What are the characteristics of successful startups?

Ranak Bansal, Charles Miele, Mehar Poreddy

2022-12-05

Abstract

Successful startups are defined by hundreds of thousands of factors, ranging from the founding team to the macroeconomic environment. We collected data on 206 FinTech startups founded between 2008 and 2015 that raised pre-seed to Series B investments and analyzed 25 variables to understand which were most consistent among successful versus failed startups. We found that the most predictive variables for startup success were active tech count and number of lead investors. We also looked at the strengths and limitations of leveraging these results to predict the future trajectory of a company.

Background Info: Venture Capital / Startups

Clark (2007) claimed that "the average person in 1800 was no better off in material terms than the average person of 10,000 or 100,000 BC." The effects of technological advancement in the 1800s and beyond, with the Industrial Revolution followed by the Great Divergence, skyrocketed economic growth. In other words, we are much better off than 200 years ago, particularly due to innovation. Kelly, Paplanikolau, Seru, and Taddy (2018) provided evidence supporting this assertion, finding that "technological progress is correlated with productivity by creating the first direct measure of technological progress that is comparable across time and space." But what entities thrust the economy forward? In recent decades, startups - companies that generate significant economic and social value - have become the main impetus.

While companies such as Amazon, Apple, Facebook, Google, Microsoft, and Tesla have faced pressing social and regulatory concerns, they have proven to be invaluable to the economy. They have all been supported by venture capital firms in the past, which provide financing for businesses that find themselves in the early stages of growth. Kaplan and Lerner (2010) estimated that roughly one-half of all true IPOs are VC-backed despite less than 0.25% receiving VC funding. Furthermore, Gornall and Strebulaev (2015) estimated that public companies that received VC funding account for 44% of research and development spending and 20% of the market capitalization of U.S. public companies.

Venture capital firms operate uniquely compared to other industries. The success of a fund raised by a VC comes down to whether they are able to invest in one to two startups that achieve success, whether it be through raising additional capital, which increases the value of the company, acquisition, or an initial public offering. These successful investments represent only 5-20% of the portfolio but typically return 10x to 100x. Therefore, they easily offset the losses from the rest of the portfolio and generate substantial profits for the entirety of the firm.

Given the high risk involved in venture capital, nearly 95% of firms are not profitable (Dean 2017). Every venture capital firm has its own M.O., which can be seen in its thesis, sourcing, and due diligence strategies. Furthermore, VCs remain discreet about these intellectual processes. Therefore, there is a lack of standardization in the industry. Given the wide variety of factors involved in a startup, ranging from the founding team to macroeconomic conditions, it is extremely difficult for the average venture capital firm to discover and invest in a successful startup. Given that 90% of startups (Krishna, Agrawal, Choudhary 2015) and 75% of venture-backed startups fail (Ghosh 2012), this process becomes even more difficult.

Several researchers have attempted to dissect this problem. In How Do Venture Capitalists Make Decisions, Gompers, Gornall, Kaplan, and Strebulaev surveyed 885 venture capitalists at 681 firms about their decision-

making process. Specifically, they analyzed deal sourcing, investment selection, valuation, deal structure, post-investment value-added, exits, internal firm organization, and relationships with limited partners. They found that VCs prioritize the management team over the product and technology. VCs also attribute the management team to the outcome of the company.

However, Okrah, Nepp, and Agbozo take a different approach in Exploring the factors of startup success and growth. They attempted to discover the factors influencing innovation and making startups attractive for financing, with an emphasis on developing nations, by looking at data in Africa and Latin America in comparison to Europe. They concluded that government policies, internal market openness, and internal market dynamics highly affected funding, thereby having an impact on startup success. Essentially, this paper argues that macroeconomic factors significantly affect a startup.

Finally, Gelderen, Thurik, and Bosma took a closer look at the human aspect of a company in Success and Risk Factors in the Pre-Startup Phase. They sampled 517 entrepreneurs and followed them over a 3-year period. In the end, 195 succeeded, and 115 were abandoned. They concluded that the entrepreneur's environment, such as their network, financial, and ecological approaches, had more of an impact than the characteristics of the entrepreneur themselves.

This literature highlights the difficulty of investing in startups. Factors such as the management team, macroeconomic conditions, and core product features all have a unique impact on the outcome of a business. It is difficult to conclude a single variable as a predictor for startup success.

This paper seeks to understand the many factors most consistent with successful startups and how they differ from failed ones. Furthermore, we aim to explore the strengths and limitations of using data to predict a company's success. We analyze data from Crunchbase, looking at a multitude of factors, including the company's investors, year founded, and more. Our goal is to navigate through the vast amount of data and research conducted in this space and explore the possibilities of leveraging data to predict the future trajectory of companies.

Data Collection / Cleaning Process

Our data collection process had four main steps:

1.) We used CrunchBase's filter feature to find companies that fit our criteria. The filters we used were:

a. FinTech companies

b. Founded in the U.S.

c. Founded between 2008 and 2015 *(we have since expanded this date range from our presentation (it was initially 2008-2012) in order to have a more complete data set)*

d. Latest funding rounds being either pre-seed, seed round, Series A, or Series B.

e. The company cannot be public

We decided to control for these variables in order to limit the number of potential confounders.

2.) We chose 25 CrunchBase columns that we felt would be most useful in answering our question. The columns that we gathered are described in the table below. Some of these metrics are numerical, while others are categorical.

3.) We exported the CrunchBase dataset to a CSV file(~700 companies), and then omitted rows that had missing columns, reducing our number of companies to 206. We also made changes to many of our numerical columns in order to make them understandable by R, such as omitting dollar signs and commas.

4.) After finishing our data collection process, we were left with a data set of 206 observations and 25 variables, but we added two other metrics - LinkedIn followers and company geographical state. In order to acquire the LinkedIn followers for each company, we used a Python web-scraping script using the selenium library. To acquire the company's founded state, we manually researched this data.

Importing our data

```
companies <- read.csv("final.csv")
success <- subset(companies, Failed == 0)
failed <- subset(companies, Failed == 1)
Regions <- read.csv('Regions.csv')
## Table of all the variables we collected along with their class.
## Note that some of the integers, such as DiversitySpotlight are true/false dummy variables.
varData <- setNames(stack(sapply(companies, class))[2:1], c('variable', 'class'))
```

```
varData$Description <- descriptions
gt::gt(varData)
```

variable	class	Description
Name	character	Name of company
LastFundingType	character	Last type of funding receieved (e.g. pre-seed, seed, series A, etc.)
CB.Rank	integer	Unique rank provided CrunchBase. Generally correlates with company success.
Website	character	Company website domain
Description	character	Short(usually one-sentence) description of company
Operating.Status	character	Identifies whether a company is active of closed
FoundedDate	integer	Year the company was founded
NumArticles	integer	Number of articles written about the company in the press
NumFounders	integer	Total number of founders
NumRounds	integer	Total number of rounds the company has raised
Founders	character	String containing each founder's first and last name, seperated by commas.
TotalFundingAmt	numeric	Total amount of funding the company has raised(USD)
WasAcquired	integer	Boolean dummy variable indicating if company has been acquired
Top5Investors	character	String containing the top 5 investors of the company, seperated by commas.
NumLeads	integer	Total number of lead investors.
NumInvestors	integer	Total number of investors
NumAcquisitions	integer	Total number of acquisitons made by the company
ActiveTechCount	integer	Total number of active technologies in the company's tech stack
HeadquartersRegion	character	Region the company operates in.
DiversitySpotlight	integer	Boolean dummy variable indicating if company has any notable 'diverse' statistics (e.g.
ClosedDate	character	Year the company has closed. If not closed, says 'Not Closed'
FullDescription	character	Longer description of the company
Failed	integer	Boolean dummy variable indicating if the company has closed AND was not acquired
LI.Followers	integer	Number of LinkedIn followers. If the company has no LinkedIn account, 0.
domain.length	integer	String length of the company's website domain.

Basic Analysis

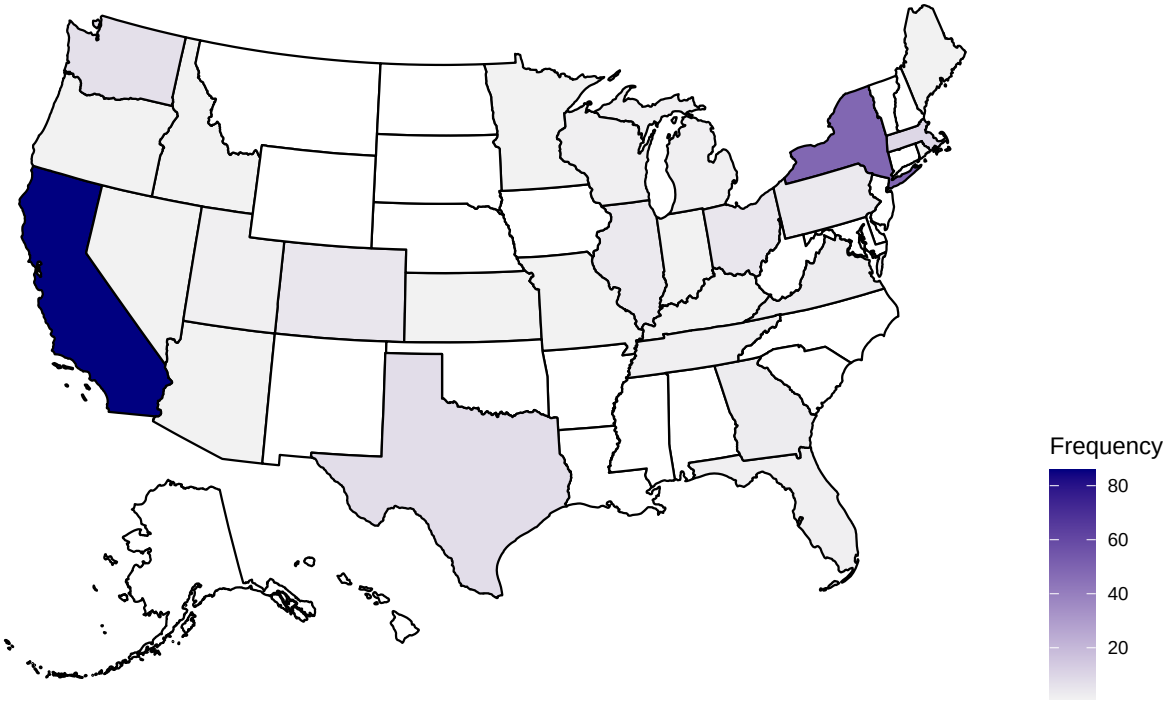
This section includes some basic data analysis that will give readers some context surrounding our data. We hope that it gives you a general idea about the data we have gathered.

