Angel Returns 2007 - 17

Carlee Price

October 31, 2017

A look at returns in the Angel space, 2007 - 2017

We've captured data from companies in the Crunchbase dataset that are US-based, were first funded with either Seed or Angel capital totalling at least $100k between Jan 01, 2007 and Sept 26, 2017. There are 6,727 rows and 19 columns. The data was processed and cleaned as a separate project, which can be seen here.

Conclusions in this report rely on a "best efforts" approach around assumptions and are always subject to change. Assumptions will always be stated and logically supported. Without these assumptions, there are no conclusions.

Here we address some data-quality issues that emerge as the work progresses.

```
#aardvark was acquired by Google in 2010 for $50Mm (verified by TechCrunch)
data$Price[which(data$Company.Name == "Aardvark")] <- 50000000
```

First a very simple, top-level view of the information.

```
table1 <- as.data.frame(table(Status))
transform(table1, Relative = prop.table(Freq))
```

```
##         Status Freq    Relative
## 1       Closed  291 0.043258510
## 2          IPO   13 0.001932511
## 3    Operating 5617 0.834993311
## 4 Was Acquired  806 0.119815668
```

Of the companies in this dataset, 4.3% were closed during the study period, 0.2% went public, 12.0% were acquired, and 83.5% were still operating at the end of the period. We can also look at the mean age for the companies that fell into each bucket.

```
ddply(data, .(Status), summarize, MeanAge=mean(Age))
```

```
##         Status   MeanAge
## 1       Closed  783.8969
## 2          IPO 1849.1538
## 3    Operating 1095.1763
## 4 Was Acquired 1016.3846
```

Some of these companies may simply have not "baked" long enough to have reached their ultimate outcome. If we subset for companies that were first funded 1,016 days before our end date we might get different figures. % of companies operating should fall.

```
data$First.Funding <- as.Date(data$First.Funding)
data.baked <- subset(data, First.Funding < '2015-12-14')
table2 <- as.data.frame(table(data.baked$Status))
transform(table2, Relative = prop.table(Freq))
```

```
##           Var1 Freq    Relative
## 1       Closed  287 0.060921248
## 2          IPO   13 0.002759499
## 3    Operating 3616 0.767565273
## 4 Was Acquired  795 0.168753980
```

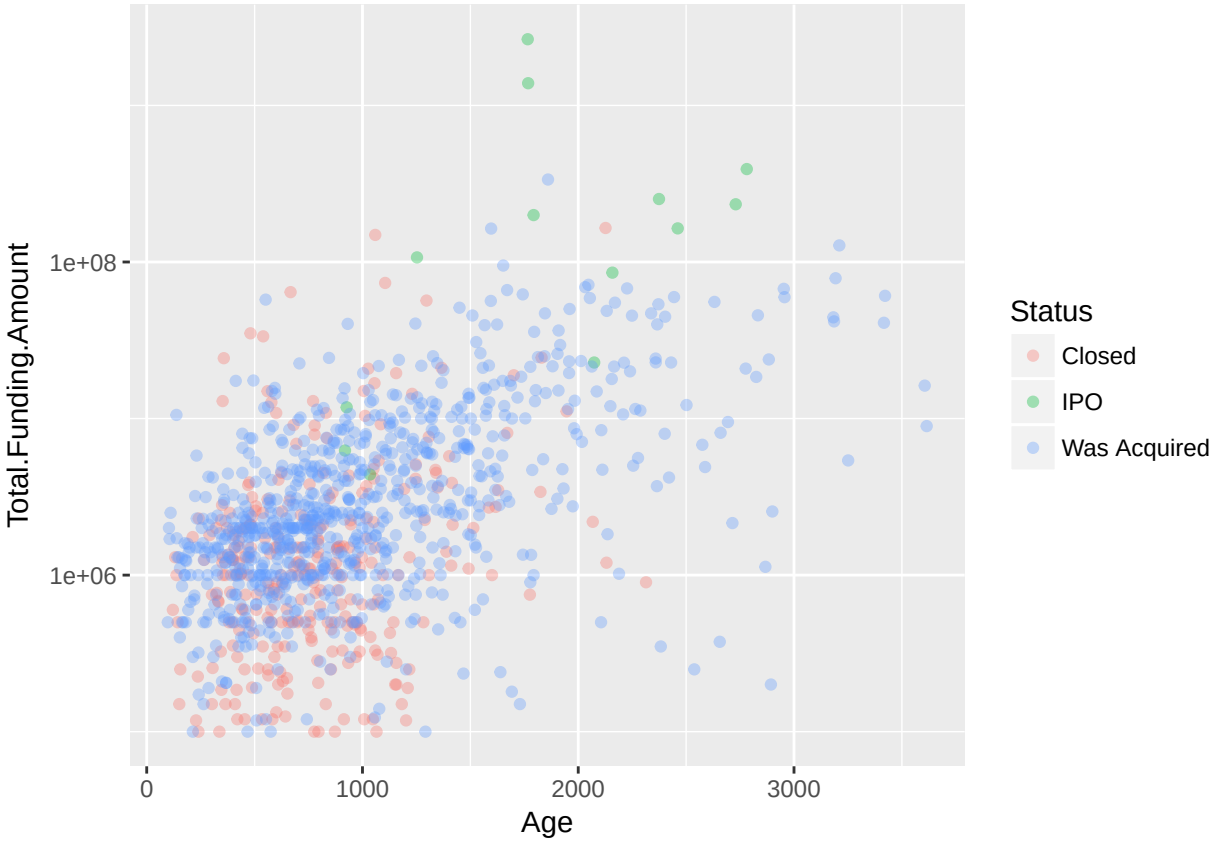
Indeed we see that the % of companies listed as Operating is lower here (76.8%) and Closeds (6.1%), Acquireds (16.9%) and IPOs (0.3%) are all higher.

Let's have a look graphically at the evolution of these companies over time. Age is on the x-axis, Total Funding amounts (log transformed) on the Y-axis. Point colour reflects the status of each company.

```
ggplot(data, aes(x = Age, y = Total.Funding.Amount, col = Status, alpha = 0.1)) + geom_jitter(alpha = 0
```

Let's strip the Operating companies out and have another look.

```
data3 <- subset(data, Status != "Operating")
ggplot(data3, aes(x = Age, y = Total.Funding.Amount, col = Status, alpha = 0.1)) + geom_jitter(alpha = 
```



We can see another interesting relationship here. The companies that executed an IPO raised SIGNIFICANTLY more on average than the other companies. Obvious, but still interesting to see the numbers: $424.2Mm on average.

```
ddply(data, .(Status), summarize, MeanTotFunding=mean(Total.Funding.Amount), SDFunding = sd(Total.Fundi
```

```
##         Status MeanTotFunding SDFunding
## 1       Closed        4273314  14990660
## 2          IPO      424194783 761287225
## 3    Operating       14221923 218774340
## 4 Was Acquired        7367820  17907494
```

There also appears to be a fairly reliable relationship between the passage of time and the total amount raised. Let's take a closer look.

```
datamod1 <- lm(Total.Funding.Amount ~ Age, data = data)
summary(datamod1)
```

```
##
## Call:
## lm(formula = Total.Funding.Amount ~ Age, data = data)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -7.811e+07 -1.658e+07 -4.478e+06  4.308e+06  1.151e+10
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) -11407831    4277207  -2.667  0.00767 **
## Age             23442       3252   7.209 6.23e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 202700000 on 6725 degrees of freedom
## Multiple R-squared:  0.007669,   Adjusted R-squared:  0.007522
## F-statistic: 51.98 on 1 and 6725 DF,  p-value: 6.234e-13
```

The slope of this line is highy significant (p is tiny), but is not full explanatory (R-squared is small). The model suggests that companies (on the whole) consume 23k in raised funds each day they're in operation, but that there are a host of other factors that affect the amount they raise aside from the passage of time.

Let's take a high-level view of this dataset. We know these companies (6,727 of them) raised $92Tn in the study period. How did activiy change through the study period? A histogram tells us that the number of companies seeking funding increased yearwise (note that 2017 is an incomplete year). We can also see that aggregate amounts raised decreased after 2011. So the average total raised (remembering that the Year here is the first year of funding, and that the total raise would include subsequent years) has been falling.

Another quick note in regard to the number of companies funded: steady and marked increase from 2007 onward reflects an increase in activity but *may also reflect characteristics of the platform from which the data was gathered*. If Crunchbase became more active during this period, and better at collecting data (which anectdotal data suggest they did) these numbers would have increased disproportionately to the increase in real activity.

