UNIVERSITY OF ST ANDREWS

A COURSE WORK ON GENETIC ANALYSIS

MODULE: GD5302

STUDENT ID: 200015689

DIGITAL HEALTH (MSc.)

Introduction

The aim of this report is to identify an unknown sample present in a patient genome sequence using DNA sequencing technique. The dataset used in the practical belongs to four patients (A, B, C and D). The raw patient data were obtained from the University of St-Andrews Marvin server as supplied by Dr. Peter Thorpe. These data were analysed in order to help identify what organism is contained in each patient sequence data and assist the clinicians in disease diagnosis and decision making. To analyse the files, several processes as discussed in each question section. These processes were performed to eliminate errors that may eventually falsify the genome analysis. This was performed by using genome assembly software discussed in the question below. Shell script was used to access the software packages in a non-interaction mode, and also Python was written to perform some calculation in the genome sequence obtained.

Question 1: Quality Control the Fastq data:

The fastq output data realised for each patient from the DNA sequencing was processed to identify how low quality bases which could affect the genome sequence analysis. Data quality control, a preprocessing technique, is often crucial in processing next-generation data [1]**.** This is to identify poor quality sequences in DNA sequencing that can easily result in suboptimal downstream analysis [2]**.** As such, there was a need to quality control the reads to avoid errors that could significantly affect the genome sequence analysis. In order to quality control the raw data, a software package known as FastQC [3] was used. This provides a modular set of analysis that can be used to check if raw sequenced data has significant errors due to poor quality reads. The FastQC was run on each raw fastq data (both forward and reversed data) of all patients (patient A to D), to access the quality scores of each read in the fastq data. The result of this process is shown in (fig 1a, 1b) for both forward and reversed fastq data for patient A. A similar output was also generated for patient B, C and D.

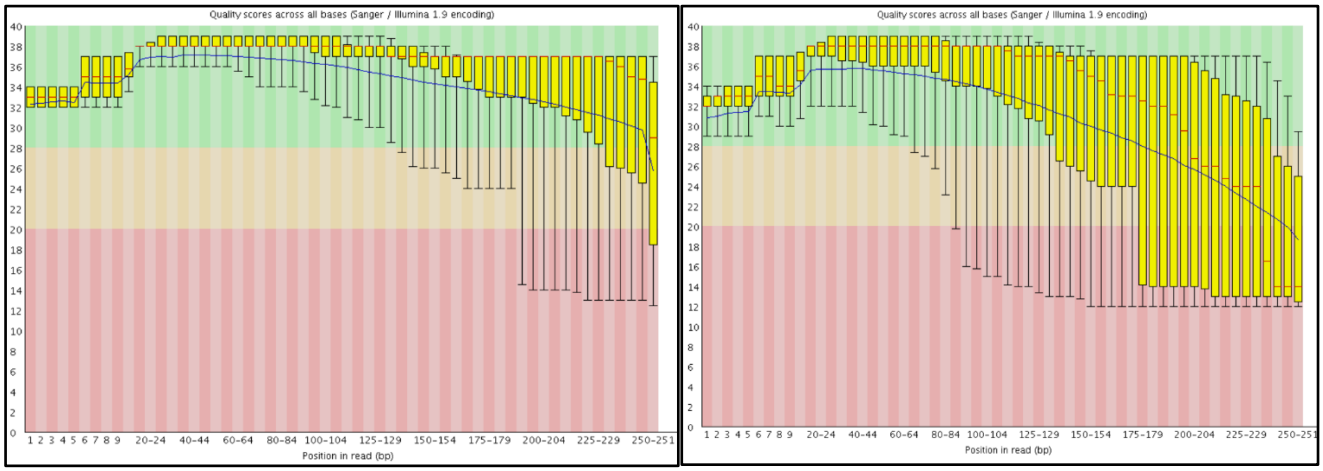


Fig 1a, 1b: A figure showing the quality score across the reads in Patient fastq data. The vertical axis represents the quality scores.

The FastQC results, as shown above, shows different quality scores for each reads in the patient data (in fastq format). The background of the results (graph) divides the quality scores into layers of good quality scores (green), reasonable quality (orange) and poor quality scores (red). The area marked red represents region of low-quality scores that can affect the genome analysis. As such, there was a need to trim them

off. To do this, a software known as Trimmomatic [4, 5] was used. For an effective trim, base pairs with quality scores below 20 were trimmed off. The value "20" was selected as it represents the region of low quality score as seen in the result above. Also, several studies have used this value for an effective trimming [6]**.** Upon successful trim, the FastQC was run to assess how well the trimming performed. The result is as shown in fig 1c and 1d.

