**Assignment 2**

Jialin Kang

e-mail: jkang58@jhu.edu

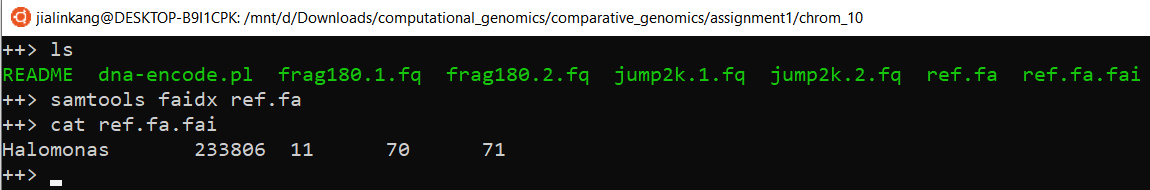
**Question 1. Coverage Analysis [20 pts]**

**Question 1a. How long is the reference genome?**

In linux, use command

‘samtools faidx ref.fa’

‘cat ref.fa.fai’



We can know the length of the reference genome is: 233806 bp

**Question 1b. How many reads are provided and how long are they? Make sure to measure each file separately**

In linux, use command

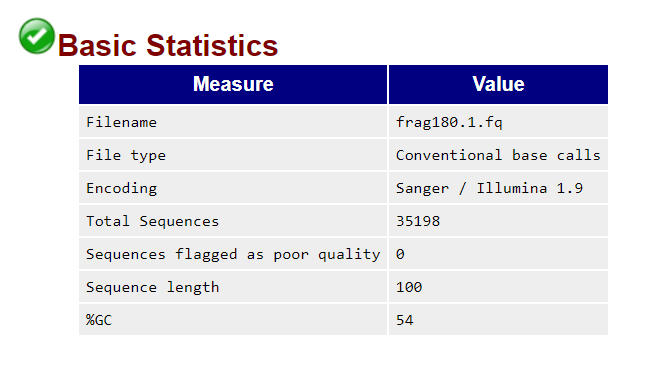
‘fastqc frag180.1.fq’

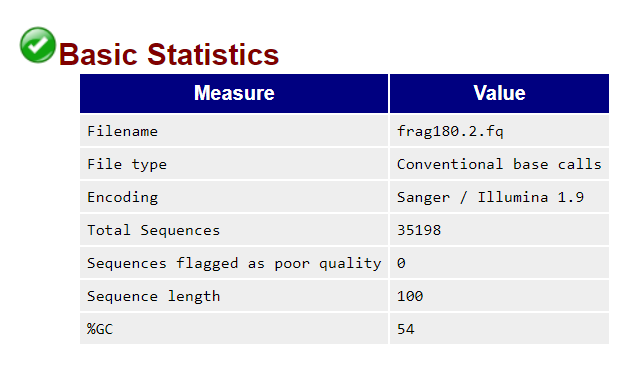
‘fastqc frag180.2.fq’

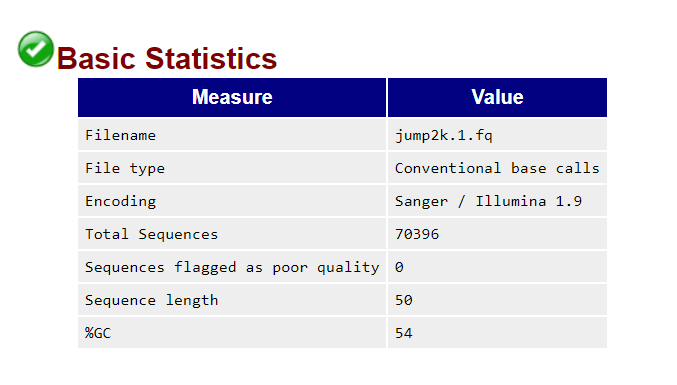
‘fastqc jump2k.1.fq’

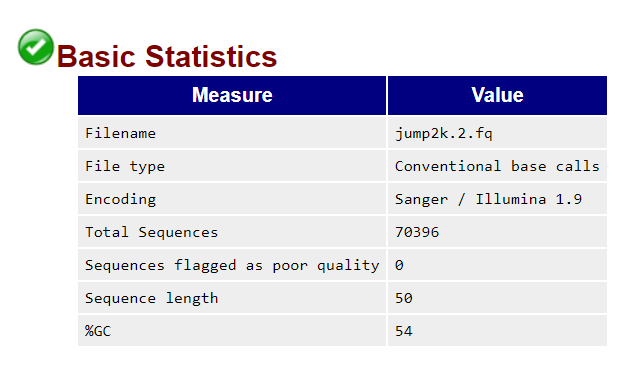
‘fastqc jump2k.2.fq’

We can get a html file and zip file of each command.









