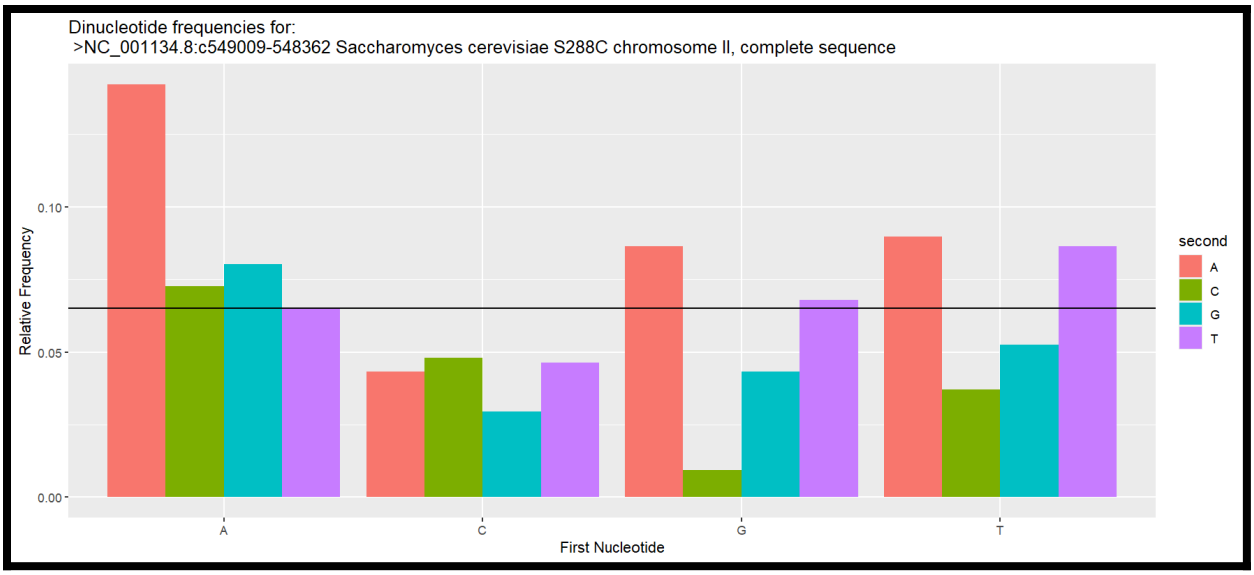
Lab 9: File input

Biol135: Introduction to Bioinformatics Programming.



Part 0: At the top of rstudio click new, but instead of creating a new script, create a new Rshiny web app. This R framework allows you to create simple web apps that can run locally on your device. A sample app will load showing a simple histogram that can be modified using a slider input. Take some time to see how the UI and the Server component of the app work together to produce the final product.

Part 1: Modify the app so that it prompts the user for a fasta file using the fileInput() function. For now don't worry about error catching, you can assume the file being entered will be a valid Fasta file containing one sequence of DNA. This will require the readFasta() function from the Seqinr package so make sure that package is installed and loaded. Note: The object saved in your workspace when using fileInput() is a data frame. To access the filename that can be passed to seqinr's read.fasta() function you must select [your file input]$datapath

Example:
In UI:

fileInput("fastaFile", "Choose fasta file")

In server

file <- input$fastaFile
fasta <- read.fasta(file$datapath)

Lab 9: File input

Biol135: Introduction to Bioinformatics Programming.

Part 2: Once the program reads the Fasta file it will quantify the dinucleotide frequencies. This should result in a dataframe containing a column for the starting nucleotide, a column for the finishing nucleotide and a column for the frequency of that dinucleotide.

Part 3: Using the data frame plot the frequencies as a grouped bar chart using ggplot2(). The x axis should be the c starting Nucleotide and each group of bars should have a uniquely colored bar for each potential Following Nucleotide. The frequencies should be relative, meaning that the value for each bar should be the number of the given transitions divided by the total number of transitions.

Part 4: Add a horizontal line that is 1/16 = 0.0625 units from the x axis. This line represents the expected frequency of each dinucleotide assuming random transitions. Add a custom x and y axis label and a graph title that includes the annotation for the fasta file. This is accessible via seqinr's getAnnot() function.

Part 5: Add a comment for each major section of your code. Since R shiny requires the filename always be app.R, add a comment with your name at the top. Upload to the dropbox in MyCourses. (Since this lab has an additional layer of complexity we will give two weeks to complete it. Submit by November 14th at 11:59 pm.

The following parts are not required but will yield a bonus:

Part 5.1: If you're app is working you may take the additional step of publishing it on shinyapps.io. This will make your app accessible via a URL. You will be prompted to download some additional packages and create an account.

Part 5.2: Modify your code such that a fasta file containing multiple sequences will yield a series of graphs. One for each sequence and the graph title indicates the unique

Part 5.3: