Dealing with Materials Data

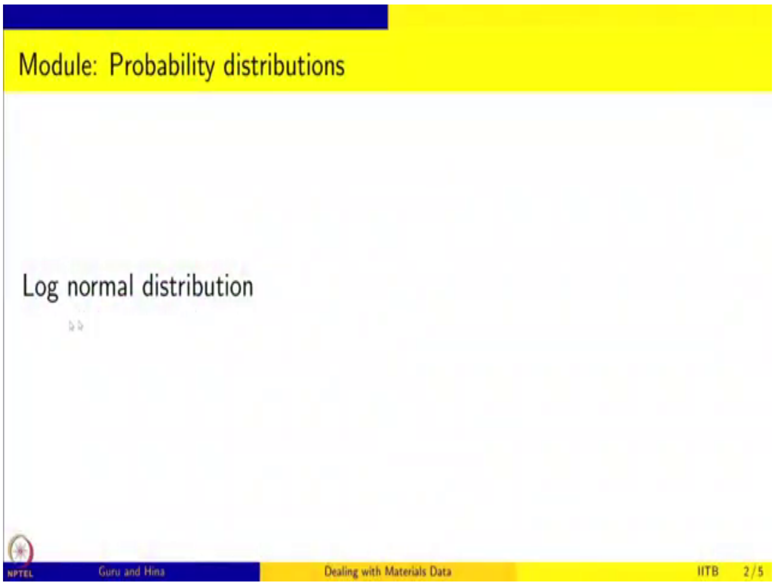
Professor M P Gururajan
Professor Hina A Gokhale
Department of Metallurgical Engineering and Materials Science
Indian Institute of Technology, Bombay

Lecture 55
Log Normal Distribution

Welcome to Dealing with Materials Data, this is a course on collection, analysis and interpretation of data from material science and engineering. We are looking at some of the R-tutorials. So, we had an introduction to R and then we learned how to describe data using R and this is the module on probability distributions.

And in this module, we have looked at discrete distributions. We have also looked at the uniform distribution which is sorry normal distribution which is a continuous distribution. And we are going to continue with continuous distributions.

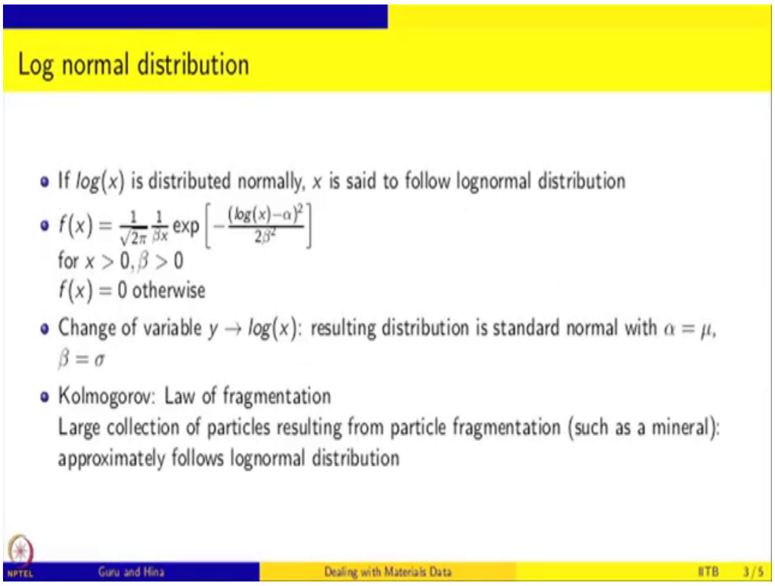
(Refer Slide Time: 0:53)



Module: Probability distributions

Log normal distribution

NPTEL Guru and Hina Dealing with Materials Data IITB 2 / 5



Log normal distribution

- If $log(x)$ is distributed normally, x is said to follow lognormal distribution
- $f(x) = \frac{1}{\sqrt{2\pi}}\frac{1}{\beta x}\exp\left[-\frac{(log(x)-\alpha)^2}{2\beta^2}\right]$
  for $x > 0, \beta > 0$
  $f(x) = 0$ otherwise
- Change of variable $y \rightarrow log(x)$: resulting distribution is standard normal with $\alpha = \mu$, $\beta = \sigma$
- Kolmogorov: Law of fragmentation
  Large collection of particles resulting from particle fragmentation (such as a mineral): approximately follows lognormal distribution

NPTEL Guru and Hina Dealing with Materials Data IITB 3 / 5

$$f(x) = \frac{1}{\sqrt{2\pi}}\frac{1}{\beta x}\exp\left[\frac{(\log(x) - \alpha)^2}{2\beta^2}\right]$$

$$for\ x > 0, \beta > 0$$

$$f(x) = 0$$

Change of variable $y \rightarrow \log(x)$: resulting distribution is standard normal with

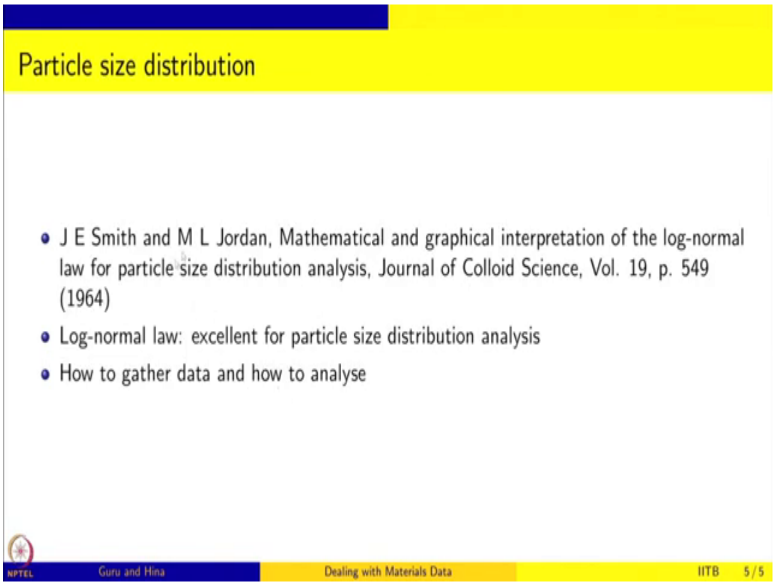
$$\alpha = \mu, \beta = \sigma$$

So Kolmogorov is the one who came up with the law of fragmentation. He showed that large collection of particles which result from particle fragmentation, you know, this is very important in mineralogy, in geology and such areas where you are trying to break and make smaller particles. And in such cases, the particles their size distribution is actually a log number. So, this is what Kolmogorov showed.

And in the case of grain size for example, sometimes it is said that the data follows log normal distribution. I am going to show you one data which comes from a paper of underwood, Ferrite grain size, which we will plot and see that it follows log normal. But if you use our fit distr plus, fit distribution plus library and try to do the fitting, you will see that it is not quite log normal.

And this is a common, in fact, many data sets that is expected to be log normal, I have verified and rarely you get good fit for log normal.

(Refer Slide Time: 03:05)



Particle size distribution

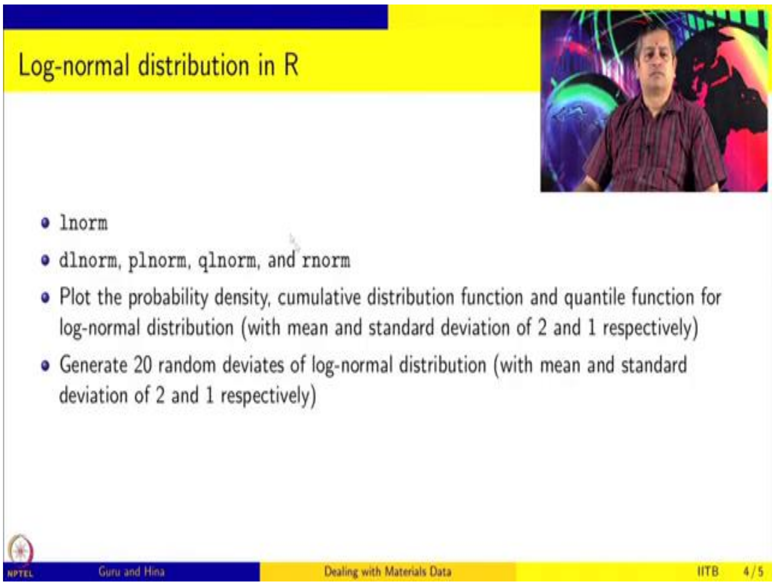
- J E Smith and M L Jordan, Mathematical and graphical interpretation of the log-normal law for particle size distribution analysis, Journal of Colloid Science, Vol. 19, p. 549 (1964)
- Log-normal law: excellent for particle size distribution analysis
- How to gather data and how to analyse

NPTEL
Guru and Hina
Dealing with Materials Data
IITB 5/5

There is one more data set which from Smith and Jordan, so it is says mathematical and graphical interpretation of log number law for particle size distribution analysis from Journal of Colloid Science. And they also say that log normal law is excellent for particle size distribution analysis.

