Binomial Distribution in R

Sharon Huang

December 1, 2017

Introduction

In the class stat133, I learned a little bit about Random Numbers and Simulations. From the lectures and labs, I knew how to use R to simulate some chance processes, like tossing a coin or rolling a die. I also learned how to use the function `sample()` to get sample.

Random Numbers and Simulations let me think about the probabilities. I think calculating probabilities is very interesting and also very useful. We might win gambling in Casino because I know the probabilites of wining. Just kidding! Therefore, in this post, I am going to introduce the function of my most favorite distribution - Binomial Distribution in R.

I will talk about Binomial Distribution's functions in R through some classic probabilities' examples, such as tossing a coin, rolling a die, drawing a card from a deck of cards, and so on, so we can understand the distribution and function more clearly.

Now, let's know about the Binomial Distribution first.

Binomial Distribution

Binomial Probability has the following properties:

- There are n repeated trials.
- All of the trials are independent, which means that the outcome on one trial does not affect the outcome on others.
- Each trial has two possible outcomes: one is success with probability p, the other is failure with probability $q = 1 - p$.

Even though the binomial probability formula is very complicated, the binomial function in R is very simple, so it is very useful for students in statistic major to obtain the data.

Example 1

a. *What is the probability of obtaining 55 heads in 100 tosses of a fair coin?*
   This distribution is binomial distribution because in this experiments, we need to flip a fair coin 100 times, which means $n = 100$.
   Everytime flipping a fair coin does not affect the rest of experiments.
   And each trial has two possible outcomes, which are head and tail, both with the same probability 0.5.
   This is the reason we can use binomial distribution to solve this problem.

In this example, we know that $n = 100$, $p = 0.5$. We also know that the number of successes, heads, out of 100 trials is $x = 55$.
In R, we can use `dbinom()` function to calculate the probability density function of X, when we know n, p, and x.

```
# P(X=55)
dbinom(x = 55, size = 100, prob = 0.5)
```

```
## [1] 0.0484743
```

```
# we can write the function like the following:
dbinom(55, 100, 0.5)
```

```
## [1] 0.0484743
```

When we use the second method, we need to remember the order inside the function: x, size, probability.

b. *What are the probabilities of obtaining heads from 0 to 55 in 100 tosses of a fair coin?*
   In this question, it means that n and p are the same with part a, but x is from 0 to 55.
   We can also use `dbinom()` function for this question.

```
# P(x=0) & P(x=1) & ... & p(x=55)
dbinom(x = 0:55, size = 100, prob = 0.5)
```

