Automated Processing of Primary Genome Analysis

- Student: Zeyu Yang (Tony) [zy2414@ic.ac.uk]
- Mentor: Andy Yates

Description

Since the success of the Human Genome Project in 2003, many novel technologies have been invented to accelerate genome sequencing speed. With the development of new sequencing machineries and their much-reduced cost, a large quantity of novel genome data is produced daily at an increasing rate. Hence, it is important to develop an automated process that analyses newly sequenced genome data efficiently.

The primary aim of this project is to construct a generic framework that would allow automated processing of new genome sequences. European Nucleotide Archive (ENA), one of the leading repository for nucleotide sequence data, has over one billion records and its collection is still expanding exponentially. ENA records a large variety of nucleotide data, from DNA sequencing machine configurations to sequence traces and annotated information. Those data gathered from different sources, such as small-scale lab sequencing and European sequencing centres, is made freely available online, and can be accessed via ENA's REST API.

GC content, the percentage of guanine (G) or cytosine (C) bases on a genome, provides vital information about this organism. Guanine (G) and cytosine (C) base pairs stack thermodynamically more favourable than adenine (A) and thymine (T) base pairs, hence facilitate the DNA's stability. GC content is varied for different organisms and can provide evolutionary insights into particular organisms. Therefore, GC analysis is a meaningful and relative easy algorithm to implement, and will allow a general framework to be constructed around it.

Common Workflow Language (CWL) will be used to construct the automation framework. CWL documents, written in JSON or YAML, are used to describe the connection of different command line tools, and it was developed in aid of scientific data analysis. Workflows described in CWL specification are easily portable and scalable in different computational environments. Because of the explicit and isolated nature of CWL tasks, CWL workflows can be containerised to allow easy deployment.

Deliverables:

A portable, scalable and modular Common Workflow Language (CWL) system that performs analysis on novel genome sequences.

Timeline:

During the community bonding period, I can devote 10 hours per week to get familiar with the necessary software tools and research the algorithms for DNA analysis. I will also get to know my mentor.

During the project period, I can devote up to 48 hours per week to work on research and coding.

14th May - 27th May: Enquire chromosome data through ENA's REST API in FASTA format. File handler to save and organise obtained data.

28th May - 10th June: Coding for the GC analysis algorithm. Research into other DNA analysis algorithms that only needs the genome (e.g. repeat masking, sequence alignment).

11th June - 24th June: Create and test CWL workflow that would produce a BigWig file with GC content results for a given genome.

25th June - 8th July: Containerised the workflow and test for portability.

9th July - 22nd July: Coding for the additional analysis algorithm. Testing the automation process with the new analysis method.

23rd July - 5th August: Research into submission and long term storage of processed data. Documenting the whole project.

6th August - 14th August: Tidy code and finalise documentation; submit final code.

Implementation

- Each sequence in ENA's database is identified with a unique ENA accession number. ENA supports REST API to allow data enquiries through URLs. So the desired genome data can be located with an accession number, and downloaded via HTTP, in FASTA format.
  E.g. Chromosome 1 of human for GRCh38 in FASTA format can be downloaded via this URL:
  https://www.ebi.ac.uk/ena/data/view/CM000663&display=fasta&download=text

- Calculate GC content of a given window size (e.g. 5 base pairs) of the sequence and create a wiggle file contains the results. The first line of the FASTA file is the description line, and the raw sequence starts from the second line, with 60 characters per line. The program will read the sequence 5 base pairs (characters) at a time, calculate a GC percentage of this window then move on to the next 5 base pairs window.

Wiggle file format contains a declaration line followed by the data block. Each line in the data block contains the chromosome position and the data (GC percentage in this case), separated by a space.
E.g. for a sequence such as “CAGCTTCTCCTCTGT” would result in the following wiggle output:

Declaration line
40930 60
40935 60
40940 40

- Using UCSC’s wigToBigWig program to create a BigWig file from the wiggle file containing the GC content output. BigWig is a compressed indexed binary file format that is useful for displaying dense, continuous data. BigWig format also allows simple calculation of a variety of summary statistics, which can be valuable for future usage.

- A CWL workflow will be developed to describe the whole process from data access to analysis to create the BigWig file.

- The CWL workflow will be containerised in a Docker container and tested for portability.

About me:

I am a final year MSci Chemistry student at Imperial College London, UK. My initial interest was in organic chemistry and total synthesis. After a summer placement at UCLA with Prof. Houk in 2017, I became interested in computational organic chemistry and the theoretical studies of reaction mechanisms, which eventually led me to bioinformatics. I have had a long interest in programming and data science. I learnt Python in my first and second year of undergraduate studies, and took ‘Introduction to Data Science in Python’ course on Coursera.

I currently hold an offer for the Wellcome Trust 4-Year PhD Programme in Theoretical Systems Biology and Bioinformatics, starting in October 2018.