And they also describe in their paper, how to gather data and how to analyze the data for log number of distributions. So, we will take data which is given in this paper and try to see if it follows log normal and also try to generate from our R-model, the data and try to see if we can compare the distribution that we generate with the empirical data and say anything about the distribution.

(Refer Slide Time: 03:56)



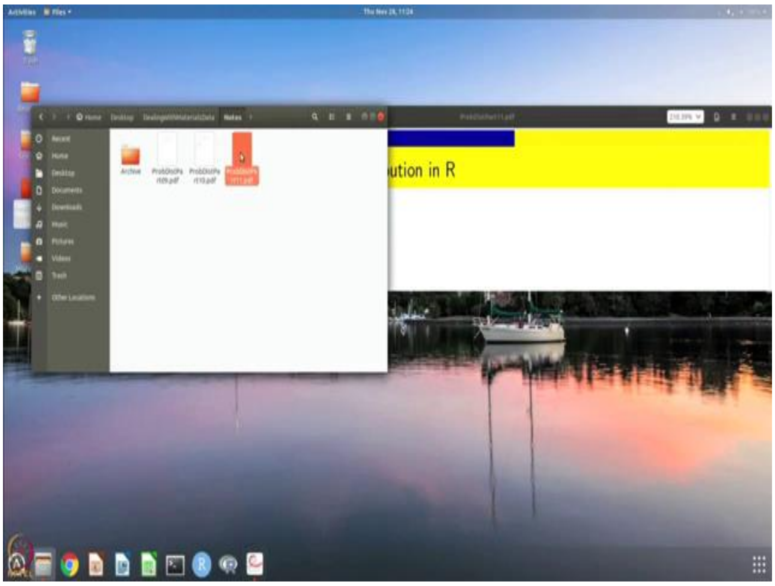
Log-normal distribution in R

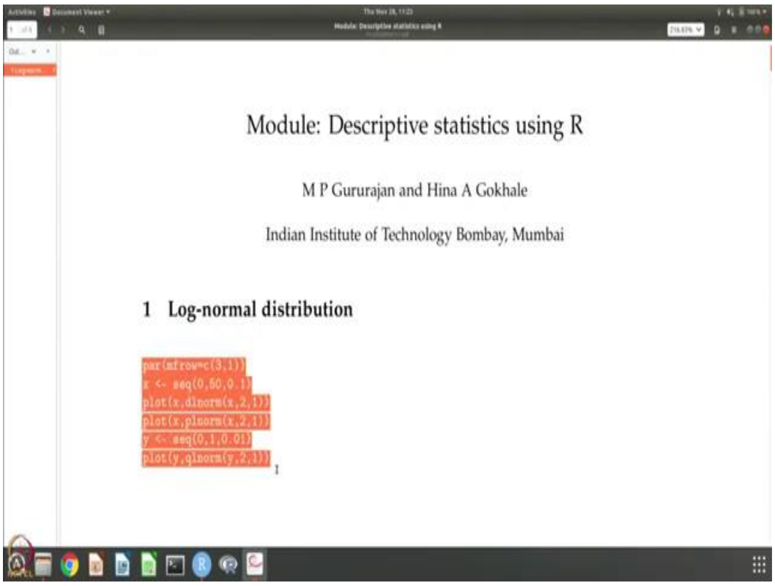
- lnorm
- dlnorm, plnorm, qlnorm, and rnorm
- Plot the probability density, cumulative distribution function and quantile function for log-normal distribution (with mean and standard deviation of 2 and 1 respectively)
- Generate 20 random deviates of log-normal distribution (with mean and standard deviation of 2 and 1 respectively)

Of course, the log normal distribution for in R, the command is lnorm. So, dlnorm, plnorm, qlnorm, rlnorm are the commands as a function calls. So, you can get the probability density cumulative distribution function and quantile function using these 3 functions. The random deviates are generated using rlnorm.

So, we are going to use standard mean of 2 and standard deviation of 1 and we are going to generate these quantities just to check. So, we will now do the R tutorial for log normal distribution

(Refer Slide Time: 04:31)





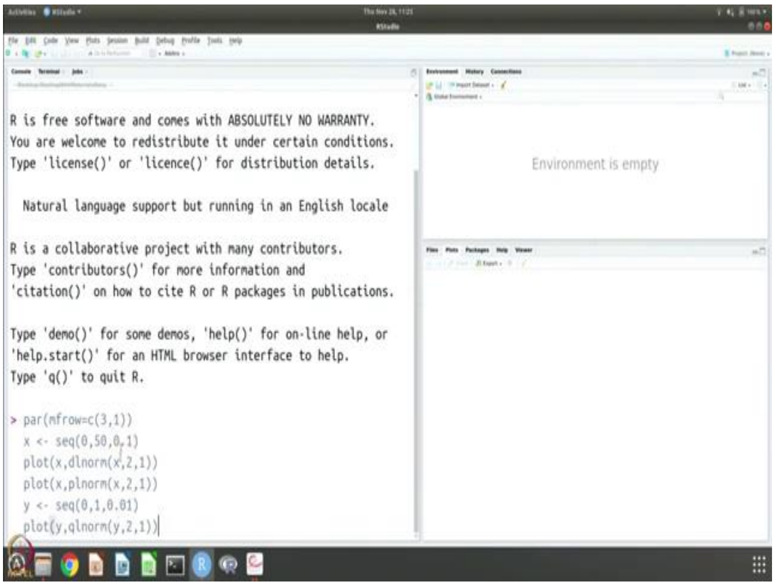
Module: Descriptive statistics using R

M P Gururajan and Hina A Gokhale

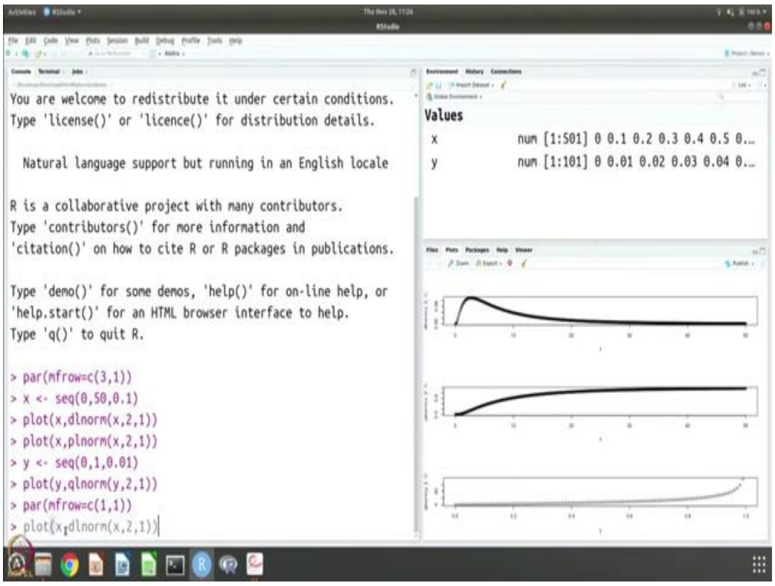
Indian Institute of Technology Bombay, Mumbai

1 Log-normal distribution

```
par(mfrow=c(3,1))
x <- seq(0,50,0.1)
plot(x,dlnorm(x,2,1))
plot(x,plnorm(x,2,1))
y <- seq(0,1,0.01)
plot(y,qlnorm(y,2,1))
```



```
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> par(mfrow=c(3,1))
  x <- seq(0,50,0.1)
  plot(x,dlnorm(x,2,1))
  plot(x,plnorm(x,2,1))
  y <- seq(0,1,0.01)
  plot(y,qlnorm(y,2,1))
```



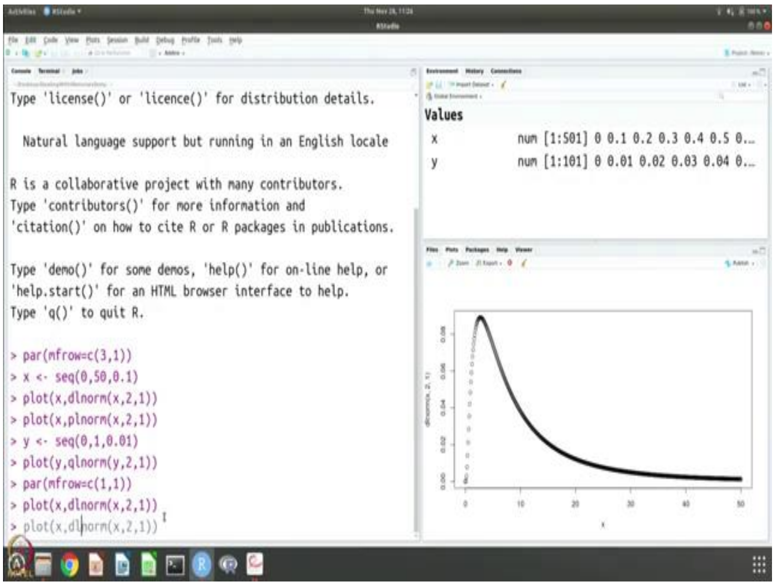
```
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> par(mfrow=c(3,1))
> x <- seq(0,50,0.1)
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> y <- seq(0,1,0.01)
> plot(y,qlnorm(y,2,1))
> par(mfrow=c(1,1))
> plot(x,dlnorm(x,2,1))
```

```
Values
x          num [1:501] 0 0.1 0.2 0.3 0.4 0.5 0...
y          num [1:101] 0 0.01 0.02 0.03 0.04 0...
```

The first exercise as usual, we are going to make 3 plots. And we are going to plot between 0 and 50. And the first one is log normal, the probability distribution function. The second one is a cumulative distribution function. And as we indicated for dlnom, the mean log is to a 2 and standard deviation log is 1.

