ISE 3293/5013 Laboratory 8
Sampling Distributions and the CLT

In this lab we will take what we have learnt already concerning sampling distributions and include ideas pertaining to the central limit theorem (CLT). We will cover the following:

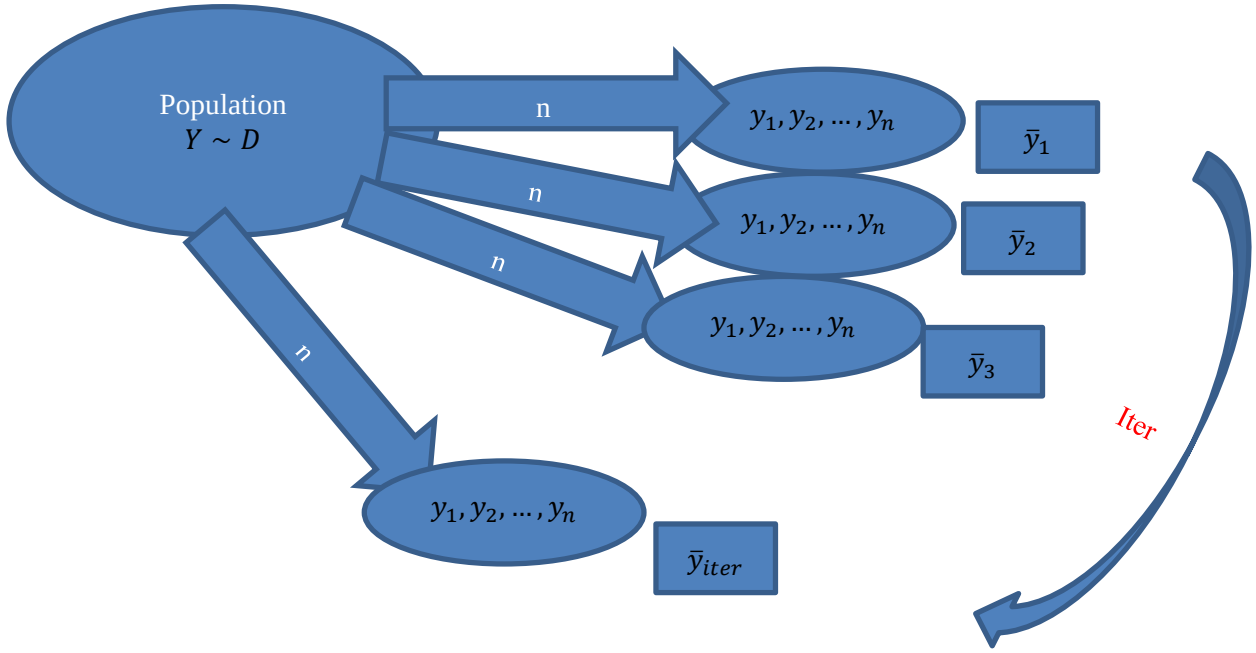
1. Sample from any distribution using r ----(), example rbinom(), rpois(), runif(), etc (see ?distributions in R)
2. How to create a statistic, usually the *sum* or the *mean*.
3. How to store the statistic.
4. How to repeat the procedure for a designated number of iterations.
5. When finished learn how to create a histogram of the statistic with other graphs.

The method for doing this will be to use a ready-made R script, adapt it and re-run it for the problems given below.

Objectives

In this lab you will learn how to:

1. Create a sample from one population.
2. Create the sample mean or sum.
3. Create sampling distributions and appropriate graphs for these particular statistics.
4. Apply the CLT and discern its limitations.



Theory: Suppose that $Y_1, Y_2, \ldots, Y_n$ are independent random variables taken from some distribution (not necessarily normal) and could be continuous or discrete.
Then the expected value of the mean and sum can be calculated as follows

$$T = Y_1 + \cdots + Y_n$$
$$\bar{Y} = \frac{T}{n} = \frac{Y_1 + \cdots + Y_n}{n}$$
$$E(\bar{Y}) = E\left(\frac{Y_1 + \cdots + Y_n}{n}\right) = \frac{nE(Y_i)}{n} = E(Y_i)$$
$$E(T) = nE(Y_i)$$

The variance of T and $\bar{Y}$ can also be found

$$V(\bar{Y}) = V\left(\frac{T}{n}\right) = \frac{1}{n^2}V(T) = \frac{nV(Y_i)}{n^2} = \frac{V(Y_i)}{n}$$
$$V(T) = nV(Y_i)$$

The CLT says that if n is large then with good approximation $\bar{Y} \sim N$ and $T \sim N$.

