

Worksheet 6.1: R you ready to learn some R?

Introduction to R, RStudio and featureCounts

Author: Taylor Jones (tajo5912@colorado.edu)

Before Class: Make sure you have RStudio installed on your computer!

<https://www.rstudio.com/products/rstudio/download/>

You want to get RStudio Desktop- Free

R is awesome for plot generation and any statistical analysis you want to do- even those that aren't related to next gen sequencing!

Useful resources:

Introduction to R: <https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf>

This is an introduction to the ins and outs of R syntax.

R for Data Science: <https://r4ds.had.co.nz/index.html>

This discusses applying R to data science with tips on exploratory data analysis.

**The Pirate's Guide to R: <https://bookdown.org/ndphillips/YaRrr/>

This is an excellent (and famous) comprehensive guide to R. I highly recommend this for learning as it starts at the basics and helps you build up skills for data analysis. It is also an easy read.

Beautiful Plotting Tips: <http://zevross.com/blog/2014/08/04/beautiful-plotting-in-r-a-ggplot2-cheatsheet-3/>

Cheat sheet for making beautiful, publication worthy plots in ggplot2!

Useful packages:

Data isn't always beautiful. To tidy data up there is a very useful package called tidyvrse with instructions in R for Data Science. Info can be found here: <https://www.tidyverse.org/>

To plot data beautifully (ggplot2 is also a part of tidyverse): <https://ggplot2.tidyverse.org/>

Section A: Getting familiar with R

- 1) You can access R through the command line or through RStudio. We will start on command line. We use this on a big cluster when a viz instance isn't available. You can write an R script and submit it with a sbatch script. Let's look at some basic R syntax. Go ahead and launch R in the command line. Do this by typing R and Entering. Notice when you do this your username will be replaced by ">" indicating you are in R.

```
tajo5912@fiji-2:~ R      Type "R" and enter to go into R
^ MY USERNAME
R version 3.6.0 (2019-04-26) -- "Planting of a Tree"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-redhat-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]
Notice there is no USERNAME here
> indicates you are in R not bash
```

- 2) Now that you are in R do a basic calculation of your choosing to test it out!

```
> 19+22
[1] 41
>
```

Here I typed 19+22(Enter) and it returned the answer to the summation which is 41.

- 3) Set some variables. In R you can set variables that represent numbers, lists, data frames etc. Let's start with numbers. To create a variable, you use the following syntax: "VARIABLE NAME <- CONTENT" For example:

```
> a<-19
> b<-22
>
```

- 4) We can perform arithmetic on these variables now that they are set. We get the same answer, 41.

```
> a+b
[1] 41
>
```

- 5) Now let's set a slightly more complex variable. Let's make a list. With a list you need to call "c" to define the datatype.

```
> list<-c(1,3,5,7,9,11,13)
```

Here I made the list. Let's call list to check what is inside.

```
> list
[1] 1 3 5 7 9 11 13
>
```

You can see it returns the list we just created.

- 6) Let's test some basic functions on this list such as sum, mean and standard deviation.

```
> sum(list)
[1] 49
> mean(list)
[1] 7
> sd(list)
[1] 4.320494
>
```

You invoke R functions by calling the function name (such as "sum") followed by your variable in parentheses (such as "list"). Some functions require more than one input, with the inputs separated by comma. Let's exit command line R and switch to RStudio to try it out!

- 7) Exit R on command line by typing q(), saving the workplace image saves the work we did.

```
> q()
Save workspace image? [y/n/c]: y
```

Section B:

Section B will be in RStudio. I have provided 2 different scripts that we will investigate. We will start with Learning_R.R in RStudio.

Open up RStudio on the viz instance. Once it is open go to File → Open File.. → Select Learning_R.R Further instructions are in this script!

The other script is featureCounts.R that will cover counting reads from bam files and metadata tables that will be used for both counts and for differential expression analysis. Note: We run featureCounts on a cluster/instance because bams are big!!