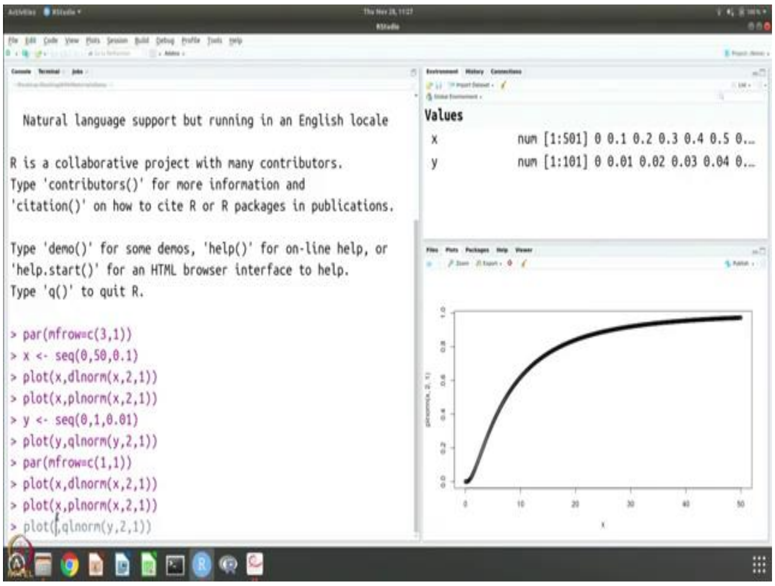
So, that is a value we are using. So, you can see the mean log 0, standard log 1. Standard deviation of log 1 is what by default it uses, but you can change those values. And of course, I am also going to do the quantile plot. So, there are going to be 3 plots. So, you can see that this is the distribution and this is the cumulative distribution function and this is the quantile plot. Of course, you can plot just the plots individually to get a better idea how they look.

(Refer Slide Time: 06:13)



```
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> par(mfrow=c(3,1))
> x <- seq(0,50,0.1)
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> y <- seq(0,1,0.01)
> plot(y,qlnorm(y,2,1))
> par(mfrow=c(1,1))
> plot(x,dlnorm(x,2,1))
> plot(x,dlnorm(x,2,1))
```

```
Values
x          num [1:501] 0 0.1 0.2 0.3 0.4 0.5 0...
y          num [1:101] 0 0.01 0.02 0.03 0.04 0...
```

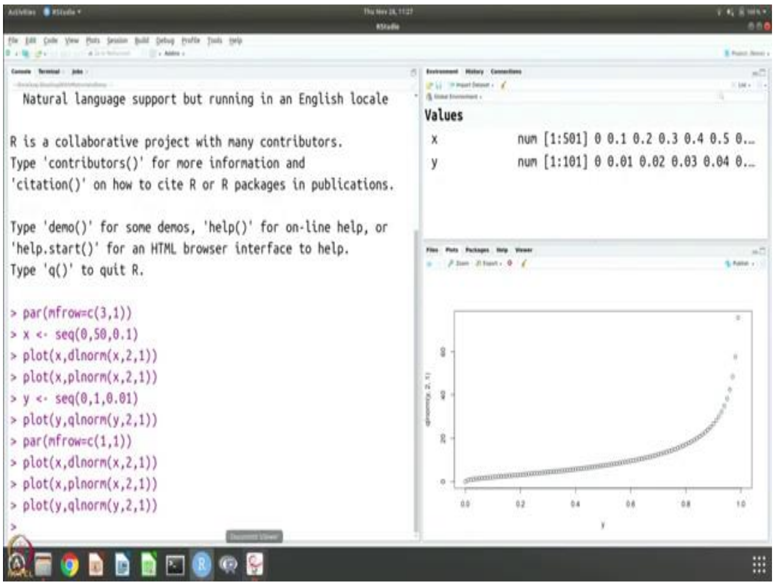


```
Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> par(mfrow=c(3,1))
> x <- seq(0,50,0.1)
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> y <- seq(0,1,0.01)
> plot(y,qlnorm(y,2,1))
> par(mfrow=c(1,1))
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> plot(y,qlnorm(y,2,1))
```

Values

```
x          num [1:501] 0 0.1 0.2 0.3 0.4 0.5 0...
y          num [1:101] 0 0.01 0.02 0.03 0.04 0...
```

Environment History Connections
Files Plots Packages Help Viewer
Zoom
Export
Publish
plnorm(x, 2, 1)
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50
x



```
Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> par(mfrow=c(3,1))
> x <- seq(0,50,0.1)
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> y <- seq(0,1,0.01)
> plot(y,qlnorm(y,2,1))
> par(mfrow=c(1,1))
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> plot(y,qlnorm(y,2,1))
>
```

Values

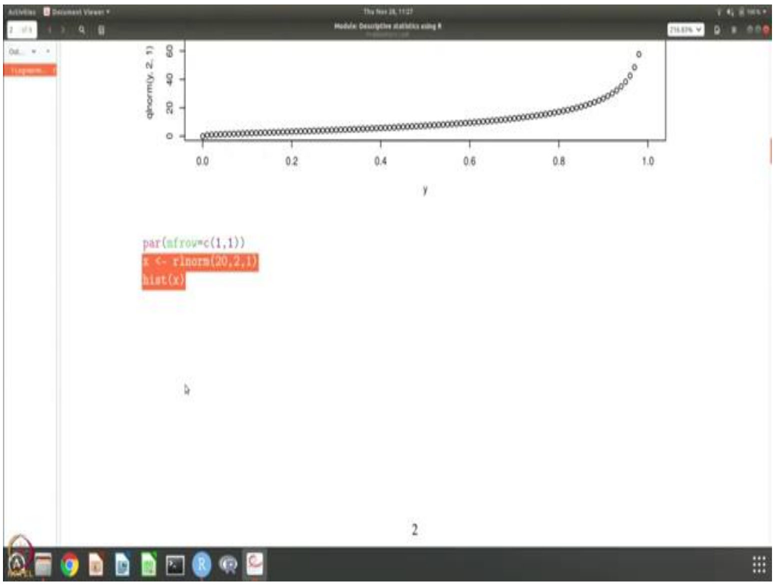
```
x          num [1:501] 0 0.1 0.2 0.3 0.4 0.5 0...
y          num [1:101] 0 0.01 0.02 0.03 0.04 0...
```

Environment History Connections
Files Plots Packages Help Viewer
Zoom
Export
Publish
qlnorm(y, 2, 1)
0
20
40
60
0.0
0.2
0.4
0.6
0.8
1.0
y

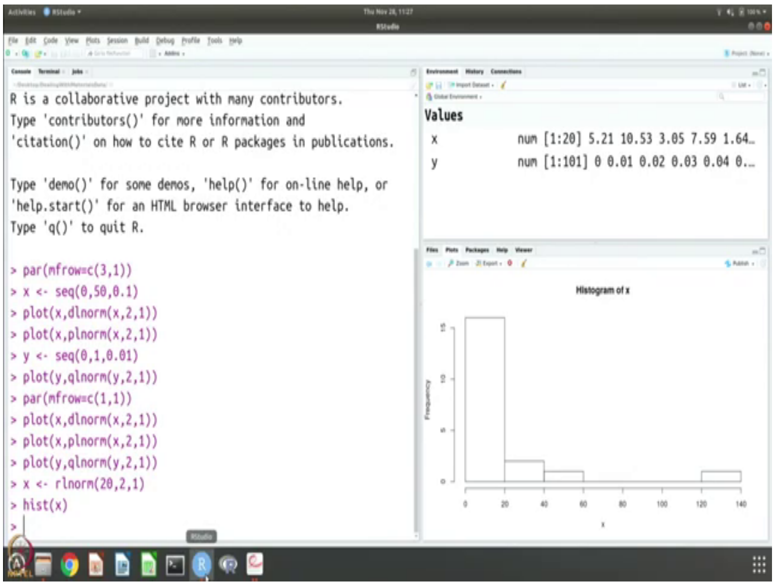
So, this is the distribution function of standard log normal distribution. So, if you see some data follows distribution like this then you expect it to be a log normal. So, that is what we are going to see, you will see many data that looks like this, but it need not be log normal because there are competing distributions which described similar kind of data is what we are going to see.

And of course, we will see the cumulative distribution function goes like that okay and the quantile function, it goes something like this, so because it is the inverse of the cumulative distribution function.