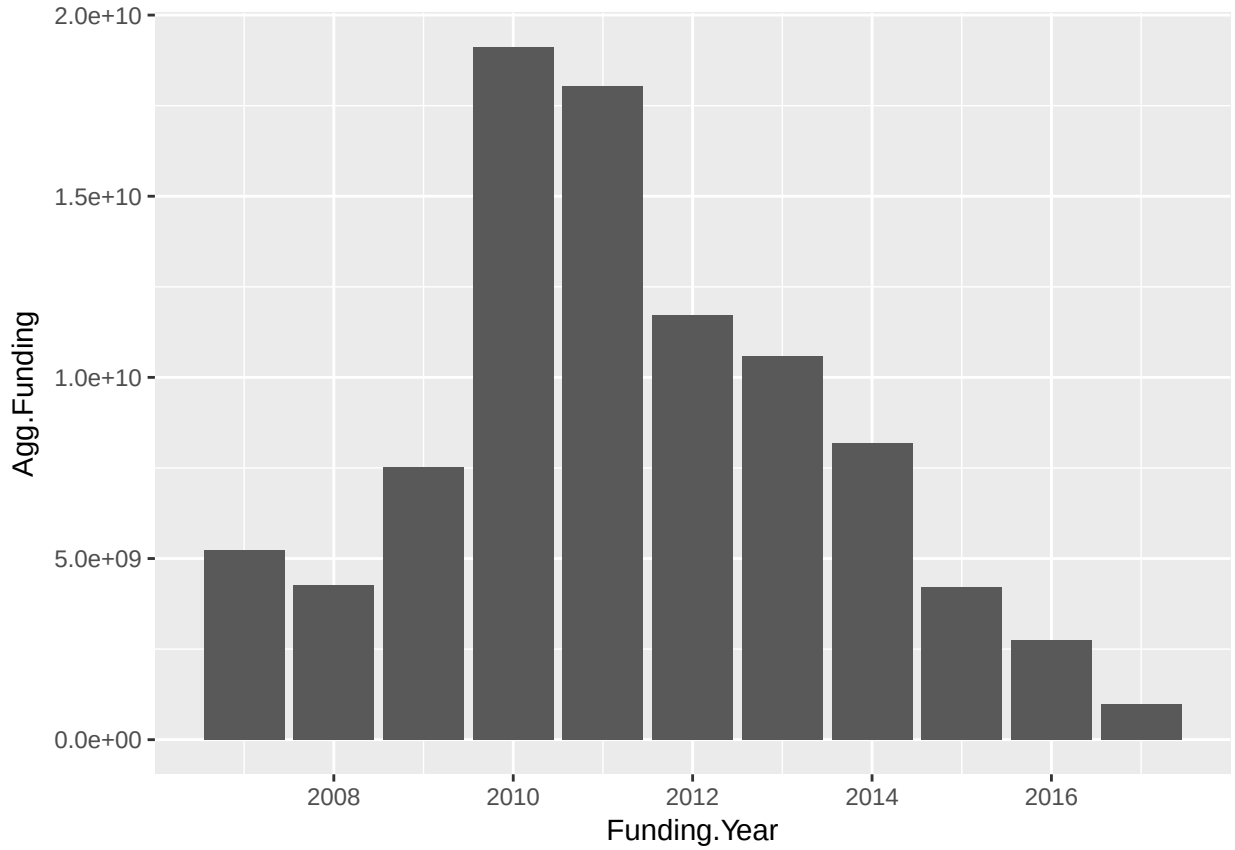
```
nrow(data)
```

```
## [1] 6727
```

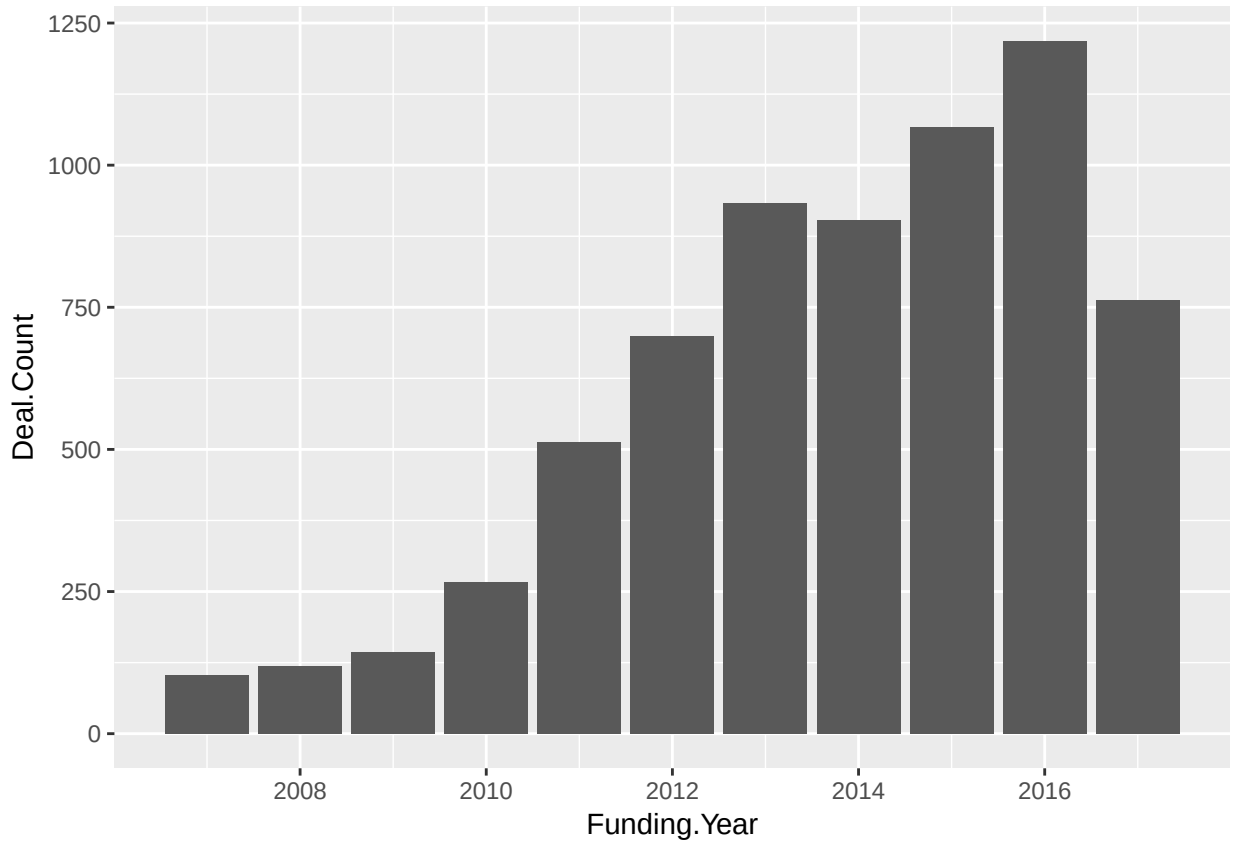
```
sum(data$Total.Funding.Amount)
```

```
## [1] 92581068983
```

```
#table1 <- as.data.frame(table(cut(data$First.Funding, breaks="year")))
data$Funding.Year <- sapply(strsplit(as.character(data$First.Funding),'-'), "[", 1)
data$Funding.Year <- as.numeric(data$Funding.Year)
table2 <- ddply(data, .(Funding.Year), summarize, Agg.Funding=sum(Total.Funding.Amount), Deal.Count = le
ggplot(table2, aes(Funding.Year, Agg.Funding)) +geom_bar(stat = "identity")
```




```
ggplot(table2, aes(Funding.Year, Deal.Count)) +geom_bar(stat = "identity")
```

