DSA5403 Lab2 Chapter 3, R and distributional theory

Jishan Luo

January 16, 2019

Contents


```
library(ggplot2)
```

Task 1

The Binomial distribution. Throw a coin "n" times and the number of successes is "x". Where p =probability of a success

a. Write the formula for p

$$p(X = x|n, p) = \binom{n}{x} \cdot p^x(1-p)^{n-x} = \frac{n!}{x!(n-x)!} \cdot p^x(1-p)^{n-x}$$

b. Using the answer to the above write your own r function dmybin() to calculate p, record this in Rmd

```
dmybin=function(x,n,p)
  {
n_x=factorial(n)/(factorial(x)*factorial(n-x))
pb=n_x*p^x*(1-p)^(n-x)
list(result=pb) #print the result of probability--pb
}
```

c. Now calculate p(X=4|n=10,p=0.5) using your function.

```
dmybin(x=4,n=10,p=0.5)
```

```
## $result
## [1] 0.2050781
```

d. Use the built in R function dbinom() to calculate the same probability.

```
dbinom(4,size=10,prob=0.5)
```

```
## [1] 0.2050781
```

e. What if we wish to calculate the cumulative probability p(X<=x|n,p), we would need to sum individual probabilities. Make a function called pmybin() that would do the job.

```
pmybin=function(x,n,p)
{
  i=0
  for (i in 0:x){
    if (i <= x){
      n_x=factorial(n)/(factorial(x)*factorial(n-x))
      pb=n_x*p^x*(1-p)^(n-x)
      i=i+1
      pb=pb+pb
```

```
    }
  }
  list(result=pb)
}
```

f. Use the function to calculate p(X<=5|n=10,p=0.5)

```
# pmybin(5,10,0.5)=0.6230469
```

g. Use the built in R function pbinom() to do the same and see whether the answers are the same.

```
pbinom(5,10,0.5)
```

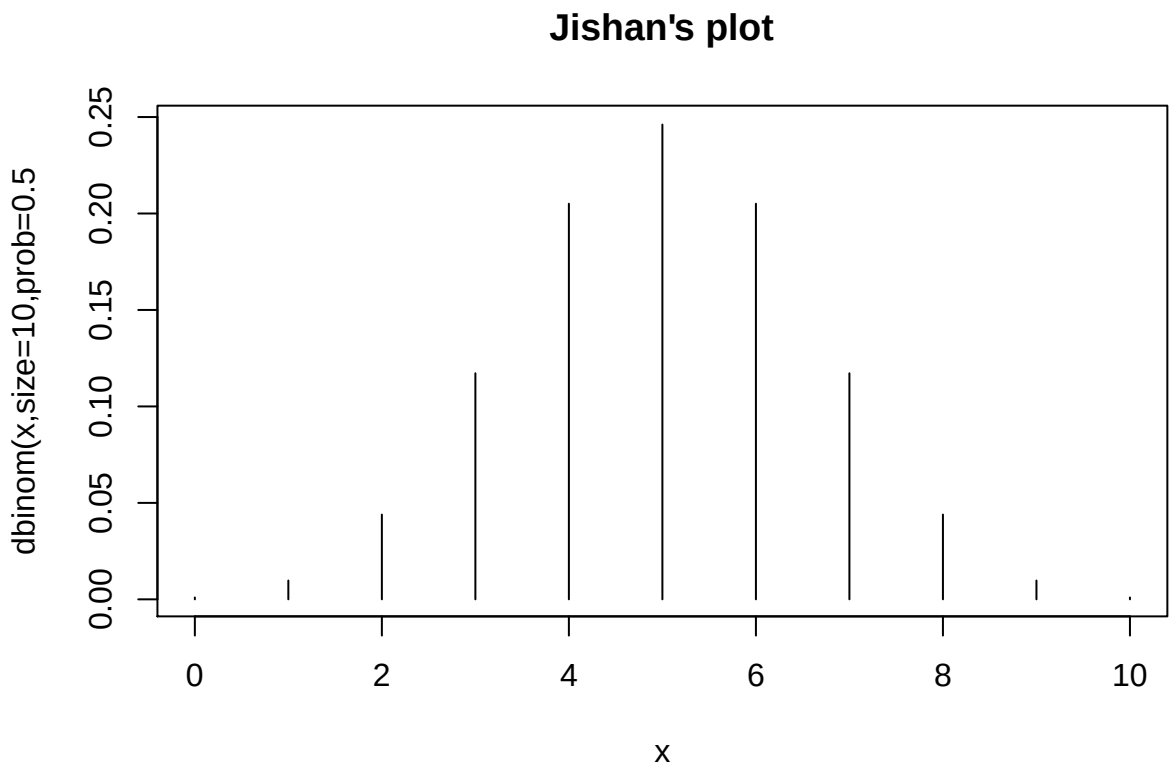
```
## [1] 0.6230469
```

```
sum(dbinom(0:5,10,0.5)) #we can see the result of two funtions are the same
```

```
## [1] 0.6230469
```