Tasks
All output made please copy and paste into **this word file**. Save and place in the dropbox when completed. Anything you are asked to make should be recorded under the question in this document. There will be two files you need to upload:

- a pdf of this document (pdf) or the word file (docx)
- a text file of all the code you used to create answers (txt)

Note: All plots you are asked to make should be recorded in this document.

- Task 1
  - Make a folder LAB8
  - Download the file "lab8.r"
  - Place this file with the others in LAB8.
  - Start Rstudio
  - Open "lab8.r" from within Rstudio.
  - Go to the "session" menu within Rstudio and "set working directory" to where the source files are located.
  - Issue the function `getwd()` and copy the output here.

```
F:/Google Drive - Saied/Courses/02 OU/11 Fundamentals of
Engineering Statistical Analysis/02 Labs/08 Lab 8
```

- Task 2
  - Create your own R file and record the R code you used to complete the lab.
  - Create a sample of size n=10 from a uniform distribution that has lower limit 0 and upper limit 5 by using runif(10,0,5). Record the results here.
  - Give the mean and variance of the uniform for the case where a=0, b=5, i.e. $\mu = \frac{a+b}{2}, \sigma^2 = \frac{(b-a)^2}{12}$
  - Use the sample you made to calculate $\bar{x}$ *and* s^2. How do they compare to the population parameters?
  - Use the above theory to write down the mean and variance of the distribution of:

- The sum T

```
$Expmean
[1] 25.00451
$Calmean
[1] 25
$Expvar
[1] 20.88104
$Calvar
[1] 20.83333
```

- The mean $\bar{Y}$

```
$Expmean
[1] 25.03822
$Calmean
[1] 25
$Expvar
[1] 21.62297
$Calvar
[1] 20.83333
```

- Below I have given the simple function `myclt()`
  - `myclt=function(n,iter){`
  - `y=runif(n*iter,0,5) # A`
  - `data=matrix(y,nr=n,nc=iter,byrow=TRUE) #B`
  - `sm=apply(data,2,sum) #C`
  - `hist(sm)`
  - `sm`
  - `}`
  - `w=myclt(n=10,iter=10000) #D`
  - Explain what the following lines do
    - A

      ```
      This line produces (n x iter) samples by
      uniform distribution of (a = 0, b = 5).
      ```

    - B

      ```
      Makes a matrix in which each column is our n
      sample.
      ```

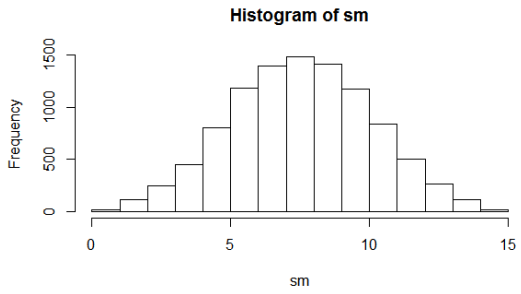
    - C

      ```
      Calculates the summation of the samples which
      are in columns
      ```

    - D

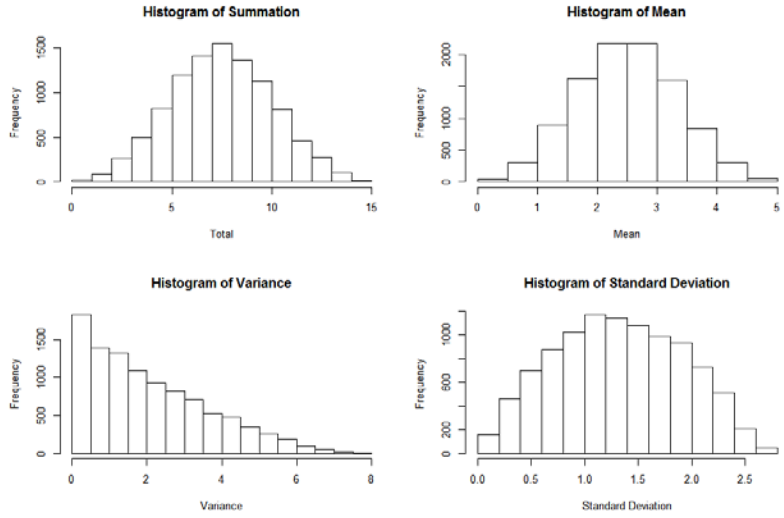
      ```
      Runs the function for 10000 times with 10
      samples in each iteration
      ```

