SLE712 Assignment 3

Due Friday 29th May (week 11)

- This assignment consists of two bioinformatics exercises.
- You may work individually or in groups of two or three, so long as there is evidence of contribution to the code repository by all members.
- Your submission will consist of a written report **AND** one GitHub repository.
- The report will be submitted as one PDF document to the CloudDeakin dropbox. Submitting in a different format will result in a 5% deduction. There is a maximum word count of 1000 words.
- The report should include a cover sheet with names, student numbers, unit code, date of submission and assignment title
- If you describe ideas and works that are not your own, you must reference your sources with in-text citations and a list of references according to the Harvard style: https://www.deakin.edu.au/students/studying/study-support/referencing/harvard
- Any further questions please contact me by discussion board or email: m.ziemann@deakin.edu.au
- This assignment is worth 20% of your total grade for the unit. A breakdown of the marks is provided:

Report	Component	Marks given (total=100)
Part 1 (50 marks)	Code works (2 marks per qn)	20
	Code documentation (README and comments)	10
	Evidence of team coding (source control)	10
	Written answers (1 mark per qn)	10
Part 2 (50 marks)	Code works for points 1 - 4; 2 marks each	8
	Code works for points 5 - 6; 6 marks each	12
	Code documentation (README and comments)	10
	Evidence of team coding (source control)	10
	Written answers for points 1-4; 1 mark each	4
	Written answers for points 5-6; 3 marks each	6

Rubric:

Component	Full mark	Half mark	No mark
Code works	The code provided on Github executes without errors and generates the correct answer	The code provided on GitHub has a slight mistake which gives an incorrect answer but there is evidence that student has used the learning materials and made an attempt	The code yields an error or there is a major mistake, or no GitHub repository was provided
Code Documentation	The repository has a detailed README that accurately describes the contents. The code contains enough comments to describe what each chunk of code is doing	There is a README and some comments but they are not detailed enough or contain inaccurate information	There was no attempt to document the repository
Evidence of team coding	All group members made numerous contributions including code, issues and documentation versioning	Each member made one contribution to the repository	Only one commit was made, or no repo was provided
Written answers	Addresses the question accurately and is consistent with code provided. Student provided a clear description of how the problem was solved.	The question was answered accurately but the method used to solve the problem was not given. Minor inconsistencies between answer and code. Minor grammar or spelling errors.	Student response did not answer the question or there are major inconsistencies between code and answer. Major grammatical and spelling errors.

Part 1: Importing files, data wrangling, mathematical operations, plots and saving code on GitHub

The purpose of this exercise will be for you to develop skills in problem solving, R coding, work together as a team using Rstudio and GitHub. You will be provided with two data files to work with: “gene_expression.tsv” and “growth_data.csv” which are available from this URL*:
https://github.com/markziemann/SLE712_files/tree/master/bioinfo_asst3_part1_files

* To download a file with R, click on “view raw” and then you can copy the URL from the address bar and then use the download.file command in R.

- For points 1-10 below
    - Describe how you solved the problem.
    - Provide the answer as directed. The answer could be a descriptive, numerical, categorical, table or chart.
- Provide a link to GitHub repository with the following:
    - The code should run without errors, and yield answers to points 1-10 below.
    - If working in a group, there needs to be evidence that all group members have made contributions to the code repository. This means that there needs to be “commits” and “issues” from each group member.
    - A README that describes the purpose of each script and their inputs and outputs.
    - The code should contain sufficient comments so that someone else can understand what each line or chunk of code is trying to achieve

---

The file “gene_expression.tsv” contains RNA-seq count data for two samples of interest.

1. Read in the file, making the gene accession numbers the row names. Show a table of values for the first six genes.
2. Make a new column which is the mean of the other columns. Show a table of values for the first six genes.
3. List the 10 genes with the highest mean expression
4. Determine the number of genes with a mean <10
5. Make a histogram plot of the mean values in png format and paste it into your report.

The file “growth_data.csv” contains measurements for tree circumference growing at two sites, control site and treatment site which were planted 20 years ago.

6. Import this csv file into an R object. What are the column names?
7. Calculate the mean and standard deviation of tree circumference at the start and end of the study at both sites.
8. Make a box plot of tree circumference at the start and end of the study at both sites.
9. Calculate the mean growth over the past 10 years at each site.
10. Use the t.test and wilcox.test functions to estimate the p-value that the 10 year growth is different at the two sites.

Part 2: Determine the limits of BLAST

In class you will be shown how to

- Download and unzip files
- Perform simple manipulations and analyses with sequence data
- Use a provided function to incorporate point mutations into a sequence
- Use provided functions to perform a BLAST search and interpret results

In this assignment we will be testing your ability to use supplied functions to perform an analysis into the limits of BLAST. Your group will be allocated **one** *E. coli* gene sequence found in the file: https://raw.githubusercontent.com/markziemann/SLE712_files/master/bioinfo_asst3_part2_files/sample.fa

For example if your Rstudio username is student71 then your sequence is 71. Each group selects just 1 sequence. Next, you will need the whole set of E. coli genes can be downloaded from this link: ftp://ftp.ensemblgenomes.org/pub/bacteria/release-42/fasta/bacteria_0_collection/escherichia_coli_str_k_12_substr_mg1655/cds/Escherichia_coli_str_k_12_substr_mg1655.ASM584v2.cds.all.fa.gz

---

- For points 1-6 below
    - Describe how you solved the problem.
    - Provide the answer as directed. The answer could be a numerical, categorical, table or chart.
- Provide a link to GitHub repository with the following:
    - The code should run without errors, and yield answers to questions 1-6 below.
    - If working in a group, there needs to be evidence that all group members have made contributions to the code repository. This means that there needs to be “commits” and “issues” from each group member.
    - A README that describes the purpose of each script and their inputs and outputs.
    - The code should contain sufficient comments so that someone else can understand what each line or chunk of code is trying to achieve

---

1. Download the whole set of *E. coli* gene DNA sequences and use gunzip to decompress. Use the makeblast() function to create a blast database. How many sequences are present in the *E.coli* set?
2. Download the sample fasta sequences and read them in as above. For your allocated sequence, determine the length (in bp) and the proportion of GC bases.
3. You will be provided with R functions to create BLAST databases and perform blast searches. Use blast to identify what *E. coli* gene your sequence matches best. Show a table of the top 3 hits including percent identity, E-value and bit scores.
4. You will be provided with a function that enables you to make a set number of point mutations to your sequence of interest. Run the function and write an R code to check the number of mismatches between the original and mutated sequence.
5. Using the provided functions for mutating and BLASTing a sequence, determine the number and proportion of sites that need to be altered to prevent the BLAST search from matching the gene of origin. Because the mutation is random, you may need to run this test multiple times to get a reliable answer.
6. Provide a chart or table that shows how the increasing proportion of mutated bases reduces the ability for BLAST to match the gene of origin. Summarise the results in 1 to 2 sentences.