h. Make the following plot, where n=10,p=0.5 except put your name on the title:

```
n=10
p=0.5
x=0:n
plot (x,dbinom(x, size=n, prob = p), type = "h",
      ylab = "dbinom(x,size=10,prob=0.5",
      main = "Jishan's plot") # plot the probabilities with n=10,p=0.5
```



Task 2

Learn how to use the four basic distributional functions dpois, pstem, rstem, qstem

a. Suppose that X~Pois(lamda). Use R and the above function types to answer the following.

- i. Find P(X=4|lamda=3)

```
dpois(4, 3)
```

```
## [1] 0.1680314
```

- ii. Find P(X<=4|lamda=3)

```
ppois(4,3,lower.tail=T)
```

```
## [1] 0.8152632
```

- iii. Find P(X>4|lamda=3)

```
1-ppois(4,3) #method 1
```

```
## [1] 0.1847368
```

```
1-sum(dpois(0:4,3)) #method 2
```

```
## [1] 0.1847368
```

```
ppois(4,3,lower.tail = F) #method 3
```

```
## [1] 0.1847368
```

- iv. Find x so that P(X<=x|lamda=3)=0.9997077

```
myfun=function(x,lam,prob){
  prob=1-(lam^x*exp(-lam))/factorial(x)
  return(x)
}
#myfun(x,3,0.99970777)
x=10
```

- v. Create a sample of size 100 from a Poisson distribution that has parameter lamda=3, Store in an object.

```
set.seed(124)
Fst=rpois(100,3)
```

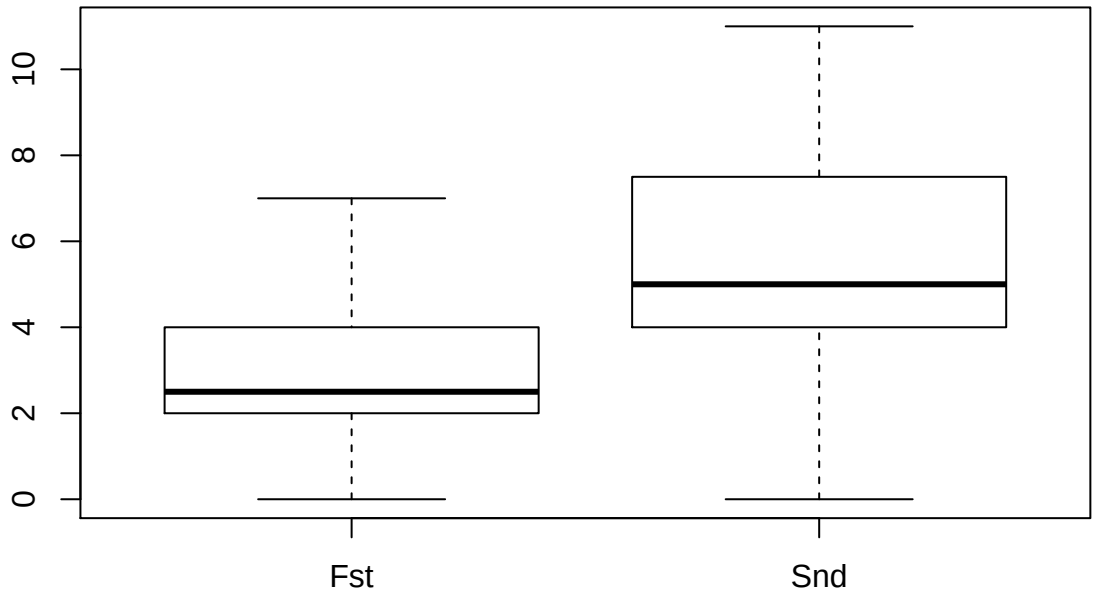
- vi. Make a second sample of size 100 from a Poisson that has parameter lamda=6, store in an object

```
set.seed(124)
Snd=rpois(100,6)
```

b. Make boxplots of the random samples you made above.