(Refer Slide Time: 07:07)



```
par(mfrow=c(1,1))
x <- rlnorm(20,2,1)
hist(x)
```



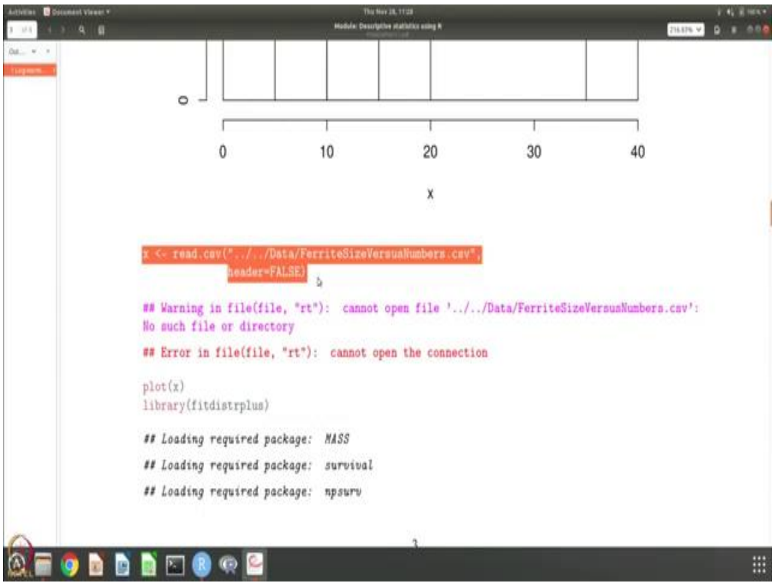
```
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> par(mfrow=c(3,1))
> x <- seq(0,50,0.1)
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> y <- seq(0,1,0.01)
> plot(y,qlnorm(y,2,1))
> par(mfrow=c(1,1))
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> plot(y,qlnorm(y,2,1))
> x <- rlnorm(20,2,1)
> hist(x)
>
```

```
Values
x          num [1:20] 5.21 10.53 3.05 7.59 1.64…
y          num [1:101] 0 0.01 0.02 0.03 0.04 0.…
```

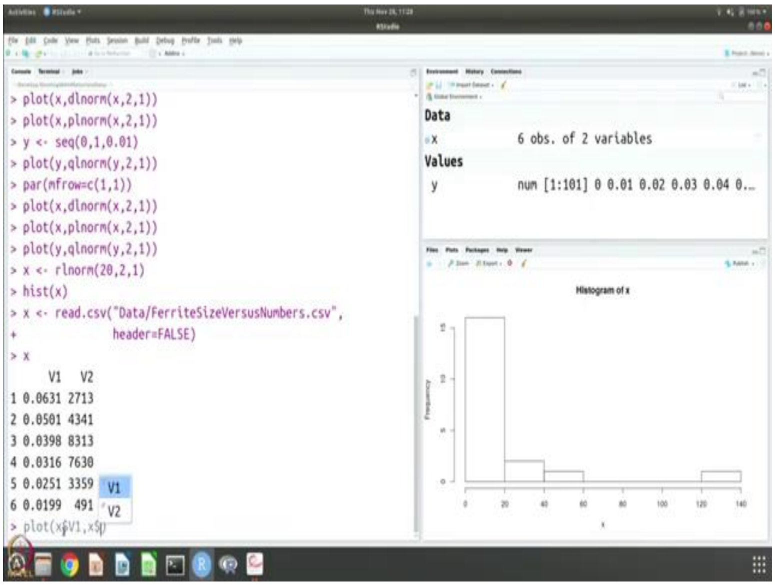
Of course, one can generate random deviates from log normal distribution. And that is what, we will do and plot that data as a histogram and here is that data. So, this generates random deviates from log normal distribution, again with the same mean and standard deviation, and then we are going to have histogram plot.

And you can see that the data goes like this. Okay so, it has a long tail but it peaks somewhere closer here in the beginning and then it goes down.

(Refer Slide Time: 07:43)



```
x <- read.csv("../../Data/FerriteSizeVersusNumbers.csv",
              header=FALSE)

## Warning in file(file, "rt"):  cannot open file '../../Data/FerriteSizeVersusNumbers.csv':
No such file or directory

## Error in file(file, "rt"):  cannot open the connection

plot(x)
library(fitdistrplus)

## Loading required package:  MASS

## Loading required package:  survival

## Loading required package:  npsurv
```



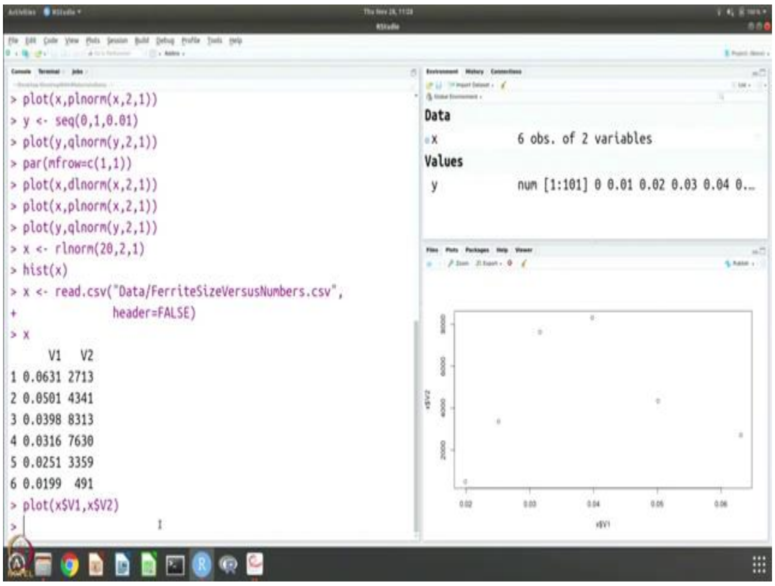
```
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> y <- seq(0,1,0.01)
> plot(y,qlnorm(y,2,1))
> par(mfrow=c(1,1))
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> plot(y,qlnorm(y,2,1))
> x <- rlnorm(20,2,1)
> hist(x)
> x <- read.csv("Data/FerriteSizeVersusNumbers.csv",
+               header=FALSE)
> x
      V1   V2
1 0.0631 2713
2 0.0501 4341
3 0.0398 8313
4 0.0316 7630
5 0.0251 3359
6 0.0199  491
> plot(x$V1,x$)
```

Data

x   6 obs. of 2 variables

Values

y   num [1:101] 0 0.01 0.02 0.03 0.04 0...



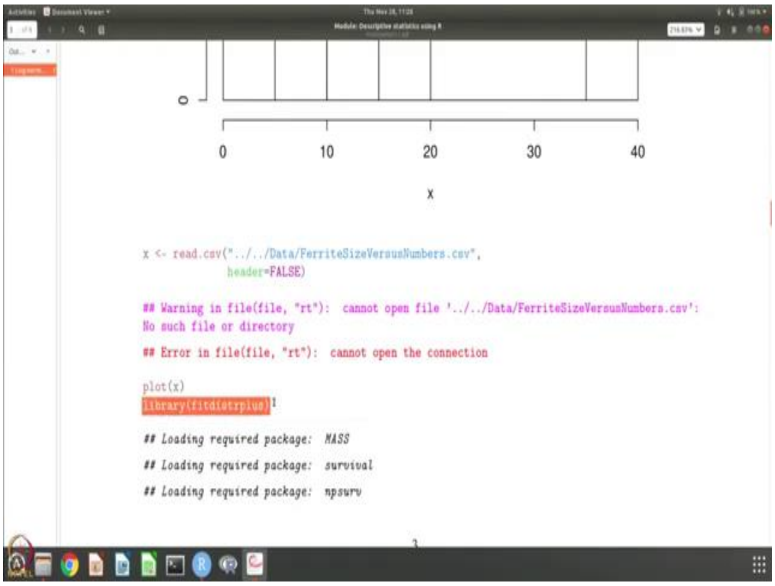
```
> plot(x,plnorm(x,2,1))
> y <- seq(0,1,0.01)
> plot(y,qlnorm(y,2,1))
> par(mfrow=c(1,1))
> plot(x,dlnorm(x,2,1))
> plot(x,plnorm(x,2,1))
> plot(y,qlnorm(y,2,1))
> x <- rlnorm(20,2,1)
> hist(x)
> x <- read.csv("Data/FerriteSizeVersusNumbers.csv",
+              header=FALSE)
> x
      V1   V2
1 0.0631 2713
2 0.0501 4341
3 0.0398 8313
4 0.0316 7630
5 0.0251 3359
6 0.0199  491
> plot(x$V1,x$V2)
>
```

Data
x   6 obs. of 2 variables
Values
y   num [1:101] 0 0.01 0.02 0.03 0.04 0...