From those result of basic statistics, we can get:

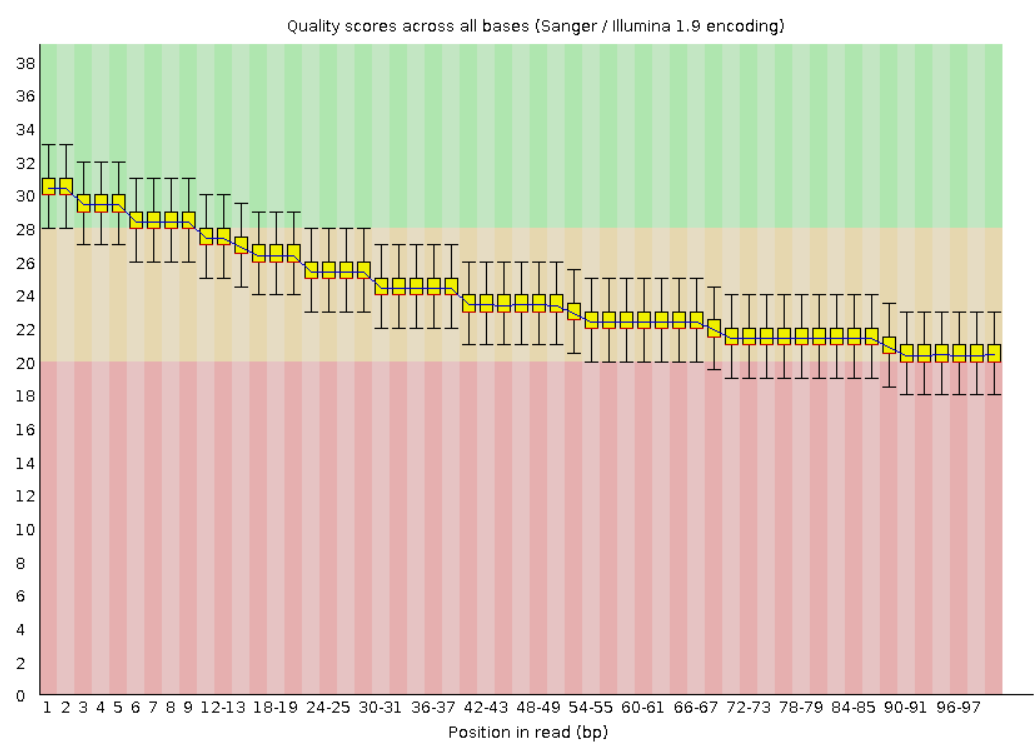
|  |  |  |  |
| --- | --- | --- | --- |
| Name | Reads | Length | Coverage |
| frag180.1.fq | 35198 | 100 | 15.05 |
| frag180.2.fq | 35198 | 100 | 15.05 |
| Jump.2k.1.fq | 70396 | 50 | 15.05 |
| Jump.2k.2.fq | 70396 | 50 | 15.04 |

