BCB 444 Fall 2015 Project 4

Forming Data Pipeline and Formatting with Perl

Part I

Due Sep. 30 by 1 p.m.

1. Introduction

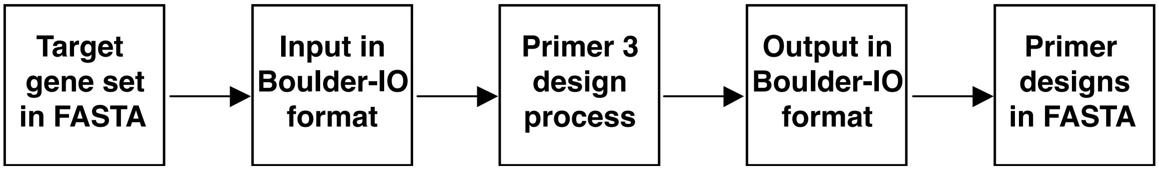
Polymerase Chain Reaction (PCR) is an important molecular biology technique that allows the amplification of targeted DNA sequences. This technique is so widely used that it is safe to say PCR is probably one of the most important molecular biology techniques invented in the past 30 years! To utilize PCR to amplify a target gene, a researcher must design a pair of DNA *primers* based on the target gene sequence. Primers are short DNA oligonucleotides that can anneal (i.e., forming hydrogen bonds) to either end of the amplification site on the target gene in order to uniquely control the amplification process. For more details about PCR, please consult your favorite molecular biology textbook or the Wikipedia. This project and the next one focus on the formatting requirements to use one of the best-known PCR primer design software — **Primer3**. It can be downloaded and installed onto your own computers if you are interested in using it later: http://primer3.sourceforge.net/, but for the purpose of this and the following project, you do not need to install Primer3 yourself.

2. Primer3

Steve Rozen and Helen Skaletsky originally developed Primer3 at the Whitehead Institute and Howard Hughes Medical Institute back in the 1990's. Primer3 predates the many other primer design software tools developed later with graphical user interface. Most people these days prefer to use either a web service or a graphical user interface application to design their primers. However, Primer3 has always been one of the best primer design software since its initial creation, and its rich functionality should not be overlooked just because its user interface is command-line driven. Actually, there are several web applications with embedded Primer3 design services if people prefer that kind of web interface: http://primer3.sourceforge.net/webif.php Nevertheless, as bioinformaticists with the knowledge of Perl, using the command-line version of Primer3 should not be too difficult and requires only a one-time setup. If we need to design tens of thousands of primer pairs for NGS target gene enrichment, it is far more efficient to use the command-line Primer3. In practice, many high-throughput genomic applications require a *pipeline* of data processing that sends input data to available software tools such as Primer3, collects the output from the tools and then converts the data to a final output format. Such a data pipeline can be easily created using Perl.

3. Input and output to Primer3

In this project we are assuming that you are in a situation where Primer3 has been downloaded and installed on a local computer and you are trying to access it using a command-line interface. You don't really need to have access to Primer3 to complete this project. Nevertheless, we have downloaded and installed Primer3 to the class account (i.e., gdcb-bcbteach.gdcb.iastate.edu) so you may test your data pipeline to using Primer3 as described below and also at the end of Project 5 description.



Primer3 was designed with a *pipeline* concept in its input and output as depicted in the picture above. The input to Primer3 is in the so-called **Boulder-IO** format. It is a very simple format consisting of various "property=value" pairs with one pair occupying one line. If an equal sign "=" stands on a line by itself, it ends a *block* of data. Each block in our project application contains the information specific to a gene in the input to Primer3. The Primer3 output will replicate the input block information it receives and also add the primer design information computed by Primer3. The three relevant input properties for each gene are as follows:

```
SEQUENCE_ID=egl-2
SEQUENCE_TEMPLATE=ataacactttattctattgaaatgtcatagaca ...
SEQUENCE_PRIMER_PAIR_OK_REGION_LIST=0,3318,0,3318
=
```

SEQUENCE_ID identifies the gene name. **SEQUENCE_TEMPLATE** is the gene sequence (in just one line; which can be very long but still must be on one line). The last property **SEQUENCE_PRIMER_PAIR_OK_REGION_LIST** defines a region on the gene sequence where Primer3 should pick new primers. Normally the primer selection region should be from 0 to the full length of the gene (i.e., length($seq)-1 because Primer3 also counts from base position 0), but for certain applications such as when there are alternative splicing of a gene, a researcher may wish to restrict primer selections to the shared common exons among all isoforms of a gene so all transcripts of the same gene can be detected by the same primer pair. More information about the last property will be discussed below.

Primer3 also employs the concept of *stream* input and output, meaning that the input to Primer3 in Boulder-IO format will not be altered, but new data produced by Primer3 will be added to the stream of data in the output. Primer3 puts its primer designs in the output using the following additional property tags:

```
PRIMER_LEFT_NUM_RETURNED=10
PRIMER_RIGHT_NUM_RETURNED=10
PRIMER_INTERNAL_NUM_RETURNED=0
PRIMER_PAIR_NUM_RETURNED=10
PRIMER_PAIR_0_PENALTY=0.642211
PRIMER_LEFT_0_PENALTY=0.414630
PRIMER_RIGHT_0_PENALTY=0.227581
PRIMER_LEFT_0_SEQUENCE=gctaacacttgacgcggcgc
```

```
PRIMER_RIGHT_0_SEQUENCE=actcgccggcttcaagctcg
PRIMER_LEFT_0=2763,20
PRIMER_RIGHT_0=2878,20
PRIMER_LEFT_0_TM=59.585
PRIMER_RIGHT_0_TM=59.772
PRIMER_LEFT_0_GC_PERCENT=65.000
PRIMER_RIGHT_0_GC_PERCENT=65.000
PRIMER_LEFT_0_SELF_ANY_TH=21.06
PRIMER_RIGHT_0_SELF_ANY_TH=15.25
PRIMER_LEFT_0_SELF_END_TH=21.06
PRIMER_RIGHT_0_SELF_END_TH=7.21
PRIMER_LEFT_0_HAIRPIN_TH=0.00
PRIMER_RIGHT_0_HAIRPIN_TH=42.60
PRIMER_LEFT_0_END_STABILITY=6.5300
PRIMER_RIGHT_0_END_STABILITY=5.0300
PRIMER_PAIR_0_COMPL_ANY_TH=11.15
PRIMER_PAIR_0_COMPL_END_TH=19.54
PRIMER_PAIR_0_PRODUCT_SIZE=116
```