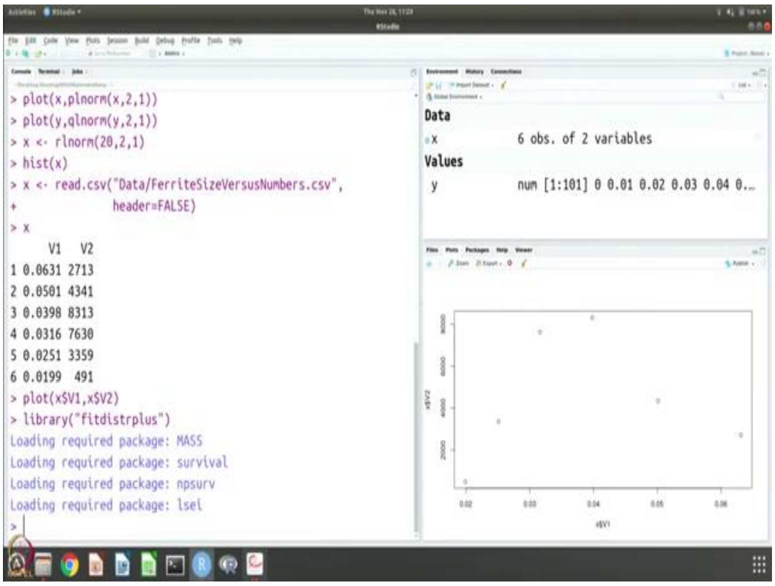
So, let us take a look at a couple of data sets. The first one that I want to use is from underwood. And so let us read that data first. So, it is for ferrite size versus numbers that is what Underwood has given. So this is the size and these are the numbers. So, if you plot, so you see that the data goes like this.

So, Underwood says that this could be expected to be log numbers normal approximately and let us check.

(Refer Slide Time: 08:21)

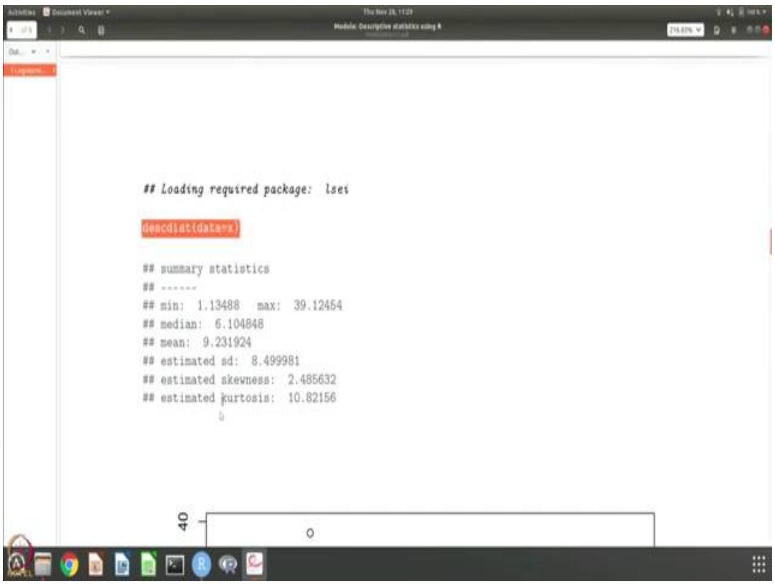


```
x <- read.csv("../../Data/FerriteSizeVersusNumbers.csv",
          header=FALSE)

## Warning in file(file, "rt"):  cannot open file '../../Data/FerriteSizeVersusNumbers.csv':
No such file or directory

## Error in file(file, "rt"):  cannot open the connection

plot(x)
library(fitdistrplus)1

## Loading required package:  MASS

## Loading required package:  survival

## Loading required package:  npsurv
```

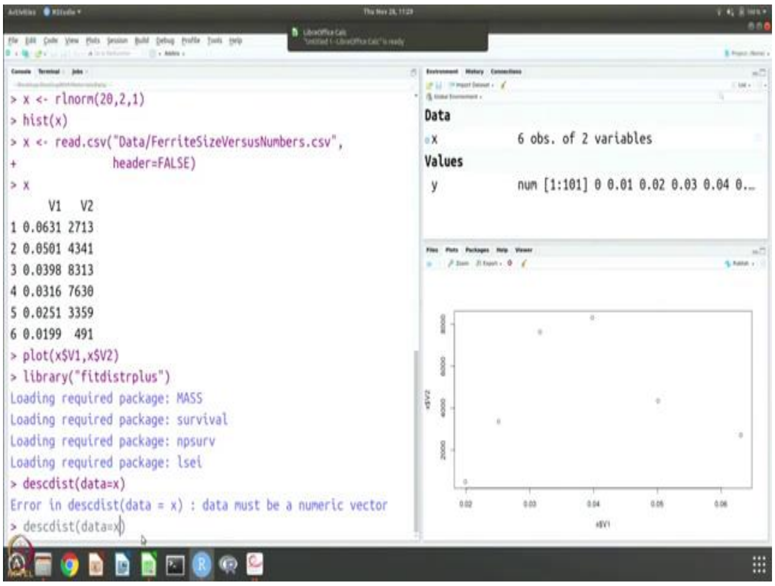


```
> plot(x,plnorm(x,2,1))
> plot(y,qlnorm(y,2,1))
> x <- rlnorm(20,2,1)
> hist(x)
> x <- read.csv("Data/FerriteSizeVersusNumbers.csv",
+              header=FALSE)
> x
      V1   V2
1 0.0631 2713
2 0.0501 4341
3 0.0398 8313
4 0.0316 7630
5 0.0251 3359
6 0.0199  491
> plot(x$V1,x$V2)
> library("fitdistrplus")
Loading required package: MASS
Loading required package: survival
Loading required package: npsurv
Loading required package: lsei
>
```

```
Data
x          6 obs. of 2 variables
Values
y          num [1:101] 0 0.01 0.02 0.03 0.04 0...
```

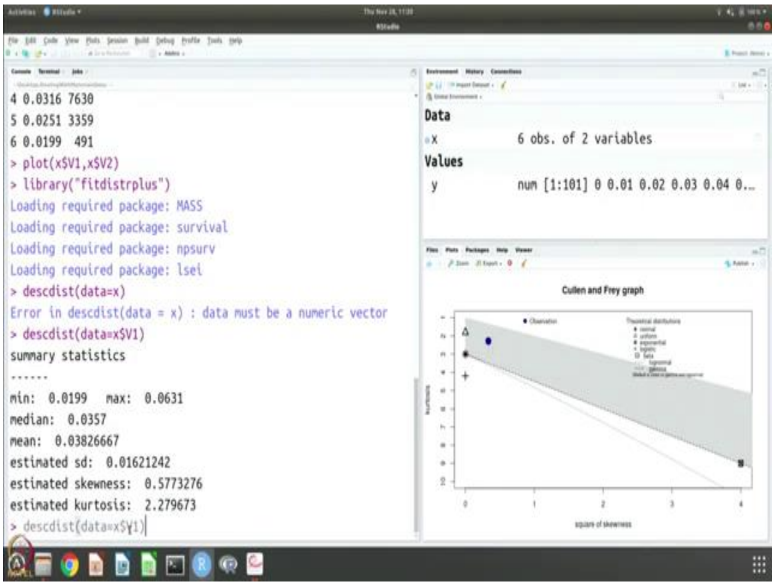


```
## Loading required package:  lsei

descdist(data=x)

## summary statistics
## ------
## min:  1.13488   max:  39.12464
## median:  6.104848
## mean:  9.231924
## estimated sd:  8.499981
## estimated skewness:  2.485632
## estimated kurtosis:  10.82156
```

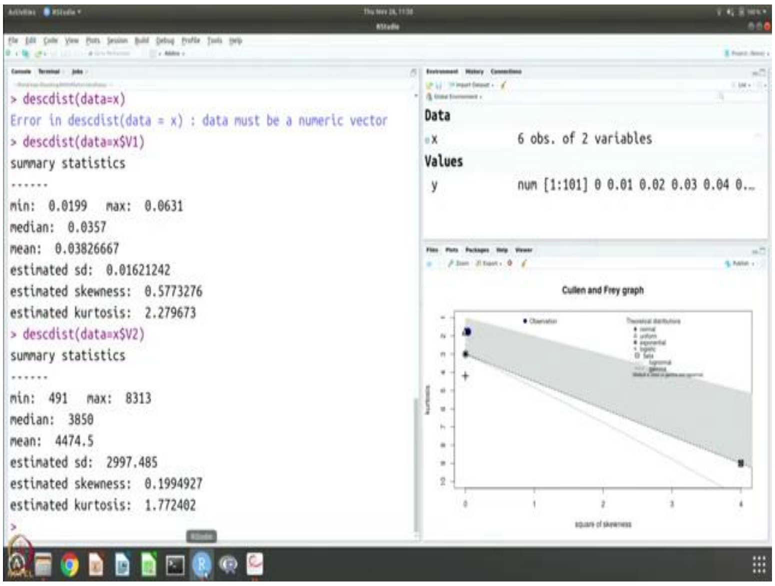


```
> x <- rlnorm(20,2,1)
> hist(x)
> x <- read.csv("Data/FerriteSizeVersusNumbers.csv",
+              header=FALSE)
> x
      V1   V2
1 0.0631 2713
2 0.0501 4341
3 0.0398 8313
4 0.0316 7630
5 0.0251 3359
6 0.0199  491
> plot(x$V1,x$V2)
> library("fitdistrplus")
Loading required package: MASS
Loading required package: survival
Loading required package: npsurv
Loading required package: lsei
> descdist(data=x)
Error in descdist(data = x) : data must be a numeric vector
> descdist(data=x)
```

```
Data
x          6 obs. of 2 variables
Values
y          num [1:101] 0 0.01 0.02 0.03 0.04 0...
```