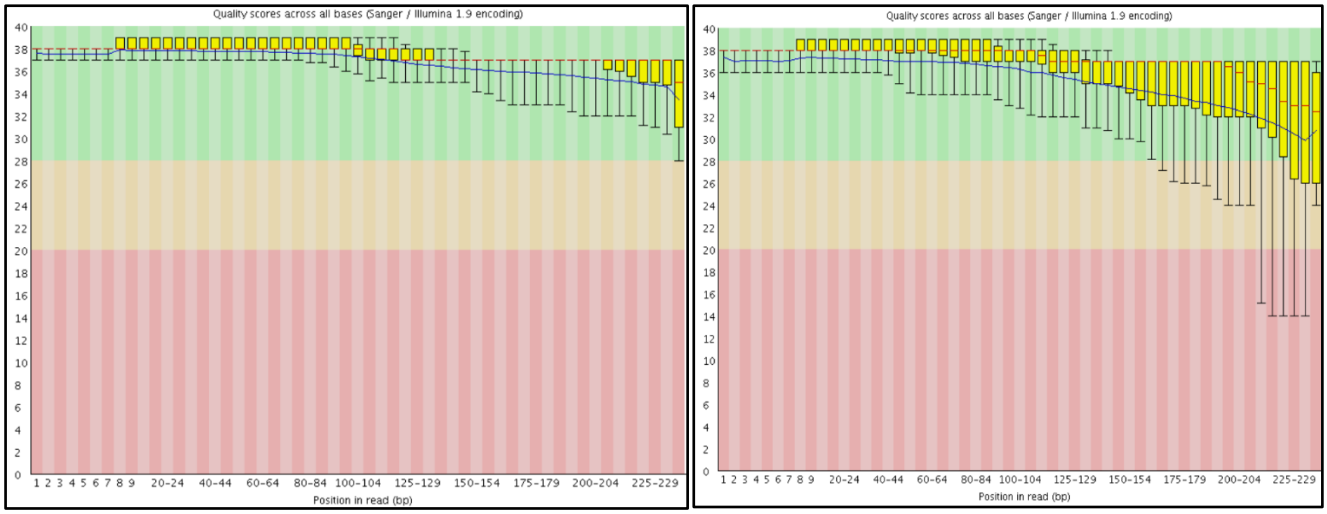
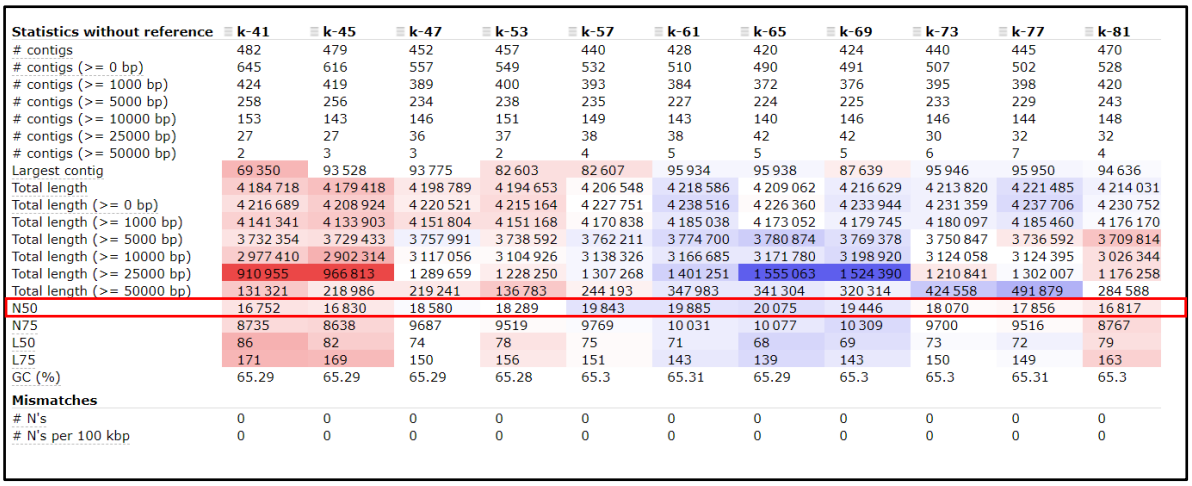


Fig 1c, 1d: A figure showing the quality score of patient A data (forward and reversed) after trimming.

As seen in the figure above, low-quality scores were removed after using the trimmomatic software package. The same process was also performed for other patients (B, C and D)

Assemble the Fastq into a genome Assembly(Question 2)

The trimmed fastq data realised above further were assembled into a longer contiguous genomic sequence. This is important to create a representation of gene sequence that matches the DNA of where the gene originated from . The fastq trimmed data contains shorter reads which are much shorter than most genomes or even most genes. To assemble the trimmed data into longer contiguous genome sequence, software packages such as Velvet, SPAdes and Unicycler was used. The essence of this is to determine the assembler which produces the best genome sequence in terms of contiguity. In order to justify the best assembler, the N50 metric was used. N50 is a measure to describe the quality of assembly of a genome sequence that are fragmented in contigs of different length. This has also been used in several studies to determine the quality of a genome assembly **.** Using Velvet [7], the trimmed fastq data (forward and reversed pairs) for each patient were assembled using different K-mer values **(**41 to 81**).** This was done to determine an optical K mer value which reproduce best result. The best result was determined using a quality assessment tool (Quast) [8], which shows a comparism of the N50 values for each K-mer. The result is as shown in fig 2a for patient A. The same approach was also adopted for patient B, C and D.