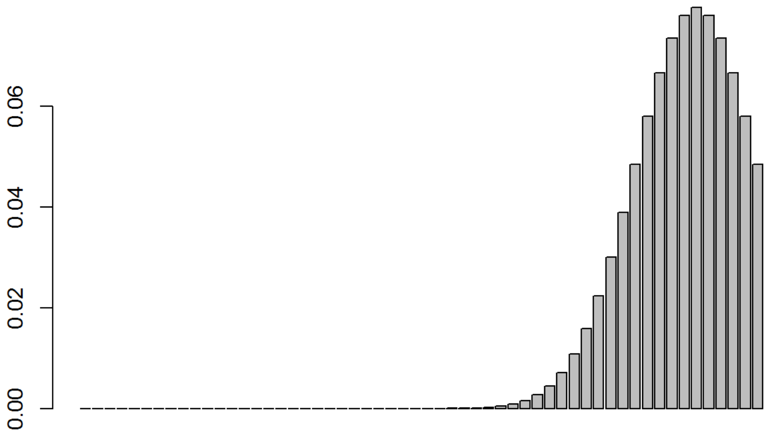
```
##  [1] 7.888609e-31 7.888609e-29 3.904861e-27 1.275588e-25 3.093301e-24
##  [6] 5.939138e-23 9.403635e-22 1.262774e-20 1.467975e-19 1.500596e-18
## [11] 1.365543e-17 1.117262e-16 8.286361e-16 5.609229e-15 3.485735e-14
## [16] 1.998488e-13 1.061697e-12 5.246031e-12 2.419003e-11 1.043991e-10
## [21] 4.228163e-10 1.610729e-09 5.783981e-09 1.961524e-08 6.293223e-08
## [26] 1.913140e-07 5.518672e-07 1.512525e-06 3.943369e-06 9.790433e-06
## [31] 2.317069e-05 5.232091e-05 1.128170e-04 2.324713e-04 4.581053e-04
## [36] 8.638557e-04 1.559739e-03 2.697928e-03 4.472880e-03 7.110732e-03
## [41] 1.084387e-02 1.586907e-02 2.229227e-02 3.006864e-02 3.895256e-02
## [46] 4.847430e-02 5.795840e-02 6.659050e-02 7.352701e-02 7.802866e-02
## [51] 7.958924e-02 7.802866e-02 7.352701e-02 6.659050e-02 5.795840e-02
## [56] 4.847430e-02
```

```
# Again, we also can write the function like the following:
dbinom(0:55, 100, 0.5)
```

```
##  [1] 7.888609e-31 7.888609e-29 3.904861e-27 1.275588e-25 3.093301e-24
##  [6] 5.939138e-23 9.403635e-22 1.262774e-20 1.467975e-19 1.500596e-18
## [11] 1.365543e-17 1.117262e-16 8.286361e-16 5.609229e-15 3.485735e-14
## [16] 1.998488e-13 1.061697e-12 5.246031e-12 2.419003e-11 1.043991e-10
## [21] 4.228163e-10 1.610729e-09 5.783981e-09 1.961524e-08 6.293223e-08
## [26] 1.913140e-07 5.518672e-07 1.512525e-06 3.943369e-06 9.790433e-06
## [31] 2.317069e-05 5.232091e-05 1.128170e-04 2.324713e-04 4.581053e-04
## [36] 8.638557e-04 1.559739e-03 2.697928e-03 4.472880e-03 7.110732e-03
## [41] 1.084387e-02 1.586907e-02 2.229227e-02 3.006864e-02 3.895256e-02
## [46] 4.847430e-02 5.795840e-02 6.659050e-02 7.352701e-02 7.802866e-02
## [51] 7.958924e-02 7.802866e-02 7.352701e-02 6.659050e-02 5.795840e-02
## [56] 4.847430e-02
```

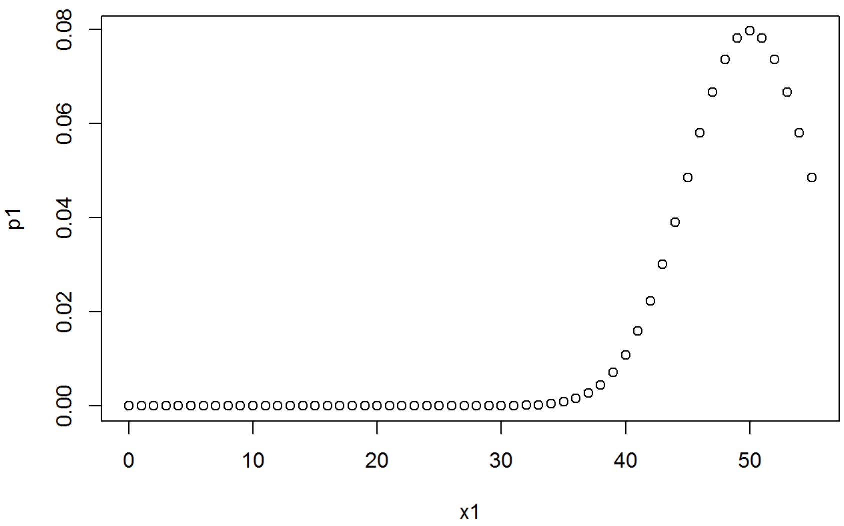
In order to see the trend more clearly, we can use barplot.

```
# barplot by P(x=0) & P(x=1) & ... & p(x=55)
barplot(dbinom(0:55, 100, 0.5))
```



If we want to see the value of x-axis and y-axis, We can use plot instead.

```
# plot by P(x=0) & P(x=1) & ... & p(x=55)
x1 <- seq(0,55,by=1)
p1 <- dbinom(x1, 100, 0.5)
plot(x1,p1)
```