Environment History Connections
Data
x 6 obs. of 2 variables
Values
y num [1:101] 0 0.01 0.02 0.03 0.04 0...
Files Plots Packages Help Viewer

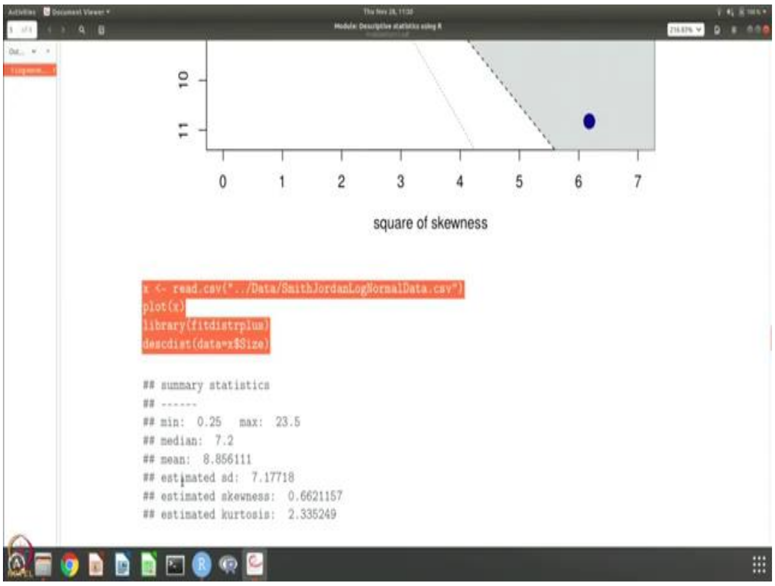
```
4 0.0316 7630
5 0.0251 3359
6 0.0199  491
> plot(x$V1,x$V2)
> library("fitdistrplus")
Loading required package: MASS
Loading required package: survival
Loading required package: npsurv
Loading required package: lsei
> descdist(data=x)
Error in descdist(data = x) : data must be a numeric vector
> descdist(data=x$V1)
summary statistics
------
min:  0.0199   max:  0.0631
median:  0.0357
mean:  0.03826667
estimated sd:  0.01621242
estimated skewness:  0.5773276
estimated kurtosis:  2.279673
> descdist(data=x$V1)
```


Environment History Connections
Data
x 6 obs. of 2 variables
Values
y num [1:101] 0 0.01 0.02 0.03 0.04 0...
Files Plots Packages Help Viewer

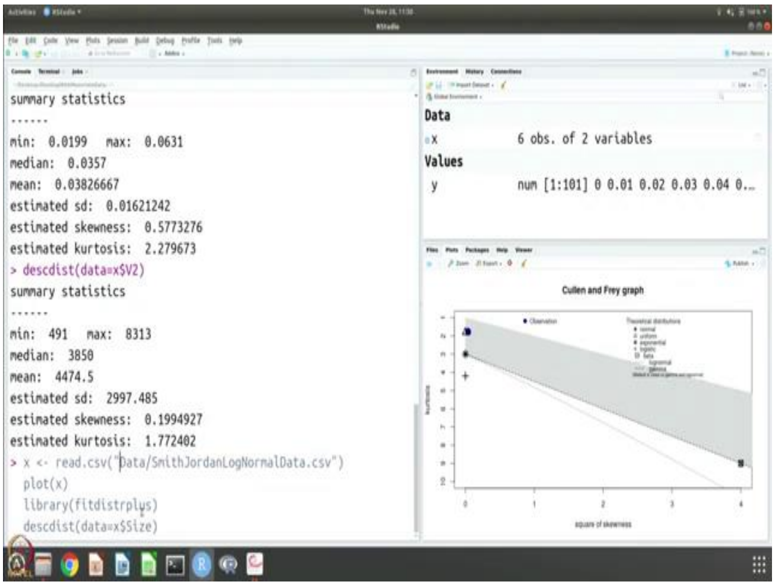
```
> descdist(data=x)
Error in descdist(data = x) : data must be a numeric vector
> descdist(data=x$V1)
summary statistics
------
min:  0.0199   max:  0.0631
median:  0.0357
mean:  0.03826667
estimated sd:  0.01621242
estimated skewness:  0.5773276
estimated kurtosis:  2.279673
> descdist(data=x$V2)
summary statistics
------
min:  491   max:  8313
median:  3850
mean:  4474.5
estimated sd:  2997.485
estimated skewness:  0.1994927
estimated kurtosis:  1.772402
>
```

So, we want to use the library fit distr plus then we want to, okay so we want to take this data and we want to check whether our data follows. As you can see, if it try to look at the data, then it does not follow log normal really, log number is somewhere here and our observation lies somewhere in beta. This is for V1, you can look at V2, in fact V2 is more or less like uniform. So, it is clear that difficult to see that this data follows log normal distribution.

(Refer Slide Time: 09:46)

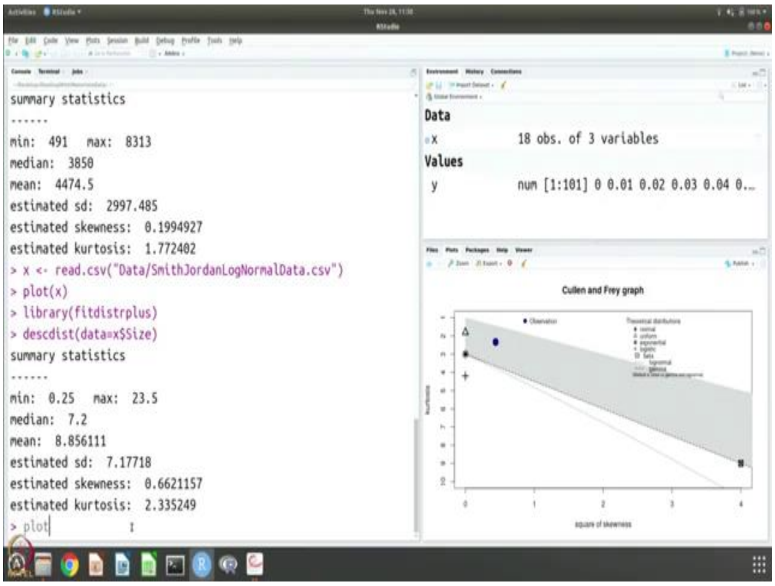


```
x <- read.csv("../Data/SmithJordanLogNormalData.csv")
plot(x)
library(fitdistrplus)
descdist(data=x$Size)

## summary statistics
## ------
## min:  0.25   max:  23.5
## median:  7.2
## mean:  8.856111
## estimated sd:  7.17718
## estimated skewness:  0.6621157
## estimated kurtosis:  2.335249
```



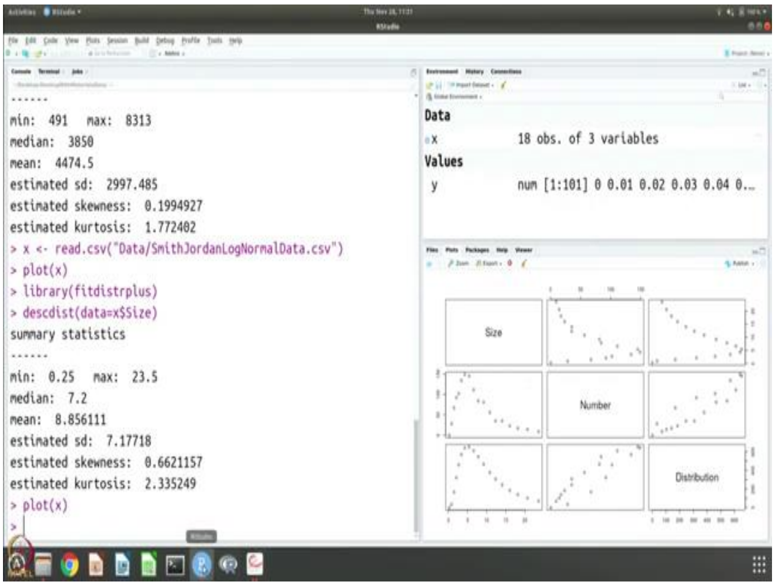
```
summary statistics
------
min:  0.0199   max:  0.0631
median:  0.0357
mean:  0.03826667
estimated sd:  0.01621242
estimated skewness:  0.5773276
estimated kurtosis:  2.279673
> descdist(data=x$V2)
summary statistics
------
min:  491   max:  8313
median:  3850
mean:  4474.5
estimated sd:  2997.485
estimated skewness:  0.1994927
estimated kurtosis:  1.772402
> x <- read.csv("Data/SmithJordanLogNormalData.csv")
  plot(x)
  library(fitdistrplus)
  descdist(data=x$Size)
```

```
Data
x          6 obs. of 2 variables
Values
y          num [1:101] 0 0.01 0.02 0.03 0.04 0...
```



