MATH 4753 Laboratory 5
Random sampling

This lab will investigate random sampling. We will start with the R function `sample()`. This will introduce you to functions and how they can be very useful in simulation. The process of simulation utilizes random sampling and allows the statistician to trial theories in practice or to carry out analyses when there are no analytical results available. There are a number of discrete distributions that you can learn about in R. In this lab you will need to THINK through the code and understand the process.

Objectives

In this lab you will learn how to:

1. Simulate from first principles.
2. Use built in r-dist() functions.
3. Make appropriate plots.

Tasks

All work is to be done on RMarkdown.
Upload completed:

- HTML
- RMD

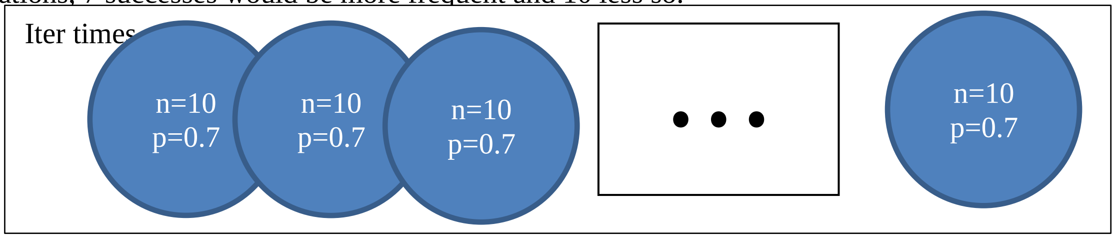
Documents.

Note: All plots you are asked to make should be constructed through RMD and knitted into your html document

- Task 1
  - Make a folder LAB5
  - Download the file “lab5.r”
  - Place this file with the others in LAB5.
  - Start Rstudio
  - Open “lab5.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()`

- Task 2
  - Make a new file for your code in RStudio editor, call it “mylab5.R” and place in it all the code you need to answer the tasks of this lab (copy and paste from lab5.R).
  - Use the hash # symbol and write your own comments in the code file explaining what the code does.
  - Using code in lab5.R –`mybin()`, simulate a binomial experiment where n=10,p=0.7, and Y=number of successes. When running this particular function you will be carrying out iter experiments of making n=10 trials with the probability of a success p=0.7 and

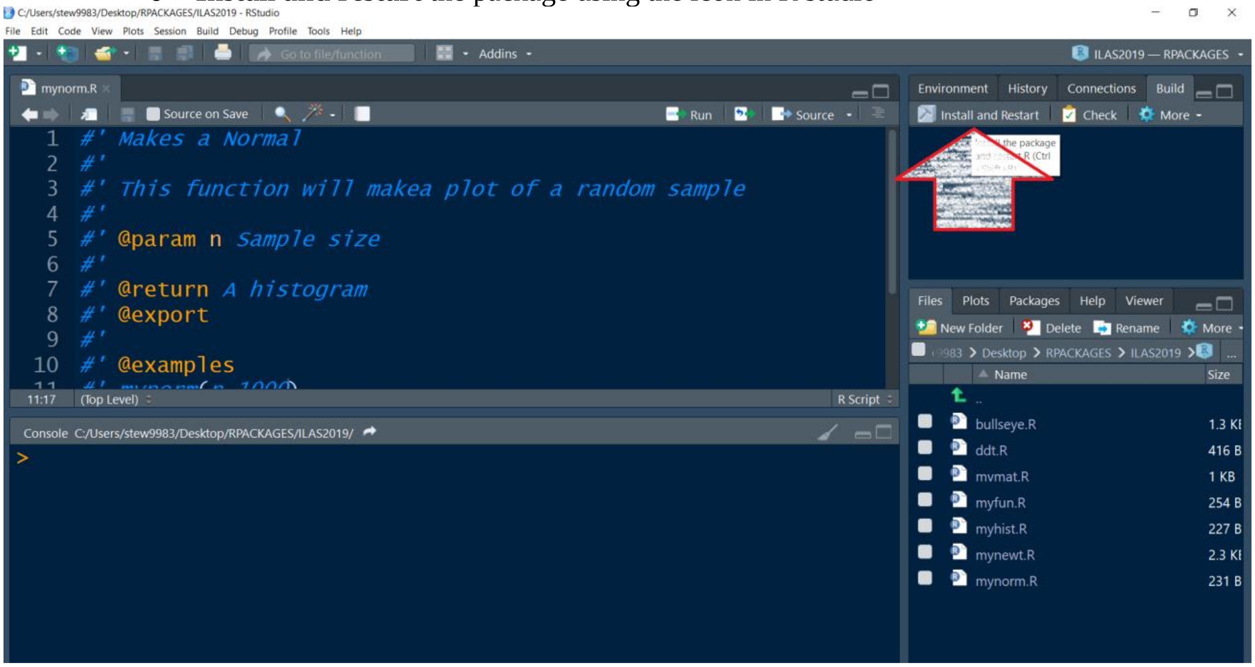
then recording the proportion of successes made over all the iterations. For example 1 success might happen on some iterations but it would have a low frequency over all the iterations, 7 successes would be more frequent and 10 less so.



  - Record the plots for each of the following (Use color)
    - 100 iterations
    - 200 iterations
    - 500 iterations
    - 1000 iterations
    - 10,000 iterations
  - For the last simulation (10,000 iterations) copy and paste the table that is produced from the function.
  - Verify that each value in the table is approximately correct using `dbinom()` – you will need to look up the function using ?`dbinom`
- Task 3
  - Suppose that there is a bag of 20 marbles, 12 white (“1”) and 8 black “0”. Using the `sample()` function create a sample of size n=5 without replacement. Make sure this appears in your document
  - Now do the same with replace=TRUE.
  - Use the function `myhyper()`, on CANVAS, assume Y = number of whites, remember sampling is without replacement, n=5. Use this code to make barplots of Y for the following number of iterations
    - 100
    - 200
    - 500
    - 1000
    - 10,000
  - For the last simulation (10,000 iterations) copy and paste the table that is produced from the function.
  - Verify that each value in the table is approximately correct using `dhyper()`
- Task 4
  - In lab5.r there is a function called `mysample()`. Look at the code and describe what you think it should do – record your response here.
  - Run the function with the following arguments `mysample(n=1000, iter=30,time=1)`
  - Describe what you see here.
  - Make sure the plot appears in your document (you need only include one of the 30 iterations)

- Task 5
  - Use R to calculate
    - $\binom{8}{4}$ – hint: Try choose()
    - $P(Y > 4), Y \sim Pois(\lambda = 2)$

  - Some more calculations in R
    - $P(Y = 10), Y \sim NegBin(p = 0.4, r = 3)$ see CANVAS which has a full explanation of this.
    - $P(Y \leq 8), Y \sim Bin(n = 15, p = 0.4)$
- Task 6
  - Create an R function by copying code from this lab.
  - Document the function using roxygen
  - Place the function into your package
  - **Install and restart** the package using the icon in R studio



C:/Users/stew9983/Desktop/RPACKAGES/ILAS2019 - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
ILAS2019 — RPACKAGES
mynorm.R
Source on Save
Run
Source
Environment
History
Connections
Build
Install and Restart
Check
More
Install the package and restart R (Ctrl+Shift+B)
11:17
(Top Level)
R Script
Console C:/Users/stew9983/Desktop/RPACKAGES/ILAS2019/
Files
Plots
Packages
Help
Viewer
New Folder
Delete
Rename
More
9983 > Desktop > RPACKAGES > ILAS2019 >

```r
#' Makes a Normal
#'
#' This function will makea plot of a random sample
#'
#' @param n Sample size
#'
#' @return A histogram
#' @export
#'
#' @examples
#' mynorm(n=1000)
```

```
>
```

Name	Size
..	
bullseye.R	1.3 KI
ddt.R	416 B
mvmat.R	1 KB
myfun.R	254 B
myhist.R	227 B
mynewt.R	2.3 KI
mynorm.R	231 B

  - Invoke the function by doing the following – suppose your function is called `myf()`
    Inside your R chunk for task 6 invoke the code:
    `ILAS2019::myf()`
    Make sure `myf()` has the necessary arguments to run.

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

- Task 7 – Extra for experts
  - Write some code that will simulate a neg binomial distribution. You could use the while() condition