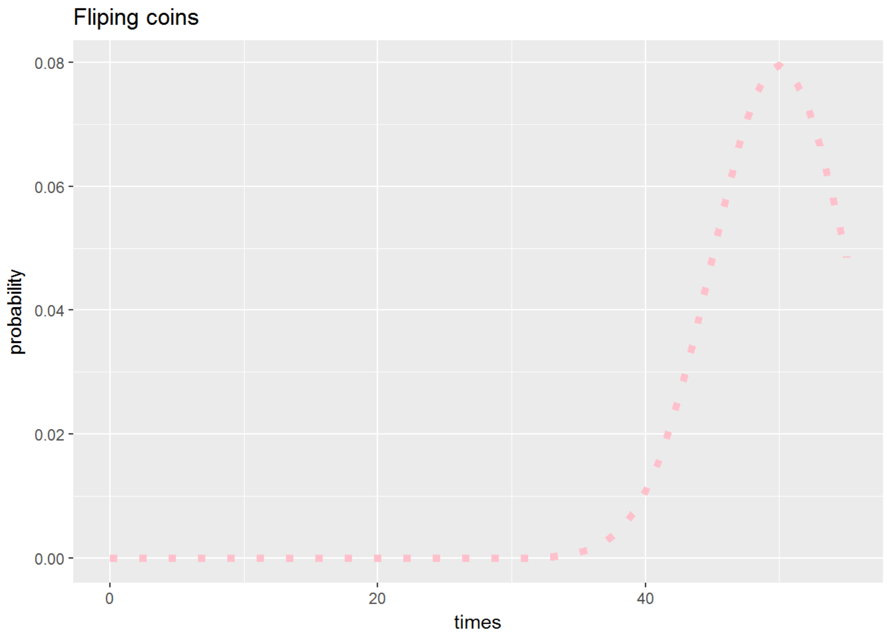


In order to make the graph more pretty and clearer, I like to use `ggplot2` which I talked about in my last post.

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.2
```

```
dat1 <- data.frame(x1, p1)
ggplot(data = dat1, aes(x = x1, y = p1)) +
  geom_line(color = "pink", size = 2, linetype="dotted") +
  labs(title="Fliping coins", x="times", y="probability")
```



In this graph, we can see that it is more likely to obtain 50 heads in 100 tosses of a fair coin.

c. *What is the probability of obtaining 55 or fewer heads in 100 tosses of a fair coin?*
   We can calculate this question base on part c. We can add all the probabilities together to get the answer.

```
# P(x<=55) by dbinom()
sum(dbinom(x = 0:55, size = 100, prob = 0.5))
```

```
## [1] 0.8643735
```

```
sum(dbinom(0:55, 100, 0.5))
```

```
## [1] 0.8643735
```

There is a better way to do it:
We can use function `pbinom()` which calculates the cumulative probability distribution function to solve this problem.

```
# P(x<=55) by pbinom()
pbinom(q = 55, size = 100, prob = 0.5, lower.tail = T)
```

```
## [1] 0.8643735
```

```
pbinom(55, size = 100, prob = 0.5)
```

```
## [1] 0.8643735
```

In `pbinom()` function, we need to verify the value q, the size n, the probability p. Also, we need to indicate that the probability we want is less than or equal to q or at least q. In R, the defalut for the "lower.tail" argument is true, so it is not necessary to include the "lower.tail" argument if the probability we want is less than or equal to q. On the contrary, we must wrtie "lower.tail = F" if we want the probability at least q.

d. *What is the probability of obtaining at least 60 heads in 100 tosses of a biased coin with chance of heads 20%, and chance or tails 80%?*
   According to the explainaiton above, it is easy to know that $q = 59$, $n = 100$, $p = 0.2$, $lower.tail = F$.

```
# P(x>59) by pbinom()
pbinom(q = 59, size = 100, prob = 0.2, lower.tail = F)
```

```
## [1] 2.515819e-18
```

The `dbinom()` and `pbinom()` functions are calculating the probabilities of some events.
There is another function, `qbinom()`, which is calculating the number of successes when we know the the size and the probability.
In other words, `qbinom()` is the R function that calculates the inverse cumulative probability distribution function of binomial distribution.

Example 2

Assume you flip a fair coin 100 times. What is the number N that, 90% of the time, the number of heads is less than or equal to N?