In this section, we analyze our variables independently. We chose this particular subset of variables because we found them to be the most insightful and intuitive. In the later sections, we analyze more variables in aggregate.

Map

```
#All startups
data <- xtabs(~State, Regions)
startups <- data.frame(data) # frequency of state converted into data frame
colnames(startups)[1] = "state" # have to rename column to state for plot_usmap() to work
```

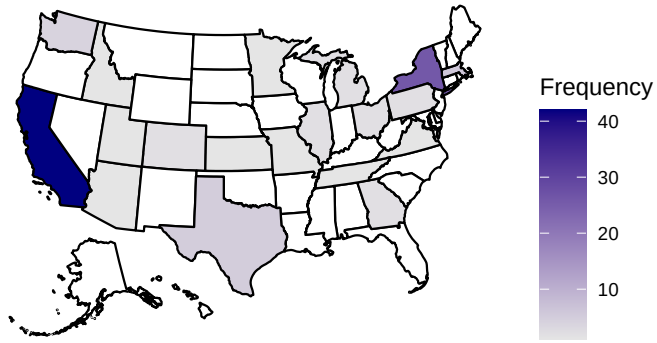
```
plot_usmap(data = startups, values = "Freq") +
  scale_fill_continuous(na.value = "white", low = "gray95", high = "navyblue",
                        name = "Frequency", label=scales::comma) +
  theme(legend.position = "right")
```



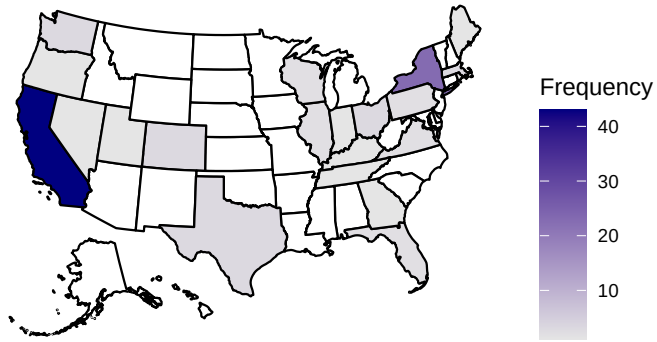

```
merged <- merge(Regions, companies, by='Name') #merged dataframes based on startup name

states <- subset(merged, select=c(Name, State, Failed))
failMap <- subset(states, Failed == 1) #subset based on success/failure
successMap <- subset(states, Failed == 0) #subset based on success/failure

#failed startups
failx <- xtabs(~State, failMap) #same process as all
failxd <- data.frame(failx)
colnames(failxd)[1] = "state"

f <- plot_usmap(data = failxd, values = "Freq") +
  scale_fill_continuous(na.value = "white", low = "gray90", high = "navyblue",
                        name = "Frequency", label=scales::comma) +
  theme(legend.position = "right") + labs(title = "failed")

#successful startups
successx <- xtabs(~State, successMap) #same process as all
successxd <- data.frame(successx)
colnames(successxd)[1] = "state"

s <- plot_usmap(data = successxd, values = "Freq") +
  scale_fill_continuous(na.value = "white", low = "gray90", high = "navyblue",
                        name = "Frequency", label=scales::comma) +
  theme(legend.position = "right") + labs(title = "success")
```

```
#plot
grid.arrange(s, f)
```

success



failed



The first map shows an overall distribution of where startups were founded, regardless of their outcome. As we expected, a majority of FinTech startups were founded in either California or New York. States such as Texas, Colorado, and Washington are also notable.

When we seperated the graphs on whether the companies are successful or not, we see no significant change, which seems to indicate that location is not a significant predictor of success; however, further statistical analysis will need to be conducted to prove this.

Most Successful VCs

```
# Function that returns a list containing the raw data of each company's top
# 5 investors. If the company had less than 5 investors, it simply showed all
# the investors the company has received data from.
getInvestorList <- function(d) {
  results <- c()
  for (i in 1:nrow(d)) {
    top5 <- str_split(d[i,'Top5Investors'], ', ')
    results = c(results, top5)
  }
  results <- Reduce(c, results)
  invTable <- table(results)

  return(invTable[order(invTable, decreasing=TRUE)][1:10])
}

successInvestors <- getInvestorList(success)
```

```
failedInvestors <- getInvestorList(failed)

# Table plot
data.frame(successInvestors) %>%
  gt() %>%
  tab_header(
    title="Successful"
  ) %>%
  opt_align_table_header(align = "left") %>%
  cols_label(
    results = "Investor",
    Freq = "Count"
  ) %>%
  cols_align(align="left")
```

Successful

Investor	Count
Y Combinator	13
Plug and Play	9
Digital Currency Group	8
Alumni Ventures	6
Andreessen Horowitz	5
FundersClub	5
500 Global	4
Blockchain Capital	4
QED Investors	4
SixThirty	4

```
# Table plot
data.frame(failedInvestors) %>%
  gt() %>%
  tab_header(
    title="Failed"
  ) %>%
  opt_align_table_header(align = "left") %>%
  cols_label(
    results = "Investor",
    Freq = "Count"
  ) %>%
  cols_align(align="left")
```

Failed

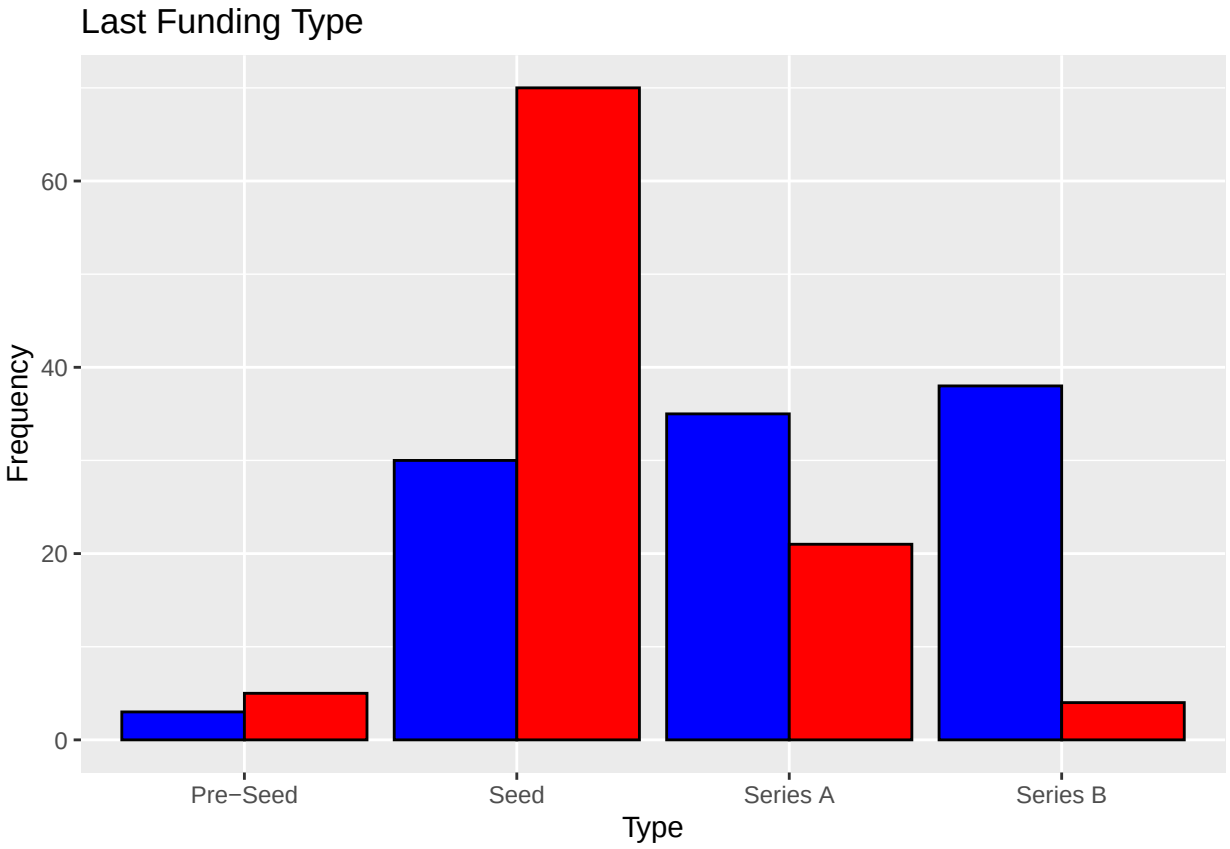
Investor	Count
Techstars	17
500 Global	9
Y Combinator	9
Right Side Capital Management	7
Techstars New York City Accelerator	4
DCVC	3

Draper Associates	3
GV	3
Khosla Ventures	3
Accel	2

These two tables showcase the most frequent investors in both successful and failed companies. Startup incubators such as Y Combinator, Techstars, and 500 Global invested in the most failed startups, as accelerators tend to invest in a larger amount of companies compared to the average venture capital firm. However, we can see that Y Combinator invests in the most succesful companies.

Funding Rounds

```
companies$LastFundingType <- as.factor(companies$LastFundingType)
ggplot(companies, aes(x=LastFundingType, fill=ifelse(Failed == 0, 'blue', 'red'))) +
  geom_histogram(binwidth=1, colour="black", position="dodge", stat = "count") +
  scale_fill_identity() +
  labs(title="Last Funding Type",
       x ="Type", y = "Frequency")
```

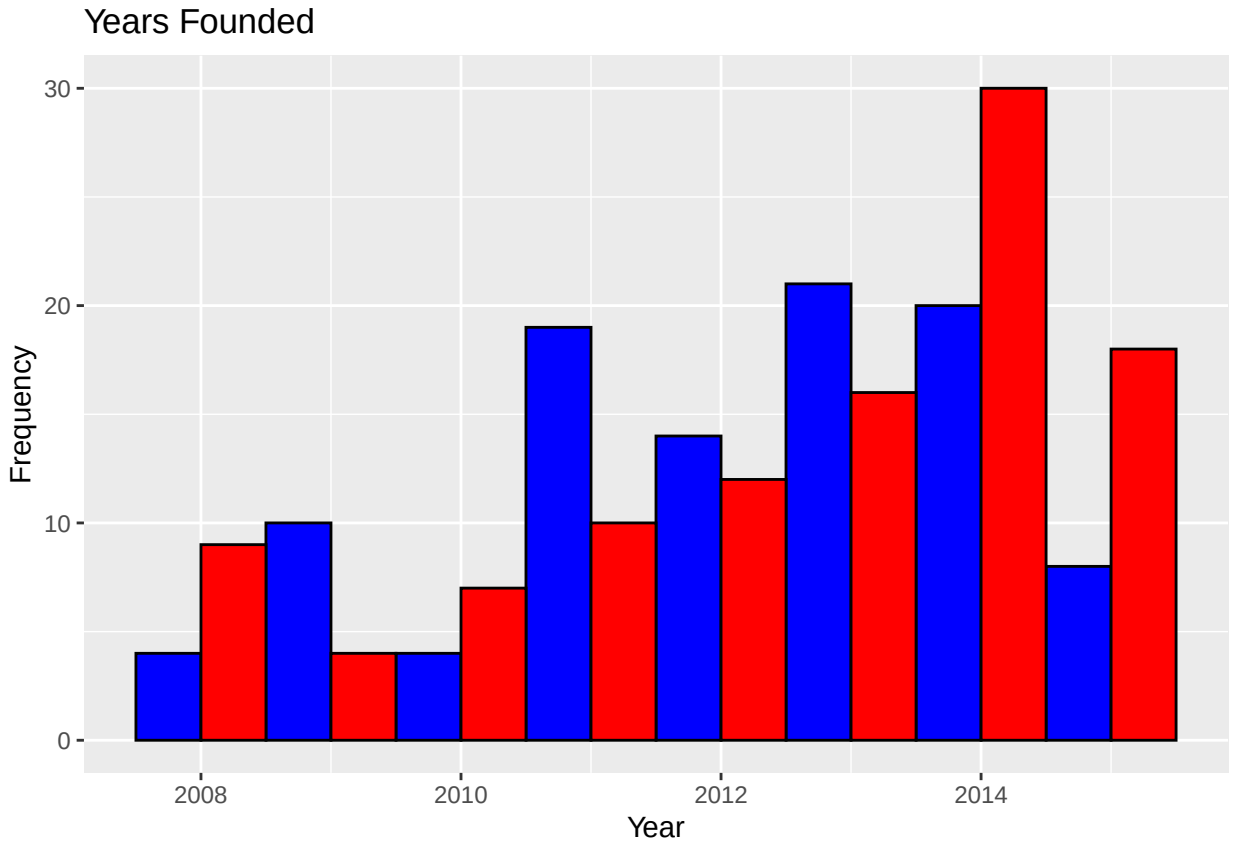


From the barplot above, we can see the distribution of the latest funding round, separated by failure (red) and success (blue). We note that most failed startups are in earlier stages (pre-seed and seed), whereas most succesful startups are in later stages (Series A and Series B). This would indicate to venture capital firms that companies in later rounds are lower risk investments, but also lower reward investments.

Year founded

```
ggplot(companies, aes(x=FoundedDate, fill=ifelse(Failed == 0, 'red', 'blue'))) +
  geom_histogram(binwidth=1, colour="black", position="dodge") +
```

```
  scale_fill_identity() +
  labs(title="Years Founded",
       x ="Year", y = "Frequency")
```



From the barplot above, we can see that a smaller amount of companies received funding during the 2008 recession because VC firms tend to be more selective in periods of economic distress. We also note that there is a large increase in the number of companies founded in the aftermath of the recession.

Multi-time founders

```
getFounderList <- function(d) {
  results <- c()
  for (i in 1:nrow(d)) {
    top5 <- str_split(d[i,'Founders'], ', ')
    results = c(results, top5)
  }
  results <- Reduce(c, results)
  results <- table(results)
  results <- results[order(results, decreasing=TRUE)]
  results <- data.frame(cbind(results))
  results$name <- rownames(results)
  return(results)
}

successFounders <- getFounderList(success)
failFounders <- getFounderList(failed)

mergedFounderList <- merge(successFounders, failFounders, by="name")
```

```r
colnames(mergedFounderList) <- c("Name", "Successful", "Failed")

data.frame(table(mergedFounderList)) %>%
  select(Name, Successful, Failed) %>%
  gt() %>%
  tab_header(
    title="Repeat Founders"
  ) %>%
  cols_align(align="left")
```

Repeat Founders

Name	Successful	Failed
Joao Abiul Menano	1	1
Sam Hopkins	1	1
Steven Muszynski	1	1

```r
# Find founders who've founded multiple successful or failed companies
multSuccessFounders <- subset(successFounders, successFounders$results > 1)
multFailFounders <- subset(failFounders, failFounders$results > 1)
colnames(multFailFounders) <- c("Frequency", "Name")
data.frame(table(multFailFounders)) %>%
  select(Name, Frequency) %>%
  gt() %>%
  tab_header("Repeat Failed Founders") %>%
  cols_align(align="left")
```

Repeat Failed Founders

Name	Frequency
Patrick Hosty	2

The two tables above show people that have founded multiple companies. We see that there are three founders who have started both failed and successful companies and that there is only one person who has founded two failed companies. Since this data is sparse, we cannot draw any meaningful conclusions.

Text Analysis

Word Cloud of Company's (Full) Description

```r
# Word cloud function
wc <- function(dtm) {
  matrix <- as.matrix(dtm)
  words <- sort(rowSums(matrix),decreasing=TRUE)
  df <- data.frame(word = names(words),freq=words)
  txt <- df
  set.seed(1000)
  wordcloud(words = df$word, freq = df$freq, min.freq = 1,
            max.words=200, random.order=FALSE, rot.per=0.35,
            colors=brewer.pal(8, "Dark2"))
}
```

```r
# Inverse document frequency function
idf <- function(dtm, og.data) {
  dtm.idf <- weightTfIdf(dtm)
  tf_matrix <- data.matrix(dtm.idf)
  tf_frame <- data.frame(tf_matrix)
  tf_frame$word <- rownames(tf_frame)
  important.words = data.frame(matrix(nrow=0, ncol=6))
  colnames(important.words) <- c('Name','1', '2', '3', '4', '5')
  for (i in 1:(ncol(tf_frame) - 1)) {
    c.name <- paste(c("X", i), collapse = "")
    description <- tf_frame[, c(c.name, 'word')]
    description <- description[order(description[,1], decreasing=TRUE), ]
    important.words[nrow(important.words) + 1, 2:6] <- description$word[1:5]
    important.words[nrow(important.words), 1] <- og.data[i, 'Name']
  }
  return(important.words)
}

main <- function(d) {
  dtm <- TermDocumentMatrix(d$FullDescription,control = list(
    removePunctuation = TRUE, stopwords = TRUE))
  wc(dtm = dtm)
  return(idf(dtm = dtm, og.data = d))
}

fail.words <- main(failed) # Returns IDF matrix
```

```
success.words <- main(success) # Returns IDF matrix
```

This data shows a word cloud of the company's full description, seperated by successful and failed companies. While we foresaw this data to be potentially useful for our analysis, both word clouds ended up looking indistinguishable from one another.