- Record the plot made when D is executed



- Using the object w, find sample estimates $\bar{w}$ *and* s_w^2 (you can use `mean()` and `var()`)

  `Some new codes have been created: mycltmean, mycltvar, mycltsd, and mycltsum.`
- Change the code in myclt() so that it produces a histogram of the sample means and releases a vector of sample means. **Hint:** You only need to change the last three lines beginning with line C.



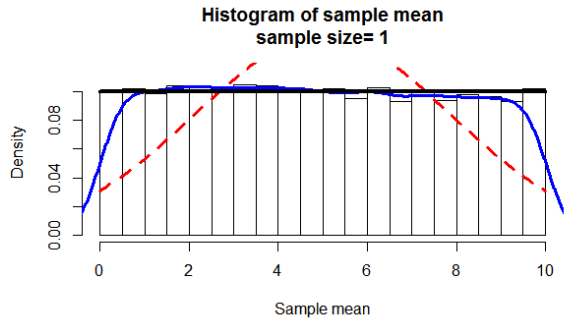
- Use this changed function and execute line D as the first step to calculate the estimates $\bar{w}$ *and* s_w^2 record these and the plot that the function makes.

- Task 3
  - We will now make a more sophisticated graph using `mycltu()`.
  - Examine the function and the comments which explain what the code does.
    - `w=apply(data,2,mean)`, how does the apply function use the 2?

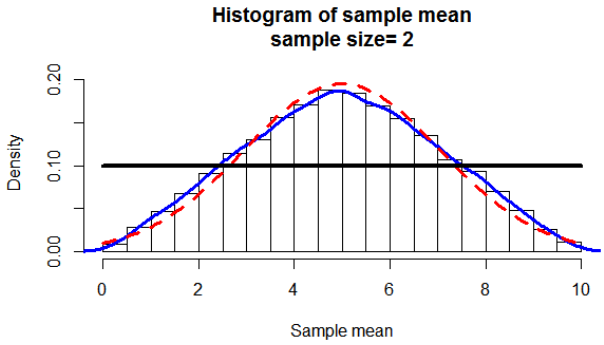
      `The second parameter, 2, is the number of matrix dimension which means the function as third parameter applies to the matrix as first parameter vertically. The function will be applied on matrix columns.`
    - How many terms are in `w`, when `mycltu(n=20,iter=100000)` is called?

      `2,000,000`
    - `curve(dnorm(x,mean=(a+b)/2,`
    - `sd=(b-a)/(sqrt(12*n))),add=TRUE,col="Red",lty=2,lwd=3)`: Explain why sd takes the formula as shown in the function.

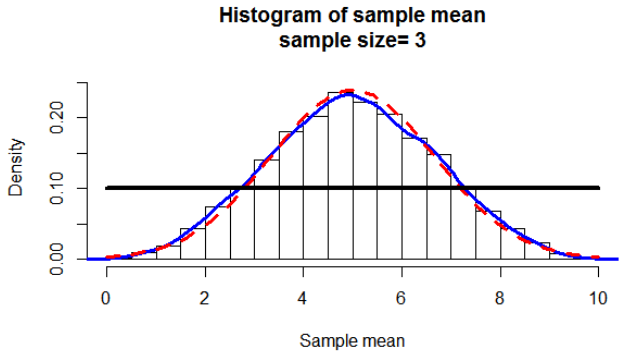
      `As proved in at the beginning of this document` $\sigma^2 = \frac{(b-a)^2}{12n}$ `and then` $sd = \sigma = \frac{b-a}{\sqrt{12n}}$
  - Record the plots using the following parameters and options
    - n=1, iter=10000, a=0,b=10