```
qbinom(p = 0.9, size = 100, prob = 0.5, lower.tail = T)
```

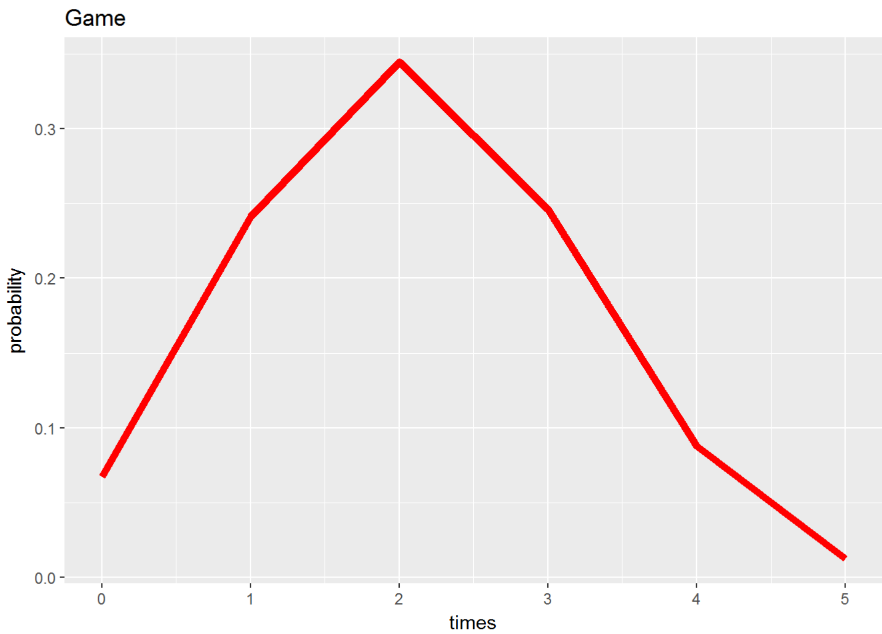
```
## [1] 56
```

I like this function a lot because I can calculate how many games I might win when I know the proabability of winning and losing a game and the times I play games.

Game time

Let's play a game: you roll a die and I roll a die. You win if the number showing on your die is strictly greater than the one on mine. If we play this game five times, what is the chance that you win at least four times?
This game seems very complicated, but if we solve it step by step, it would be simply.
We know that $n = 5$, $q = 3$, $lower.tail = F$, so we are missing probability.
In the games, there are 36 possible outcomes, and the outcomes that the number showing on your die bigger than mine are 15. Therefore, $p = 5/12$.
Based on the data that we know, we can draw a graph to show the chance that you win.

```
# plot by P(x=0) & P(x=1) & ... & p(x=5)
x2 <- seq(0, 5, by = 1)
p2 <- dbinom(x2, 5, 5/12)
dat2 <- data.frame(x2, p2)
ggplot(data = dat2, aes(x = x2, y = p2)) +
  geom_line(color = "red", size = 2) +
  labs(title="Game", x="times", y="probability")
```



And it is very easy to get the answer.

```
# P(x>3) by pbinom()
pbinom(q = 3, size = 5, prob = 5/12, lower.tail = F)
```

```
## [1] 0.1004694
```

```
# If you want to know more about binomial distribution, you can use the help command
help("Binomial")
```

Conclusion

In this post, we talked about the Binomial Distribution Functions in R, which are:

- `dbinom()`
- `pbinom()`
- `qbinom()`

Obviously, calculating Binomial Distribution in R is much easier.
I hope you can remember the main functions of Binomial Distribution in R and understand the Binomial Distribution well after reading this post.
Thank you for reading! I hope you like this post and like Binomial Distribution.

Reference

Binomial Distribution in R
Probability Distribution in R
Binomial Distribution
Basic Probability Distributions
Basic Probability Distributions in R
ggplot cheatsheet
Probability