

**Essentials of Data Science with R Software – 1**  
**Professor Shalabh**  
**Department of Mathematics and Statistics**  
**Indian Institute of Technology Kanpur**  
**Lecture No. 37**  
**Geometric Distribution in R**

Hello friends, welcome to the course Essentials of Data Science with R Software – 1; in which we are trying to understand the fundamentals of probability theory and statistical inference; which are possibly going to be useful in data science. So, you can recall that in the last lecture, we had understood the geometric distribution. Now, the next question is how you can compute different types of probabilities? How you can compute the quantiles? How can you compute the CDF? How you can generate the random numbers from the geometric distribution in the R software?

That is the question which we are going to handle in this lecture. So, now I am sure that you will be very comfortable with this type of lecture, because the things are going to be on the similar lines; what we have done in the case of discrete, uniform, binomial, Poisson etc. But, here in this lecture there is going to be one difference; the difference is when you took or when you consider the binomial or Poisson distribution, you had the parameters say  $n$  and  $p$  in the binomial and  $\lambda$  in the Poisson distribution. And you use  $n$ ,  $p$ ,  $\lambda$  as such; but when you are trying to use this geometric distribution in R.

So, one of the value which we have used in defining the probability mass function  $k$ . The use of  $k$  is going to be a little bit different than in the theory; so that is why you have to be very careful. Although I will try to make it clear here, but the reason I told you that you should not be over confident. So, let us try to begin our lecture.

(Refer Slide Time: 02:16)

**Geometric Distribution:**  
A discrete random variable  $X$  is said to follow a geometric distribution with parameter  $p$  if its PMF is given by

$$P(X = k) = p(1 - p)^{k-1}, k = 1, 2, 3, \dots$$

The mean (expectation) and variance are given by

$$E(X) = \frac{1}{p}$$
$$Var(X) = \frac{1}{p} \left( \frac{1}{p} - 1 \right)$$

*Handwritten notes:*  $k = 0, 1, 2, 3, \dots$   
 $k \rightarrow$  in our notations  
 $\rightarrow k-1$  in the notations of R s/w.

So, now geometric distribution is defined like as follows. A discrete random variable  $X$  is said to follow a geometric distribution with parameter  $p$ . If its probability mass function is given by, probability of  $X$  equal to  $k$ . Please make a note here which is equal to probability  $p(1 - p)^{k-1}$ ; and  $k = 1, 2, 3, \dots$ . This is what we have considered in the theory. And in R you will see means, I will try to show you, this  $k$  starts from 0, 1, 2, 3 and so on. So, that is why whatever you are using here as say  $k$  in our notations; this is going to be  $k$  minus 1 in the notations of R software.

If you just keep this thing in mind, there is no problem; and the mean and variance that you had already reported. That they are  $1/p$  as the mean and the variance is  $\frac{1}{p} \left( \frac{1}{p} - 1 \right)$ .

(Refer Slide Time: 03:33)

**Geometric Distribution: In R**

Value  $\rightarrow$  *density* *geometric*

dgeom gives the density, *dgeom ( )*

pgeom gives the distribution function,

qgeom gives the quantile function (smallest value  $x$  such that CDF  $F(x) \geq p$ ), and

rgeom generates random deviates.

Invalid prob will result in return value NaN, with a warning.

Now, let us try to understand what are the commands, which are going to give you the different type of density, CDF, quantile etc. in the geometric distribution. So, `dgeom` means density, and `geom` means geometric. This command gives that density that you know that is the probability `prob`. This gives the CDF, `qgeom` this gives us the quantile function; and `rggeom` this generates the random numbers. Definitely if you try to use some invalid `prob` that will result in value `NaN`, with a warning.

(Refer Slide Time: 04:28)

**Geometric Distribution: In R**

Note that the definition of  $X$  in R slightly differs from our definition. In R,  $k$  is the number of failures before the first success.

This means we need to specify  $(k - 1)$  in the `dgeom` function rather than  $k$ .

Details: The geometric distribution with `prob = p` has density  $p(x) = p(1-p)^x$  for  $x = 0, 1, 2, \dots, 0 < p \leq 1$ .