Statistics without reference	k-41	k-45	k-47	k-53	k-57	k-61	k-65	k-69	k-73	k-77	k-81
# contigs	482	479	452	457	440	428	420	424	440	445	470
# contigs (>= 0 bp)	645	616	557	549	532	510	490	491	507	502	528
# contigs (>= 1000 bp)	424	419	389	400	393	384	372	376	395	398	420
# contigs (>= 5000 bp)	258	256	234	238	235	227	224	225	233	229	243
# contigs (>= 10000 bp)	153	143	146	151	149	143	140	146	146	144	148
# contigs (>= 25000 bp)	27	27	36	37	38	38	42	42	30	32	32
# contigs (>= 50000 bp)	2	3	3	2	4	5	5	5	6	7	4
Largest contig	69 350	93 528	93 775	82 603	82 607	95 934	95 938	87 639	95 946	95 950	94 636
Total length	4 184 718	4 179 418	4 198 789	4 194 653	4 206 548	4 218 586	4 209 062	4 216 629	4 213 820	4 221 485	4 214 031
Total length (>= 0 bp)	4 216 689	4 208 924	4 220 521	4 215 164	4 227 751	4 238 516	4 226 360	4 233 944	4 231 359	4 237 706	4 230 752
Total length (>= 1000 bp)	4 141 341	4 133 903	4 151 804	4 151 168	4 170 838	4 185 038	4 173 052	4 179 745	4 180 097	4 185 460	4 176 170
Total length (>= 5000 bp)	3 732 354	3 729 433	3 757 991	3 738 592	3 762 211	3 774 700	3 780 874	3 769 378	3 750 847	3 736 592	3 709 814
Total length (>= 10000 bp)	2 977 410	2 902 314	3 117 056	3 104 926	3 138 326	3 166 685	3 171 780	3 198 920	3 124 058	3 124 395	3 026 344
Total length (>= 25000 bp)	910 955	966 813	1 289 659	1 228 250	1 307 268	1 401 251	1 555 063	1 524 390	1 210 841	1 302 007	1 176 258
Total length (>= 50000 bp)	131 321	218 986	219 241	136 783	244 193	347 983	341 304	320 314	424 558	491 879	284 588
N50	16 752	16 830	18 580	18 289	19 843	19 885	20 075	19 446	18 070	17 856	16 817
N75	8735	8638	9687	9519	9769	10 031	10 077	10 309	9700	9516	8767
L50	86	82	74	78	75	71	68	69	73	72	79
L75	171	169	150	156	151	143	139	143	150	149	163
GC (%)	65.29	65.29	65.29	65.28	65.3	65.31	65.29	65.3	65.3	65.31	65.3
Mismatches											
# N's	0	0	0	0	0	0	0	0	0	0	0
# N's per 100 kbp	0	0	0	0	0	0	0	0	0	0	0

Fig 2a: Quality Assessment check (in terms of contigs) to determine an optimal K-mer value for patient A.

As shown in the figure above, the optimal K-mer value for patient A is 65, which an N50 value of 20075. Same process was performed for patient B, C and D, and an optimal K-mer value of 45, 77 and 81 respectively, were obtained.

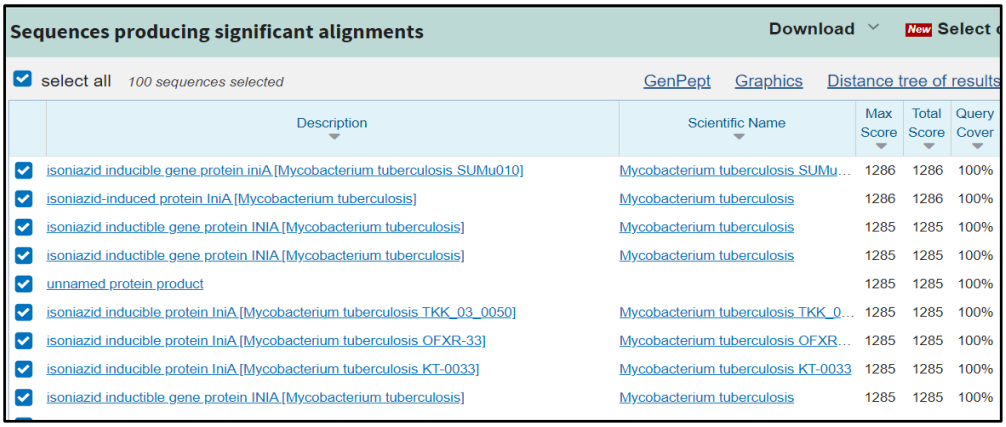
Also, the trimmed data for each patient was assembled using SPAdes [9]**.** The SPAdes assembler perform assemblies by using a different list of K-mer values automatically inferred by the software. The range of K-mer values are compared, and the best result is return as Scaffolds.fasta. Finally, the trimmed fastq data was also assembled using Unicycler [10]. Unicycler function mainly as a spade optimizer, and selects the best assembled genome sequence (assembly.fasta) through a wider range of K-mer value. In conclusion, to get the best assembled genome (in terms of contigs), the best result for Velvet (contigs.fa), SPAdes (scaffolds.fasta), and Unicycler (assembly.fasta) were compared using Quast as shown in fig 2b, for patient A. From the result, it can be deduced that SPAdes had the largest N50 value, and thus, concluded to be the best assembler. The same process was carried out for patients B, C and D, and SPAdes was the best for all.