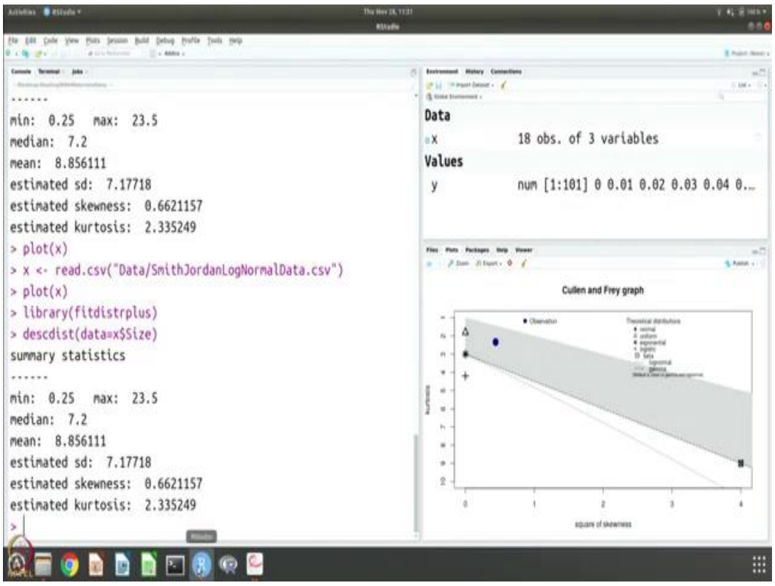
```
summary statistics
------
min:  491   max:  8313
median:  3850
mean:  4474.5
estimated sd:  2997.485
estimated skewness:  0.1994927
estimated kurtosis:  1.772402
> x <- read.csv("Data/SmithJordanLogNormalData.csv")
> plot(x)
> library(fitdistrplus)
> descdist(data=x$Size)
summary statistics
------
min:  0.25   max:  23.5
median:  7.2
mean:  8.856111
estimated sd:  7.17718
estimated skewness:  0.6621157
estimated kurtosis:  2.335249
> plot
```

```
Data
x          18 obs. of 3 variables
Values
y          num [1:101] 0 0.01 0.02 0.03 0.04 0...
```



```
......
min:  491   max:  8313
median:  3850
mean:  4474.5
estimated sd:  2997.485
estimated skewness:  0.1994927
estimated kurtosis:  1.772402
> x <- read.csv("Data/SmithJordanLogNormalData.csv")
> plot(x)
> library(fitdistrplus)
> descdist(data=x$Size)
summary statistics
------
min:  0.25   max:  23.5
median:  7.2
mean:  8.856111
estimated sd:  7.17718
estimated skewness:  0.6621157
estimated kurtosis:  2.335249
> plot(x)
>
```

```
Data
x            18 obs. of 3 variables
Values
y            num [1:101] 0 0.01 0.02 0.03 0.04 0...
```



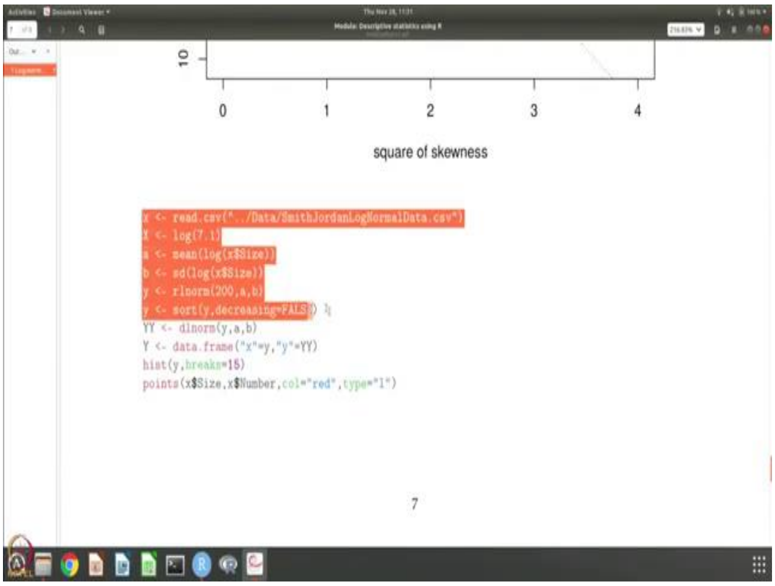
```
......
min:  0.25   max:  23.5
median:  7.2
mean:  8.856111
estimated sd:  7.17718
estimated skewness:  0.6621157
estimated kurtosis:  2.335249
> plot(x)
> x <- read.csv("Data/SmithJordanLogNormalData.csv")
> plot(x)
> library(fitdistrplus)
> descdist(data=x$Size)
summary statistics
------
min:  0.25   max:  23.5
median:  7.2
mean:  8.856111
estimated sd:  7.17718
estimated skewness:  0.6621157
estimated kurtosis:  2.335249
>
```

```
Data
x            18 obs. of 3 variables
Values
y            num [1:101] 0 0.01 0.02 0.03 0.04 0...
```

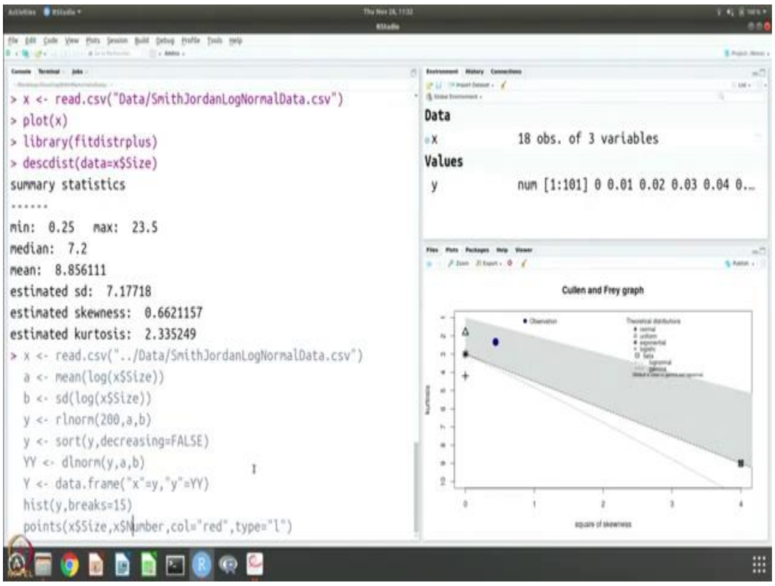
And there is another data, which is from Smith and Jordan, like I told you and let us try to load that data and see what happens okay, so we want to read the Smith-Jordan log normal data. We want to plot x and then we are going to use fit distribution plus library and describe the data of size okay.

So, again here again the data seems to be in the beta, it is not really in, however if you look at the data, so you can see that it does look like log normal distribution very nicely right. So, even though it looks nicely like this, when we try to do the fit distr plus you see that it says that the data is not really following log normal, log number means it should have been somewhere here but observation falls somewhere in beta. So, this is a problem, it is very difficult to actually know.

(Refer Slide Time: 11:01)

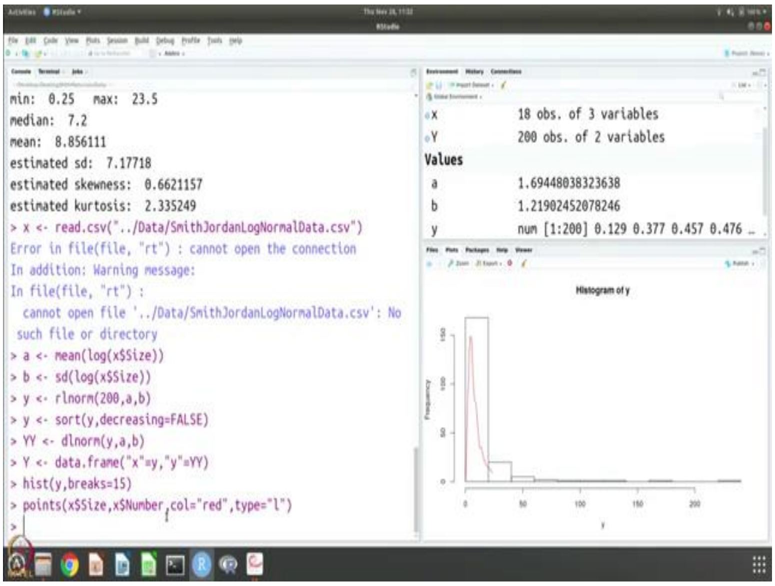


```
x <- read.csv("../Data/SmithJordanLogNormalData.csv")
X <- log(7.1)
a <- mean(log(x$Size))
b <- sd(log(x$Size))
y <- rlnorm(200,a,b)
y <- sort(y,decreasing=FALSE)
YY <- dlnorm(y,a,b)
Y <- data.frame("x"=y,"y"=YY)
hist(y,breaks=15)
points(x$Size,x$Number,col="red",type="l")
```