[Note that the PMF we considered is  $p(1-p)^{x-1}$ ]

*we have considered*

So, now let us try to first understand, before we try to understand the parameters inside the parenthesis; that there is a slight difference in the definition of geometric distribution in R, in comparison to what we have done. In R this  $k$  is the number of failures before the first success. This means we need to specify  $k$  minus 1; instead of  $k$  of our notation in the functions like `dgeom`.

So, in the case of geometric distribution in the R software, it is defined as the function `prob`; that will be used for  $p$ . But, that will define in the following way. That  $p(x) = p(1-p)^x$ ; where  $x = 0, 1, 2, \dots$  and so on. That is exactly what I told you in the earlier. What we had considered it was something like  $p(1-p)^{x-1}$ ; so we have considered, so be very careful.

(Refer Slide Time: 05:49)

**Geometric Distribution: In R**

Arguments

`x, q` vector of quantiles representing the number of failures in a sequence of Bernoulli trials before success occurs.

`p` vector of probabilities.

`n` number of observations.

`prob` probability of success in each trial.  $0 < \text{prob} \leq 1$ .

`lower.tail` logical; if **TRUE** (default), probabilities are  $P[X \leq x]$ , otherwise,  $P[X > x]$ .

So, now the values that you have to give inside the parenthesis, they will be like as here. `x` and `q` they are the vector of quantiles, `p` is the vector of probabilities, `n` the number of observations, `prob` is the probability of success in each trial. And `lower.tail`, you know that if this is a logical variable; it takes value **TRUE** or **FALSE**. And if that is **TRUE**, you are going to compute as PDF like probability  $X$  less than equal to `x`. And if you try to take `lower.tail` equal to **FALSE**, then it will compute probability  $X$  greater than `x`.

(Refer Slide Time: 06:23)

**Geometric Distribution: : In R**

We use the following command to draw a curve

```
k = 0:100
prob = dgeom(x=k-1, prob=0.67)
plot(x = k, y = prob, main = "Geometric
Probability Mass Function") #plot outcomes vs.
their probabilities
```

So, let us try to first understand the application of this thing, and then I will try to introduce how are you going to use these values inside the parenthesis in different commands. Intentionally if you try to see, I have not written that how you have to give the `dgeom` inside something like this.

Because there was a confusion, so that is why I have postponed it to the example, from where you are going to use it. So, now as usual what we have done earlier, we simply try to plot the probabilities. And in order to plot it, we are going to take the value of  $k$  starting from 0 to 100; but you can see here that in this case, your  $x$  will be actually  $k$  minus 1.

So, but according to the definition what we have learnt? It should have been  $k$ ; but it does not make any difference. So, I am trying to take care the value of  $k$  and then we are trying to choose here prob equal to 0.67; who can take anything. And then we are trying to compute the different probabilities, and we are trying to plot the probabilities with respect to the value of  $x$ . And the title here will be geometric probability mass function and so on. So, if you try to see this plot will come like this; that initially the probabilities are here like this. But, after sometime they will become almost parallel to the  $x$ -axis close to 0.

(Refer Slide Time: 08:03)

**Geometric Distribution: Example in R**  
**Example:** An urn contains  $N$  white and  $M$  black balls. Balls are randomly selected, one at a time, until a black one is obtained. If we assume that each selected ball is replaced before the next one is drawn, then we already have found the probability that

(a) exactly  $n$  draws are needed as  $P(X = n) = \frac{MN^{n-1}}{(M+N)^n}$

(b) at least  $k$  draws are needed as  $P(X \leq k) = (1 - p)^{k-1}$  where  $p = \frac{M}{M+N}$ .

Let  $N = 5$ ,  $M = 10$ , then  $p = \frac{10}{15} \approx 0.67$ .

We compute the probabilities using R software.

So, now what we do that we try to consider the same example that we considered in the last lecture; and we try to compute those probabilities using the R software. So, in that example we had considered that there is an urn, which has  $N$  white and  $M$  black balls. And blacks and the balls are randomly selected one at a time, until a black one is obtained. And if we assume that each selected ball is replaced before the next one is drawn; then we already had found the

probability that there are exactly  $n$  draws needed. So, that was probability  $x$  equal to  $n$ , that was obtained here like this.

