SEIS 631 Final Project: Number of Products vs. Sales Size

Richard Liu

05/2020

Introduction

What:

In a hypothetical scenario, I want to find out how many $1M products I need to initiate in order to have 95% confidence that at least one of them will reach $20M size. Assuming the probability of every $1M incremental sales is at XX%. All sales increases for each product are independent from each other. Success is defined as sales increase by $1M for each target size. Failure is defined as sales does not exceed more than $1M.

Why:

This hypothetical scenario would help project planning through statistical means without having to predict which specific product(s) will be the next to reach $20M sales milestone. In reality, we can't predict the long term sales outcome of any given product anyway due to their high variance and low population (n=1).

Minimal scenario:

I explore both distributions but found no such solution exist in any of the two proposed functions

Optimistic scenario:

I explore both distributions but found one of the two distributions provides the solution

Ambitious scenario:

I explore both distributions but found both of the two distributions can provide the solution

How:

I will try to explore using the following distributions to study the probability:

A. Geometric Distribution

B. Negative Binomial Distribution

A. Geometric Distribution requires Bernoulli Random Variables.

Criteria for Bernoulli Random Variables:

- Each sales increment can be thought as a trial
- A product is labeled as a success if its sales increase is larger than $1M for each trial and failure is when the sales increase is less than $1M in the same trial
- Probability of each $1M sales increase is a constant (p=XX%)
- Each trial has only two possible outcomes: success (>$1M) or failure (<$1M)

Key equation: P(success on the 20-th trial)=(1-p)^(20-1)*p

B. Negative Binomial Distribution criteria:

1. The trials are independent: sales increment is independent
2. Each trial outcome can be classified as a success (>$1M) or failure (<$1M)
3. The mean probability of success (p) for each trial is the same (yes. p=XX%)
4. The last trial much be a success (yes, N-th trial is success, N+1 trial is a failure)

Key Equation: P(20th success on the 21st trial)=p^20*(1-p)

Assumption #1:

- The probability of success of each product are a random variable with a known distribution.

Assumption #2:

- The probability of success of each product are a random variable from a chi-square model.
- What R-function do I use to generate a set of random variables that has normal distribution? rgeom()
- Generate 10 random p with mean of 80% and sdtev of +/-30% rnorm(10, mean=0.8, sd=0.3)
- Question 1: how to generate 20 random numbers with a geometric distribution with a mean and a stdev? rgeom(number, probability)
- Question 2: how to import a table from excel into R?

Use myData <- read.csv(file='Final_Project_Rliu.csv', header=TRUE) or myData <- read_csv(file='Final_Project_Rliu.cs
in tidyverse.

- Question 3: how to define the confidence interval of 95% in this context when the P distribution is hypothetically known as a function of the sales size, i.e. decreasing p as sales increases?

Solved by qgeom(percentile, probability)

- Question 4: why is plot() not working for the imported data file? The data stays as strings due to % and does not appear to be numeric. How to convert it into numeric?

This is solved by cleaning the data in excel before importing.

- Question 5: is there a way to make the neg binomial solution work?

Probably not after I gave it a try.

Load the data set from CSV file.

```
setwd("C:/Users/RICHARD/Downloads")

## myData <- read.csv(file='Final_Project_Rliu.csv', header=TRUE)

library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.3
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## Warning: package 'ggplot2' was built under R version 4.1.3
```

```
## Warning: package 'tibble' was built under R version 4.1.3
```

```
## Warning: package 'tidyr' was built under R version 4.1.3
```

```
## Warning: package 'readr' was built under R version 4.1.3
```

```
## Warning: package 'purrr' was built under R version 4.1.3
```

```
## Warning: package 'dplyr' was built under R version 4.1.3
```

```
## Warning: package 'forcats' was built under R version 4.1.3
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
myData <- read_csv(file='Final_Project_Rliu.csv')
```

```
## New names:
## * `` -> ...11
```

```
## Rows: 20 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (8): Sales_Bracket, Observed_P, Observed_Failure, Cal_P_Geometric, Indiv...
## lgl (3): Calculated_P_Neg_Bino, Individual_p_Neg_Bino, ...11
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
summary(myData)
```