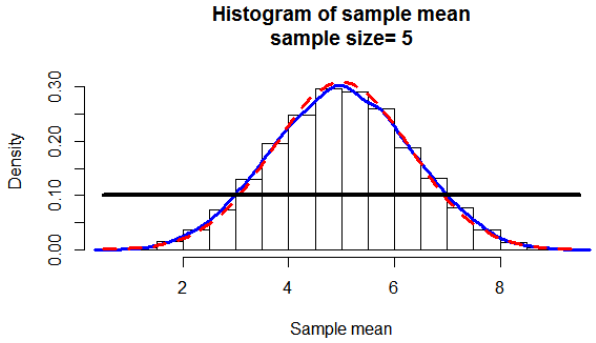
- n=2, iter=10000, a=0,b=10



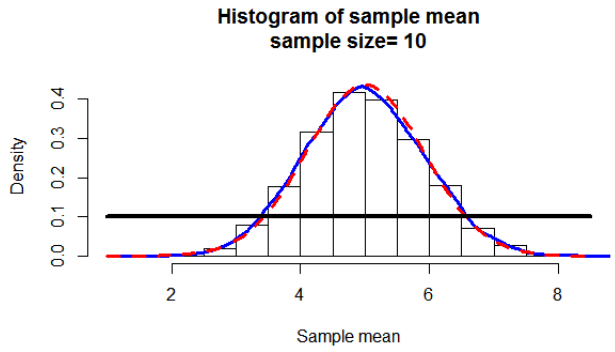
- n=3,iter=10000,a=0,b=10



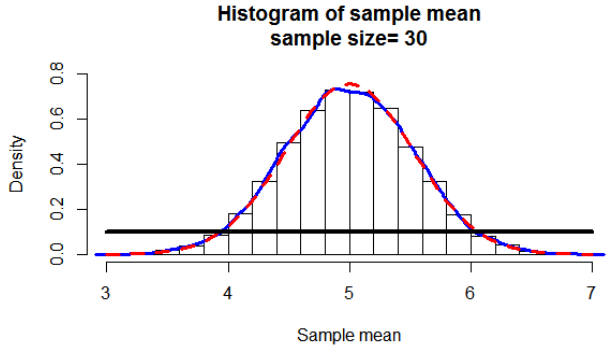
- n=5,iter=10000,a=0,b=10



- n=10, iter=10000,a=0,b=10



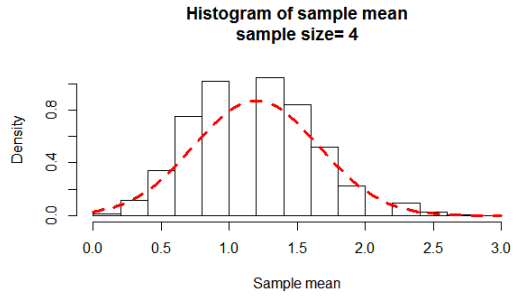
- n=30,iter=10000,a=0,b=10



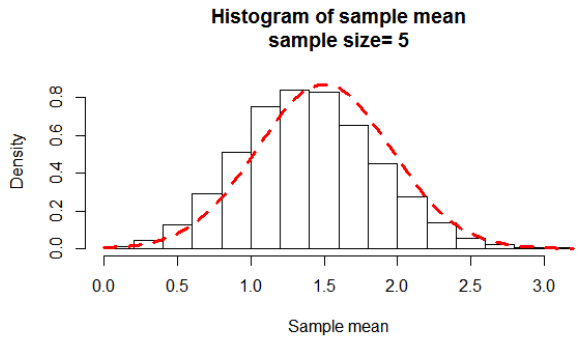
  - What do you conclude?

```
As n increases it approximately follows Normal distribution. Even after n = 3, a good shape of normal distribution is observable. Y ~ N as n increases.
```