And at least  $k$  draws are needed that was probability  $x$  less than equal to  $k$ , which was obtained here like this; where, this  $P$  was  $M/(M+N)$ . So, now we try to choose the two values of  $M$  and  $N$ , as capital  $N$  is equal to 5 and capital  $M$  is equal to 10. So, this  $p$  probability comes out to be close to 0.67; and now we try to compute this probability using the R software.

(Refer Slide Time: 09:13)

**Geometric Distribution: Example in R**

(a) We find the probability of exactly  $n = 4$  draws are needed as  $P(X = 4)$  by using the Geometric distribution.

```
> dgeom(x=4-1, prob=0.67)
[1] 0.02407779
```

computes the density of geometric distribution with  $p = 0.67$  as

$$P(X = 4) = 0.67(1 - 0.67)^{4-1} = 0.02407779$$

R Console

```
> dgeom(x=4-1, prob=0.67)
[1] 0.02407779
> .67*(1-.67)^3
[1] 0.02407779
```

So, suppose we want to find the probability of exactly  $n$  equal to 4 draws are needed; and for that I would like to use the command in the R software. So, basically we want to compute probability that  $X$  equal to 4. So, now you can see here, you are trying to use the command `dgeom` to compute the density. But, now for the  $x$  you are not giving as 4; but you are trying to give 4 minus 1, this is  $k$  minus 1 and this is here your  $k$ . This is what we have to keep in mind, and after that you have to simply that write the value of probability as `prob` equal to 0.67; which is the parameter of the geometric distribution.

And if you try to compute it; it will come out to be like this. And in case if you try to compute the same probability manually also using this expression; you can see here that this both these expressions are going to give you the same outcome, as shown on this screenshot. You can see this and here this.

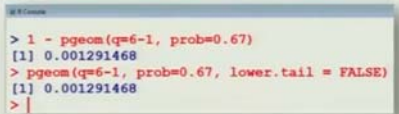
(Refer Slide Time: 10:19)

**Geometric Distribution: Example in R**

`pgeom(q, prob, lower.tail = TRUE)` calculate the CDF  
 $F(q) = P(X \leq q)$  at any point  $q$ .  $P(X \leq 6) \rightarrow 6-1$

(b) We find the probability of at least 6 draws are needed with  
 $p = 0.67$  as  $P(X \leq 6 - 1) = 1 - F(6 - 1)$ ; then we write  
> `1 - pgeom(q=6-1, prob=0.67)`  
[1] 0.001291468

or equivalently  
> `pgeom(q=6-1, prob=0.67, lower.tail = FALSE)`  
[1] 0.001291468



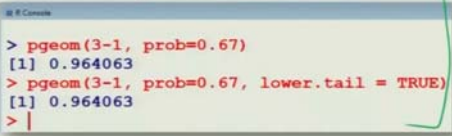
So, now we try to compute the CDF, so for that we have the command `pgeom`,  $q$ , and then you have to give the probability. And you have an option to give lower dot tail equal to true or false; just like as we have used earlier. So, this will give you the CDF  $F(q)$  equal to probability  $X$  less than equal to  $q$ , at any point  $q$ . So, when you are trying to describe this command, then you have to be careful when you are trying to choose the value of  $q$ . So, that is again going to be something like the  $k$  minus 1 type of issue.

So, suppose we want to find out the probability of at least 6 draws are needed with  $p$  equal to 0.67. Then that means we want probability that  $x$  less than equal to 6; but, we are trying to specify this 6 as in the R software as 6 minus 1. So, that means we have to write probability that  $X$  less than equal to 6 minus 1, which is equal to 1 minus  $F$  of 6 minus 1. And using the command `1 - pgeom` with  $q$  equal to 5, and  $prob$  equal to 0.67; we can compute this value.