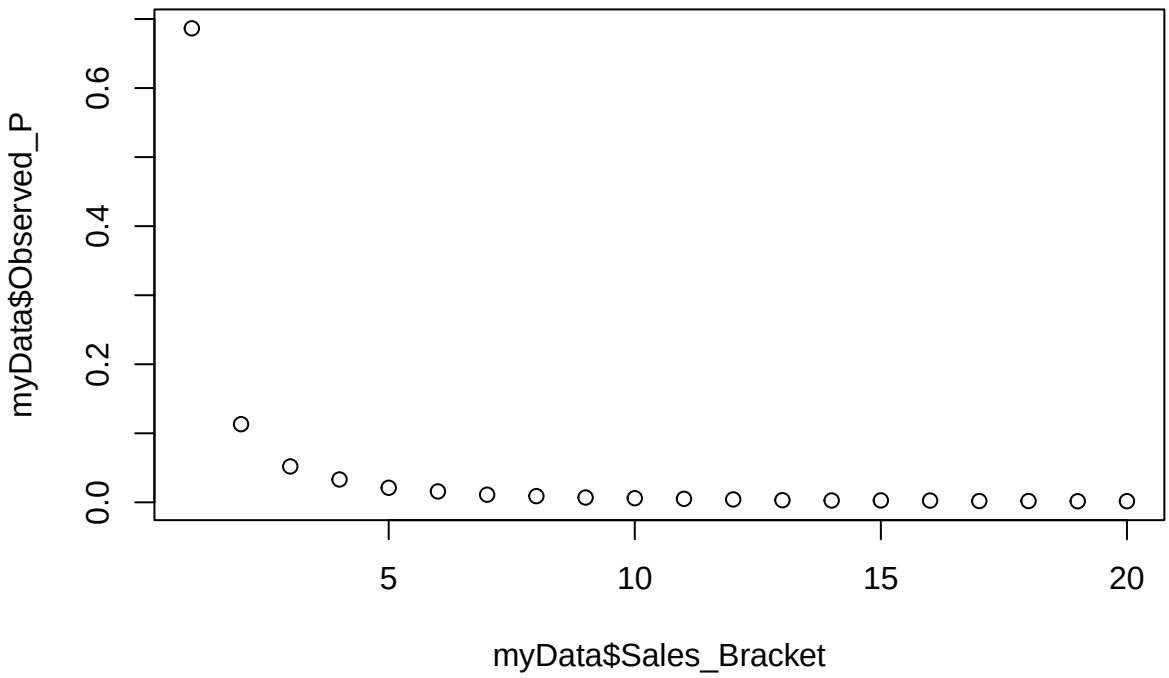
```
##  Sales_Bracket     Observed_P       Observed_Failure Cal_P_Geometric
##  Min.   : 1.00   Min.   :0.001600   Min.   :0.3135   Min.   :0.001600
##  1st Qu.: 5.75   1st Qu.:0.002675   1st Qu.:0.9829   1st Qu.:0.002575
##  Median :10.50   Median :0.005550   Median :0.9944   Median :0.005250
##  Mean   :10.50   Mean   :0.049120   Mean   :0.9509   Mean   :0.047765
##  3rd Qu.:15.25   3rd Qu.:0.017125   3rd Qu.:0.9973   3rd Qu.:0.015725
##  Max.   :20.00   Max.   :0.686500   Max.   :0.9984   Max.   :0.686500
##  Individual_p_Geometric Power_Model_Distribution Calculated_P_Neg_Bino
##  Min.   :0.001600       Min.   :0.001300         Mode :logical
##  1st Qu.:0.002675       1st Qu.:0.002325         FALSE:20
##  Median :0.005550       Median :0.004900
##  Mean   :0.049120       Mean   :0.044355
##  3rd Qu.:0.017125       3rd Qu.:0.016775
##  Max.   :0.686500       Max.   :0.560200
##  Individual_p_Neg_Bino      Mean           STDEV          ...11
##  Mode :logical         Min.   :  1.0   Min.   :  0.82   Mode:logical
##  FALSE:20              1st Qu.: 59.0   1st Qu.: 59.00   NA's:20
##                        Median :181.5   Median :181.00
##                        Mean   :240.7   Mean   :240.39
##                        3rd Qu.:373.8   3rd Qu.:373.50
##                        Max.   :625.0   Max.   :624.00
```

```
head(myData)
```

```
## # A tibble: 6 x 11
##   Sales_Bracket Observed_P Observed_Failure Cal_P_Geometric Individual_p_Geomet~
##           <dbl>      <dbl>            <dbl>           <dbl>                <dbl>
## 1             1     0.686             0.314          0.686                0.686
## 2             2     0.113             0.887          0.100                0.113
## 3             3     0.052             0.948          0.0468               0.052
## 4             4     0.0331            0.967          0.0299               0.0331
## 5             5     0.0211            0.979          0.0194               0.0211
## 6             6     0.0158            0.984          0.0145               0.0158
## # ... with 6 more variables: Power_Model_Distribution <dbl>,
## #   Calculated_P_Neg_Bino <lgl>, Individual_p_Neg_Bino <lgl>, Mean <dbl>,
## #   STDEV <dbl>, ...11 <lgl>
```