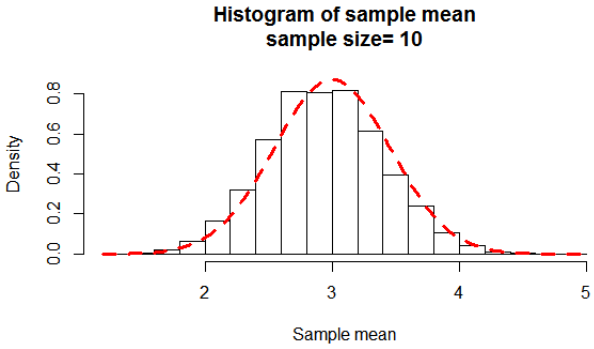
- Task 4
  - We will now make samples from a binomial distribution using `mycltb()`.
  - Make graphs for the following parameters and options
    - n=4,iter=10000,p=0.3



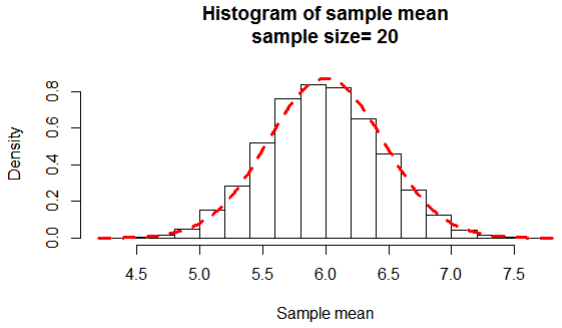
    - n=5,iter=10000,p=0.3



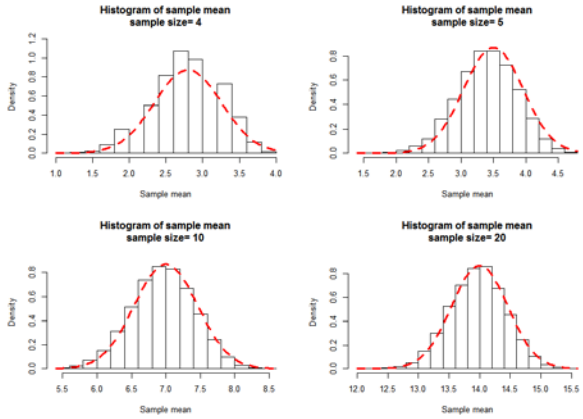
- n=10,iter=10000,p=0.3



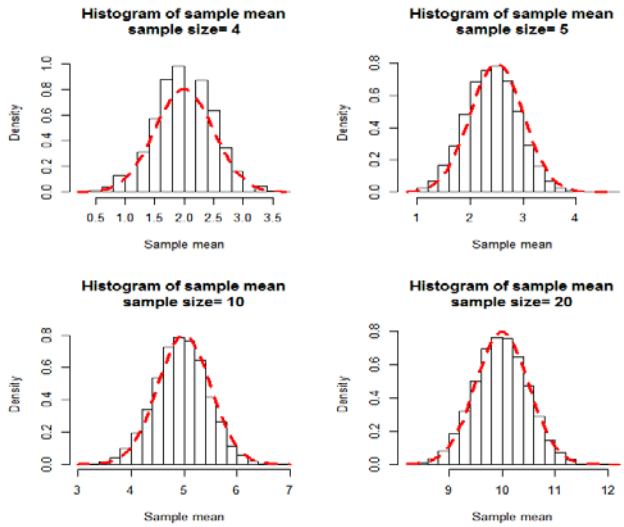
- n=20,iter=10000,p=0.3



o Do the same, except use p=0.7



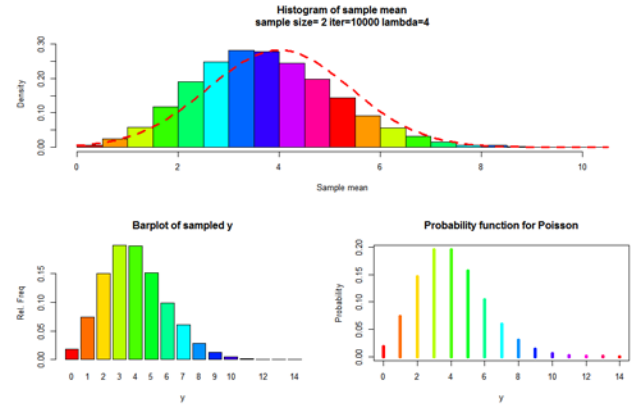
o Do the same again this time with p=0.5