Statistics without reference	Velvet	Spades	Unicycler
# contigs	420	188	175
# contigs (>= 0 bp)	490	323	188
# contigs (>= 1000 bp)	372	149	169
# contigs (>= 5000 bp)	224	110	129
# contigs (>= 10000 bp)	140	92	109
# contigs (>= 25000 bp)	42	59	60
# contigs (>= 50000 bp)	5	29	27
Largest contig	95 938	173 491	135 702
Total length	4 209 062	4 292 219	4 246 532
Total length (>= 0 bp)	4 226 360	4 328 775	4 249 308
Total length (>= 1000 bp)	4 173 052	4 264 477	4 241 443
Total length (>= 5000 bp)	3 780 874	4 173 763	4 140 061
Total length (>= 10000 bp)	3 171 780	4 042 498	3 995 175
Total length (>= 25000 bp)	1 555 063	3 497 599	3 192 076
Total length (>= 50000 bp)	341 304	2 462 894	2 051 288
N50	20 075	57 251	46 914
N75	10 077	30 881	25 507
L50	68	24	29
L75	139	49	60
GC (%)	65.29	65.41	65.36
Mismatches			
# N's	0	1237	0
# N's per 100 kbp	0	28.82	0

Fig 2b: Quality Assessment check (in terms of contigs) to determine the best assembly software package.

Predict the genes from the Genome Assembly (Question 3)

Upon selecting the best-assembled genome sequence as discussed above, the assembled genome was annotated to identify and label the relevant features (such as protein coding genes) on the genome sequence [11]**.** To carry out this process, a software package called PROKKA [12] was used. The result from PROKKA produces 10 files, all with a comma prefix. However, only four (4) files were of relevance in this report. They include .fna (fasta file of original nucleotide), faa (fasta file of translated coding genes), .fnn (fasta file of all genome features), and . gbk/gbf (Genbank file containing sequence and annotation of protein genes). The .faa file was blasted using a basic local alignment search tool, Blastp [13], to have an overview of the labelled encoded protein in the gene, and organism with similar protein gene. As shown in figure 3a, the result of blastp for patient tells that the protein code in the patient gene correlates with that of mycobacterium. The same approach was also taken for patient B,C and D. Also, Figure 3b shows a summary of the total number of features in the assembled data and the labelled protein (rRNA, tmRNA, tRNA).

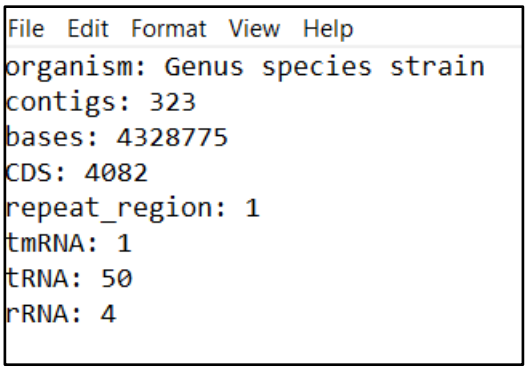


Sequences producing significant alignments Download New Select

select all *100 sequences selected* GenPept Graphics Distance tree of results

Description	Scientific Name	Max Score	Total Score	Query Cover
isoniazid inducible gene protein iniA [Mycobacterium tuberculosis SUMu010]	Mycobacterium tuberculosis SUMu…	1286	1286	100%
isoniazid-induced protein IniA [Mycobacterium tuberculosis]	Mycobacterium tuberculosis	1286	1286	100%
isoniazid inductible gene protein INIA [Mycobacterium tuberculosis]	Mycobacterium tuberculosis	1285	1285	100%
isoniazid inductible gene protein INIA [Mycobacterium tuberculosis]	Mycobacterium tuberculosis	1285	1285	100%
unnamed protein product		1285	1285	100%
isoniazid inducible protein IniA [Mycobacterium tuberculosis TKK_03_0050]	Mycobacterium tuberculosis TKK_0…	1285	1285	100%
isoniazid inducible protein IniA [Mycobacterium tuberculosis OFXR-33]	Mycobacterium tuberculosis OFXR…	1285	1285	100%
isoniazid inducible protein IniA [Mycobacterium tuberculosis KT-0033]	Mycobacterium tuberculosis KT-0033	1285	1285	100%
isoniazid inductible gene protein INIA [Mycobacterium tuberculosis]	Mycobacterium tuberculosis	1285	1285	100%

Fig 3a: A figure showing an overview of the labelled encoding protein in patient A assembled gene.



File Edit Format View Help

```
organism: Genus species strain
contigs: 323
bases: 4328775
CDS: 4082
repeat_region: 1
tmRNA: 1
tRNA: 50
rRNA: 4
```

Figure 3b: A Figure showing the overall summary of annotated features in the assembled genome of patient A.