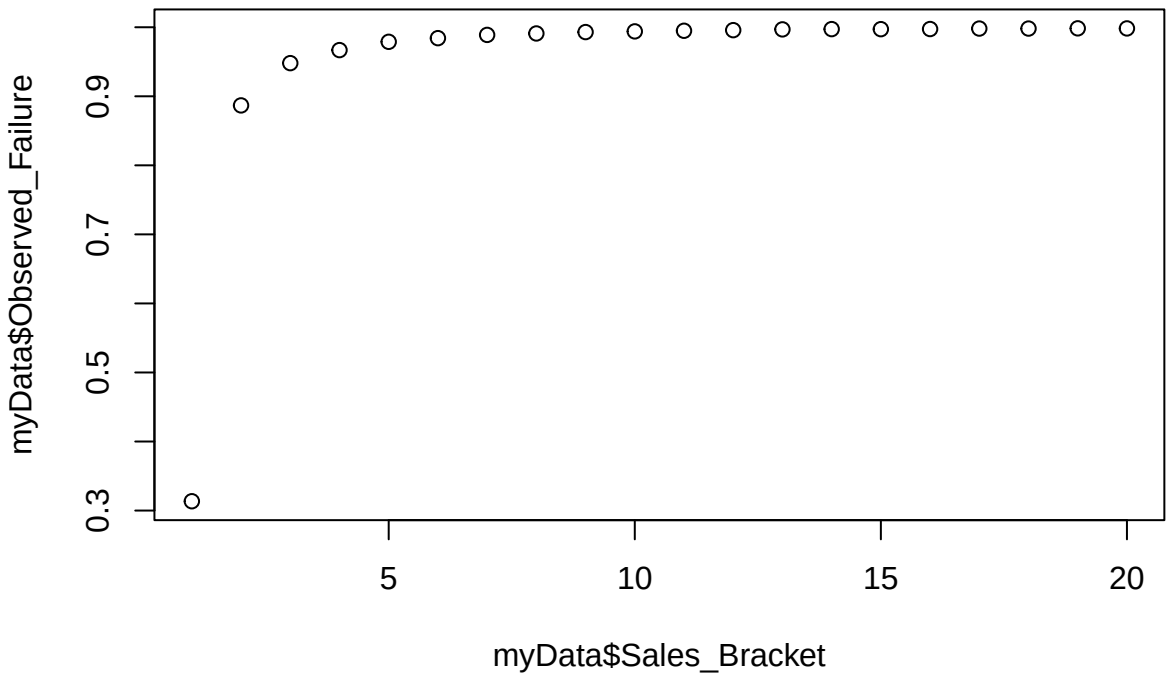
Plotting observed probability vs sales.

```
plot(x=myData$Sales_Bracket,y=myData$Observed_P)
```



Plotting observed Failure rate vs sales.

```
plot(x=myData$Sales_Bracket,y=myData$Observed_Failure)
```




```
?plot
```

```
## starting httpd help server ... done
```

Load the data frame with the following different variables to compare:

x-Sales

y1-Observed probability (hypothetical)

y2-Observed failure (1-P)