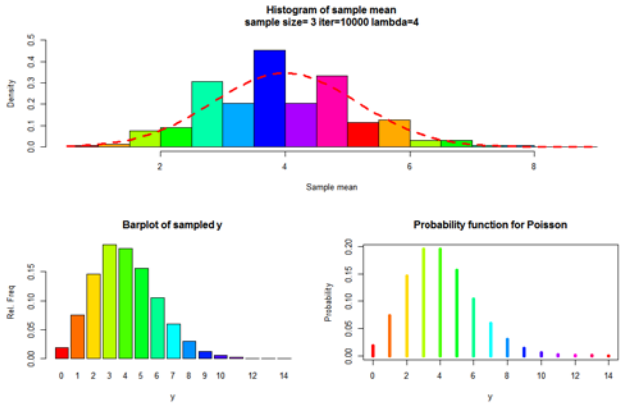
- What do you conclude?

```
As the sample size n increases it tends to become more normally distributed. It also show that the population distribution does not affect the result. It doesn't matter what is the population distribution, the distribution of the sample's mean would be normally distributed.
```

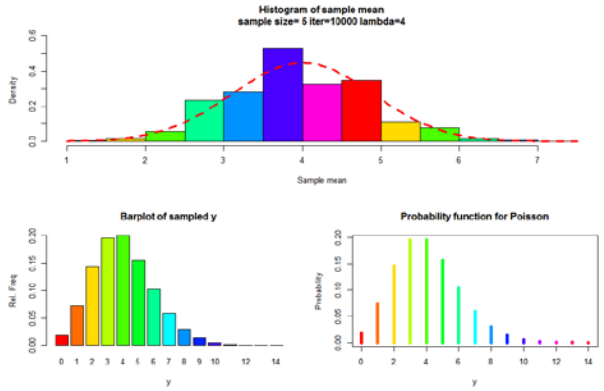
- Task 5
  - This task will need to be recorded
  - This time we will make use of the Poisson distribution using `mycltp()`.
  - Make graphs for the following parameters and options
    - n=2, iter=10000,lambda=4