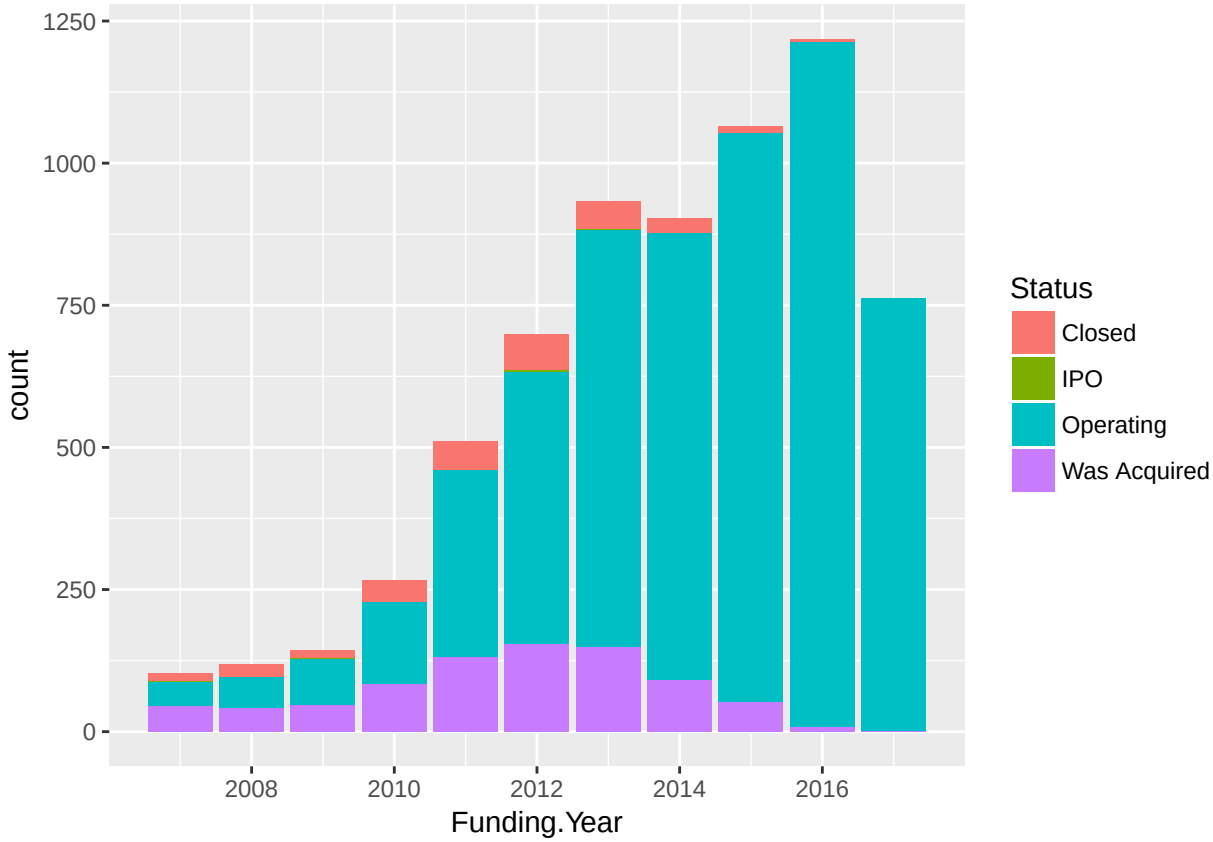



```
table2$Avg.Total.Raise = table2$Agg.Funding/table2$Deal.Count
table2
```

```
##    Funding.Year Agg.Funding Deal.Count Avg.Total.Raise
## 1          2007  5222865216        103        50707429
## 2          2008  4277429891        119        35944789
## 3          2009  7523782650        144        52248491
## 4          2010 19112976367        267        71584181
## 5          2011 18045498299        512        35245114
## 6          2012 11715169795        699        16759900
## 7          2013 10571810281        933        11330986
## 8          2014  8188872354        904         9058487
## 9          2015  4200430344       1066         3940366
## 10         2016  2737023629       1218         2247146
## 11         2017   985210157        762         1292927
```

We can also look by year at what the outcomes for these companies were: will tell us where exit activity begins to drop off and may indicate whether any particular years brought more success than others.

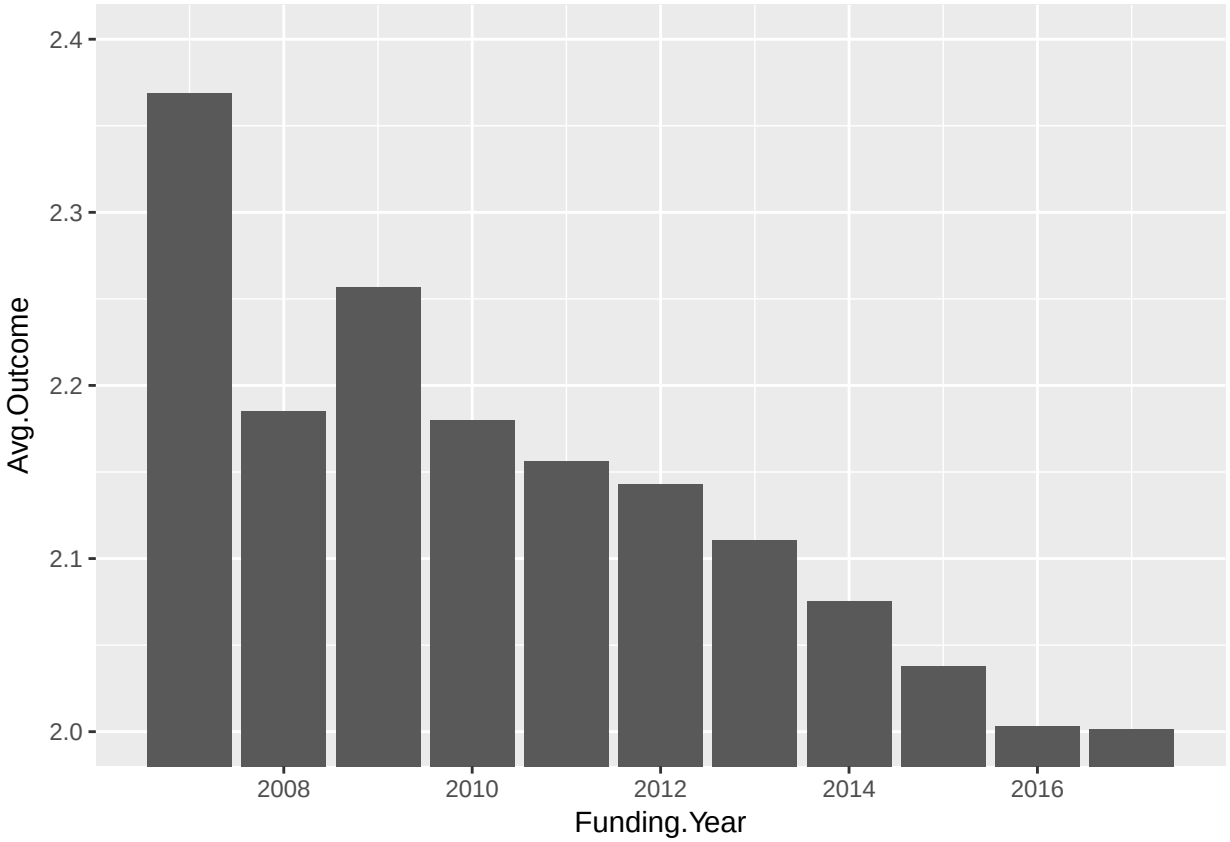
```
ggplot(data, (aes(Funding.Year))) + geom_bar(aes(fill=Status))
```



So indeed it looks like a cohort needs at least four years to begint o see exits, acquisitions, etc. Meaning any company in our set of 2013 or early may not be at a stage yet to be judged on outcome. We can also use factor levels to judge outcomes across a vintage of companies.

So we suspect that the most successful companies (based on their success at fundraising at least) were 2010 vintage. In order to have a closer look, we refactor Status in order make numeric determinations around outcomes. This factor list is ordered, and higher = better outcome.

```
data$Status <- ordered(data$Status, levels = c("Closed", "Operating", "Was Acquired", "IPO"))
table3 <- ddply(data, .(Funding.Year), summarize, Avg.Outcome=mean(as.numeric(Status)))
ggplot(table3, aes(Funding.Year, Avg.Outcome)) +geom_bar(stat = "identity") + coord_cartesian(ylim=c(2,
```



We can see that 2007 and 2009 were actually the most "successful" years for Seed-funded companies. Some of these from later vintages will need more time to successfully resolve, but for 6+ year old companies we should know with a level of confidence the outcome.