y3-Calculated probability (P) based on Geometric Distribution

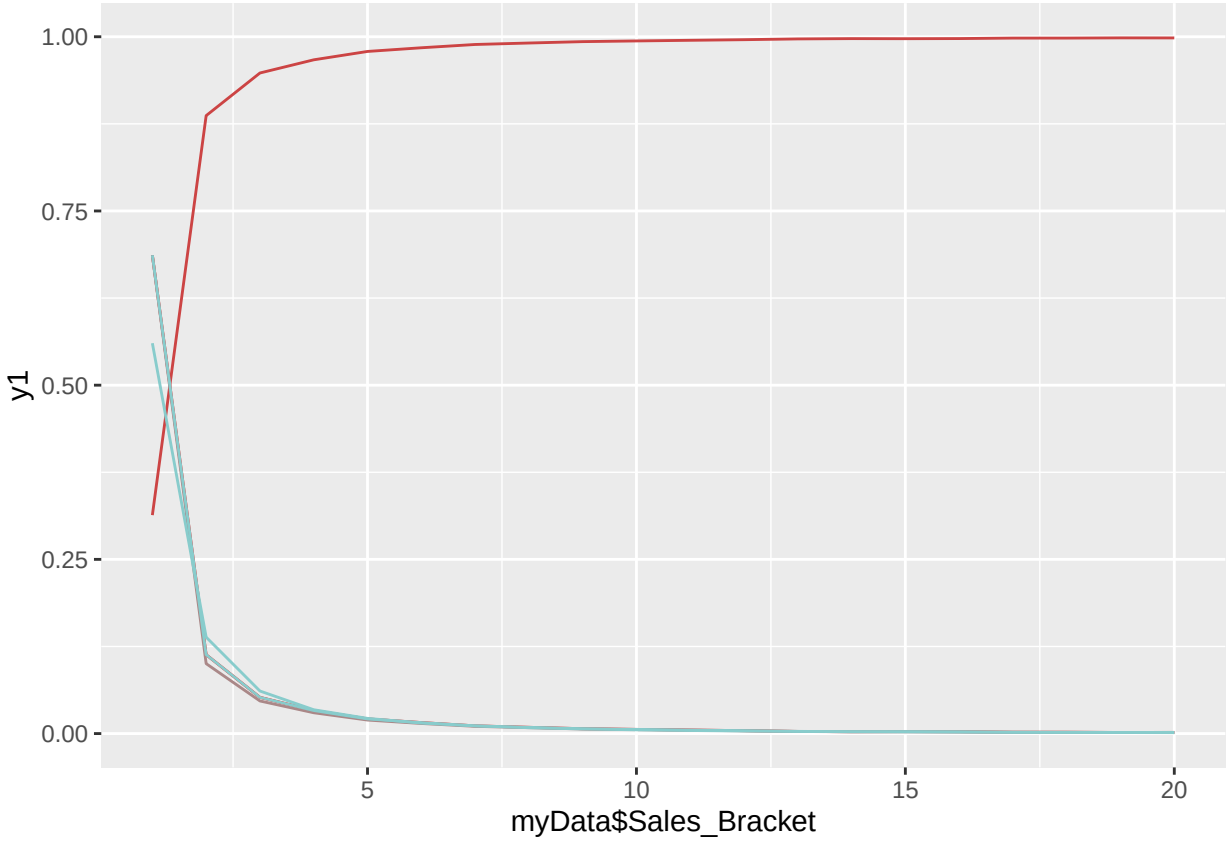
y4-Individual average probability (p) based on Geometric Distribution

y5-Power Law Distribution (Regression)

```
base.r.df <- data.frame(x=myData$Sales_Bracket, y1=myData$Observed_P,
                        y2=myData$Observed_Failure, y3=myData$Cal_P_Geometric,
                        y4=myData$Individual_p_Geometric,
                        y5=myData$Power_Model_Distribution)
```

Plot up the y1 through y5 all on the same chart:

```
ggplot(base.r.df) +

  geom_line(aes(x=myData$Sales_Bracket, y=y1), color="#FF0000") +

  geom_line(aes(x=myData$Sales_Bracket, y=y2), color="#CC4444") +

  geom_line(aes(x=myData$Sales_Bracket, y=y3), color="#AA8888") +

  geom_line(aes(x=myData$Sales_Bracket, y=y4), color="#88CCCC")  +

  geom_line(aes(x=myData$Sales_Bracket, y=y5), color="#88CCCC")
```



Analyzing the number of products needed to reach 20M at different confidence levels as measued by different target percentiles.

From the above data set, products above $20M or above is about p=2%.

At probability of 0.50 (50% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(0.5,0.02, lower.tail = TRUE)
```

```
## [1] 34
```

At probability of 0.60 (60% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(0.6,0.02, lower.tail = TRUE)
```

```
## [1] 45
```

At probability of 0.70 (70% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(0.7,0.02, lower.tail = TRUE)
```

```
## [1] 59
```

At probability of 0.80 (80% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(0.8,0.02, lower.tail = TRUE)
```

```
## [1] 79
```

At probability of 0.85 (85% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(0.85,0.02, lower.tail = TRUE)
```

```
## [1] 93
```

At probability of 0.90 (90% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(0.90,0.02, lower.tail = TRUE)
```

```
## [1] 113
```

At probability of 0.95 (95% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(0.95,0.02, lower.tail = TRUE)
```

```
## [1] 148
```

At probability of 0.975 (97.5% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(0.975,0.02, lower.tail = TRUE)
```

```
## [1] 182
```

At probability of 0.99 (99% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(0.99,0.02, lower.tail = TRUE)
```

```
## [1] 227
```

At probability of 1.00 (100% percentile), how many trials to get to the $20M at p=2%.

