

```
clf = RandomForestClassifier().fit(x_train, y_train)
clf = LogisticRegression().fit(x_train, y_train)
```

We can then get the predictive accuracy of the algorithms. We will start off with our Logistic Regression algorithm. We get a surprisingly low predictive accuracy.

```
prediction = clf.predict(x_test)
print(accuracy_score(y_test, prediction))
```

```
0.6317117117117117
```

Let's investigate first by looking at the regression coefficients of our numerical variables.

	Features	Coeff
0	Number of Articles	-0.226716
1	Number of Founders	0.085051
2	Number of Funding Rounds	-0.119695
3	Years Active	-0.033986
4	Last Funding Amount Currency (in USD)	0.140943
5	Last Equity Funding Amount Currency (in USD)	0.136104
6	Total Equity Funding Amount Currency (in USD)	-0.434140
7	Total Funding Amount Currency (in USD)	0.096672
8	Number of Lead Investors	0.407628
9	Number of Investors	0.406038
10	BuiltWith - Active Tech Count	0.069158
11	SEMrush - Monthly Visits	0.434858
12	SEMrush - Average Visits (6 months)	-0.144095
13	SEMrush - Visit Duration	0.076154
14	SEMrush - Page Views / Visit	0.022347

We can’t interpret Logistic Regression Coefficients directly. We must transform them from log odds to odds ratios. We can do this by simply taking the exponent of the Coefficient. Doing this will give us a list of odds ratios (Ma, 2019).

	Features	Coeff
0	Number of Articles	0.831914
1	Number of Founders	1.069555
2	Number of Funding Rounds	0.881407
3	Years Active	0.952841
4	Last Funding Amount Currency (in USD)	1.142543
5	Last Equity Funding Amount Currency (in USD)	1.167490
6	Total Equity Funding Amount Currency (in USD)	0.655415
7	Total Funding Amount Currency (in USD)	1.030235
8	Number of Lead Investors	1.572957
9	Number of Investors	1.520251
10	BuiltWith - Active Tech Count	1.098097
11	SEMrush - Monthly Visits	1.229853
12	SEMrush - Average Visits (6 months)	1.080684
13	SEMrush - Visit Duration	1.045093
14	SEMrush - Page Views / Visit	1.030916

We have to continue to be careful with our interpretation since all our numerical variables were scaled in the Data Preparation Step. For example for the Years Active feature, we can't speak of the number of years but instead we must say standardized units of the Years Active feature (Ma, 2019).

Thankfully the results are easy to interpret since they are odds. 2 standout Features are Number of Lead Investors and Number of Investors. A one standardized unit increase in the Number of Lead Investors feature increases the likelihood a company is a Series A by 57%, while holding all other variables constant. Number of Investors increases the likelihood by 52%.

Surprisingly Years Active is not positively correlated with business success. One would assume the longer a company is alive the more likely it is to reach Series A funding, but counterintuitively each standardized unit increase in the Years Active feature actually decreases the likelihood of a company reaching Series A by 5%.

Also counterintuitively is Number of Articles is negatively correlated with a company reaching Series A. One would think that if a company has a large number of articles written about them in the media they would be a Series A company, but this does not seem to be the case based on this analysis.

The last stand out predictor variable is SEMrush - Monthly visits which increases the likelihood a company is Series A by 22% for each standardized unit increase. This makes sense since more monthly visits means the more popular a company is. The more popular it is the more likely it is successful. The rest of the variables are relatively close to 1 and therefore do not offer much predictive power.

We can now look at our categorical variables which consist of 105 features, since we One Hot Encoded them in our Data Preparation step. The three variables with the highest predictor power were if a company was in the Biotech industry, or was based in San Francisco or in Los Angeles.

51	x1_Biotechnology	1.967300
39	x0_San Francisco Bay Area, West Coast, Western US	1.967951
21	x0_Greater Los Angeles Area, Inland Empire, We...	1.990274

Since all categorical variables are binary we can interpret this as a company based in San Francisco has a 96% more likelihood of Reaching Series A than a company who is not. Biotechnology giving a company such a high likelihood of success is surprising. A company's chance of success being increased dramatically by being based in San Francisco and Los Angeles is not surprising at all. These are gigantic metropolitan areas and have very large financial and business institutions which make the environment very conducive to business success.

A company being based in the European Union and Latin America offers the least predictive power of all the variables.

3	x0_Central America, Latin America	1.000000
8	x0_European Union (EU)	1.012651

The variables which drastically lower the likelihood of a company having Reached Series A is if the number of employees is 1-10 or if the company is based in the Southern United States. A company having 1-10 employees is not surprisingly correlated with a low likelihood of success since small companies with 1-10 employees are usually startups that have not yet reached product market fit. Surprisingly, being based in the Southern United States is correlated with negative business success.

	Features	Coeff
95	x3_1-10	0.289874
7	x0_East Coast, Southern US	0.452790
87	x1_Travel and Tourism	0.605738
28	x0_Greater Philadelphia Area, Great Lakes, Nor...	0.650095
41	x0_Tampa Bay Area, East Coast, Southern US	0.651052
76	x1_Navigation and Mapping	0.656270

Also some variables of interest are shown above. Being based in Philadelphia and Tampa Bay is associated with negative business outcomes. As is being in the Travel and Navigation industries.