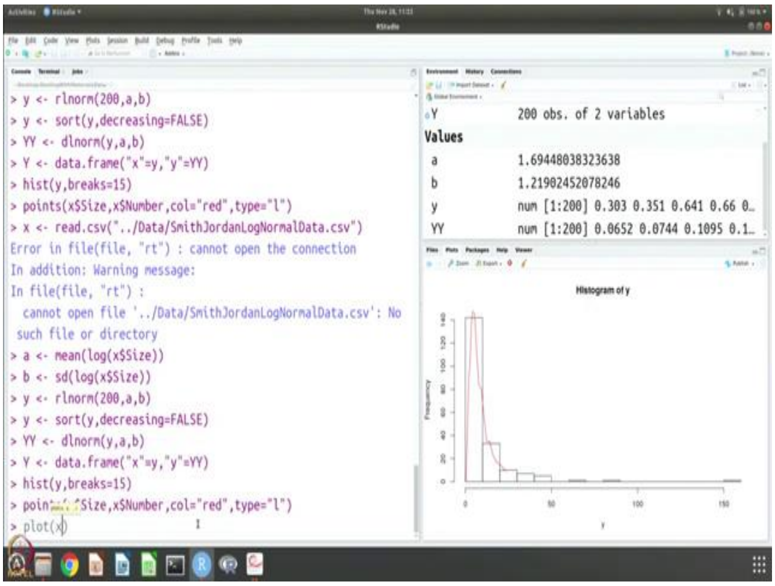
```
> x <- read.csv("Data/SmithJordanLogNormalData.csv")
> plot(x)
> library(fitdistrplus)
> descdist(data=x$Size)
summary statistics
------
min:  0.25   max:  23.5
median:  7.2
mean:  8.856111
estimated sd:  7.17718
estimated skewness:  0.6621157
estimated kurtosis:  2.335249
> x <- read.csv("../Data/SmithJordanLogNormalData.csv")
  a <- mean(log(x$Size))
  b <- sd(log(x$Size))
  y <- rlnorm(200,a,b)
  y <- sort(y,decreasing=FALSE)
  YY <- dlnorm(y,a,b)
  Y <- data.frame("x"=y,"y"=YY)
  hist(y,breaks=15)
  points(x$Size,x$Number,col="red",type="l")
```

```
Data
x          18 obs. of 3 variables
Values
y          num [1:101] 0 0.01 0.02 0.03 0.04 0...
```



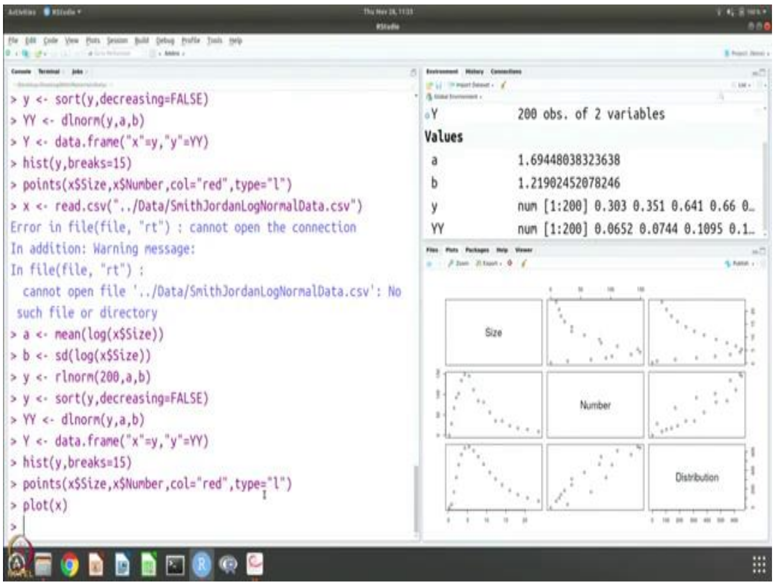
```
min:  0.25   max:  23.5
median:  7.2
mean:  8.856111
estimated sd:  7.17718
estimated skewness:  0.6621157
estimated kurtosis:  2.335249
> x <- read.csv("../Data/SmithJordanLogNormalData.csv")
Error in file(file, "rt") : cannot open the connection
In addition: Warning message:
In file(file, "rt") :
  cannot open file '../Data/SmithJordanLogNormalData.csv': No
 such file or directory
> a <- mean(log(x$Size))
> b <- sd(log(x$Size))
> y <- rlnorm(200,a,b)
> y <- sort(y,decreasing=FALSE)
> YY <- dlnorm(y,a,b)
> Y <- data.frame("x"=y,"y"=YY)
> hist(y,breaks=15)
> points(x$Size,x$Number,col="red",type="l")
>
```

```
x          18 obs. of 3 variables
Y          200 obs. of 2 variables
Values
a          1.69448038323638
b          1.21902452078246
y          num [1:200] 0.129 0.377 0.457 0.476 ...
```



```
> y <- rlnorm(200,a,b)
> y <- sort(y,decreasing=FALSE)
> YY <- dlnorm(y,a,b)
> Y <- data.frame("x"=y,"y"=YY)
> hist(y,breaks=15)
> points(x$Size,x$Number,col="red",type="l")
> x <- read.csv("../Data/SmithJordanLogNormalData.csv")
Error in file(file, "rt") : cannot open the connection
In addition: Warning message:
In file(file, "rt") :
  cannot open file '../Data/SmithJordanLogNormalData.csv': No
 such file or directory
> a <- mean(log(x$Size))
> b <- sd(log(x$Size))
> y <- rlnorm(200,a,b)
> y <- sort(y,decreasing=FALSE)
> YY <- dlnorm(y,a,b)
> Y <- data.frame("x"=y,"y"=YY)
> hist(y,breaks=15)
> poin[illegible]Size,x$Number,col="red",type="l")
> plot(x)
```

```
Y               200 obs. of 2 variables
Values
 a              1.69448038323638
 b              1.21902452078246
 y              num [1:200] 0.303 0.351 0.641 0.66 0…
 YY             num [1:200] 0.0652 0.0744 0.1095 0.1…
```



```
> y <- sort(y,decreasing=FALSE)
> YY <- dlnorm(y,a,b)
> Y <- data.frame("x"=y,"y"=YY)
> hist(y,breaks=15)
> points(x$Size,x$Number,col="red",type="l")
> x <- read.csv("../Data/SmithJordanLogNormalData.csv")
Error in file(file, "rt") : cannot open the connection
In addition: Warning message:
In file(file, "rt") :
  cannot open file '../Data/SmithJordanLogNormalData.csv': No
 such file or directory
> a <- mean(log(x$Size))
> b <- sd(log(x$Size))
> y <- rlnorm(200,a,b)
> y <- sort(y,decreasing=FALSE)
> YY <- dlnorm(y,a,b)
> Y <- data.frame("x"=y,"y"=YY)
> hist(y,breaks=15)
> points(x$Size,x$Number,col="red",type="l")
> plot(x)
>
```

```
Y               200 obs. of 2 variables
Values
 a              1.69448038323638
 b              1.21902452078246
 y              num [1:200] 0.303 0.351 0.641 0.66 0…
 YY             num [1:200] 0.0652 0.0744 0.1095 0.1…
```

And there are other competing distributions, which will also give and something like beta, which, by changing parameters you can fit the data well might do that. So, it is really difficult sometimes to know which is the right distribution that the data follows. Even though if you know for physical reasons that the data is expected to follow a distribution that is the distribution you should use.

So, we are again going to take a look at the Smith Jordan log normal data and I am going to calculate the mean and standard deviation of the size data and I am going to generate random deviates with that mean and standard deviation from the log normal. And then I am going to plot it then I am willing to plot the data.

Then we will see whether there is a better matching that we can see. And of course, you can see that the histogram of data that I generated with the same mean and standard deviation looks like this and our data also looks like this. So, it does look like we have data, so every time I run you get a different distribution because the random deviates are different. So, you can see that every time the deviates that you generate seem to fit very well the data which is not surprising.

Because from by looking at the data for example, you can see that it looks like the log normal. So, in the case of grain size and such fragmented particle size etc it is expected that the distribution is log normal. So, it is always useful to try to see how closely does the log normal distribution described the data, so log normal is an important distribution.

So, it is used especially in areas like this, where there is reason to believe, based on some of the theories like kolmographs, law fragmentation, for example, that the data is expected to follow log normal distribution. But sometimes for example, grain size there are other competing distributions that will describe what is happening. We have also seen in some cases, the grain sizes, it was very different. We have seen data while we were doing descriptive statistics.

So, especially grain size set, data is very difficult to say that it should always follow log normal. But in other cases where you expect log normal, you will try to fit the data to log normal and see, even though if you do blindly and try to fit the data to available distributions, there are other competing distributions which will show up and probably show that they have better fit to your data.

So, it depends on your needs and purposes. If you know for sure that the data should follow given distribution that is what you should try to fit for. If you just try to get a description does not matter whatever distribution that you can get, then of course, you can explore and find the distribution that describes your data the best.

So, this is log normal distribution and like I said, I have found it very difficult to find any data that if you use fit dist r plus will show that it is log normal. So, I am not sure unless maybe if you just generate random deviates and give it to fit d i s t r plus it will show that it is log normal. In all other cases, I have found that there are always competing distributions and most of the times that is beta that it shows to be better fitting. But it is a good exercise for you to go look up for data.

As part of this course, you should also tried yourself to go look for data or generate some such data and try to do the analysis and see if you can get better data that fits log normal distribution. Thank you.