IDF Matrix

```
head(success.words) %>%
  gt() %>%
  tab_header("Successful")
```

Successful

Name	1	2	3	4	5
Wefunder	amount	crowd	disruptive	dollars	hundred
R3	dlt	collaboration	regulated	markets	trust
Ripple	ripple	competitive	correspondent	counterparty	crosscurrency
Synapse	synapsefi	fintech	banking	deposit	innovations
Ripio	bitcoins	argentina	bitpagos	brazil	chile
BitPay	bitpay	payment	bitcoin	api	support

```
head(fail.words) %>%
  gt() %>%
  tab_header("Failed")
```

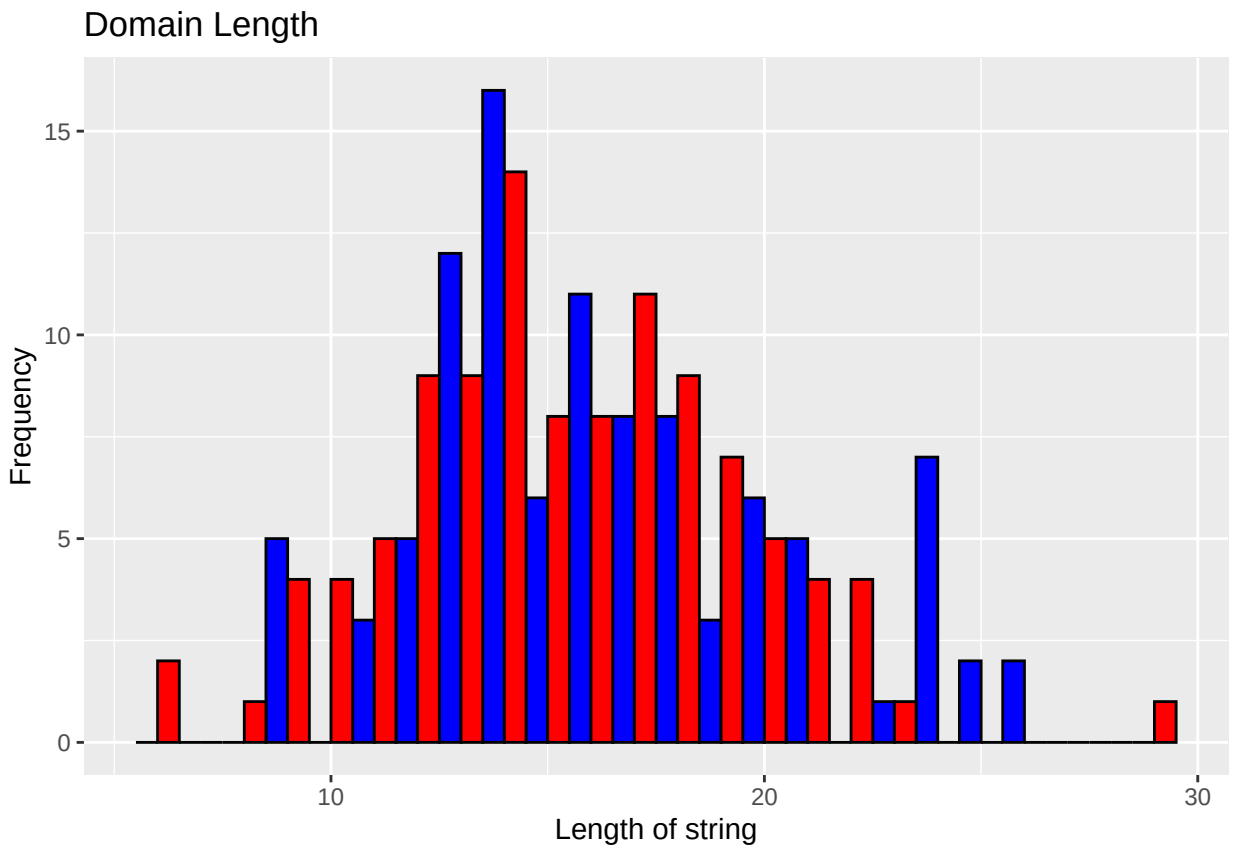
Failed

Name	1	2	3	4	5
Neighborly	broadband	infrastructure	economic	fiber	neighborly
Crowdfunder	crowdfunder	earlystage	index	ventures	fund
Instavest	instavest	—	share	alongside	arise
qplum	accounts	team	investing	investment	finance
Kapitall	kapitall	generation	investing	ideas	llc
Clara Lending	started	clara	empowered	families	life's

This data shows an IDF matrix of each company's "full description" column. The IDF matrix essentially tells us the most unique words in each company's description. While we thought this data might include interesting insights into a company's value proposition, we found it difficult to parse, with most companies looking relatively homogeneous.

Website Domain Length

```
ggplot(companies, aes(x=domain.length, fill=ifelse(Failed == 0, 'red', 'blue'))) +
  geom_histogram(binwidth=1, colour="black", position="dodge") +
  scale_fill_identity() +
  labs(title="Domain Length",
       x ="Length of string", y = "Frequency")
```

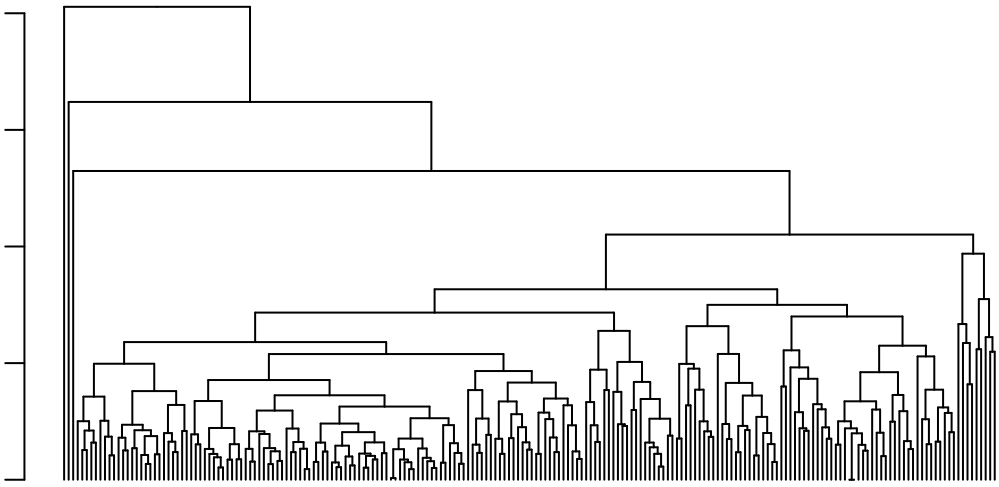


This histogram shows the distribution of the each company's website domain length, separated between successful and failed. This plot seems to indicate that a website's domain is not a significant predictor of success; however, further statistical analysis will need to be conducted to prove this.

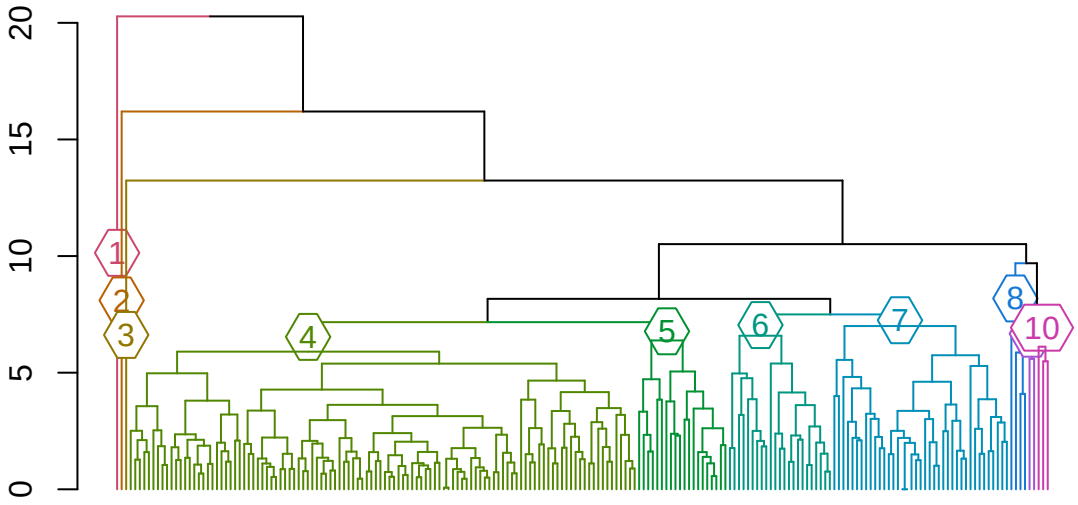
Dendrogram

```
# take only relevant numeric variables
data_numeric <- companies[,c(7,8,9,10,12,13,15,16,17,18,20,24,25)]

# scale the data
data_scaled = scale(data_numeric, center=TRUE, scale=TRUE)

# create distance matrix
data_distance_matrix = dist(data_scaled, method='euclidean')

# create hcluster
```

```
hier_data = hclust(data_distance_matrix, method='complete')
names <- companies$Name[hier_data$order]
hier_data <- as.dendrogram(hier_data)

# Create a basic dendrogram to visualize the groupings
labels(hier_data) <- ""
plot(hier_data, cex=0.5, labels = FALSE)
```



```
# Make labels blank
labels(hier_data) <- ""

# Color code the dendrogram based on groups
dend_h <- heights_per_k.dendrogram(hier_data)
par(mfrow = c(1,1))
hier_data <- color_branches(hier_data, k = 10, groupLabels = TRUE)

# Plot the dendrogram
plot(hier_data)
```




```
# A cut dendrogram
hier_data %>% plot(ylim = c(3, 20))
```


```
# Returns first 5 names of companies in given sub-cluster
labels(hier_data) <- names

cluster1 = cutree(hier_data, k=10)

# How many companies are in each group/cluster?
summary(factor(cluster1))
```

```
##   1   2   3   4   5   6   7   8   9  10
##  20   4   1  39   2  23 112   3   1   1
```

```
# Function to print the first five companies of every cluster, if there are
# even 5 companies
clustSummary <- function(cluster, n) {
  return (head(names(which(cluster == n))))
}

# Cluster's 1 - 5 are successful companies
clustSummary(cluster1, 1)
```

```
## [1] "Wefunder"    "EquityZen"   "blooom"      "Modo"        "Crowdfunder"
## [6] "Edge"
```

```
clustSummary(cluster1, 2)
```

```
## [1] "R3"             "BitPay"         "Ellevest"       "Finix Payments"
```

```
clustSummary(cluster1, 3)
```

```
## [1] "Ripple"
```

```
clustSummary(cluster1, 4)
```

```
## [1] "Synapse" "Propel"  "Ascent"  "Gem"     "Okcoin"  "Coinme"
```

```
clustSummary(cluster1, 5)
```

```
## [1] "Ripio"      "Stocktwits"
```

```
# Clusters 6, 7, and 8 are borderline companies (borderline of success / fail)
clustSummary(cluster1, 6)
```

```
## [1] "LedgerX"    "Sentieo"    "Neighborly" "Riskalyze"  "Prism"
## [6] "BehavioSec"
```

```
clustSummary(cluster1, 7)
```

```
## [1] "Thinknum"        "Savana"          "BillingPlatform" "Linqto"
## [5] "QuantConnect"    "MovoCash"
```

```
clustSummary(cluster1, 8)
```

```
## [1] "SimpleNexus"    "Even Financial" "LendEDU.com"
```

```
clustSummary(cluster1, 9)
```

```
## [1] "Fintiv"
```

```
clustSummary(cluster1, 10)
```

```
## [1] "Venmo"
```