Now that we have done both numerical and categorical interpretations for our Logistic Regression model we can move on to Random Forest and see if we get similar or different results.

We can start with the prediction accuracy. Our Random Forest Classifier was more accurate than our Logistic Regression with around 75% accuracy.

```
prediction = clf.predict(x_test)
print(accuracy_score(y_test, prediction))
```

```
0.7517117117117117
```

Now we can look at the feature importances of the Random Forest model. Thankfully feature importances are much more straightforward to interpret than Logistic Regression coefficients. A High feature importance simply means the feature contributes to more of the variation of the dependent variable (Grover, 2017). We can start with our numerical variables.

	Features	Importances
1	Number of Founders	0.024027
8	Number of Lead Investors	0.029433
2	Number of Funding Rounds	0.040937
14	SEMrush - Page Views / Visit	0.052447
13	SEMrush - Visit Duration	0.052987
10	BuiltWith - Active Tech Count	0.057095
11	SEMrush - Monthly Visits	0.057427
9	Number of Investors	0.058394
0	Number of Articles	0.059084
12	SEMrush - Average Visits (6 months)	0.059184
3	Years Active	0.078031
6	Total Equity Funding Amount Currency (in USD)	0.095683
7	Total Funding Amount Currency (in USD)	0.096262
5	Last Equity Funding Amount Currency (in USD)	0.117051
4	Last Funding Amount Currency (in USD)	0.121959

This is surprising as our Random Forest gives us the opposite results of our Logistic Regression model and considers Number of Founders and Number of Lead Investors as the least important features with little predictive power. We have to remember that a small Feature importance does not mean that a company is less likely to be successful, but that the variable has little predictive power in determining whether the company is successful or not (Grover, 2017).

The Last Funding Amount Currency (in USD) and Last Equity Funding Amount Currency (in USD) are the variables with the most predictive power. This makes sense as companies which have reached Series A will tend to have larger funding.

Our categorical variables are more similar to our Logistic Regression model. Both our Logistic Regression and Random Forest model agree that a company being based in Latin America has little predictive power in determining if it will be successful.

	Features	Importances
26	x0_Greater Philadelphia Area, East Coast, Nort...	0.000000
3	x0_Central America, Latin America	0.000046

Here are our top 5 categorical variables

25	x0_Greater New York Area, East Coast, Northeas...	0.021610
91	x2_$1M to $10M	0.022597
0	x0_Asia-Pacific (APAC)	0.023699
53	x1_Commerce and Shopping	0.024790
95	x3_1-10	0.044514

Like our Logistic Regression Model, our Random Forest model agrees that a company having between 1-10 employees is one of the most important variables for prediction. Being Based in a metropolitan area like New York also has a relatively large influence on the target variable. Similar to our Logistic Regression Analysis, being based in San Francisco has a large predictive power.

39	x0_San Francisco Bay Area, West Coast, Western US	0.018629
68	x1_Health Care	0.018860

Since we have 104 categorical features they all have a relatively small effect on the dependent variable. For example “x3_1-10” is our top feature importance but it only accounts for about 4.4% of the total value of the feature importances. Our 2nd most important feature “x1_Commerce and Shopping”” only accounts for about 2.4%.

This concludes our Analysis. We can now move on to the Data Summary.

Data Summary and Implications

We can now begin to summarize the insights we have uncovered based on our analysis.

We can begin by looking at the Location the company is based in. Our Logistic Regression Model found a high likelihood of success if a company was based in San Francisco. This makes sense as we can see examples of very successful companies that are based in San Francisco such as Google, Apple and Facebook just to name a few. Our Random Forest model also assigned a high feature importance score to the San Francisco location variable as well. Random Forest also showed being in a large metropolitan area like New York had a large effect on the target variable. Being based in Latin America seemed to not have any influence on the target variable and this was the case with both the Logistic Regression and Random Forest Model. Our Logistic Regression found that being based out of the South in the United States had a big negative correlation with success and this was not the case with our Random Forest Model.

Next we can discuss industry. Our Logistic Regression model found a very high probability of success for companies in the Biotechnology industry and very low likelihood of success for companies in the Travel and Navigation industries. Our Random Forest model did not agree with this and found a company being in the Ecommerce industry to have a lot of predictive power about its success or lack thereof.

Both our Random Forest and Logistic Regression model found a company having 1-10 employees to be a very important data point. Our Logistic Regression

model found having 1-10 employees decrecred a company's chance of success by around 72%. We can conjecture that the less employees a company has the less work it is able to do which decreases its chance of success.

2 standout numerical features from the Logistic regression analysis were Number of Investors and Number of Lead Investors. This however doesn't offer us any novel insight into business success. We can't say that having a large number of Investors causes business success. A large number of investors is simply correlated with companies reaching Series A. It is likely that Series A companies simply have more investors due to the fact that by definition they raised equity capital from investors.