Building out returns estimates for Acquired Companies

We can measure returns in a number of ways. We have good detail on the total amount of money that went IN to these companies. We have some information of what they generated on exit. Using just what has been reported as fact, so just acquisitions that happened at disclosed values, investors got 16.5% of their money back. When we add IPOs to the mix, the number increases to 78%.

```
#compare disclosed acquisition prices with total funds deployed in the space.
sum(data$Price, na.rm = TRUE)/sum(Total.Funding.Amount)
```

```
## [1] 0.16549
```

```
(sum(data$Price, na.rm = TRUE)+sum(data$Valuation.at.IPO, na.rm = TRUE))/sum(Total.Funding.Amount)
```

```
## [1] 0.7805841
```

But this isn't the entire picture. Of the 806 companies in our dataset that were acquired, 706 are missing pricing information.

```
length(which(data$Status == "Was Acquired"))
```

```
## [1] 806
```

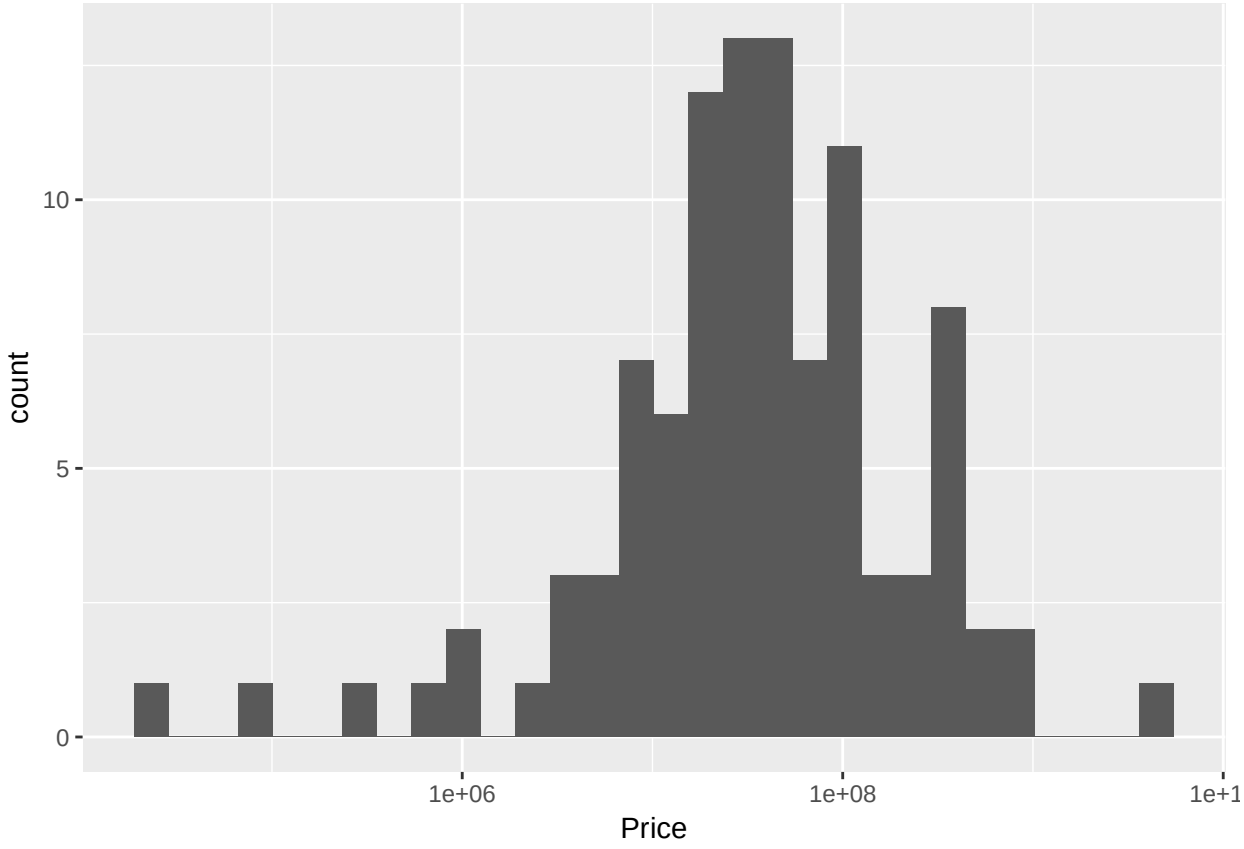
```
sum(is.na(data$Price[which(data$Status == "Was Acquired")]))
```

```
## [1] 705
```

We can make fair assumptions based on what we know from our acquired/disclosed set regarding where these other transactions may have been priced. Without populating this missing data, we will not be able to draw broad conclusions about returns in the space. First, let's start calculating returns using the information we have.

```
#data2 = priced acquired
#data3 = unpriced acquired
#subset data to include just the priced/acquired companies
data2 <- subset(data, Status == "Was Acquired" & Price > 1)
data3 <- subset(data, Status == "Was Acquired" & is.na(Price))
data4 <- subset(data, Status != "Was Acquired")
#at what prices were these companies acquired?
ggplot(data2, aes(Price)) +geom_histogram() + scale_x_log10()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

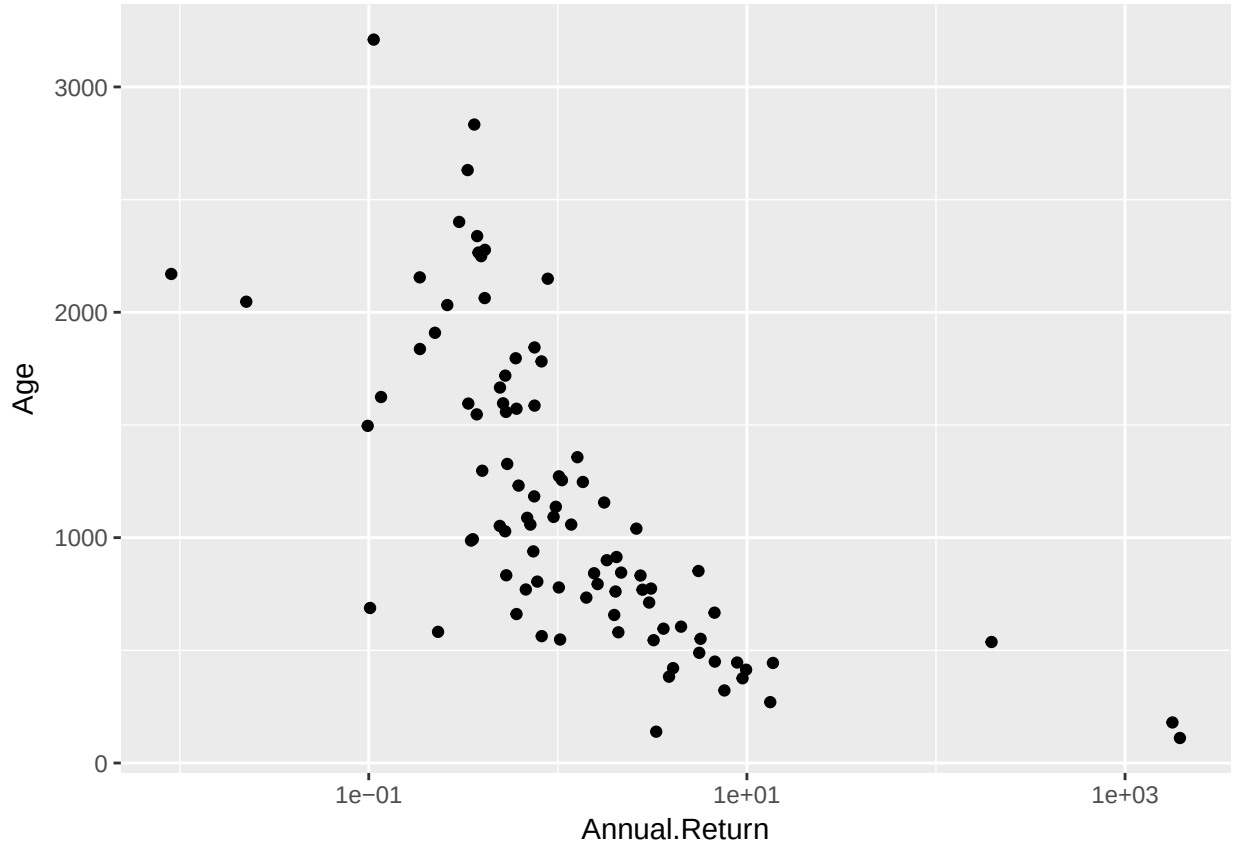



```
#calculate total return
data2$Total.Return <- data2$Price/data2$Total.Funding.Amount
#ggplot(data2, aes(Total.Return, Age)) +geom_point() + scale_x_log10()
#annualized return
data2$Annual.Return <- ((data2$Total.Return) ^ (365.25/as.integer(data2$Age))) - 1
par(mfrow=c(2,2))
ggplot(data2, aes(Annual.Return, Age)) +geom_point() + scale_x_log10()
```