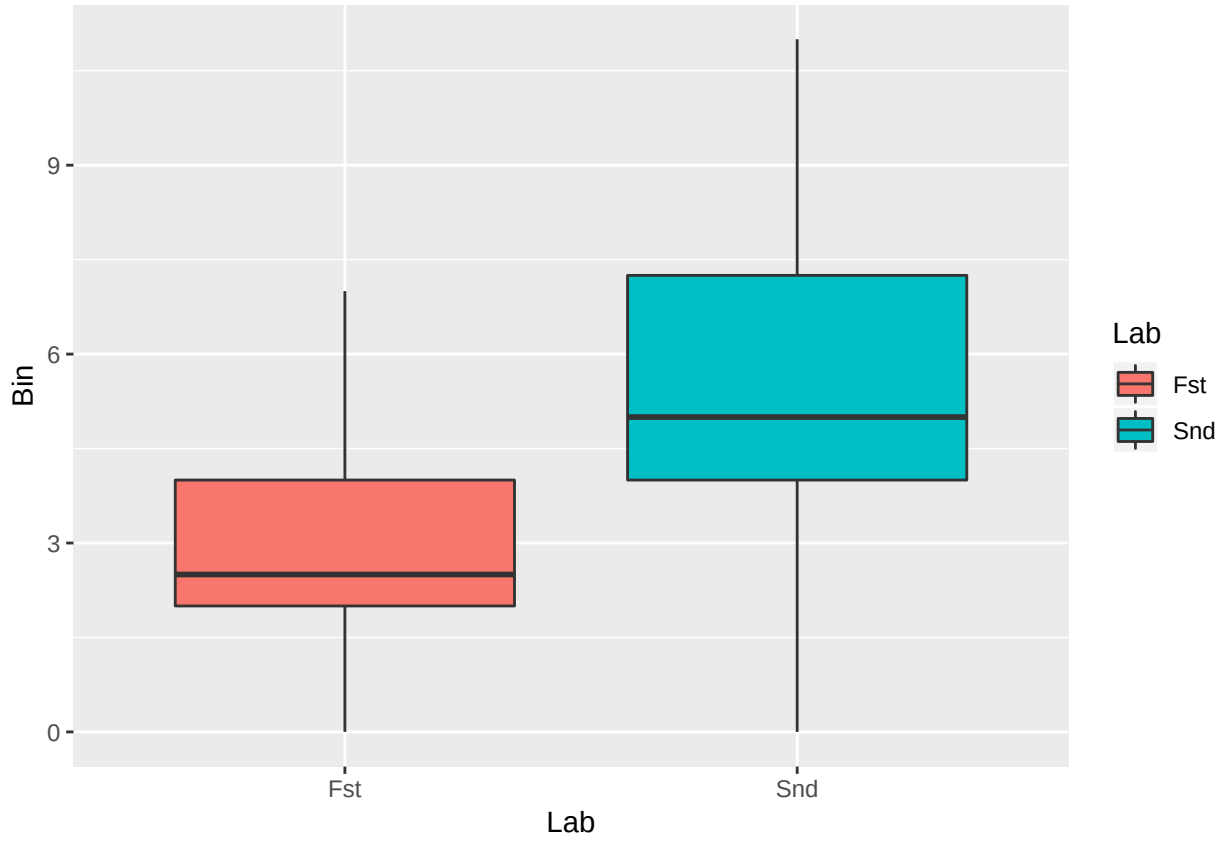
- i. We will make a data frame of the data. Call the first group "Fst" and the second group "Snd". All data in the first group have to have "Fst" associated with them etc.

```
df=data.frame(Bin =c(Fst,Snd), Lab=rep(c("Fst","Snd"),c(100,100))) # create a datafame "df" with "Fst"
boxplot(data.frame(Fst,Snd))
```