**Question 1c. How much coverage do you expect to have?**

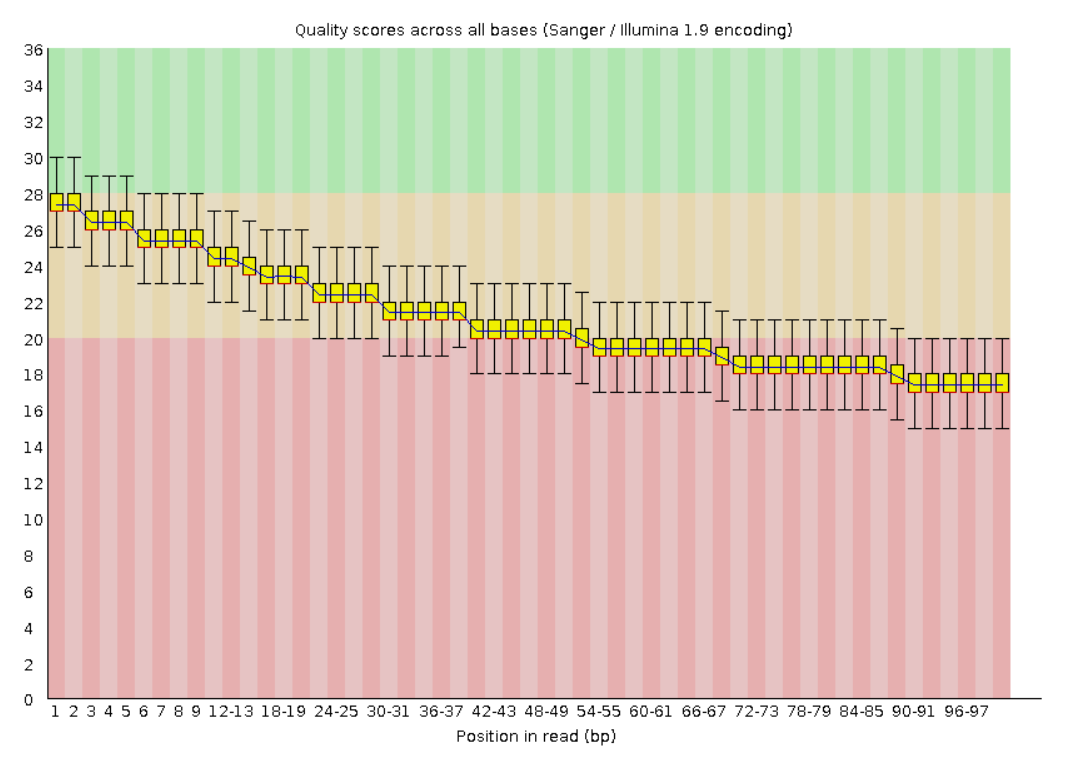
See the above chart. So the excepted coverage is 15.