```
## Warning in self$trans$transform(x): NaNs produced
```

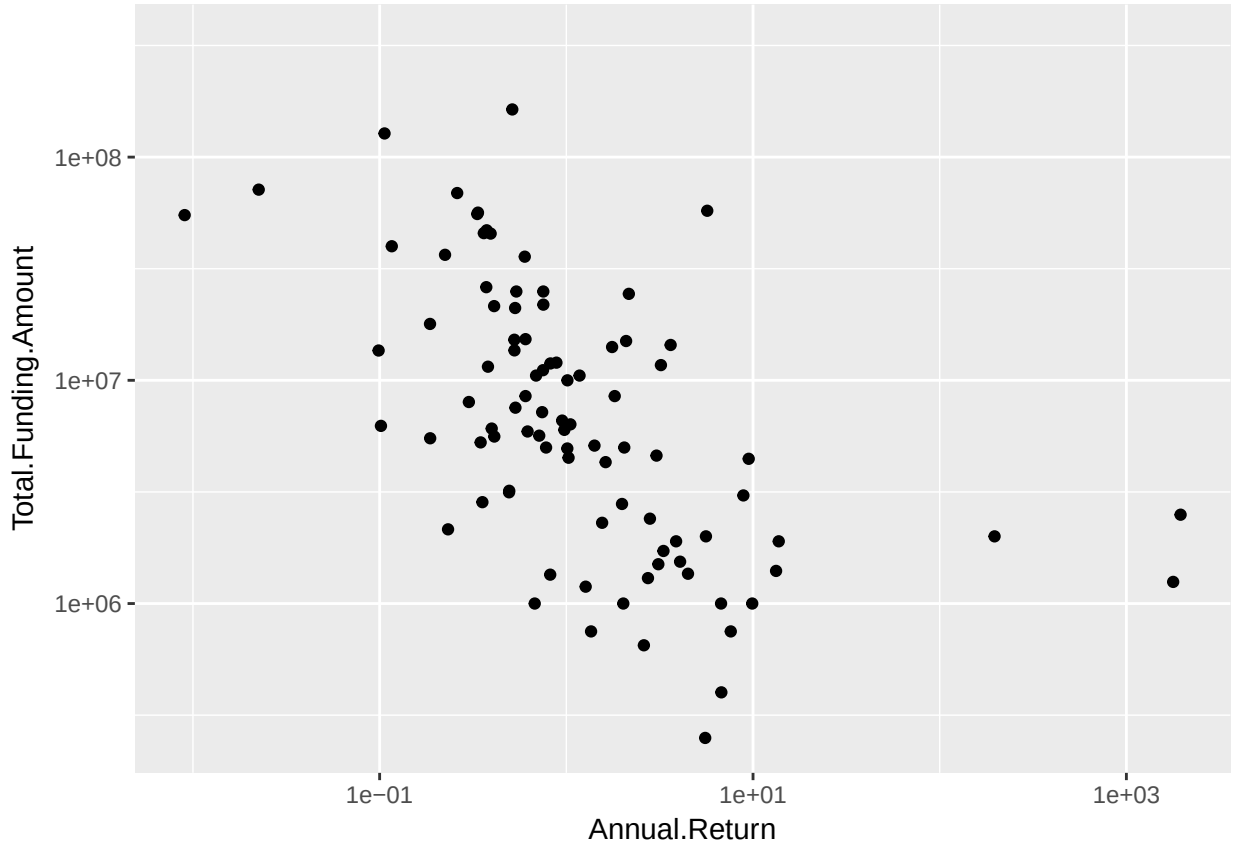
```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Removed 10 rows containing missing values (geom_point).
```




```
ggplot(data2, aes(Annual.Return, Total.Funding.Amount)) +geom_point() + scale_x_log10() + scale_y_log10
```

```
## Warning in self$trans$transform(x): NaNs produced
## Warning: Transformation introduced infinite values in continuous x-axis
## Warning: Removed 10 rows containing missing values (geom_point).
```




```
par(mfrow=c(1,1))
```

It doesn't look like funding amounts are going to tell us much about returns. Let's verify.

```
datamod2 <- lm(Annual.Return ~ Total.Funding.Amount, data = data2)
summary(datamod2)
```

```
##
## Call:
## lm(formula = Annual.Return ~ Total.Funding.Amount, data = data2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##  -49.75  -46.12  -43.61  -37.49 1902.94
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            4.944e+01  2.910e+01   1.699   0.0925 .
## Total.Funding.Amount -4.427e-07  6.461e-07  -0.685   0.4948
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 262.5 on 99 degrees of freedom
## Multiple R-squared:  0.00472,    Adjusted R-squared:  -0.005333
## F-statistic: 0.4695 on 1 and 99 DF,  p-value: 0.4948
```

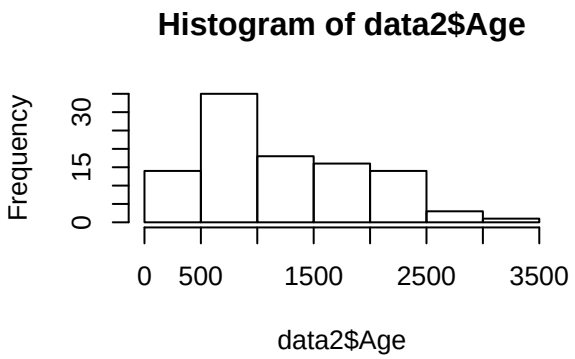
As suspected, the returns numbers are unrelated to funding amounts. Raising more money does not improve the return. Nor does the passage of time.

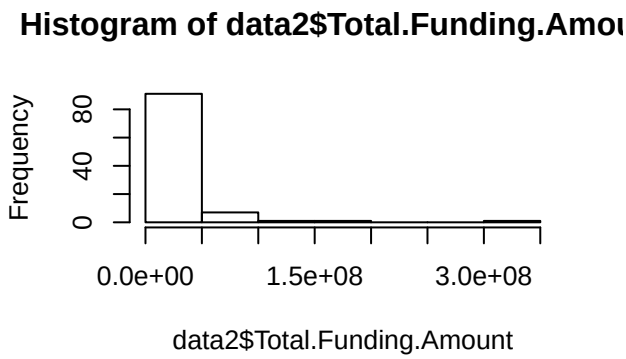
```r
datamod3 <- lm(Annual.Return ~ Age, data = data2)
summary(datamod3)
```

```
##
## Call:
## lm(formula = Annual.Return ~ Age, data = data2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -130.30  -80.74  -52.14   14.93 1815.22
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 145.75518   51.29368   2.842  0.00545 **
## Age          -0.08737    0.03701  -2.361  0.02018 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 256 on 99 degrees of freedom
## Multiple R-squared:  0.05331,    Adjusted R-squared:  0.04374
## F-statistic: 5.574 on 1 and 99 DF,  p-value: 0.02018
```

Perhaps if we break the sample into those who raised slightly less money (< median for the group, or 6.5Mm) and those who raised more, we might start to add some granularity here. Let's then see if returns are different amoung these two acquired subgroups.

```r
par(mfrow=c(2,2))
hist(data2$Age)
hist(data2$Total.Funding.Amount)
par(mfrow=c(1,1))
```






```
cutpoint <- summary(data2$Total.Funding.Amount)[3]
#break the group into two at the median
data2a <- subset(data2, data2$Total.Funding.Amount > cutpoint)
data2b <- subset(data2, data2$Total.Funding.Amount <= cutpoint)
#test if the population mean Annual.Return between the two samples is different
t.test(data2a$Annual.Return, data2b$Annual.Return)
```

```
##
##  Welch Two Sample t-test
##
## data:  data2a$Annual.Return and data2b$Annual.Return
## t = -1.5435, df = 50.001, p-value = 0.129
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -181.98547   23.82887
## sample estimates:
##  mean of x  mean of y
##  0.7039327 79.7822332
```

There is a difference: companies that raise less, return more. Although this calculation is impacted by a single transaction: Mapsense, which was acquired for 10X the invested amount just 110 days after funding, for a 2670% Annualized Return. Significant distortions can occur when annualizing short-horizon numbers. For this very reason, it's valuable to create buckets of investment rather than individual companies.

But back to our work estimating Price fields for our Acquired Undisclosed companies.