The dendrograms above shows the relationship between the companies in our dataset. We started by removing all factor variables and the numeric variables that were obvious predictor sof successful and failed companies.

We sorted the dendrogram into 10 groups and noted the commonalites in each group.

We note that groups 1 - 6 contain startups that were more succesful, with the margin of success decreasing as the group number increases.

Groups 7 contains a majority of the failed companies, and some successful companies that had metrics similar to the failed companies (low funding amounts, few articles written, etc.)

Groups 8 - 10 contain successful companies that are different from the companies in groups 1 - 6. For example, Venmo, located in group 10, has by far the highest number of articles written.

PCA

We conducted a principle-component-analysis to reduce the dimensionality in our data set and have a more intuitive statistical summary.

The first PCA examines how each company is related to one another. This means that the loading scores of each individual PC tells us how heavy each variable (e.g. number of founders, LinkedIn followers, etc.) is weighted. We then conduct a second PCA exploring the opposite relationship (this will be discussed in further depth below).

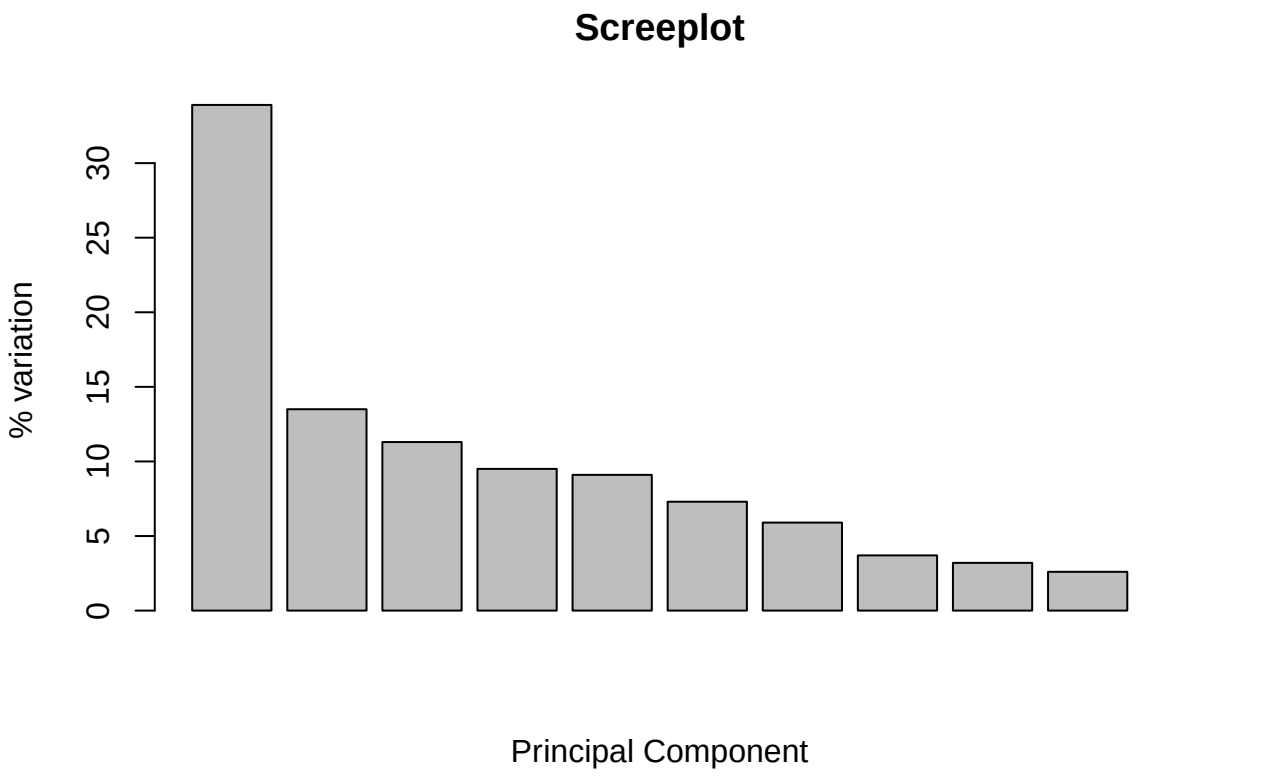
Setup

```
PCA.data <- companies
row.names(PCA.data) <- PCA.data$Name

PCA.data <- subset(PCA.data, select=-c(Name, LastFundingType,
                                       Website, Description, Operating.Status,
                                       Founders, Top5Investors, HeadquartersRegion,
                                       ClosedDate, FullDescription, Failed,
                                       DiversitySpotlight, WasAcquired, CB.Rank,
                                       TotalFundingAmt))


# Function for plotting on GG plot. Takes in data and PCs(int) to plot
pca.gg <- function(d, n1, n2, onCompanies) {
  # Variation
  p.var <- d$sdev^2
```

