Project Report

03-701: Practical Computing for Biologists Fall 2022
A python script to examine a bacterial genome Andrew ID: wanzix

1 Introduction

In the fields of molecular biology and genetics, a genome is all genetic information of an organism[1]. Sequencing technologies are now able to return whole genome sequences of individual species. From the analysis of gene sequencing, we can get a lot of valid genetic information that can be applied to predict gene expression. For example, from the DNA sequence, we can obtain the order of a protein by transcription and translation.

Python is a powerful tool not only in building web development but also in scientific analysis. Python has a wide range of packages for genetic analysis including Biopython and other data analysis packages such as NumPy, Pandas, etc.

In this Python project, I imported Biopython to implement an examination of a specific bacteria genome: Escherichia coli JE86-ST05 DNA. With scanning DNA strands and translation of open reading frames, the script returns the potential expressed protein sequence. Then we also calculated predicted molecular mass and blast 5 protein coding sequences at NCBI with a return of most similar hits.

2 Methods

2.1 Retrieve sequences

The first part of code is to retrieve basic sequence information from NCBI, including description, id, and name.

2.2 Scan open reading frames

Import `re` package and use regular expression to find stop codons in the sequence. Then we "cut" the sequence to obtain open reading frames. For those with the number of codons larger than 50, we ignore them.

2.3 Translation

Apply stardard table 1 in `Biopython` package to translate open reading frames into protein sequences. Two `for-loops` create six different frames. The internal `for-loop` translates each frame according to a selected translation table and returns a chain of amino acids in which each termination codon is encoded as `*`. By setting `max_pro_len` we can define the maximum amino acid chain length for a protein to be detected.

2.4 Molecular mass calculation

Import `SeqUtils` from `Bio`, then call build-in `molecularmweight` function. This function calculates molecular mass of protein sequences. By divided by 1000, we change the unit to kD.

2.5 BALST

Through using BLAST search, finds regions of similarity between biological sequences[2]. First we access NCBI's BLAST server, then we range through each element in the list of BLAST results to return the most similar hits. To implement this, we range over all the *score* attribute of each result and find the largest value.

2.6 Code structure

Table 1. Code structure

Task	Package	Description
Retrieving sequences	Bio.Entrez Bio.SeqIO Bio.Seq	Import Entrez from Bio to access NCBI.
Scan open reading frames	re	Use regular expression to find stop codons. Find stop codon to 'cut' sequence.
Translation		Obtain the complement or reverse complement of a Seq object using its built-in methods.
Molecular mass Calculation	Bio.SeqUtils	Calculate molecular mass and change unit.
BLAST	Bio.Blast Bio.Blast.NCBIWWW Bio.Blast.NCBIXML	Find largest score in the NCBI for selected protein sequence.

3 Results

3.1 Open reading frames

The original open reading frames could be very large since we select complete genome of a specific E.coli, so we modified the sequence and return part of the whole genome. This part of script uses a seq as input and return a slice of open reading frames with a restricted number of codons by recognizing stop codons.

3.2 A translated protein sequence

The result of protein sequence is the order of translated protein sequence, and we also return the length of each sequence as long as the strand and frame in it to give a clearer insight. We ignore protein sequence with length larger than 1000 to be better prepared for BLAST. Our output is like the following line:

```
[output]SFSF...FSF - length 4, strand 1, frame 0
```

3.3 Molecular mass of proteins

We return the list of predicted molecular mass in each protein sequence.

3.4 BLAST

The results for manually-choosen protein sequence are listed in the following table:

Table 2. Most similar hits with 5 protein coding sequences

Protein sequence	Sequence of most similar hits	Length	score
SFSF	sequence: gb|ACC99433.1| maturase K, partial [Elleanthus caricoides]	434	2005.0
LQWAICLCVD	gb|ACC99433.1| maturase K, partial [Elleanthus caricoides]	434	2005.0
KKSL	gb|ACC99433.1| maturase K, partial [Elleanthus caricoides]	434	2005.0
QQLLNWLPAVSKLKFY	gb|ACC99433.1| maturase K, partial [Elleanthus caricoides]	434	2005.0
LRSLNTLTNIGIAHRQIKITEYTTSMKRISTTITTTITITTGNGAG	gb|EAP3818702.1| thr operon leader peptide [Salmonella enterica] >gb|ECS4318266.1| thr operon leader peptide [Salmonella enterica subsp. enterica serovar Typhimurium var. 5-] >tpg|HAB6922902.1| thr operon leader peptide [Salmonella enterica subsp. enterica serovar Typhi str. CT18] >gb|EAV2688376.1| thr operon leader peptide [Salmonella enterica] >tpg|HAD4260575.1| thr operon leader peptide [Salmonella enterica subsp. enterica serovar Typhi str. CT18]	23	79.0

4 Future expansions and improvements

Since we have multiple sequences, we can apply methods in `Biopython` multiple sequence alignment objects to the alignment of two or more sequences together to mark their similarities. To implement this, we can import `Bio.AlignIO` so that read and write sequence alignment files.

5 Reference

[1]Roth, Stephanie Clare (1 July 2019). "What is genomic medicine?". Journal of the Medical Library Association. University Library System, University of Pittsburgh. 107 (3): 442–448. doi:10.5195/jmla.2019.604. ISSN 1558-9439. PMC 6579593. PMID 31258451.
[2] Allesina, Stefano; Wilmes, Madlen. Computing Skills for Biologists: A Toolbox (p. 212). Princeton University Press. Kindle Edition.