```
qgeom(1.00,0.02, lower.tail = TRUE)
```

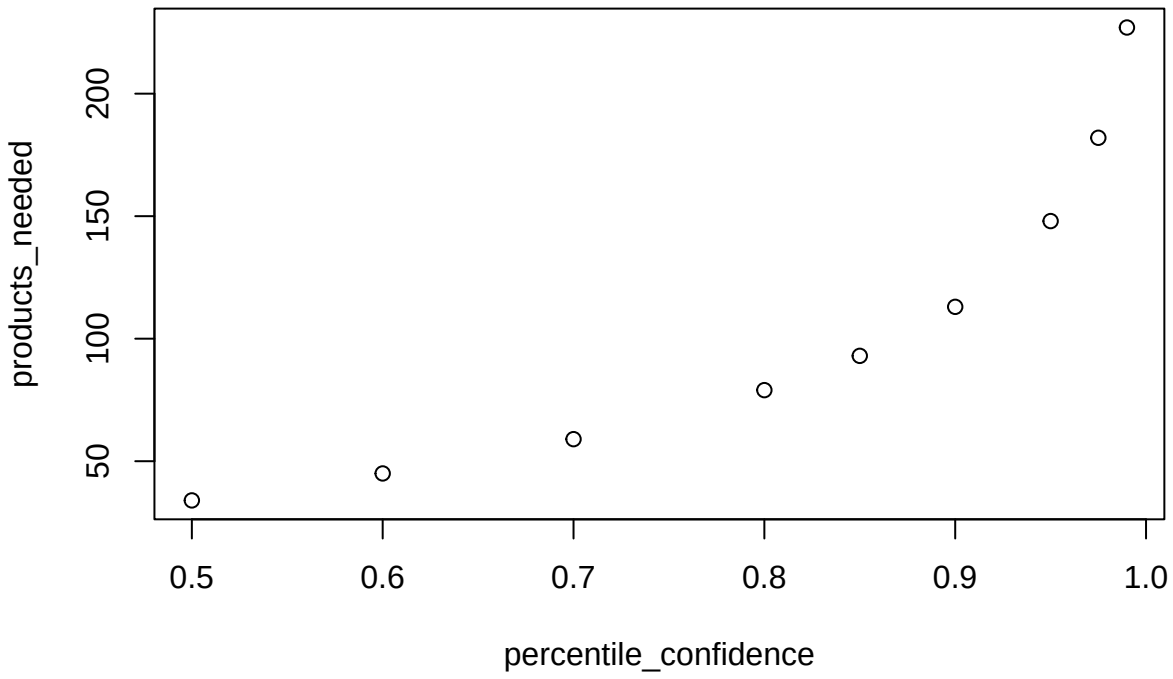
```
## [1] Inf
```

With this set of simulation trials, one can clearly show that, given a geometric distribution, achieving a $20M product sales milestone requires multiple products to allow at least one of them randomly emerge over time and the percentile (confidence) level in sales with number of products required: higher percentile (confidence) level requires more products to be started concurrently.

It also make sense that there is 100% garrenty in the success so it would be realistic to ask for a 100% percentile or 100% confidence. As the old saying goes: anything is possible and it all comes down the probability distributions.

Plot number of products needed vs. percentile to visualize the distribution

```
percentile_confidence<-c(0.5,0.6,0.7,0.8, 0.85, 0.9, 0.95, 0.975, 0.99)
products_needed<-c(34,45,59,79,93,113,148,182,227)

plot(x=percentile_confidence,y=products_needed)
```



Analyzing the probability of exceeding 20M sales vs. number of products to be initiated.

To turn the question around, if one chooses to run just 1 product based on the best criteria just to save time and resource, what probability of that product will be exceeding the 20M? Assuming the p=2% for 20M

with geometric distribution.

```
pgeom(1,0.02, lower.tail = TRUE)
```

```
## [1] 0.0396
```

What about if he chooses 2 of his best products? what probability of at least one product will be exceeding 20M?