The 2 standout numerical features in our Random Forest model were Last Funding Amount Currency (in USD) and Last Equity Funding Amount Currency (in USD). This however doesn't offer us any deep business insights, since as mentioned in the Research Question section, companies that have reached Series A already have higher funding than companies that have not.

Data points mentioned above are important to a business's success but we have to keep in mind these data points do not include important but impossible to quantify personal characteristics of the founders that are also important to success such as passion and determination (Gerber, 2013).

A recommended course of action for aspiring business owners is to consider moving to San Francisco or other large metropolitan areas to increase chances of business success. This is not only backed up by our data analysis but also recommended by Investopedia, who claim that San Francisco is very conducive to business success because it has a large resource pool, legal support and culture of

innovation and risk taking (Seth, 2019). Another possible course of action would be to hire as many employees as possible, this may not be practical in all circumstances but something to consider.

For anyone looking to expand on this analysis there are 2 recommended courses of actions. The first is finding a bigger dataset to run the analysis. We analyzed roughly 12,000 companies, however this is not even a small fraction of the 5.6 million firms in the United States alone (SBEcouncil, 2020). A bigger dataset might uncover more fruitful insights. Also another course of direction could be to redefine the success criteria. We used a Series A funding round as our success criteria, but maybe with supplemental data a new criteria such as profit or growth rate might be used.

Sources:

1. Loizos, C., 2019. *Techcrunch Is Now A Part Of Verizon Media*. [online] Techcrunch.com. Available at: <https://techcrunch.com/2019/04/25/a-quick-look-at-how-fast-series-a-and-seed-rounds-have-ballooned-in-recent-years-fueled-by-top-investors/#:~:text=And%20the%20results%2C%20while%20not,million%20in%202017%2C%20says%20Wing.> [Accessed 3 December 2020].
2. Glazer, A., 2019. *Predicting Startup Failures Using Classification*. [online] Medium. Available at: <https://towardsdatascience.com/predicting-startup-failures-using-classification-8e11d2703e0a> [Accessed 3 December 2020].
3. Hayes, A., 2020. *How Mergers And Acquisitions – M&A Work*. [online] Investopedia. Available at: <https://www.investopedia.com/terms/m/mergersandacquisitions.asp> [Accessed 3 December 2020].
4. Seibel, M., 2018. *Product Market Fit And Fundraising: Fundraising, Product Market Fit, Seed Round*. [online] YC Startup Library. Available at: <https://www.ycombinator.com/library/5y-product-market-fit-and-fundraising> [Accessed 3 December 2020].
5. Investopedia. 2020. *What Is More Important For A Business, Profitability Or Growth?*. [online] Available at: <https://www.investopedia.com/ask/answers/020415/what-more-important-business-profitability-or-growth.asp#:~:text=it%20with%20growth.-,The%20Bottom%20Line,relates%20to%20a%20particular%20company.> [Accessed 27 November 2020].
6. Gerber, S., 2013. [online] Business.com. Available at: <https://www.business.com/articles/3-qualities-important-startup-founders/> [Accessed 27 November 2020].
7. Sba.gov. 2020. [online] Available at: <https://www.sba.gov/sites/default/files/Business-Survival.pdf> [Accessed 26 November 2020].
8. Brownlee, J., 2020. *How To Calculate Feature Importance With Python*. [online] Machine Learning Mastery. Available at:
9. Crunchbase | Knowledge Center. 2020. *Where Does Crunchbase Get Their Data?*. [online] Available at: <https://support.crunchbase.com/hc/en-us/articles/360009616013-Where-does-Crunchbase-get-their-data-> [Accessed 26 November 2020].
10. Brownlee, Jason. "Why One-Hot Encode Data In Machine Learning?". *Machine Learning Mastery*, 2017, https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/.
11. Saini, Rahul. "Feature Scaling- Why It Is Required?". *Medium*, 2019, https://medium.com/@rahul77349/feature-scaling-why-it-is-required-8a93df1af310.
12. Brownlee, Jason. "A Gentle Introduction To Imbalanced Classification". *Machine Learning Mastery*, 2019, https://machinelearningmastery.com/what-is-imbalanced-classification/.
13. Ma, Ying. "Interpreting The Impact Size Of Logistic Regression Coefficients". *Medium*, 2019, https://medium.com/ro-data-team-blog/interpret-the-impact-size-with-logistic-regression-coefficients-5eec21baaac8.
14. Grover, Prince. "Intuitive Interpretation Of Random Forest". *Medium*, 2017, https://medium.com/usf-msds/intuitive-interpretation-of-random-forest-2238687cae45.
15. Seth, Shobhit. "Why Is Silicon Valley A Startup Heaven?". *Investopedia*, 2019, https://www.investopedia.com/articles/personal-finance/061115/why-silicon-valley-startup-heaven.asp.
16. "Small Business & Entrepreneurship Council". *Sbecouncil.Org*, 2020, https://sbecouncil.org/about-us/facts-and-data/#:~:text=According%20to%20data%20from%20the

,the%20United%20States%20in%202016.&text=Firms%20with%20fewer%20than%20500,99.7%20percent%20of%20those%20businesses.&text=Firms%20with%20fewer%20than%20100%20workers%20accounted%20for%2098.2%20percent.