**Question 1d. Plot the average quality value across the length of the reads.**

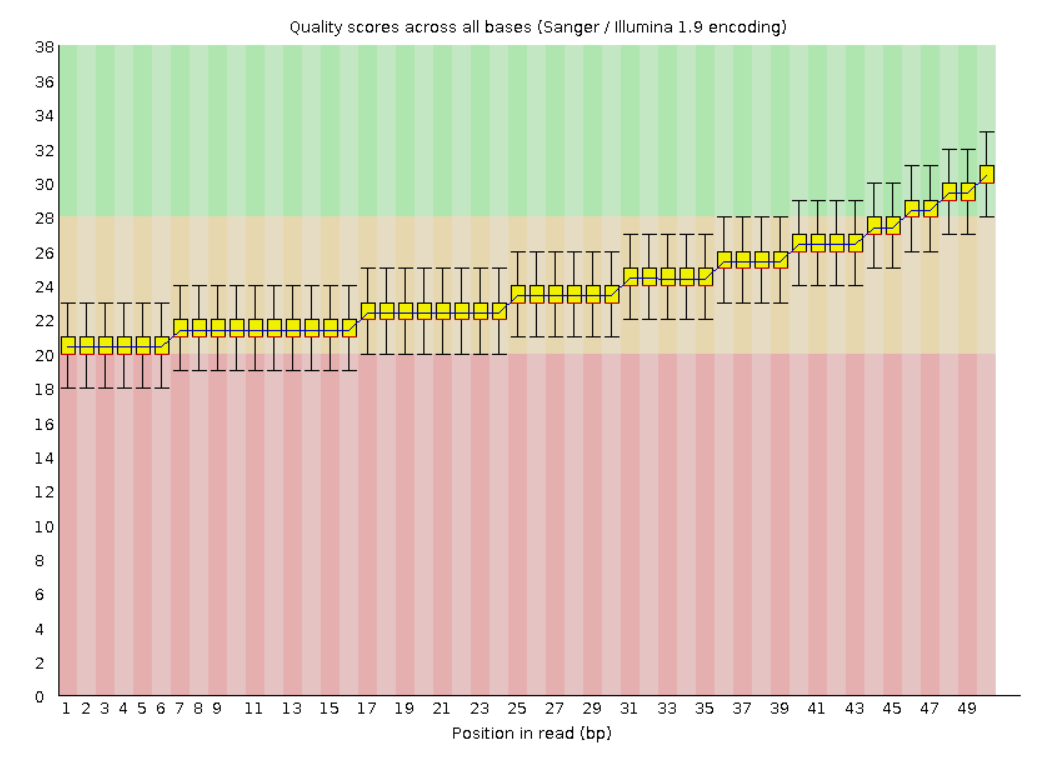
frag180.1.fq



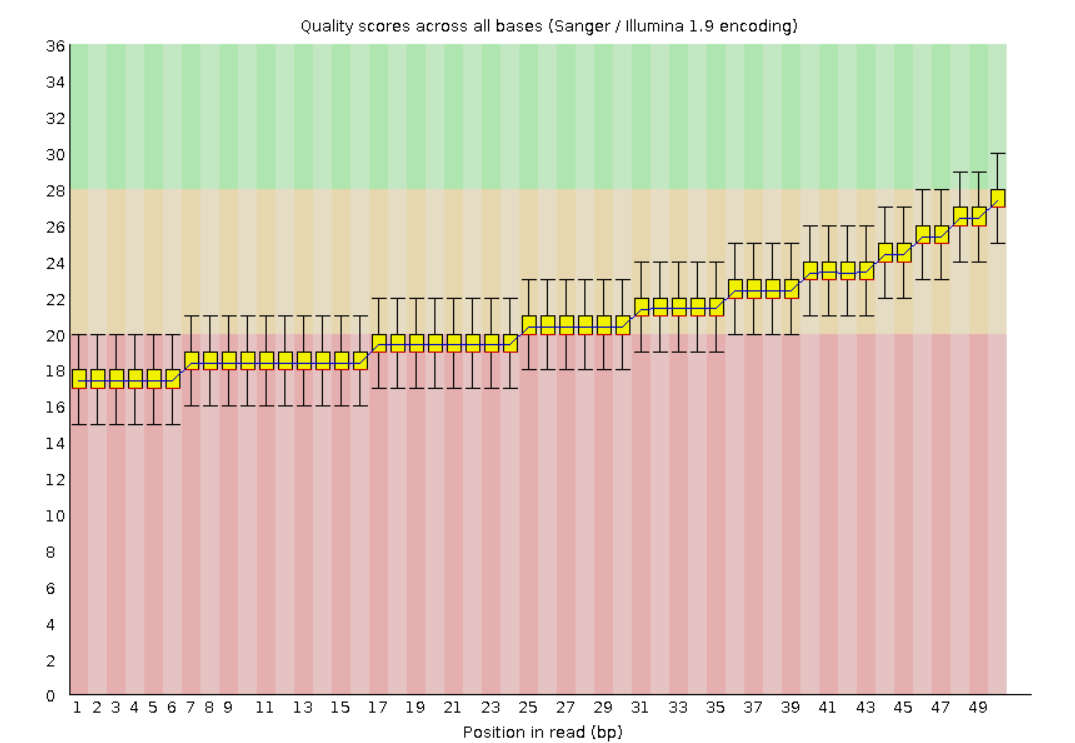
frag180.2.fq



Jump.2k.1.fq



Jump.2k.2.fq



**Question 2. Kmer Analysis**

**Question 2a. How many kmers occur exactly 50 times?**

Use the command:

jellyfish count -m 21 -s 100M -t 10 -C jump2k.1.fq jump2k.2.fq frag180.1.fq frag180.2.fq

jellyfish histo mer\_counts.jf

from the result we can know:

1062 kmers occur exactly 50 times

**Question 2b. What are the top 10 most frequently occurring kmers?**

Use command:

jellyfish dump -c -t mer\_counts.jf > kmer\_count.fasta

sort -rn -k2 kmer\_count.fasta > sort.txt

head -n 10 sort.txt

CGCCCACTAATTAGTGGGCGC 94

CCCACTAATTAGTGGGCGCCG 94

GCAGGAATTGAACCTGCGACC 93

GCCCACTAATTAGTGGGCGCC 92

ACGGCGCCCACTAATTAGTGG 92

GGCAGGAATTGAACCTGCGAC 88

GCGCGCCCGGCAGGAATTGAA 87

CGCGCCCGGCAGGAATTGAAC 87

GCGCCCGGCAGGAATTGAACC 86

AGGTCGCAGGTTCAATTCCTG 86

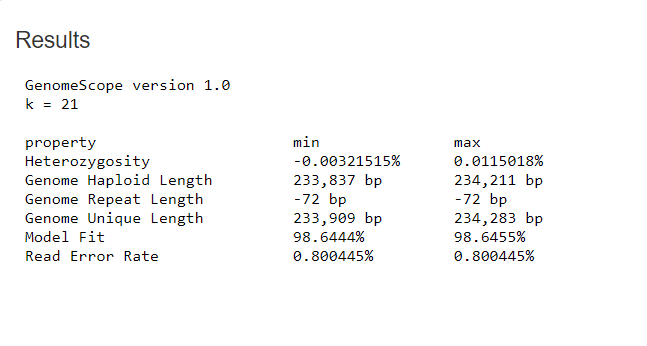
**Question 2c. What is the estimated genome size based on the kmer frequencies?**

Use command:

jellyfish histo -t 10 mer\_counts.jf > reads.histo

put the reads.histo file into genomescope website

the genomescope result link: <http://genomescope.org/analysis.php?code=YDrXUbdH3pNuOZz2nw4m>



From the results table, we can know that the min Genome haploid length is: 233837 bp

the length of the reference genome is: 233806 bp and the estimation is : 233837~234283, the estimation is very near to the really result. It is a good estimation to the reference genome.

**Question 3. De novo assembly**

**Question 3a. How many contigs were produced?**

Sudo apt install spades

spades --pe1-1 frag180.1.fq --pe1-2 frag180.2.fq --mp1-1 jump2k.1.fq --mp1-2 jump2k.2.fq -o asm -t 4 -k 31

grep -c '>' contigs.fasta

we the number of contigs is 4

**Question 3b. What is the total length of the contigs?**

samtools faidx contigs.fasta

datamash sum 2 < contigs.fasta.fai

we can get the total length of contigs is 234596

**Question 3c. What is the size of your large contig?**

sort -rn -k2 contigs.fasta.fai

from the sort result we can know the large contig size is 105834

**Question 3d. What is the contig N50 size?**

import sys

def fafile2dict():

    '''

    read a single FASTA file (SHH.fa) into a dictionary object

    and calculate the contig N50 size of this FASTA file

    run as :

    python3 N50.py < ./asm/contigs.fasta

    Rerurn

    --------------

    N50:int

    the N50 number of this FASTA file

    --------------

    '''

    # read the file context into a dict

    line = sys.stdin.readline().replace('\n','')

    seq = {}

    while line != '':

        if line[0] == '>':

            name = line.replace('>','')

            seq[name] = ''

        else:

            seq[name] += line.replace('\n','').strip()

        line = sys.stdin.readline()

    val\_list = []

    for val in seq.values():

        val\_list.append(len(val))

    val\_list.sort(reverse=True)

    val\_half = sum(val\_list)/2

    for i in range(len(val\_list)):

        if val\_half > 0:

            val\_half -= val\_list[i]

        else:

            N\_50 = val\_list[i-1]

            break

    return N\_50

if \_\_name\_\_ == "\_\_main\_\_":