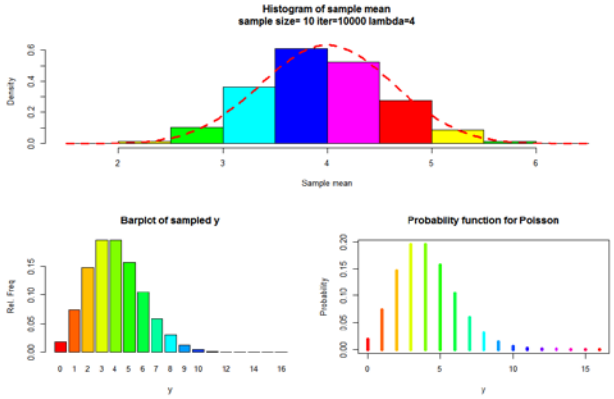
    - n=3,iter=10000,lambda=4



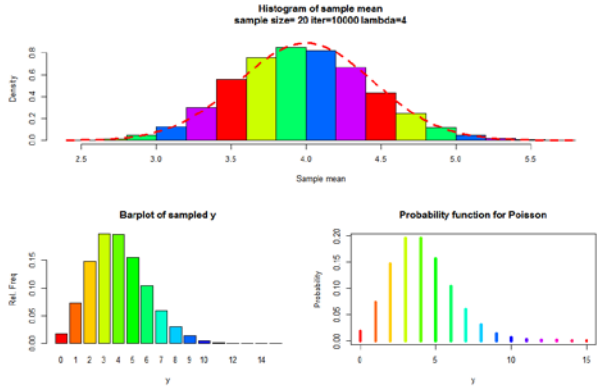
- n=5,iter=10000,lambda=4



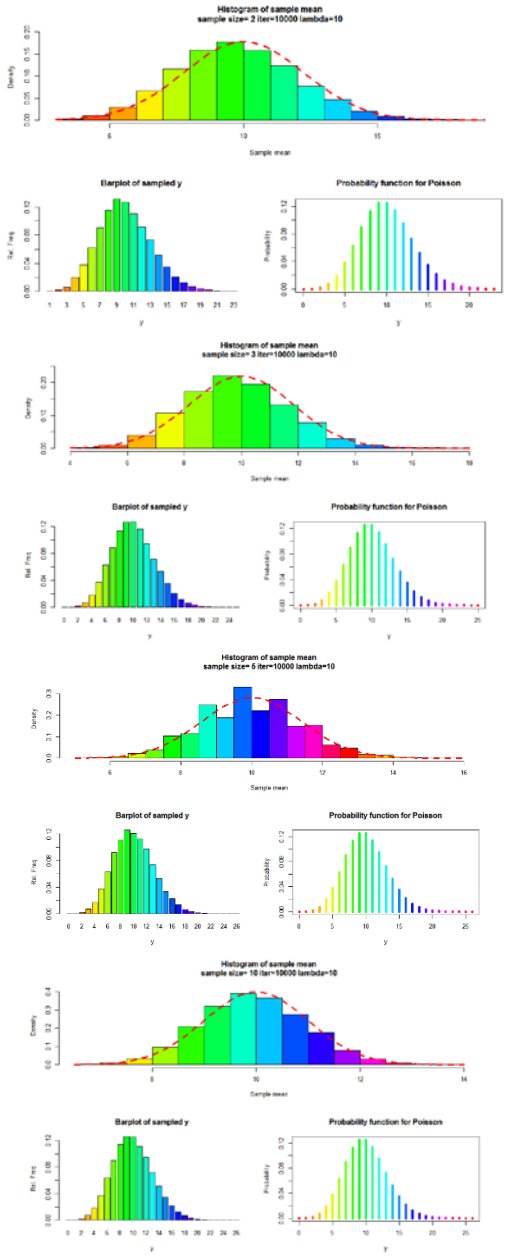
- n=10,iter=10000,lambda=4

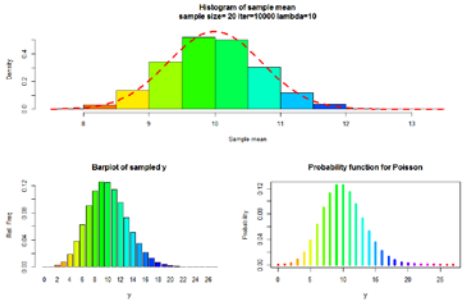


- n=20,iter=10000,lambda=4



- Do the same for lambda=10.





- Now record all of TASK 5 with BBFLASHBACK
- Place the .fbr file into the lab 8 dropbox

################### LAB FINISHES HERE ################################

- Task 6: Extra for experts!
  - Repeat task 5 but only after you have changed appropriate code to make `mycltp()` produce the sampling distribution for the sum rather than the mean.
  - Verify (by using the function) that the sampling distribution of the sum is approximately normal when n is large.

```
mycltpsum = function(n, iter, lambda = 10, ...){

  y = rpois(n*iter, lambda = lambda)
  data = matrix(y, nr= n, nc = iter, byrow = TRUE)
  w = apply(data, 2, sum)
  param = hist(w, plot = FALSE)

  ymax = 1.1 * max(param$density)

  layout(matrix(c(1,1,2,3), nr = 2, nc = 2, byrow =
TRUE))

  hist(w, freq = FALSE,  ylim = c(0,ymax), col = rainbow
(length(param$mids)),
       main=paste("Histogram of sample sum","\n", "sample
size= ",n," iter=",iter," lambda=",lambda,sep=""),
       xlab="Sample mean")
  curve(dnorm(x, mean = n * lambda, sd = sqrt(lambda *
n)),
        add = TRUE, col = "Red", lty = 2, lwd = 3) # add
a theoretical curve

 barplot(table(y)/(n*iter),   col   =   rainbow(max(y)),
main="Barplot of sampled y", ylab ="Rel. Freq",xlab="y" )
  x=0:max(y)

plot(x,dpois(x,lambda=lambda),type="h",lwd=5,col=rainbow(
max(y)),
       main="Probability    function    for    Poisson",
ylab="Probability",xlab="y")
}
```

```
windows()
mycltpsum(n = 50,iter = 10000, lambda = 10)
```