Calculate the GC content of the assembled genome (Question 4)

The GC content of the best assembled genome sequence was computed using a python library Biopython [14]. A function called SeqIO in the biopython library was used to parse the fasta file of the assembled genome sequence in order to compute the GC% content in each patient. The GC content summarizes the percentage of Nitrogenous base of Guanine (G) and Cytosine (C) Nucleotides present in DNA. The result of the GC% content for each patient (A, B, C, and D) is as shown in table 1.0. From the result, it can be seen that patient B and D have closely related GC% content, whereas, for patient A and C, the GC% content showed a wide range difference. Following the result, it is assumed that patient B and D should have similar disease or disease, the reason being that they have close range of GC% (3%-5% difference). This assumption conforms with a study conducted by [15] which reveals that taxonomically related organisms shows a close match or range of GC% (3% - 5%). However, according to a study conducted by [16] suggested that organism connected with each other GC content ranges from 55%-70% as GC content and this could be due to different variation. This article also argued that a high amount GC content could potentially cause bias which can potentially bring about certain difficult to sequence the DNA appropriately via for instance Illumina. Therefore further research is required in this area to ensure optimal and a universal GC%.

Table 1.0: A table showing the GC% Content in the assembled genome sequence of each patient.

:

	GC%
patient_A	65.34
patient_B	57.19
patient_C	32.91
patient_D	52.32

Parsing Genome Assembly using the Fasta File (Question 5)

This next step requires computing the percentage of Adenosine, Thymine, Cytosine and Guanine (ATCG) percentage of the assembled sequence. Further statistics such as ATCG percentage present in each patient Fasta file was computed. This was to check the ATCG were constituents present in the .fasta file. To do this, the best assembly genome was parsed using Seql0 from the biopython; in order to compute the ATCG percentage. However, in computing the ATCG, for all patients, it was observed that some of the patient DNA sequences contained a very small percentage of an unknown base pair. The result of this is as shown in Table 2.0.

Table 2.0: A table showing the ATCG% in the assembled genome sequence of each patient. Also, it shows the percentage of unknown base pairs (N) found in each patient assembled genome sequence.

	A%	T%	C%	G%	N%
patient_A	17.30	17.33	32.60	32.74	0.03
patient_B	21.35	21.46	28.46	28.73	0.00
patient_C	34.05	33.04	16.30	16.61	0.00
patient_D	23.34	24.34	25.23	27.09	0.00

As show in the table, patient A contained 0.03% of unknown base pairs, whereas patient B to D had 0%. A plot was further plotted to have a better representation of the ATCG%. The result is as shown in fig (5a). from the result, it can be observed that patient B and D had very close percentage of G and C necleotides respectively.

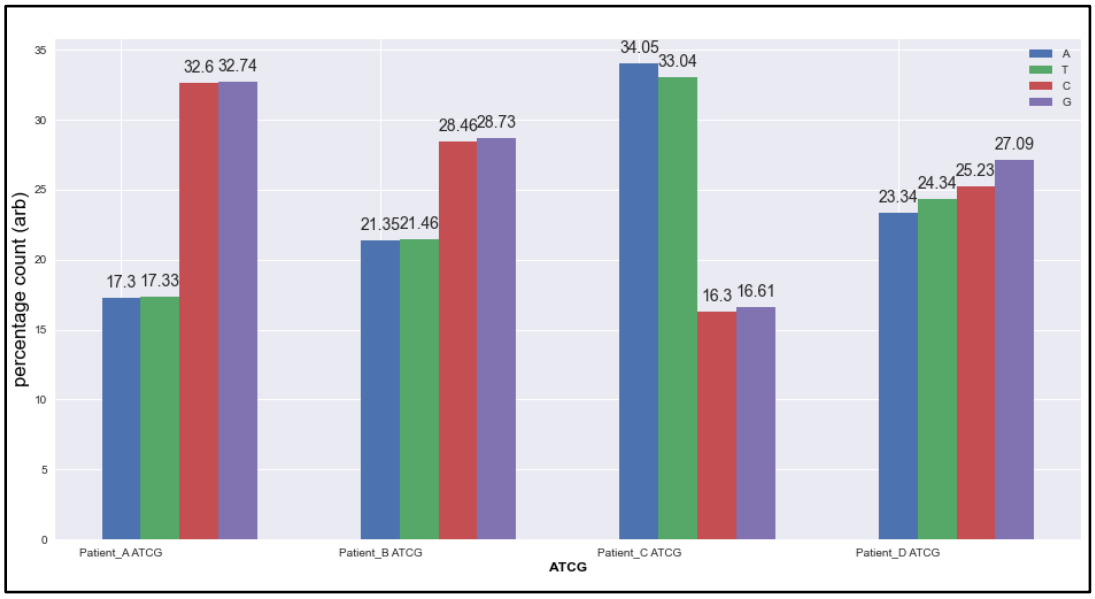


Fig 5a: A plot showing a percentage of ATCG content across assembled sequence of each patient, where patient A is marked with blue, patient B is marked with green, patient C is marked with red and patient D is marked with purple.

Calculating the length of fragments of genome assemblies using different assemblers (Question 6)

In this step, computing the length of all fragments in each fasta file is performed. To was conducted to help justify which assembler package was the best for each of the patient (A, B, C and D). A Python program was created to compute the length of each fragment produced by the various assemblers. The best assembler was selected by assessing the assembler, which generated the longest length of the sequence (fragments length). As shown in fig(6a), the best assembler was selected to be SPAdes for patient A, as it was found to havea a maxium fragment length than Velvet, and unicycler. Likewise for patient B,C and D, as seen in fig(6b,6c and 6d). SPAdes was found to produce a genome sequence with the maximum fragment length above Velvet and Unicycler. This can be compared with the result produced by Quast software tool as shown in fig(**2b)** for patient A.

