Demultiplexing Project

[rlancio5@talapas-ln1 Demultiplexing\_Proj]$ pwd

/projects/bgmp/rlancio5/Bi621/Demultiplexing\_Proj

Table of files and their contents

Each file befins with prefix “1294\_S1\_L008\_”

|  |  |  |  |
| --- | --- | --- | --- |
| R1\_001.fastq.gz | R2\_001.fastq.gz | R3\_001.fastq.gz | R4\_001.fastq.gz |
| Read 1 | Index 1 | Index 2 | Read 2 |

Files:

Input Files = 2Read. 2 Index

Total Input = 4 Files

Output Files: 24 Fw Reads. 24 Rv Reads. 2 Index Hopping. 2 LowQual/ nonmatch

Total Output = 52 Files

Unit Tests: suffix test2.fq

1 record dual matched

1 nonmatching index

1 record index hopped

1 N record

1 lowqual record

SudoCode:

Store the filehandles for each index file plus the index hopped files and the error/lowqual files in a dictionary. Iterate through the dict and store opened files in variables via file = open(“p”,’w’). Then to write to file use file.write.

Read the two sequence files and two index files simultaneously with gzip to open zipped files and zip to read them simultaniously.

Select for the lines that contain the index/read sequences in the fastq files. Store 4 lines of each file or 1 total record in memory. Then write a set of conditionals to send records to specific bucket output files based on the status of the two indexes. For each record going in a bucket the sequence of the index pair must be concatenated to the header.

If the indexes are known, and each index is identified as ours and a correct complement to its counterpart then the records from R1 and R4 get bucketed into one of their 48 index files.

If the indexes are both identified as ours but not complementary to each other then the R1 and R2 records go into their respective index hopped file.

If either of the indexes cannot be identified because of error or low quality then the record pertaining to the low quality/ error would be placed in its respective R1 or R4 LowQual/ nonmatch bins. A minimum quality score cutoff metrics must be devised for the indexes, and for the biological reads. Indexes should have more strict quality scores cutoffs since there are fewer. Potentially a cutoff of 28 for the index

At the end iterate through barcode dict and use file.close.

Questions:

High Level Functions: What make them high level?

ReverseComplement:

Paramater: String

Finds the reverse complement of a sequence

Use string slicing to obtain the reverse sequence

To get complement create a dict with each nucleotide as a key and its complement as a val. Then iterate through sequence and say if key in dict append Val to string containing reverse sequence.

Return: reverse complement sequence

Header\_create:

Parameter: header, index 1, index 2

Creates headers for the output files by appending the index pair to the header

QualScore:

Parameter: qual score characters

Reads a sequence of quality scores and determines if it dips below the cutoff