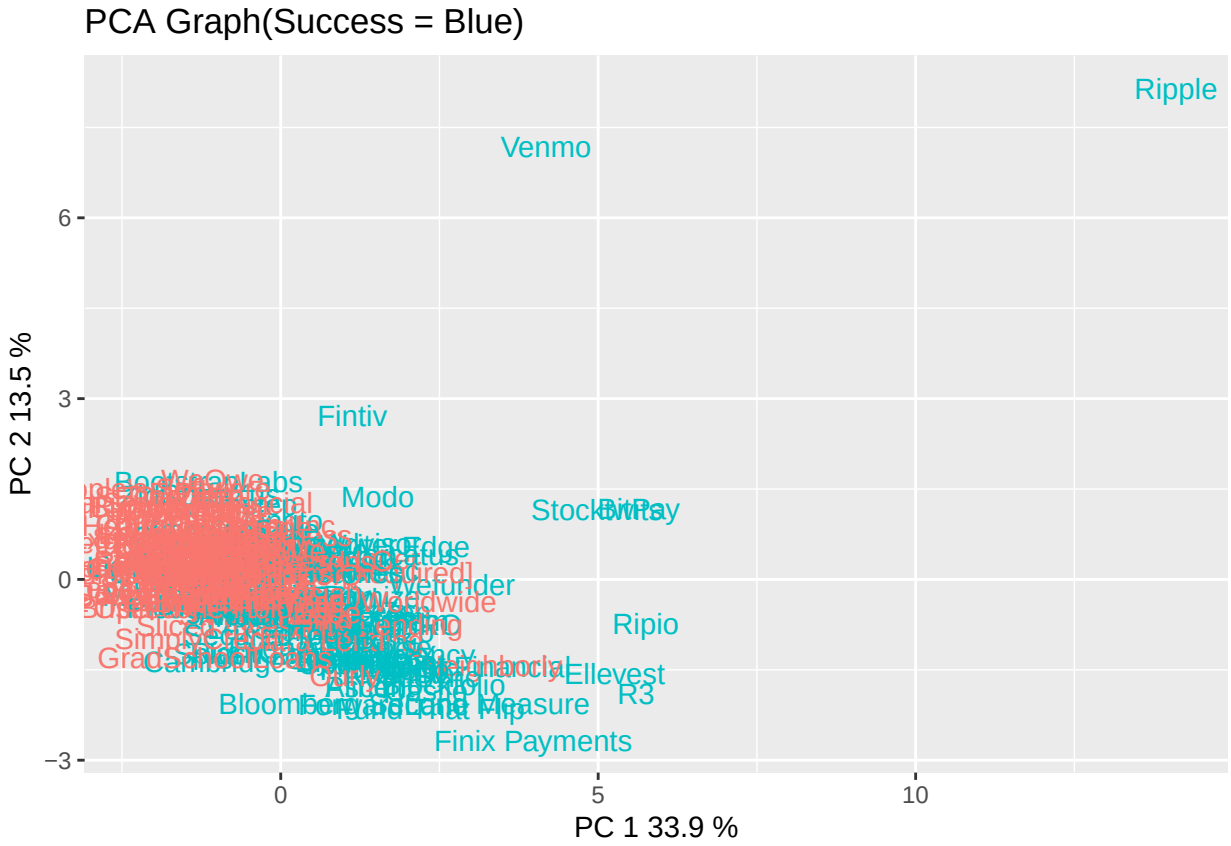
```
  p.var.per <- round(p.var / sum(p.var)*100, 1)

  if (onCompanies) {
    # For coloring
    successSubData <- subset(companies, Failed == 0)$Name
    # Big plot
    ggD <- data.frame(Sample=rownames(d$x),
                      X=d$x[,n1],
                      Y=d$x[,n2],
                      Success = rownames(d$x) %in% successSubData
    )
  } else {
    ggD <- data.frame(Sample=rownames(d$x),
                      X=d$x[,n1],
                      Y=d$x[,n2]
    )
  }


  if (onCompanies) {
    ggplot(data=ggD, aes(x=X, y=Y, label=Sample)) +
      geom_text(aes(colour = ifelse(Success, 'Red', 'Blue')), show.legend = FALSE) +
      xlab(paste("PC", n1, p.var.per[n1], "%", sep = " ")) +
      ylab(paste("PC", n2, p.var.per[n2], "%", sep=" ")) +
      ggtitle("PCA Graph(Success = Blue)")
  } else {
    ggplot(data=ggD, aes(x=X, y=Y, label=Sample)) +
      geom_text() +
      xlab(paste("PC", n1, p.var.per[n1], "%", sep = " ")) +
      ylab(paste("PC", n2, p.var.per[n2], "%", sep=" ")) +
      ggtitle("PCA Graph")
  }
}

# Loading scores
load_scores <- function(d, n) {
  loading_scores <- d$rotation[, n]
  col_scores <- abs(loading_scores)
  c_ranked <- sort(col_scores, decreasing = TRUE)
  top_5 <- names(c_ranked[1:5])
  return(d$rotation[top_5, 1])
}

# Scree plot
s_plot <- function(d){
  pca.var <- d$sdev^2
  pca.var.per <- round(pca.var/sum(pca.var)*100, 1)
  barplot(pca.var.per, main="Screeplot", xlab="Principal Component",
          ylab="% variation")
}
```

Company Analysis

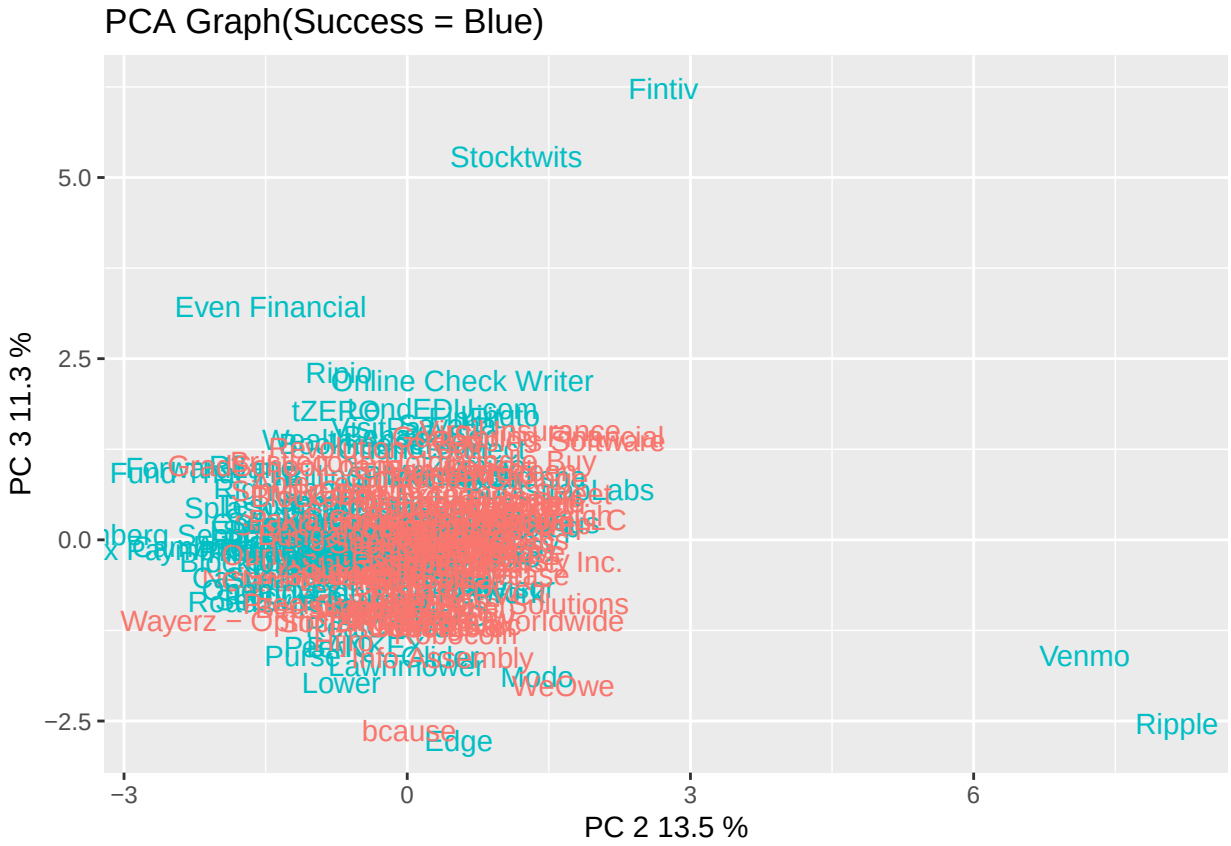
```r
# PC1
PCA1 <- prcomp(PCA.data, scale=TRUE)

# Scree plot
s_plot(PCA1)
```




```r
# Plot
pca.gg(PCA1, 1, 2, TRUE)
```




```
pca.gg(PCA1, 2, 3, TRUE)
```




```
# Which variables were most influential on where the companies were
# plotted for PC1(x-axis?)
load_scores(PCA1, 1)
```

```
##    NumInvestors       NumRounds        NumLeads    LI.Followers ActiveTechCount
##       0.4471584       0.4420980       0.3875268       0.3867992       0.3290921
```

```
# Which variables were most influential on where the companies were
# plotted for PC2(y-axis?)
load_scores(PCA1, 2)
```

```
##     FoundedDate     NumArticles    LI.Followers        NumLeads ActiveTechCount
##      0.07434462      0.31899158      0.38679916      0.38752680      0.32909209
```

```
load_scores(PCA1, 3)
```

```
##     NumFounders NumAcquisitions     FoundedDate   domain.length ActiveTechCount
##      0.13144411      0.19280390      0.07434462     -0.18647473      0.32909209
```

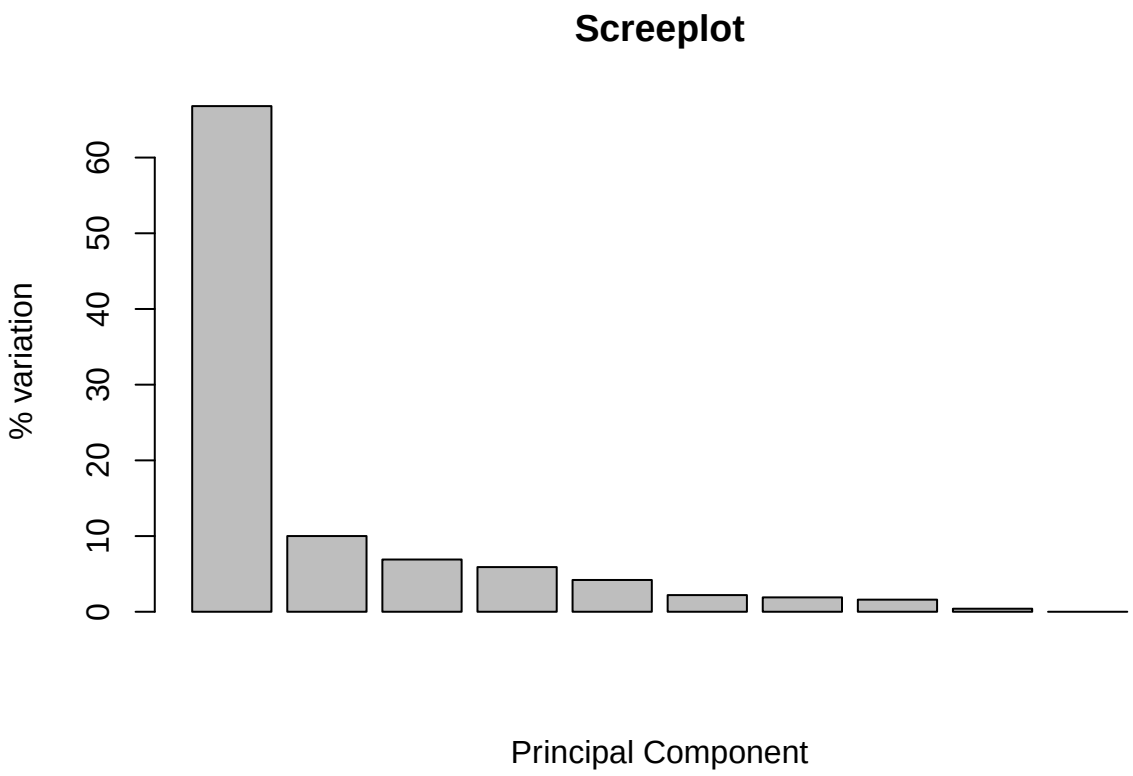
Looking at PC1, it seems that successful companies tend to be directed more towards the right. If we analyze this PC's loading scores, it seems that number of investors, number of rounds, number of lead investors, LinkedIn followers, and active tech count contribute to a company's success. One particular variable that we found interesting was the number of leads. This is notable because most companies, regardless of their status, tend to have one lead. Investments with multiple leads, however, tend to be more successful. This is likely because the lead investor is doing substantial research on the startup, so having more leads results higher quality due-diligence.