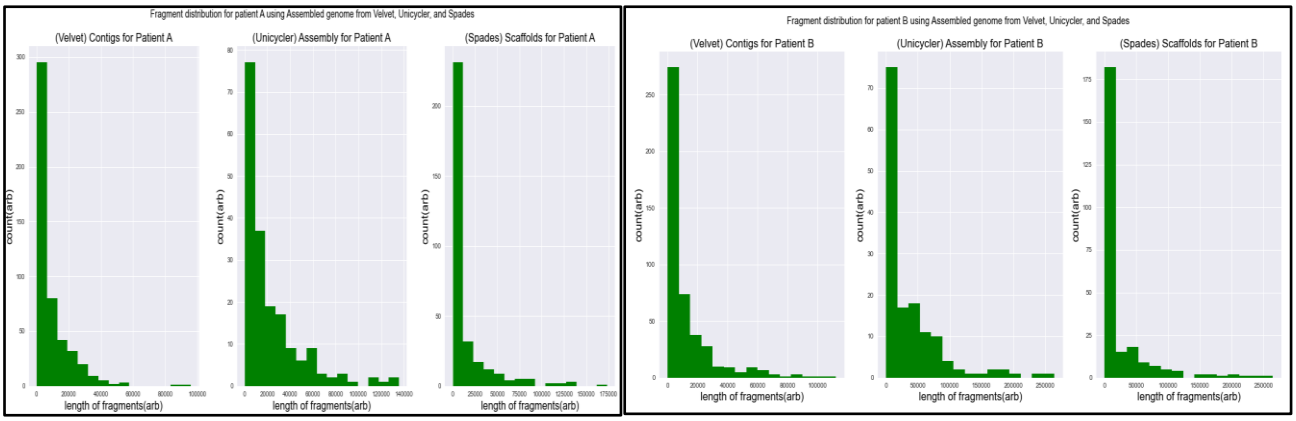


Fig 6a, 6b: A figure showing a histogram of length of fragment of assembled genome sequence for patientA and B, to justify which assembler is thee best.

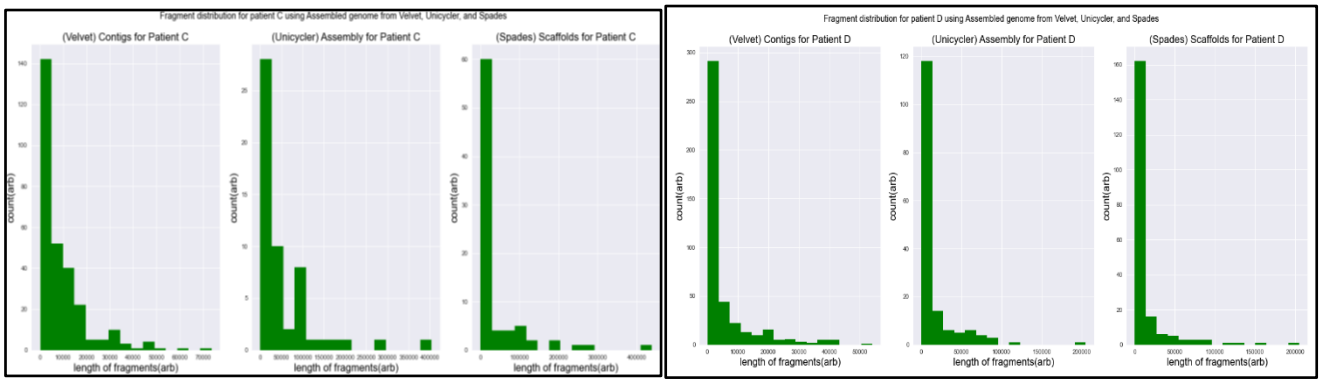
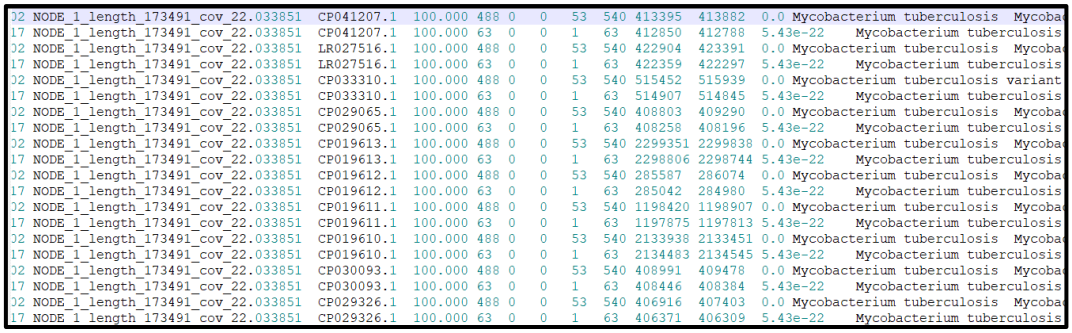


Fig 6c, 6d: A figure showing a histogram of length of fragment of assembled genome sequence for patientC and D, to justify which assembler is thee best.

