13/01/2021

## Report: Data Science and Machine Learning for the Biosciences

(To access my GitHub repository containing Jupyter notebook, script, data and report files, follow link provided. GitHub repository URL: <https://github.com/jag245/Data-Science-Assessment>)

**Introduction:**

My PhD subject area falls under the remit of “Agriculture and Food Security”, and bears the title, “A genomic approach to understand insecticide toxicokinetics in a global crop pest.” As such, genomic analyses are highly likely to form a significant component of this project. For this reason, I have elected to design an elementary piece of software using Python to perform comparative genomic analysis between nucleotide sequence data.

The peach potato aphid, or *Myzus persicae*, is the globally significant crop pest I am studying. It exhibits unusually high levels of resistance to important insecticides, even if they are relatively new, exemplified by rapidly emerging insecticide resistance in agricultural systems across the world. Exeter University has a recently established living library of over 110 *M. persicae* clones,1 sampled from across the globe. Therein are numerous aphids demonstrating resistance or susceptibility to a variety of insecticides. It is therefore of interest to perform comparative genomic study between those aphids which do exhibit resistance to a specific insecticide, and those which do not. Particularly within annotated genes, events such as single nucleotide polymorphisms (SNPs) may be representative of different alleles between different aphid clones. Also additional or missing regions of the genome, also known as indels, may point to extra or absent genetic material which may play a role in insecticide resistance and its development. Identifying the presence of potential genomic differences between aphid clones, whether in the guise of SNPs, indels or other forms, may lead to the elucidation of new targets for insecticides and the identification of important resistance alleles.

For these reasons, a program to compare the above differences is likely something I will require over the course of my PhD.

**Objectives**:

**Original aim**: The aim for my code will be to take two custom fasta nucleotide sequence files from two *M. persicae* clones and compare them in a pairwise analyses at equivalent positions along their sequence. Fasta files that align from the first contigs of either clone will be created for this comparison. The program will be designed to output a file recording whether the nucleotides at each index position in the sequence are the same (100% identity) or different (0% identity). For instance, if at index position [0], “clone O” has an “A” and “clone G006” has a “G”, then this will be recorded as a difference, and should both nucleotides be the same, then this will also be recorded. In the event of a nucleotide difference, the program will also output the difference between nucleotides, separated by a colon (i.e. A:G). Insertions or deletions will also be represented as ‘gaps’ in the output file, should such gaps be present when the sequence data are aligned.

**Modified aim:** Due to difficulties in finding equivalent contigs between genomic assembly data from two different aphid clones (O clone and G006 clone) as well as the large file sizes involved and issues regarding exact alignment of sequence data prior to running any code, a change was made to the initial sequence data files.

As the purpose of this program was to create an output file including any differences in nucleotide sequence between two genomic sequence data files, then neither its functionality nor its applicability to my PhD project are dependent on the input files. Therefore two more tractable smaller genomic files were selected from isolates of the *Barley yellow dwarf virus* (BYDV luteovirus), a cereal pathogen transmitted by an aphid vector. These NCBI sourced genomic sequence data belonged to; Ker-II isolate K439 (complete genome) 2, and Ker-III isolate K460 (complete coding sequence, CDS)3**.**

Prior to entry into the program, the two nucleotide fasta files required alignment due to an unequal length of sequence data with non-equivalent positional indices. An EMBL EMBOSS nucleotide alignment was run,4 aligning corresponding indices, and substituting dashes (-) for gaps due to missing nucleotides in either sequence. This permitted detection of the potential presence of any insertions or deletions in addition to potential SNPs.

**Python Code:**

The annotated notebook and data files are included in the zip-file/on GitHub, however here included is a brief description of the program.

On obtaining two nucleotide text files for each BYDV isolate following EMBOSS alignment, these are first opened and assigned a file handle.5 Using the .read() method, both files are then read into a variable as a string. For ease of comparison, each string is converted into a list using the list() function. A while loop satisfied by the condition of the lengths (len()) of these lists being equal (2169 nucleotides) is opened, to ensure the program proceeds only if files are uploaded correctly and can be equivalently compared. The program the proceeds to enter a nested “for” loop, wherein each nucleotide of the Ker II K439 isolate is enumerated, assigning it an index value. For each index value of the Ker II isolate, the nucleotide at that position is compared against the same index of the Ker III isolate. Then, beginning an “if-elif-elif” statement, if nucelotides are found to be equal, “100% identity” is appended to an external empty list entitled “nucleotide\_identity”. The index at the same position is appended to another external empty list entitled “nucleotide\_position\_index.” The “if-elif-elif” statement proceeds to evaluate whether either nucleotide string has a “-“ instead of a nucleotide, if so appending “gap” and the index position to the same lists. The statement concludes by assessing whether the nucleotides are not equal to one another, outputting “0% identity” and the index position to the two lists respectively if this is the case.

The loop iterates though every pair of nucleotides this way, returning two lists containing sequence identity and index position. To ultimately output this as an Excel file, these lists are firstly combined via entry into a dictionary. The “pandas” data analysis library is then imported and used to convert the dictionary into a dataframe. Finally this is outputted to an Excel file using the “df.to\_excel” function.

**Outcomes:**

A brief analysis of the output files shows that 20.7% (3s.f.) (450/2169) of sequence comparison between BYDV isolates results in gaps (“-“ characters) after pairwise alignment. The indices of these positions in the sequences are therefore of interest to look for indels, and perhaps some are due to the difference in sequence completion between the complete genome of the Ker II isolate and the complete CDS of the Ker III isolate. Also, 18.1% (3s.f.) (392/2169) of nucleotides are found to differ between isolates. These indices and their difference in nucleotide residues will be of interest, particularly within annotated coding regions to look for potentially different alleles between isolates. For further analysis, outputted index values could be examined using a genome viewer (e.g.IGV) to identify SNPs or other features in areas of interest.

**Conclusion:**

The program was run as a proof of concept for larger more complex eukaryotic genomic files from *M. persicae*, for when they are in a format more tractable for analysis, and differences in insecticide resistance have been fully quantified in the lab. However, the core program functionality will be applicable further down the road, and when modified to handle additional file input requirements, a version of this program could be employed to compare protein coding sequence data between resistant/susceptible aphid clones and as stated in the outcomes, signpost further bioinformatic analyses with a genome browser.

**References:**

1. Bass, C. *et al.* Worldwide patterns of genetic diversity underpinning the evolution of resistance to natural and synthetic insecticides in the crop pest Myzus persicae. *Unpublished* (2020).

2. Barley yellow dwarf virus Ker-II isolate K439, complete genome - Nucleotide - NCBI. https://www.ncbi.nlm.nih.gov/nuccore/NC\_021481.1?report=fasta&from=2872&to=4875.

3. Barley yellow dwarf virus Ker-III isolate K460 RNA-dependent RNA polym - Nucleotide - NCBI. https://www.ncbi.nlm.nih.gov/nuccore/NC\_043123.1?report=fasta&from=2683&to=4566.

4. EMBOSS Needle < Pairwise Sequence Alignment < EMBL-EBI. https://www.ebi.ac.uk/Tools/psa/emboss\_needle/.

5. File Handling in Python - Python Tutorial - OverIQ.com. https://overiq.com/python-101/file-handling-in-python/.