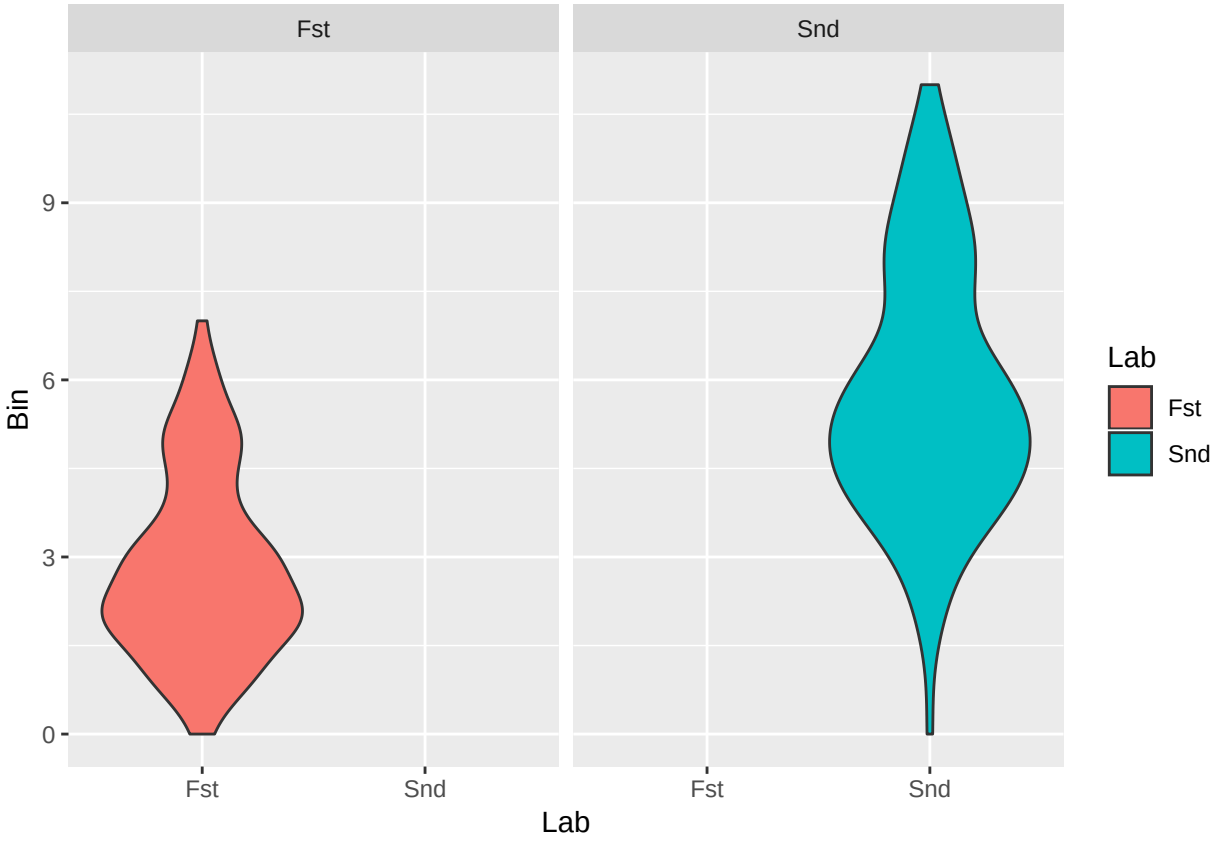
- ii. See Laboratory2.R for some exemplar code using ggplot

```
g = ggplot(df, aes(x = Lab, y = Bin, fill = Lab)) +
  geom_boxplot()
g #boxplot withe dataframe we created
```



c. Make violin plots of the same using ggplot

```
e = ggplot(df, aes(x = Lab, y=Bin, fill = Lab)) +
  geom_violin(aes(y = Bin)) +facet_wrap(~Lab)
e # violin plot
```


Fst
Snd
9
6
3
0
Bin
Fst
Snd
Fst
Snd
Lab
Lab
Fst
Snd