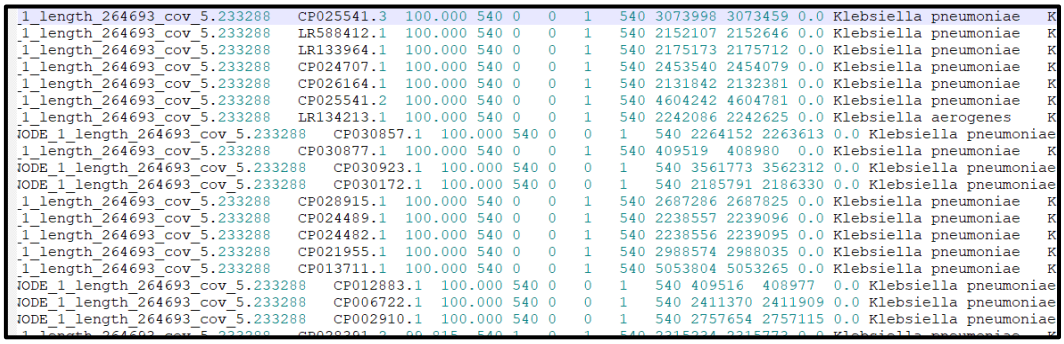
Sequence similarity search to identify the organisms (Question 7)

Finally, the best-assembled genome sequence was analysed using a similarity sequence search in order to identify the organism present in the assembled genome sequence. To do this, a software package known as blastn [13] was used. The software has the capability to identify organism present within assembled genome sequence by comparing it's a sequence to a public database of genome sequences. On complete comparison, a similarity match for the identified organism(s) is returned, which shows how related the sequence found on the assembled genome sequence matches that of the identified organism. Based on the similarity check result from blastn as shown in fig(7a, 7b,7c and 7d) The result for fig(7a) indicates that patient A genome sequence have a similarity match of 100.0 with that of Myobacterium. Thus, it can be concluded that patient A has Myobacterium pathogen. Also, for patient B seen in fig (7b), it can be seen that the genome sequence of patient B shows a similarity check of 99.815 to 100 with Klebsiella Pneumonia. Thus, it can be concluded that patient B has the klebsiella Pneumonia pathogen. Furthermore, for patient C; (see Fig 7c), it can be seen its genome sequence has a similarity of 81 – 96.82% with Staphylococcus aureus. Thus, it can be deduced that patient C has Staphylococcus aureus. Finally, for patient D (see fig 7d), the similarity check result showed a similarity match of 99-100 with Neisseria gonorrhoea. Thus, it can be concluded that patient D has Neisseria gonorrhoea. A summary of this can be found in Table 3.0.



