DSA5403 Lab2 Chapter 3, R and distributional theory

Jishan Luo

January 16, 2019

Table of 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**

1. Write the formula for p
2. 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  
}

1. 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

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

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

## [1] 0.2050781

1. 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)  
}

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

# pmybin(5,10,0.5)=0.6230469

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

pbinom(5,10,0.5)

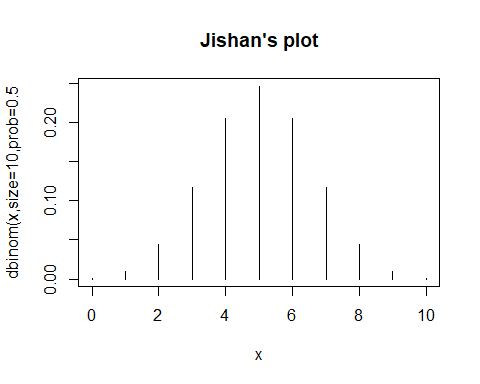
## [1] 0.6230469

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

## [1] 0.6230469

1. 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**

1. Suppose that X~Pois(lamda). Use R and the above function types to answer the following.
   1. Find P(X=4|lamda=3)

dpois(4, 3)

## [1] 0.1680314

* 1. Find P(X<=4|lamda=3)

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

## [1] 0.8152632

* 1. 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

* 1. 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,3,0.9997077)  
x=10

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

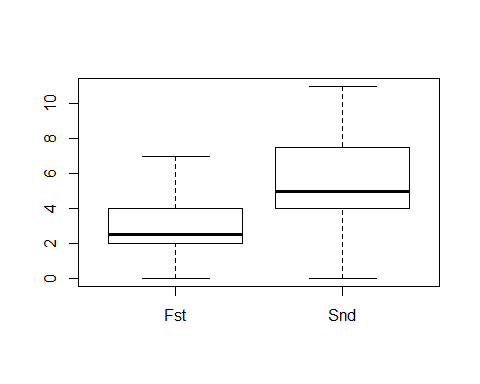
set.seed(124)  
Fst=rpois(100,3)

* 1. 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)

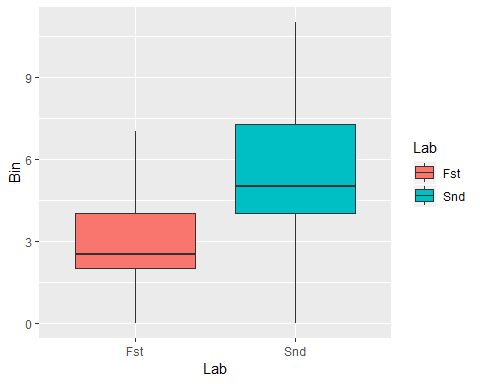
1. Make boxplots of the random samples you made above.
   1. We will make a data frame of the data. Call the first group “Fst” and the second group “Snd”. All data in the first group have to have “Fst” associated with them etc.

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



* 1. 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



1. 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