CrunchBase Variable Analysis

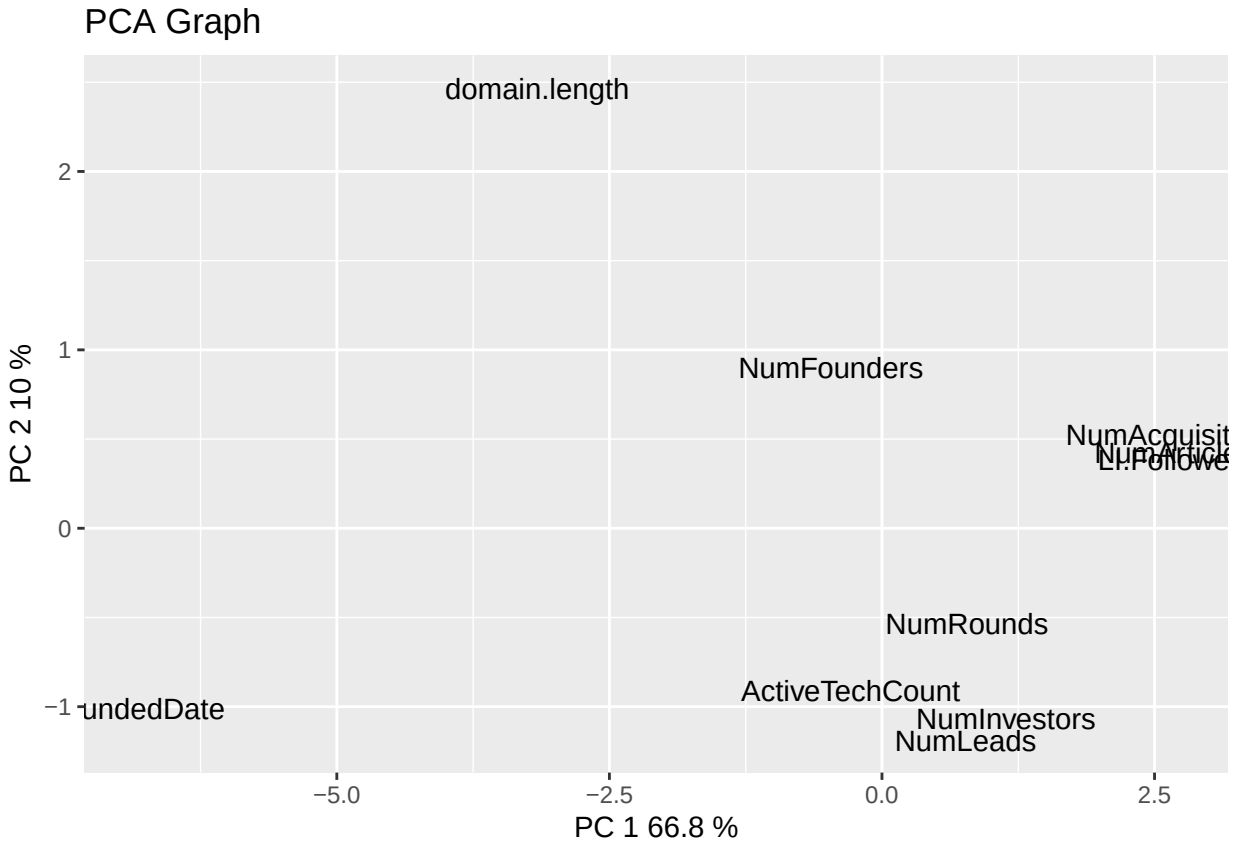
In the second PCA, we transposed our data, which allowed us to examine the relationship between each variable of the company data set. The loading scores in this PCA were not as intuitive, as it told us the

companies that had the most significant weight when determining the variable's PC value.

```
# Transposes our data, such that we can perform PCA analysis on the company variables themselves
PCA.data2 <- t(data.matrix(PCA.data))

# Normalizing each row s.t. we get skewed PCs
# In-line function subtracts the minimum from each column &
# divides by the difference between maximum and minimum
PCA.data2 <- t(apply(PCA.data2, 1, function(x)(x-min(x))/(max(x)-min(x))))
PCA2 <- prcomp(PCA.data2)

# Scree plot
s_plot(PCA2)
```




```
# Graph
pca.gg(PCA2, 1, 2, FALSE)
```




```
# What companies were most influential on where the variables were plotted for PC1?
load_scores(PCA2, 1)
```

```
##      GradSchoolLoans Cambridge Blockchain         SimplyCredit
##           -0.1171376           -0.1151948           -0.1124733
##     SocialMatters.ai     OpenDoor Trading
##           -0.1102746           -0.1066377
```

```
# PC2?
load_scores(PCA2, 2)
```

```
##             R3 Finix Payments       Ellevest         Castle            Gem
##    -0.04060651    -0.06835720    -0.05959453    -0.07819433    -0.04183070
```

Looking at the graph of PC1 and PC2, we can see how similar each predictor is to each other. Some of the notable similarities are Number of Articles and LinkedIn followers. This similarity can be defined as the "social/hype" factor of the company, and is a good indicator of how well-known a startup is. The second notable similarity is between NumInvestors, NumLeads, and NumRounds. This similarity can be defined as the "investment" factor of the company, and indicates that the three predictors tell you similar information about the company.

Lasso Regression

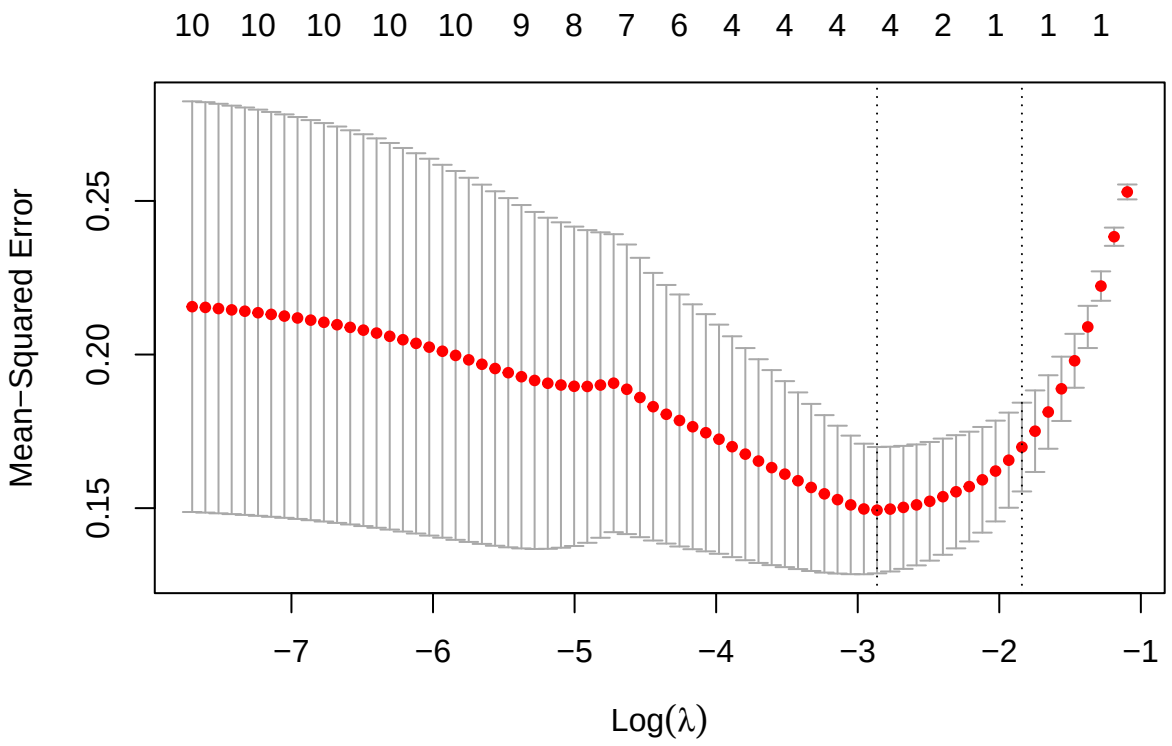
We decided to perform a LASSO model because it would give us insight into which variables were most important in the company data set. This would, hypothetically, help a VC separate the signal from the noise when analyzing a startup.

Moreover, the Lasso model adds a penalty term, which effectively makes it a more robust model than our previous model, the logistic regression. This felt especially relevant given that we were working within such an uncertain industry such as venture capital.