```
datamod4 <- lm(Annual.Return ~ Age, data = data2b)
summary(datamod4)
```

```
##
## Call:
## lm(formula = Annual.Return ~ Age, data = data2b)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -248.30 -143.39  -85.43   -3.40 1692.44
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 287.4838   102.3813   2.808  0.00714 **
## Age          -0.2581     0.1116  -2.313  0.02499 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 350.9 on 49 degrees of freedom
## Multiple R-squared:  0.0984, Adjusted R-squared:   0.08
## F-statistic: 5.348 on 1 and 49 DF,  p-value: 0.02499
```

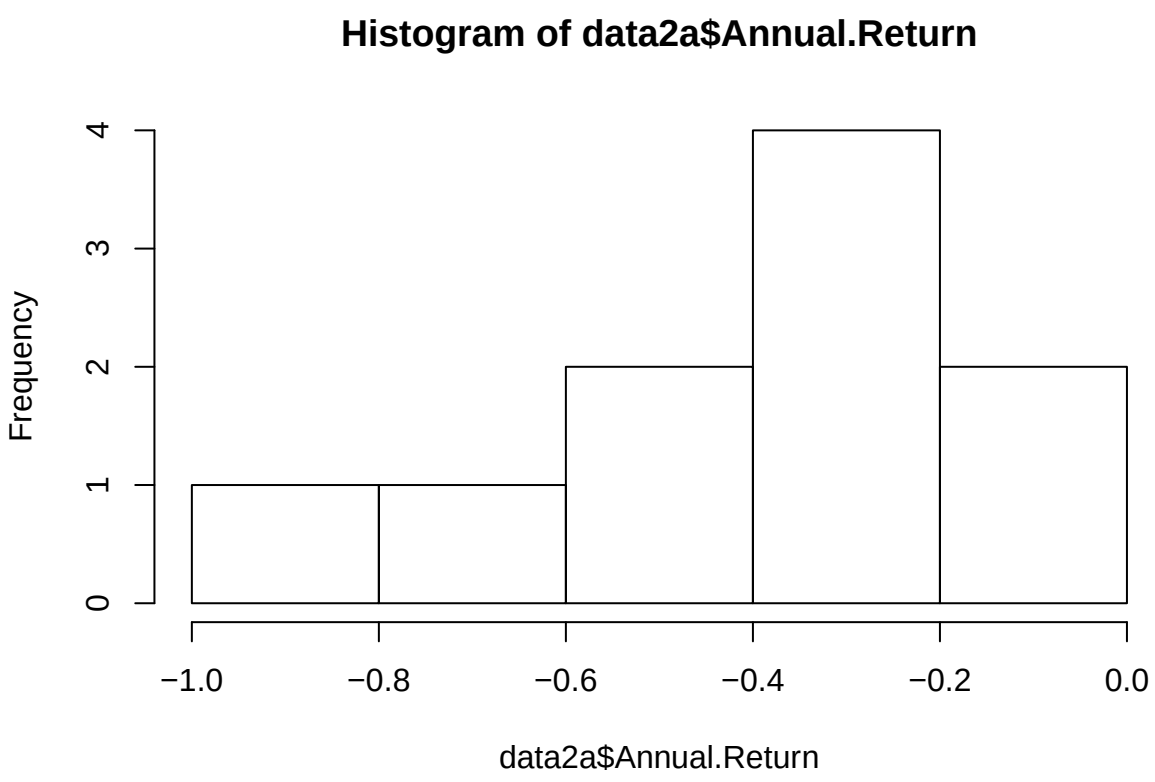
Interesting – here we see a negative slope, companies effectively generating lower returns the older they are. Let's use this model to populate returns for our those companies in our data3 group. We're going to arbitrarily reduce that intercept slightly, to account for the fact that undisclosed acquisitions are likely to be less favourable for investors (who like to publicise their wins).

Here is what we know about our acquired companies. First, 10% generated negative returns, generating exit values that were below their total fundraised amount. These negative returns are roughly normally distributed.

```
sum(data2$Annual.Return < 0)/nrow(data2)
```

```
## [1] 0.0990099
```

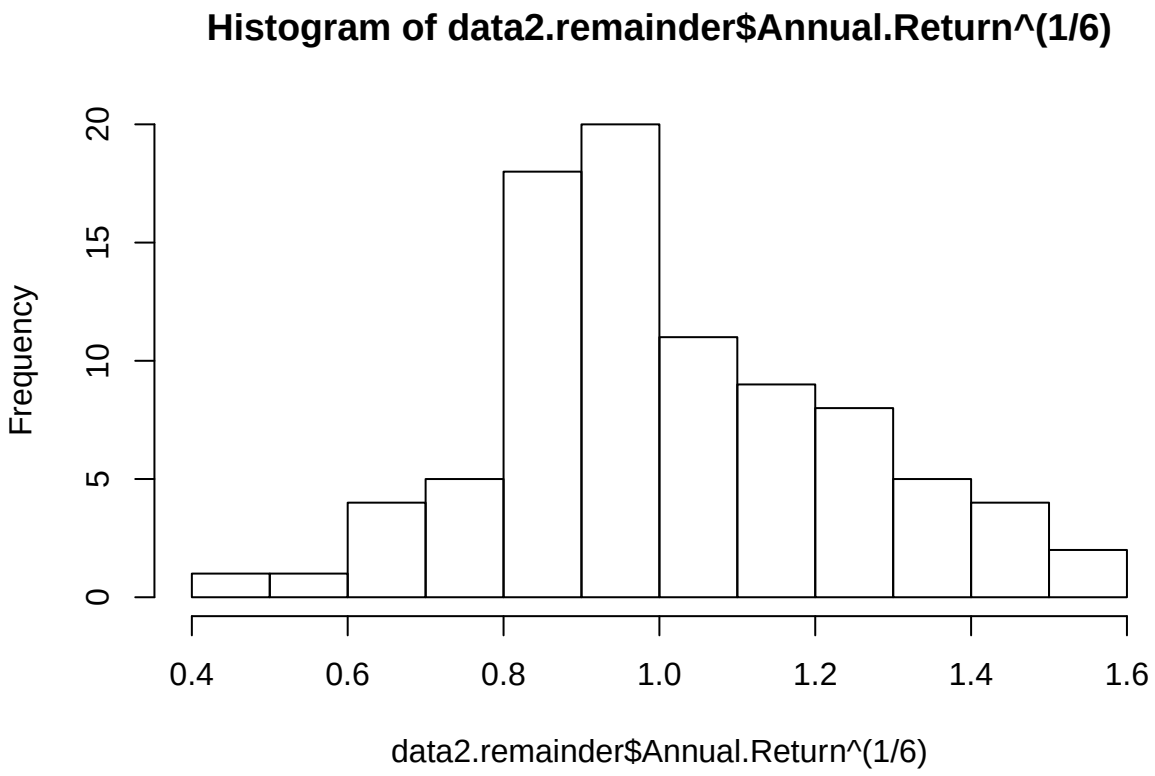
```
data2a <- subset(data2, data2$Annual.Return < 0)
data2.remainder <- subset(data2, data2$Annual.Return >= 0)
#check here if the distribution is kind of normal-ish
hist(data2a$Annual.Return)
```