Now, and in case if you do not want to use this command of `1 - F`, then you can use here lower dot tail equal to `FALSE`. And then you can write down the command as `pgeom` is equal to 6 minus 1 and  $prob$  is equal to 0.67; and it is going to give you the same outcome like this one. So that is up to you, whatever you want to use.

(Refer Slide Time: 12:02)

```
Geometric Distribution: Example in R  
We find the probability of at most 3 draws are needed with  $p = 0.67$  as  
 $P(X \leq 3) = F(3)$ ; then we write  
> pgeom(3-1, prob=0.67)  
[1] 0.964063  
or equivalently  
> pgeom(3-1, prob=0.67, lower.tail = TRUE)  
[1] 0.964063
```



10

Now, we try to find out one more probability that we find the probability of at most 3 draws. So, then in that case, you essentially need here probability  $X$  less than equal to 3, which is equal to simply  $F(3)$ . And now you know how to you have to give instead of writing  $F(3)$ , you have to actually write here  $F(3)$  minus 1 is in place of 3. So, that will become here `pgeom` and this will become a 3 minus 1; and `prob` is equal to 0.67.

And this probability you can obtain from the R software like this 0.96 and so on. Or, means equivalently if you want to do it here using the `lower.tail = TRUE` options; you will get the same thing. And this is the screenshot, so you can see whether you use `lower.tail = TRUE` or not. That is not really going to make any difference, this is the default value.

(Refer Slide Time: 13:01)

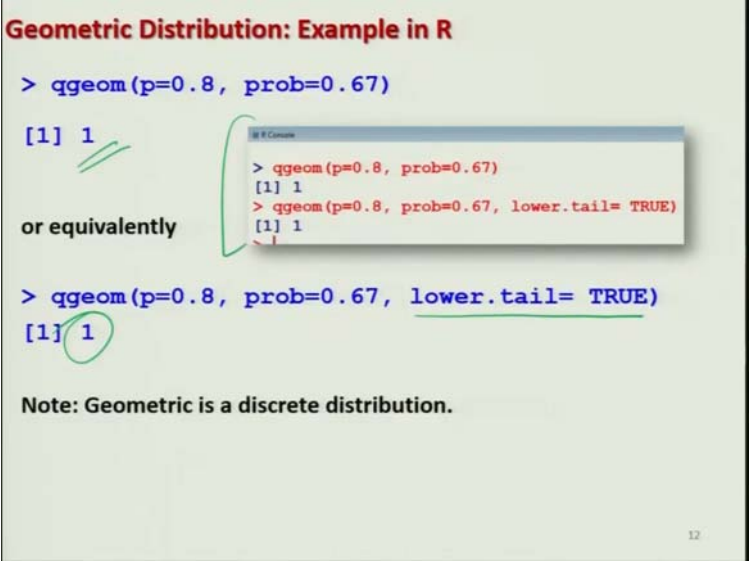
```
Geometric Distribution: Example in R  
qgeom(p, prob, lower.tail = TRUE)  
calculates the quantile which is defined as the smallest value  $x$  such  
that  $F(x) \geq p$ , where  $F$  is the CDF  $F(x) = P(X \leq x)$  at any point  $x$ .  
  
For example, suppose we want to determine the 80% quantile  $q$  which  
describes that  $P(X \leq q) \geq 0.8$  can be obtained by the command  
qgeom(p=0.8, prob=0.67)
```

11



Now similarly, if you want to compute the quantiles; so for that you have to use the command `qgeom`. And then here `p`, then probability and lower dot tail remains the same as we have used earlier. And suppose we want to determine the 80 percent quantile; so this is the value `q`, which is going to describe that probability  $X$  less than equal to `q`, is greater than or equal to 0.8. So, this can be obtained by the command `qgeom`, `p` is equal to 0.8 and `prob` equal to 0.67.

(Refer Slide Time: 13:41)



**Geometric Distribution: Example in R**

```
> qgeom(p=0.8, prob=0.67)
[1] 1
```

or equivalently

```
> qgeom(p=0.8, prob=0.67, lower.tail= TRUE)
[1] 1
```

**Note: Geometric is a discrete distribution.**

12

So, you can see how are we going to do it here. So, you write this thing and then you will get the value 1; or equivalently if you want to use the option lower dot tail equal to TRUE; even then you will get the same value. This is the screenshot and then you can see that you are getting here the value 1; because geometric is a discrete distribution.