Lastly, we preferred to use Lasso over Ridge because Lasso was able to reduce a variable's weight completely down to zero.

```
lassoReg <- function(colsToRemove) {
  trainingData <- companies[sample(nrow(companies), nrow(companies) / 2), ]
  X <- trainingData[!(names(trainingData) %in% colsToRemove)]

  X <- data.matrix(X)
  Y <- trainingData$Failed

  # Finding lambda value
  cross_val_model <- cv.glmnet(X, Y, alpha=1)

  best_lambda <- cross_val_model$lambda.min
  print(best_lambda)

  #produce plot of test MSE by lambda value
  plot(cross_val_model)
  best_model <- glmnet(X, Y, alpha = 1, lambda = best_lambda)

  y_pred <- predict(best_model, s = best_lambda, newx = X)

  #find SST and SSE
  sst <- sum((Y - mean(Y))^2)
  sse <- sum((y_pred - Y)^2)

  #find R-Squared
  rsq <- 1 - sse/sst
  rsq

  # Best model explains {rsq * 100}% of the variation in the response values
  # of the training data
  # Rsq seems to vary between 50 and 65%. Somewhat low but...

  # Which factors are most important?
  # Extracts non-zero coefficients
  print(coef(best_model))
}


colsToRemove <- c("Failed", "Description", "FullDescription",
                  "ClosedDate", "Operating.Status",
                  "WasAcquired", "Founders", "Name", "Website",
                  "Top5Investors", "HeadquartersRegion", "CB.Rank",
                  "NumRounds", "LastFundingType")

#Console output are the non-zero(i.e. "significant" variables)
# Graph output shows optimal lambda value
lassoReg(colsToRemove)
```

```
## [1] 0.05706993
```




```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                s0
## (Intercept)         7.992094e-01
## FoundedDate         .
## NumArticles        -1.427753e-05
## NumFounders         .
## TotalFundingAmt    -3.202982e-10
## NumLeads           -1.718038e-02
## NumInvestors        .
## NumAcquisitions     .
## ActiveTechCount    -8.757926e-03
## DiversitySpotlight  .
## LI.Followers        .
## domain.length       .
```

The first line of output is the optimal Lambda value produced by the glmnet function. This value gives us the "best" model that still lies within one standard error of the *optimal* Lambda value.

The Lasso model shows that the company's active tech count and number of lead investors had the most significant impact on predicting a company's success. Surprisingly, the total funding amount and number of articles were considered to be not as important as the former variables.

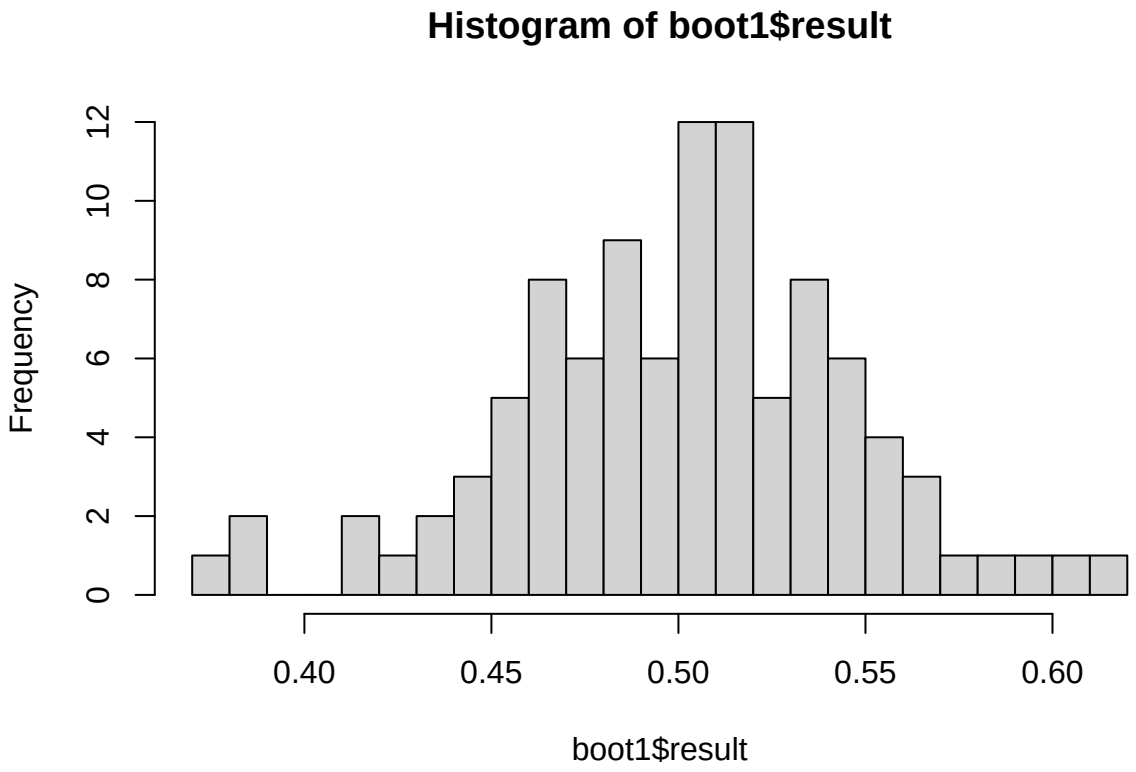
While the Lasso dramatically reduces the complexity(i.e. the *variance*) of our predictive model, but what is the cost?

To answer this question & get a more accurate measure of our the error of our Lasso model, we decided to conduct a bootstrap simulation to see the average error of our predictive model.

Bootstrap

```
# This is a simplified version of the lassoReg function defined above.
boot <- function(colsToRem) {
  # Random sample - make training data
```

```
  trainingData <- companies[sample(nrow(companies), nrow(companies) / 2), ]
  X <- trainingData[!(names(trainingData) %in% colsToRemove)]
  X <- data.matrix(X)
  Y <- trainingData$Failed

  # Finding lambda value
  cross_val_model <- cv.glmnet(X, Y, alpha=1)
  best_lambda <- cross_val_model$lambda.min
  best_model <- glmnet(X, Y, alpha = 1, lambda = best_lambda)
  y_pred <- predict(best_model, s = best_lambda, newx = X)

  # Random sample - make testing data
  testingData <- companies[sample(nrow(companies), nrow(companies) / 2), ]
  testX <- testingData[!(names(testingData) %in% colsToRemove)]
  Y <- testingData$Failed

  # Testing
  testingData$Prediction <- round(predict(best_model, s = best_lambda, newx = X))

  # Accuracy
  misPredictedRatio <- nrow(subset(testingData, (Prediction != Failed))) / nrow(testingData)
  return(misPredictedRatio)
}

# Run 100 times
boot1 = do(100)*{
  boot(colsToRemove)
}

# Summary statistics
hist(boot1$result, 30, title="Mispredicted Ratio for Bootstrap")
```




```
mean(boot1$result)
```

```
## [1] 0.5008738
```

```
sd(boot1$result)
```

```
## [1] 0.04411698
```

After conducting the bootstrap, we found that the Lasso model *mis*-predicted the outcome of our testing data about 50% of the time. While this number is much higher than the model's error on our training data, it is still significantly lower than the aforementioned high failure-rates of most VC firms.

Conclusion

After conducting all the previous experiments, it appears as though some of the most important predictors of a startups' success are its active tech count and the number of lead investors the company has.

This finding was supported by the first principle-component-analysis we conducted, and also the Lasso regression model. One other interesting insight we discovered in the company data was the relationship between each variable we measured our companies on. These relationships can be useful for VCs to contextualize large swaths of data about a company.

Nonetheless, we found many variables in our data set to not be practically significant, such as its location, funding year, and textual description.

References

Clark, G. (2007). Genetically Capitalist? The Malthu- sian Era, Institutions and the For- mation of Modern Preferences. https://faculty.econ.ucdavis.edu/faculty/gclark/papers/Capitalism%20Genes.pdf

Dean, T. (2017, June 1). The meeting that showed me the truth about VCs. TechCrunch. https://techcrunch.com/2017/06/01/the-meeting-that-showed-me-the-truth-about-vcs/

Ghosh, S. (2012, December 10). Why Most Venture-Backed Companies Fail - News - Harvard Business School. Www.hbs.edu. https://www.hbs.edu/news/Pages/item.aspx?num=214

Gompers, P. A., Gornall, W., Kaplan, S. N., & Strebulaev, I. A. (2016). How Do Venture Capitalists Make Decisions? SSRN Electronic Journal. https://doi.org/10.2139/ssrn.2801385

Kelly, B. T., Papanikolaou, D., Seru, A., & Taddy, M. (2018). Measuring Technological Innovation over the Long Run. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.3279254

Krishna, A., Agrawal, A., & Choudhary, A. (2016, December 1). Predicting the Outcome of Startups: Less Failure, More Success. IEEE Xplore. https://doi.org/10.1109/ICDMW.2016.0118

Okrah, J., Nepp, A., & Agbozo, E. (2018). Exploring the factors of startup success and growth. The Business and Management Review, 9. https://cberuk.com/cdn/conference_proceedings/2019-07-14-09-58-17-AM.pdf

van Gelderen, M., Thurik, R., & Bosma, N. (2006). Success and Risk Factors in the Pre-Startup Phase. Small Business Economics, 26(4), 319–335. https://doi.org/10.1007/s11187-004-6837-5