```
#so we'll need 10% of our data3 group to have negative returns also, which we populate to be normalish
neg.returns <- sample(1:nrow(data3), (0.1 * nrow(data3)), replace=F)
data3.negs <- data3[c(neg.returns),]
data3.negs$Inferred.Return <- rtruncnorm(nrow(data3.negs), a=-1, b=0, mean=mean(data2a$Annual.Return),
data3.remainder <- data3[-c(neg.returns),]
```

Of our remaining companies, we have 3 that show Annual Returns $> 100\%$ (up to nearly 2000%). We'll take these out in order to have a reasonable look at returns.

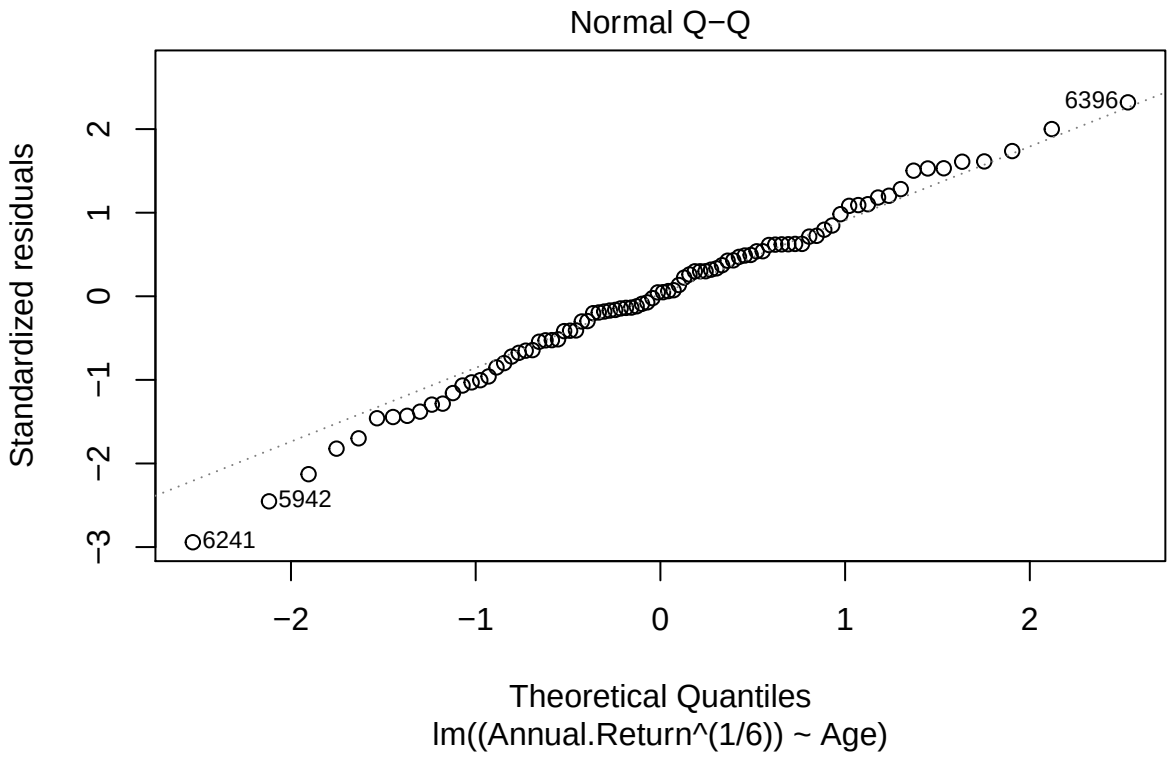
```
#remove our unhelpful outliers
data2.remainder <- data2.remainder[-c(which(data2.remainder$Company.Name %in% c("Mapsense", "Komand", "
data2.smallpos <- subset(data2.remainder, data2.remainder$Annual.Return < 2)
#we see a normalish distribution if we take the 1/6 root of the remaining returns
hist(data2.remainder$Annual.Return^(1/6))
```




```
#so we use that in our regression
datamod5 <- lm((Annual.Return^(1/6)) ~ Age, data = data2.remainder)
summary(datamod5)
```

```
##
## Call:
## lm(formula = (Annual.Return^(1/6)) ~ Age, data = data2.remainder)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.45145 -0.08745  0.00735  0.09436  0.35452
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.299e+00  3.419e-02  38.007  < 2e-16 ***
## Age         -2.394e-04  2.508e-05  -9.547  3.8e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1548 on 86 degrees of freedom
## Multiple R-squared:  0.5145, Adjusted R-squared:  0.5089
## F-statistic: 91.14 on 1 and 86 DF,  p-value: 3.797e-15
```

```
#QQ Plot here looks good
plot(datamod5, which = 2)
```



THIS NEEDS A LOT OF WORK

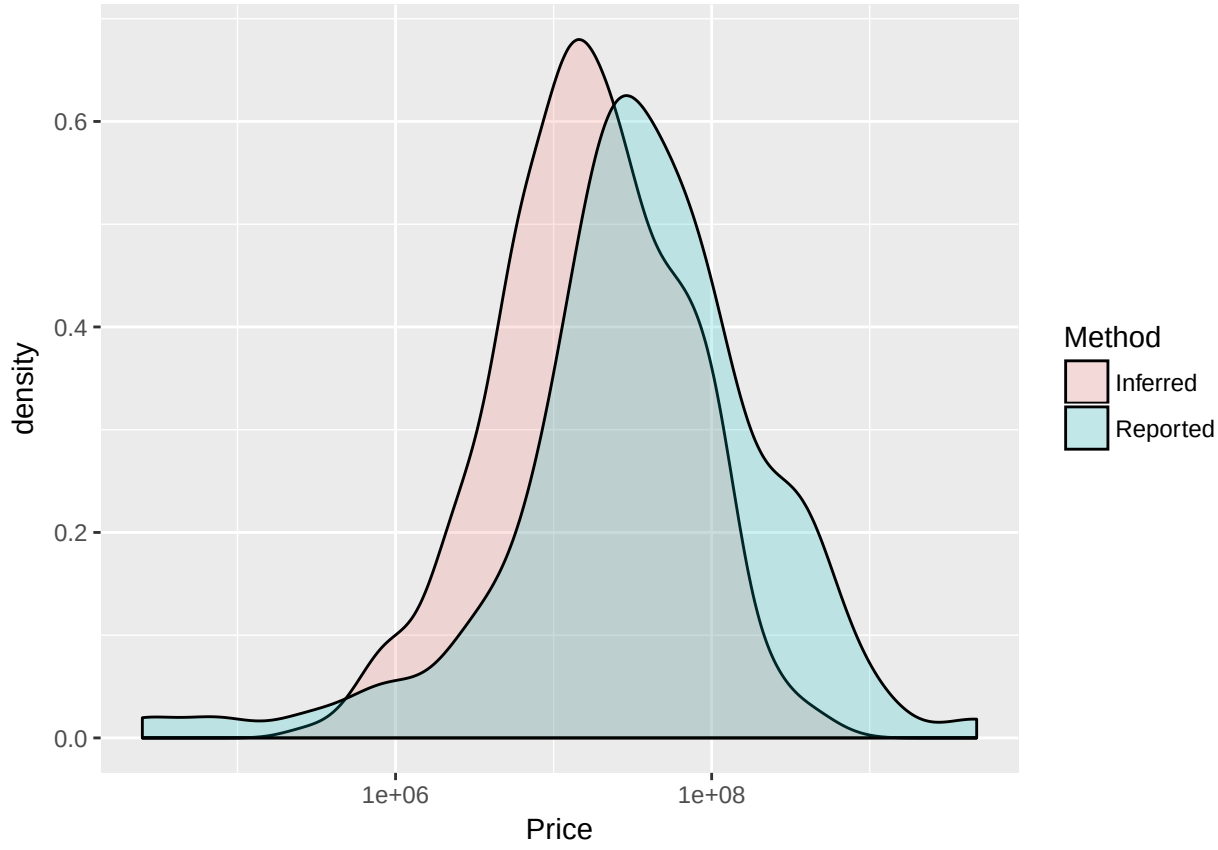
```
#we arbitrarily shift the regression line on our unpopulated set unitwise by a factor of .85 to
#acknowledge that the returns here are lower categorically than for our priced/acquired group
data3$Inferred.Return <- ((datamod5$coefficients[1]) + datamod5$coefficients[2]*data3$Age)^6
summary(data3$Inferred.Return)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.006673 0.941400 1.740000 1.786000 2.528000 4.313000
```

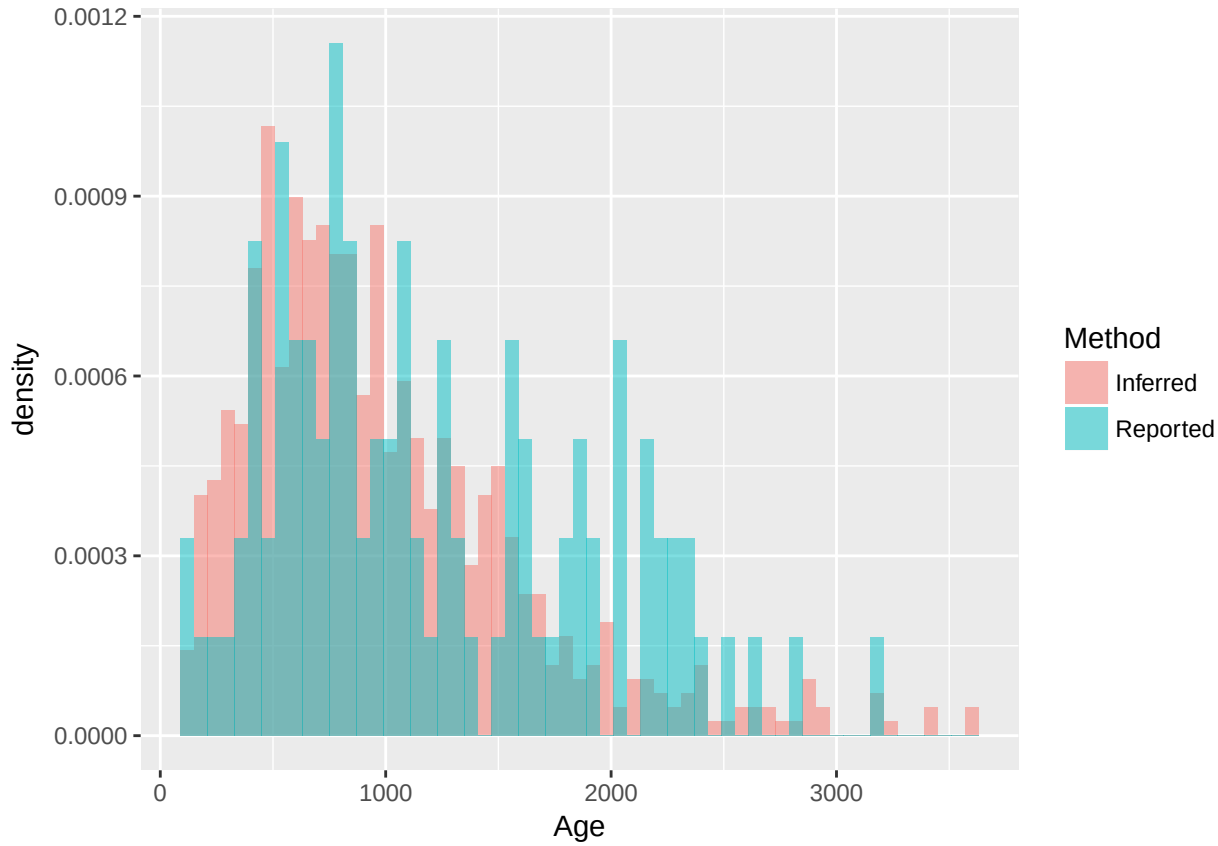
```
summary(data2$Inferred.Return)
```