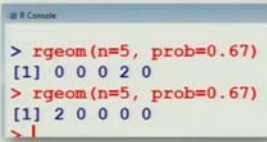
(Refer Slide Time: 14:05)

### Geometric Distribution: Example in R

`rgeom(n, prob)`  
generates `n` random numbers from geometric distribution with parameter `prob`.

For example, suppose we want to generate 5 random numbers from a geometric distribution with  $p = 0.67$  which can be obtained by the command

```
> rgeom(n=5, prob=0.67)
[1] 0 0 0 2 0
> rgeom(n=5, prob=0.67)
[1] 2 0 0 0 0 ✓
```



13

And similarly, if you want to generate some random numbers from geometric distribution, that is very straight forward. So, if you want to generate small `n` number of observation from the geometric probability mass function; then the command here is `rgeom`. And you have to write inside the parenthesis, the number of observation `n` and the `prob`, which is the probability. So, suppose if you want to generate 5 random numbers from a geometric distribution with  $p$  equal to 0.67. So, that can be simply obtained by `rgeom` and `n` is equal to 5 and `prob` is equal to 0.67; and it will give you such 5 values.

And if you try to repeat it, this will give you say once again means some other values, which are 5. And they will resemble as if they are drawn from this geometric probability distribution, where the  $p$  is equal to 0.67.

(Refer Slide Time: 15:04)

### Geometric Distribution : Mean and Variance

`rgeom(n, prob)` generates `n` random numbers from geometric distribution with  $p = \text{prob} = 0.67$

$$E(X) = \frac{1}{p} = 1.5$$

$$\text{Var}(X) = \frac{1}{p} \left( \frac{1}{p} - 1 \right) = 0.75$$

Now we generate the random numbers and calculate their mean and variance as follows:

```
x = rgeom(n, prob=0.67)
mean(x)
var(x)
```

Compare the simulated mean and variance with the theoretical mean and variance.

14

Now, in case if you want to do a sort of simulation for the geometric distribution that we have done earlier in binomial and Poisson distribution also. So, I will explain you, but I will request you to do it yourself. At least now you have to develop a sort of confidence that what you are going to observe. Well, I can explain you the outcome, but then you have to see whether this is matching with this outcome or not.

So, suppose that in this case,  $p$  is equal to 0.67 and expected value of  $x$  that is mean is  $1$  upon  $p$ ; which is 1.5, and variance of  $X$  can be computed as 0.75. Now, you try to generate  $n$  number of observation from this distribution, with prob equal to 0.67. And try to find out their mean and variance. And try to compare that what happens to the values of mean and variance, when you are trying to estimate them on the basis of a random sample. Where, when the true values are like 1.5 and 0.75. The outcome that you will see that as you try to increase the value of  $n$ , from 10 to 100 to 1000 to 10000; the value of mean and variance will be approaching towards the true value.

So, so now we stop in this lecture and now I am sure that this type of lecture will appear to be very easy to you; because I am trying to show you the same thing what I have shown you earlier. Only the commands are changing and you can see one thing that the commands are not difficult to remember.  $r$ ,  $q$ ,  $p$  etc. they are or  $d$  they are going to indicate the same quantity up to now, in all that distribution.

So, now I expect you that you try to look into your books and try to take some simple data. And from there try to read that data and try to compute the probabilities. Once you can compute such probabilities only say 10 observation or 20 observations, you will be more confident. And if you are not getting a data, I will say simply try to generate 20 observations from a given geometric distribution and try to compute different types of probability. Probability that for example,  $x$  less than 2,  $x$  greater than 2, probability that  $x$  is lying between 4 and 5 etc. etc.

Now, the time has come, where you have learned so many things, and I am sure that you should be very conversant in computing these things and understanding these things. And the ultimate is you have to match between the theory and the computation; that is the success of the data science. So, we try to practice it and I will see you in the next lecture; till then good bye.