```
pgeom(2,0.02, lower.tail = TRUE)
```

```
## [1] 0.058808
```

What about if he chooses 4 of his best products? what probability of at least one product will be exceeding 20M?

```
pgeom(4,0.02, lower.tail = TRUE)
```

```
## [1] 0.0960792
```

What about if he chooses 8 of his best products? what probability of at least one product will be exceeding 20M?

```
pgeom(8,0.02, lower.tail = TRUE)
```

```
## [1] 0.1662522
```

What about if he chooses 16 of his best products? what probability of at least one product will be exceeding 20M?

```
pgeom(16,0.02, lower.tail = TRUE)
```

```
## [1] 0.2906782
```

What about if he chooses 34 of his best products? what probability of at least one product will be exceeding 20M?

```
pgeom(34,0.02, lower.tail = TRUE)
```

```
## [1] 0.5069254
```

What about if he chooses 45 of his best products? what probability of at least one product will be exceeding 20M?

```
pgeom(45,0.02, lower.tail = TRUE)
```

```
## [1] 0.6051797
```

What about if he chooses 59 of his best products? what probability of at least one product will be exceeding 20M?

```
pgeom(59,0.02, lower.tail = TRUE)
```

```
## [1] 0.7024469
```

What about if he chooses 79 of his best products? what probability of at least one product will be exceeding 20M?

```
pgeom(79,0.02, lower.tail = TRUE)
```

```
## [1] 0.8013511
```

What about if he chooses 93 of his best products? what probability of at least one product will be exceeding 20M?

```
pgeom(93,0.02, lower.tail = TRUE)
```

```
## [1] 0.8502899
```

What about if he chooses 113 of his best products? what probability of at least one product will be exceeding 20M?

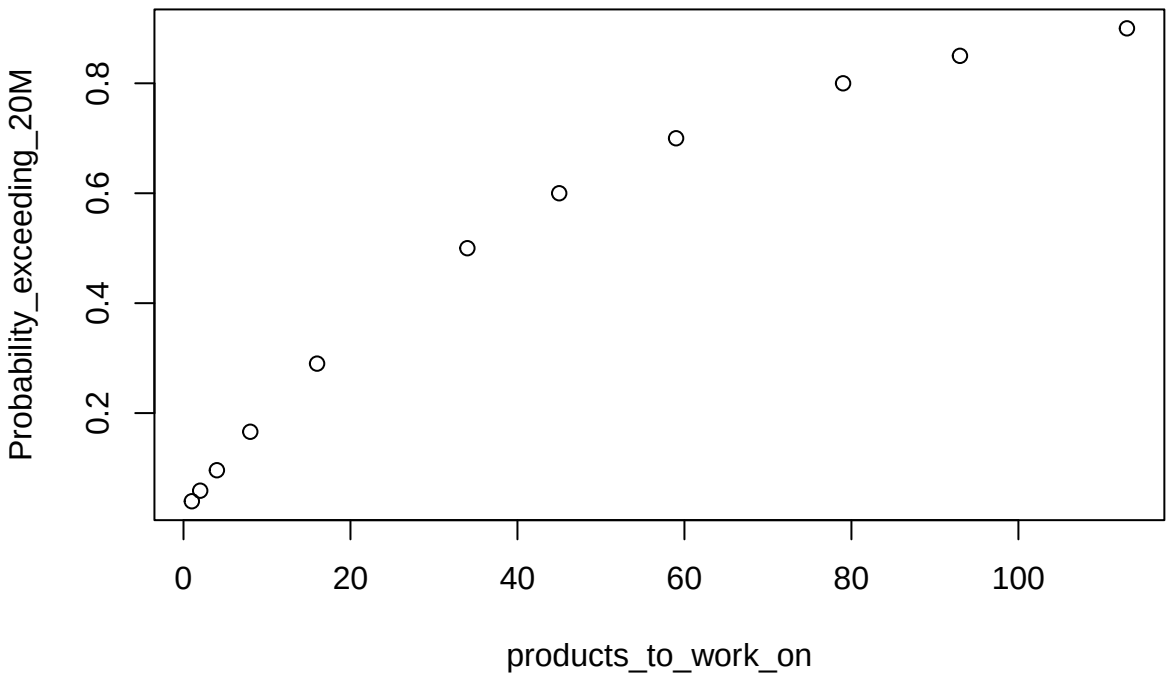
```
pgeom(113,0.02, lower.tail = TRUE)
```