```
## Length  Class   Mode
##      0   NULL   NULL
```

```
data3$Inferred.Price <- (data3$Total.Funding.Amount * (1 + data3$Inferred.Return) ^ (as.integer(data3$Ag
data3$Method <- "Inferred"
data2$Method <- "Reported"
plot1 <- subset(data3, select = c("Inferred.Return", "Inferred.Price", "Age", "Method", "Total.Funding.A
plot2 <- subset(data2, select = c("Annual.Return", "Price", "Age", "Method", "Total.Funding.Amount"))
colnames(plot1) <- c("Return", "Price", "Age", "Method", "Funding.Total")
colnames(plot2) <- c("Return", "Price", "Age", "Method", "Funding.Total")
plot3 <- rbind(plot1, plot2)
ggplot(plot3, aes(Price, fill = Method)) + geom_density(alpha = 0.2) + scale_x_log10()  #THIS IS A GREA
```




```
ggplot(plot3, aes(Age, fill = Method)) + geom_histogram(alpha = 0.5, aes(y = ..density..), position = '
```

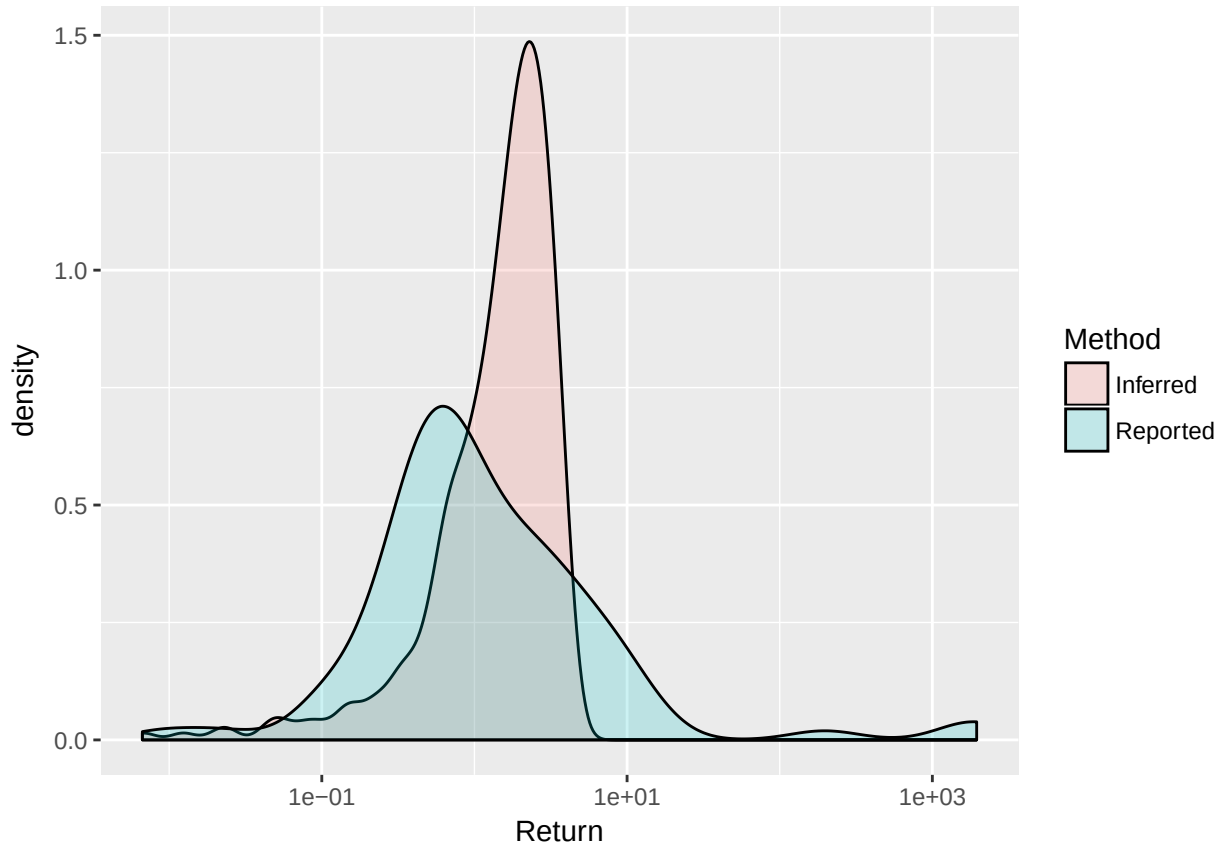



```
ggplot(plot3, aes(Return, fill = Method)) + geom_density(alpha = 0.2) + scale_x_log10()  #THIS IS A GRE
```

```
## Warning in self$trans$transform(x): NaNs produced
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Removed 10 rows containing non-finite values (stat_density).
```



If we look at the scenario we've created here, we're saying that those companies that were acquired at "some price" returned *as a group* 3.0X the money investors put in. Compared to *as a group* those companies that were acquired at disclosed prices, which returned 7.4X. Is this fair? Consider the companies that were acquired-undisclosed are as a group younger than the others.

```
sum(data3$Inferred.Price)/sum(data3$Total.Funding.Amount)
```

```
## [1] 6.674219
```

```
sum(data2$Price)/sum(data2$Total.Funding.Amount)
```

```
## [1] 7.425942
```

```
mean(data3$Inferred.Return)
```

```
## [1] 1.78627
```

```
mean(data2$Annual.Return)
```

```
## [1] 40.63456
```

Then we reassemble our groups to have a look at portfolio & structures.

```
data2$Total.Return <- NULL
data3$Price <- data3$Inferred.Price
data3$Inferred.Price <- NULL
colnames(data3)[21] <- "Annual.Return"
#colnames(data2) == colnames(data3)
data5 <- rbind(data2, data3)
data4$Annual.Return <- 0
```

```
data4$Price <- 0
data4$Method <- NA
data4$Price[which(!is.na(data4$Valuation.at.IPO))] <- data4$Valuation.at.IPO[which(!is.na(data4$Valuati
#colnames(data4) == colnames(data5)
data6 <- rbind(data4, data5)
```

Testing Theories on Portfolio size

Now that we have estimates for all of our acquired companies, and we know what the returns would have looked like had we bought the entire cohort of entrants each year (with the $9Tn we've got stashed under the mattress) let's size these portfolios down and see how the returns dispersion of the set reacts.

Monte Carlo Simulation:

```
data7 <- subset(data6, data6$First.Funding <= '2013-01-01')
runs <- 10000
#simulates a portfolio of 10 companies, returns the probability of 1X return (0% IRR).
portfolio.sim <- function(){
  row.nos <- sample(1:nrow(data7), 10, replace=F)
  portfolio.denom <- 0
  portfolio.numer <- 0
    for (i in row.nos){
      denom <- data7$Total.Funding.Amount[i]
      numer <- data7$Price[i]
      portfolio.denom <- portfolio.denom + denom
      portfolio.numer <- portfolio.numer + numer}
  portfolio.return <- portfolio.numer/portfolio.denom
return(portfolio.return > 1)
}
mc.prob <- sum(replicate(runs,portfolio.sim()))/runs
mc.prob
```

```
## [1] 0.4773
```