Properties in the list above that are highlighted by **bold font face** are output from Primer3 that are important to us. The first four properties about `*_RETURNED` are primer design summaries for each input gene. The rest of the properties include the number _0_ in names are information about the first pair of primers. The same set of properties is repeated up to 10 times with numbers 1, 2 ... 9 to represent the 10 primer pairs chosen for each target gene by Primer3.

Our jobs in this project are to format the proper input from a FASTA file to Primer3. In the next project we are going to extract Primer3 designed primers from its output and save that to another FASTA file. To format proper input to Primer3, a front-end data pipeline must convert target gene sequences in the common FASTA format to the special Boulder-IO format recognized by Primer3. To extract primers designed by Primer3, a back-end data pipeline must convert the Boulder-IO output back to the FASTA format containing the paired primer sequences. The front-end and the back-end of the data pipeline involving Primer3 are the two projects we are going to do. In this project, we first work on the input data pipeline to Primer3.

4. Formatting input to Primer3

In this project, we are given an input target gene set "target_genes.seq" in the common FASTA format and we need to write a Perl program that can convert the gene set into the appropriate Primer3 Boulder-IO input format. The following is a typical FASTA file partial content:

```
>WBGene00006810|unc-78|C04F6.4|C04F6.4b.1|coding 68 1608
cgtttgattttattgattttcttccttcgatcttttggattcctgctcaatttttacaat
ttttgcagaatgtcggaattctctcaaaccgcattgttcccgtctttgcctcgcactgct
agaggaactgccgtagttctcggcaacacccccgctggagacaagattcaatattgtaac
ggaacatcggtttatactgtaccagttggaagcctaaccgacaccgaaatctacactgag
...
```

In this input gene set, the gene name is long and contains multiple parts, but only the second part in the gene name, (e.g., in the above example, "unc-78"), needs to be copied to the Boulder-IO input to Primer3 as gene name. In addition, two *optional* integer numbers may be given in the FASTA gene header line to denote the restricted region for primer design. If a gene header line lacks these two numbers, it is assumed that the whole gene sequence (0 – length($seq)-1) can be used for primer design. Therefore, the above input gene example should be converted to the following Boulder-IO input block:

```
SEQUENCE_ID=unc-78
SEQUENCE_TEMPLATE=cgtttgattttattgattttcttccttcga... (one long line)
SEQUENCE_PRIMER_PAIR_OK_REGION_LIST=68,1541,68,1541
=
```

In the input gene header line, the two numbers denote the left and right coordinates of the primer design region. On the SEQUENCE_PRIMER_PAIR_OK_REGION_LIST property line, however, there are four integer values "left,length,right,length" (note: no space between the numbers and commas) that denote the starting region of left primer selection, its length, the starting region of right primer selection, and its length. In the example above, the length is obtained by the following calculation: 1608-68+1=1541. In this project, the left and right primers share the same allowable design region, thus in the example above their design regions both start at base 68 and are 1541 base-pair long.

For your validation, a sample Primer3 input file "primer3_input.txt" corresponding to the "target_genes.seq" file is also provided. Your project should produce the same content in "primer3_input.txt" when given the "target_genes.seq" file as input (don't get confused, "primer3_input.txt" should match the **output** of your program, which takes "target_genes.seq" as **input**). Note that at the beginning of the "primer3_input.txt" file there are some global property settings with **`P3_*`** and **`PRIMER_*`** name prefixes which instruct Primer3 how it should design all its primers. These properties appear only once at the very beginning of the input to Primer3 and thus should be copied from the sample file to your Perl program and replicated in its output to Primer3.

5. Project submission and grading

A skeletal implementation of Project 4 has been provided. You may find it helpful to get you started. To submit your project, upload your code to Blackboard Learn project area. To facilitate grading, please name your Perl project file project_4.pl. You should also add Perl comments in your code to identify yourself (e.g., your full name and University ID) and to explain your implementation. It is not necessary but it may be helpful to also include a README.txt file in your submission or to write its content in the Blackboard Learn submission form to provide more information or any special consideration about your project implementation.

The TA will grade your project after the deadline, so you may submit multiple times before the deadline. If your project works on the sample data provide, your score should be no less then 15 (out of 20 total for the project). However, We will certainly test your code on other input data not made available to you to see if your code is general enough and is not making any assumption on some fixed characteristics of the input (e.g., the 5th gene contains designated primer design region, which may be different in our test files). Keep in mind that it is THIS project description that officially defines how your project code should behave and will be graded, not the sample input and output files provided. Therefore, after successfully testing your project on the provided sample data files, you may wish to create your own additional test files by changing the "target_genes.seq" input data file to test for different input data. If you have any question about this project, post your project related questions to the Project discussion forum on Blackboard Learn.