```
## [1] 0.9000523
```

With this set of simulation trials, one can clearly show that, given a geometric distribution, the probability of achieving at least one product for $20M sales increases gradually with increasing number of products being worked on. This is making perfect sense.

Plot number of products needed vs. probability to visualize the distribution

```
Probability_exceeding_20M<-c(0.0396,0.0588, 0.096, 0.166, 0.29, 0.5,0.6,0.7,0.8, 0.85, 0.9)
products_to_work_on<-c(1,2,4,8,16,34,45,59,79,93,113)

plot(x=products_to_work_on,y=Probability_exceeding_20M)
```



Calculate the Mean and Standard Deviation for selected sales targets

equations to be used: mean = 1/p, SD = SQRT(1-p)/p

For $1M products (p=0.69)

```
p1<-0.69
c(p1)
```

```
## [1] 0.69
```

```
mean1<-1/p1
c(mean1)
```

```
## [1] 1.449275
```

```
SD1<-sqrt(1-p1)/p1
c(SD1)
```

```
## [1] 0.8069224
```

For $5M products (p=0.021)

```
p5<-0.021
c(p5)
```

```
## [1] 0.021
```

```
mean5<-1/p5
c(mean5)
```

```
## [1] 47.61905
```

```
SD5<-sqrt(1-p5)/p5
c(SD5)
```

```
## [1] 47.11639
```

For $10M products (p=0.006)

```
p10<-0.006
c(p10)
```

```
## [1] 0.006
```

```
mean10<-1/p10
c(mean10)
```

```
## [1] 166.6667
```

```
SD10<-sqrt(1-p10)/p10
c(SD10)
```

```
## [1] 166.1659
```

For $20M products (p=0.002)

```
p20<-0.002
c(p20)
```

```
## [1] 0.002
```

```
mean20<-1/p20
c(mean20)
```

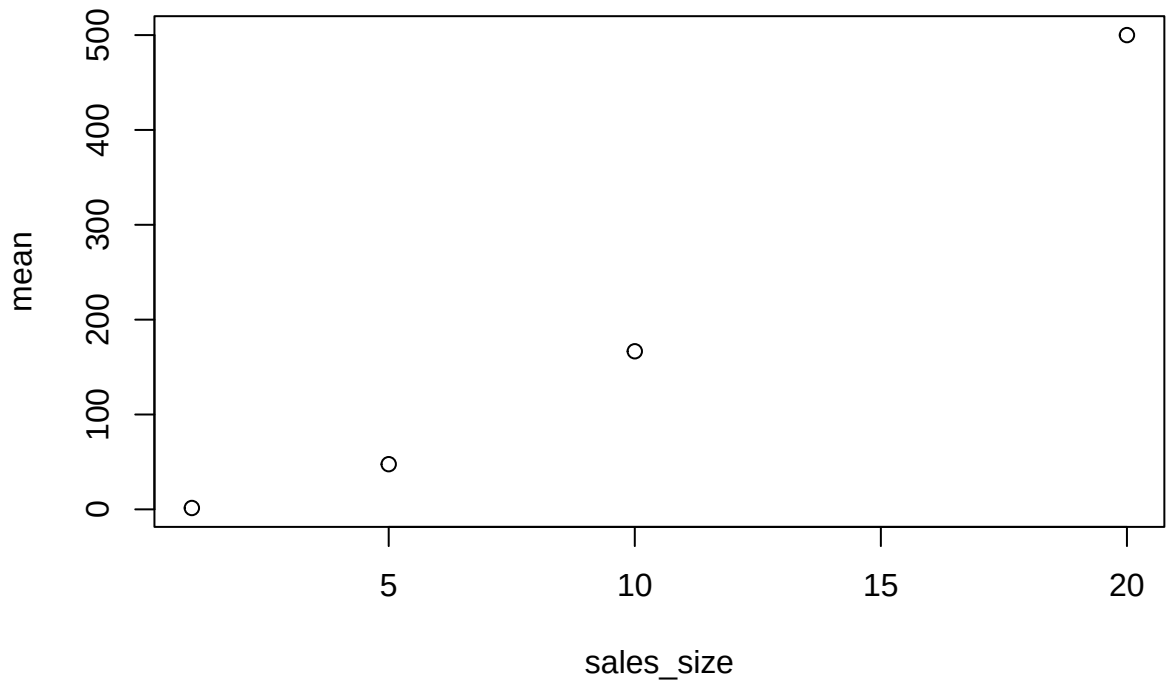
```
## [1] 500
```