    N\_50 = fafile2dict()

    print('N50 is ', N\_50)

the N50 of configs.fasts is: 47851

**Question 4. Whole Genome Alignment**

**Question 4a. What is the average identify of your assembly compared to the reference?**

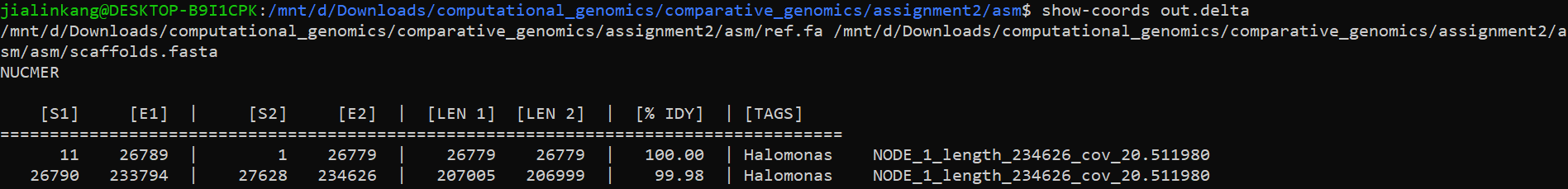
Run the command:

dnadiff ./ref.fa ./asm/scaffolds.fasta

nucmer ./ref.fa ./asm/scaffolds.fasta

show-coords out.delta

the result is as followings:



And from the out.report file we can know:

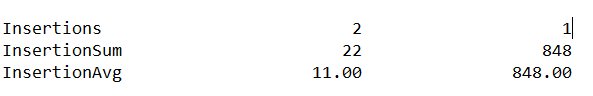
the average identify is 99.98(1-to-1) and 99.98(M-to-M)

**Question 4b. What is the length of the longest alignment**

The longest alignment is 207005.

**Question 4c. How many insertions and deletions are in the assembly?**

There is 1 insertion in the assembly.



The insertion length is 848 bp.

No deletions.

**Question 5. Decoding the insertion**

**Question 5a. What is the position of the insertion on the reference?**

The position is scaffolds.fasta NODE\_1\_length\_234626\_cov\_20.511980 26780-27627

Question 5b. How long is the novel insertion?

848bp

**Question 5c. What is the DNA sequence of the encoded message?**

The DNA sequence is:

>NODE\_1\_length\_234626\_cov\_20.511980:26780-27627

CTAACATTCGTCGGTGATGCTTTCATTCCTTGCTGTCCTAAGTCCACTCTGTATCAATGG

CTAGCGTATGCAAGTACAATAGGTCGACCGGCGCAGCGTCGTGTAGGCTTGCCTGTCAGG

ACTAACACAGTTATCACTTATGGTAATCCACCAGGTCGAACGGCGCAACTTCAGCGACTC

CCCACTATCCGGATGGCAACATTTCCGACGGCTAATAGGCTGTAAGGCATTTAATCCCCC

AAGTCATAAAGTAAACCAGGACTCACTTCCCCACGCACAACTACTATCATCCGCCCAGAT

ATAGACGAACAACGCCACCGCGTTCAACCTGTACACCTTCTGAACGTAGCCGAGGCAGAT

ATGACTACCCGCAACACGACCGTTATTCCTAGCTTATGTAATGCTTGGCGGCTGAGCGGA

GCCGCGTCCATTCGTGCAGTAAGACCAACGGACAGGATTAGTTTATGTCGAGAGGGCCGC

CTTGAATGCGTCCGATCCTCGGTACCGCTTTCAATATTGCAAGAAACCAATCAAACCTAC

GCTGCGGCCGCGCGGAATATCTGGCCCCAATCCACCAGGCGTGGAGTCGTGAAAGAAACA

CTTATTAAATGCTTGGATGCGGGGAGGCTATTATGCTCAGATTATTTAAGGAAGTTCCGA

CAAAAGGACTTAAGTCTGGTATGTCCTACAGCCCAACGGCGGCAATTCTAATCTGTATCC

CTGGCTCACGTCTGACAGTAATCTAAGCCAAACTTCCTTTCTGGACGAATGAAGAGCGGC

ATAGCGTATTTCCATCACCCCGTCCCAGCGTATTAAAGTAGCATCGTAATCTAGGATTGC

ATGTAAGG

**Question 5d. What is the secret message?**

samtools faidx ./asm/scaffolds.fasta NODE\_1\_length\_234626\_cov\_20.511980:26780-27627 > seq.fa

./dna-encode.pl -d seq.fafol

The secret message is:

Congratulations to the Spring 2020 JHU Applied Genomics course... Keep on looking for little green aliens