```
SD20<-sqrt(1-p20)/p20
c(SD20)
```

```
## [1] 499.4997
```

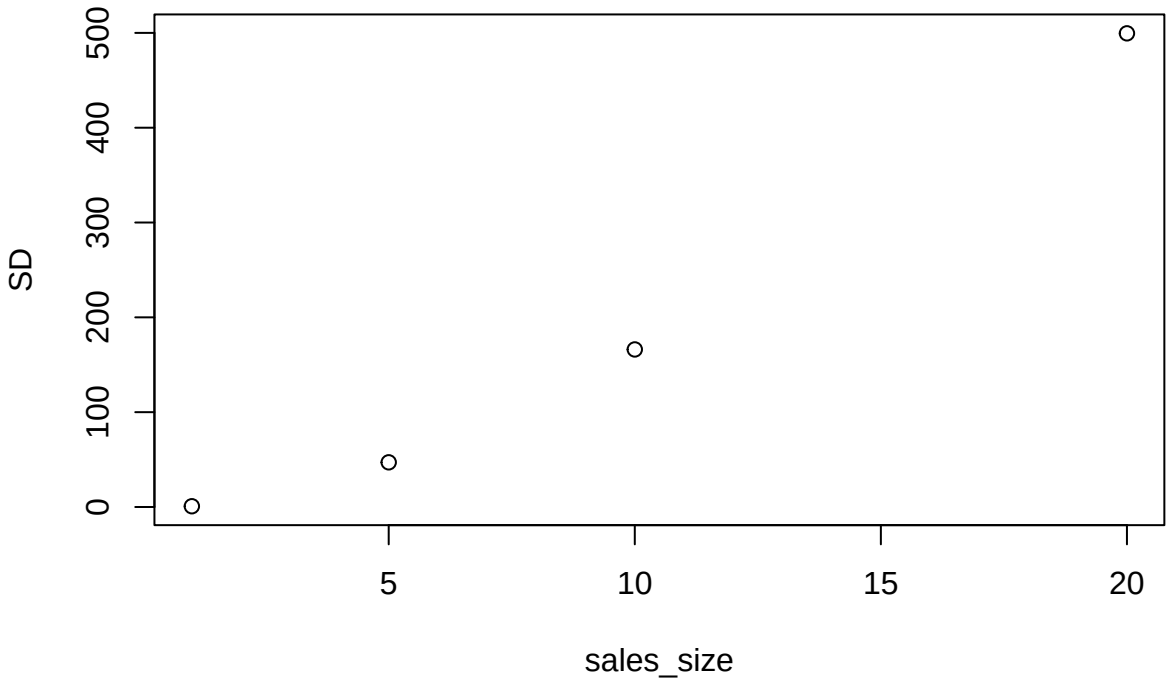
Plot the mean (Y) vs. sales size (X)

```
sales_size<-c(1, 5, 10, 20)

mean<-c(mean1,mean5,mean10,mean20)

plot(x=sales_size, y=mean)
```



Plot the Standard Deviation (Y) vs. sales size (X)

```
sales_size<-c(1, 5, 10, 20)

SD<-c(SD1,SD5,SD10,SD20)

plot(x=sales_size, y=SD)
```



Observation from above two plots: the calculated mean and calculated standard deviation appears to be identical? Any reason why this is the case?

Conclusions:

From above statistical analysis, one can conclude the following:

1) The geometric distribution can accurately describe the probability distribution of the sales size of a given product portfolio. Negative binomial distribution did not work in this case.

2) The probability of at least 1 product exceeding certain sales size (e.g. 20M) scales with the number of products that are commercialized.

3) The probability of running few products will result in lower probility and larger variance, i.e. inability to accurately predict what will happen with the small number of products.

4) The analysis on the mean and standard deviation shows that both variables scale with the product sales size, which means that more products are needed to be commercialized in order to hit a target sales size. At the same time, the standard deviation also increases with the increased target sales size.

5) The analysis proves once more that anything is possible, all it comes down to is the distribution of probabilities in each scenario. In order to have corresponding confidence level in the expected outcome, the statistical knowledge and analysis using appropriate distribution is the key to predict the most likely outcome.

Open questions from this analysis to be further explored. Any suggestion(s) are welcome.

1) In this case, why the mean (1/p) and standard deviation [sqrt(1-p)/p] are found to be the same for each target sales size? Are they always the same for other geometric distributions?

2) If the p is known for a target sales size, what is the STDEV if only 1 product is initiated to aim for that sales size? and how to calculate it for a random variable with geometric distribution?