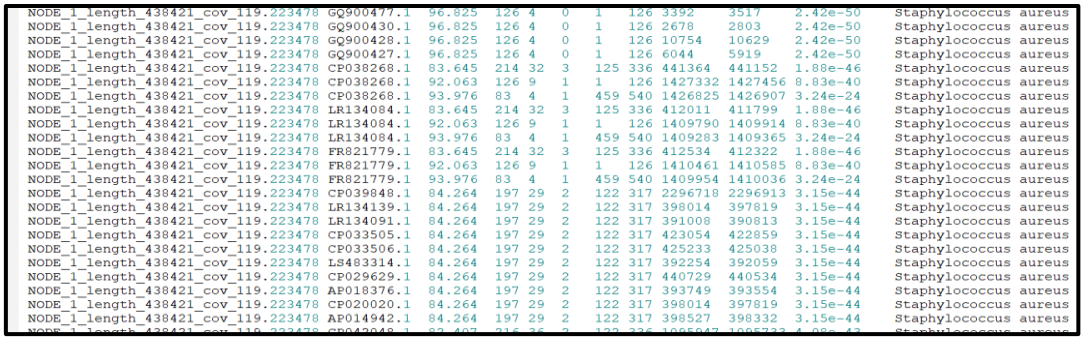
92 NODE_1_length_173491_cov_22.033851	CP041207.1	100.000	488	0	0	53	540	413395	413882	0.0	Mycobacterium tuberculosis	Mycoba
17 NODE_1_length_173491_cov_22.033851	CP041207.1	100.000	63	0	0	1	63	412850	412788	5.43e-22	Mycobacterium tuberculosis	
92 NODE_1_length_173491_cov_22.033851	LR027516.1	100.000	488	0	0	53	540	422904	423391	0.0	Mycobacterium tuberculosis	Mycoba
17 NODE_1_length_173491_cov_22.033851	LR027516.1	100.000	63	0	0	1	63	422359	422297	5.43e-22	Mycobacterium tuberculosis	
92 NODE_1_length_173491_cov_22.033851	CP033310.1	100.000	488	0	0	53	540	515452	515939	0.0	Mycobacterium tuberculosis variant	
17 NODE_1_length_173491_cov_22.033851	CP033310.1	100.000	63	0	0	1	63	514907	514845	5.43e-22	Mycobacterium tuberculosis	
92 NODE_1_length_173491_cov_22.033851	CP029065.1	100.000	488	0	0	53	540	408803	409290	0.0	Mycobacterium tuberculosis	Mycoba
17 NODE_1_length_173491_cov_22.033851	CP029065.1	100.000	63	0	0	1	63	408258	408196	5.43e-22	Mycobacterium tuberculosis	
92 NODE_1_length_173491_cov_22.033851	CP019613.1	100.000	488	0	0	53	540	2299351	2299838	0.0	Mycobacterium tuberculosis	Mycoba
17 NODE_1_length_173491_cov_22.033851	CP019613.1	100.000	63	0	0	1	63	2298806	2298744	5.43e-22	Mycobacterium tuberculosis	
92 NODE_1_length_173491_cov_22.033851	CP019612.1	100.000	488	0	0	53	540	285587	286074	0.0	Mycobacterium tuberculosis	Mycoba
17 NODE_1_length_173491_cov_22.033851	CP019612.1	100.000	63	0	0	1	63	285042	284980	5.43e-22	Mycobacterium tuberculosis	
92 NODE_1_length_173491_cov_22.033851	CP019611.1	100.000	488	0	0	53	540	1198420	1198907	0.0	Mycobacterium tuberculosis	Mycoba
17 NODE_1_length_173491_cov_22.033851	CP019611.1	100.000	63	0	0	1	63	1197875	1197813	5.43e-22	Mycobacterium tuberculosis	
92 NODE_1_length_173491_cov_22.033851	CP019610.1	100.000	488	0	0	53	540	2133938	2133451	0.0	Mycobacterium tuberculosis	Mycoba
17 NODE_1_length_173491_cov_22.033851	CP019610.1	100.000	63	0	0	1	63	2134483	2134545	5.43e-22	Mycobacterium tuberculosis	
92 NODE_1_length_173491_cov_22.033851	CP030093.1	100.000	488	0	0	53	540	408991	409478	0.0	Mycobacterium tuberculosis	Mycoba
17 NODE_1_length_173491_cov_22.033851	CP030093.1	100.000	63	0	0	1	63	408446	408384	5.43e-22	Mycobacterium tuberculosis	
92 NODE_1_length_173491_cov_22.033851	CP029326.1	100.000	488	0	0	53	540	406916	407403	0.0	Mycobacterium tuberculosis	Mycoba
17 NODE_1_length_173491_cov_22.033851	CP029326.1	100.000	63	0	0	1	63	406371	406309	5.43e-22	Mycobacterium tuberculosis	

Fig (7a): A representation of similarity match for patient A assembled genome.



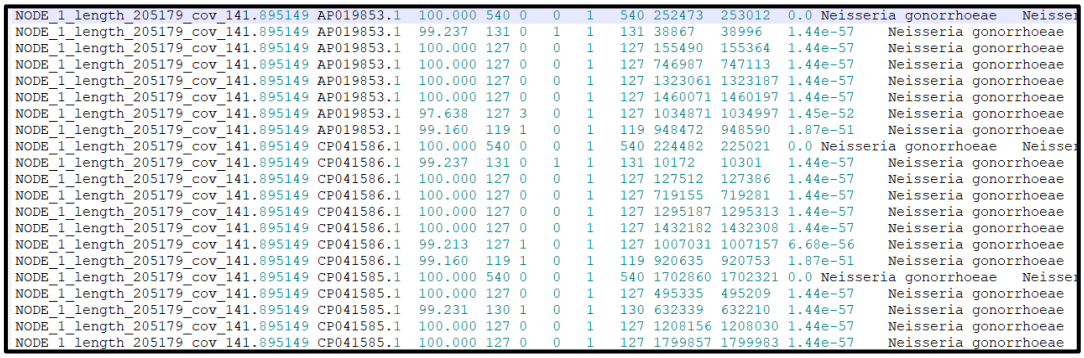
_1_length_264693_cov_5.233288	CP025541.3	100.000	540	0	0	1	540	3073998	3073459	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	LR588412.1	100.000	540	0	0	1	540	2152107	2152646	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	LR133964.1	100.000	540	0	0	1	540	2175173	2175712	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	CP024707.1	100.000	540	0	0	1	540	2453540	2454079	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	CP026164.1	100.000	540	0	0	1	540	2131842	2132381	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	CP025541.2	100.000	540	0	0	1	540	4604242	4604781	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	LR134213.1	100.000	540	0	0	1	540	2242086	2242625	0.0	Klebsiella aerogenes	K
NODE_1_length_264693_cov_5.233288	CP030857.1	100.000	540	0	0	1	540	2264152	2263613	0.0	Klebsiella pneumoniae	
_1_length_264693_cov_5.233288	CP030877.1	100.000	540	0	0	1	540	409519	408980	0.0	Klebsiella pneumoniae	K
NODE_1_length_264693_cov_5.233288	CP030923.1	100.000	540	0	0	1	540	3561773	3562312	0.0	Klebsiella pneumoniae	
NODE_1_length_264693_cov_5.233288	CP030172.1	100.000	540	0	0	1	540	2185791	2186330	0.0	Klebsiella pneumoniae	
_1_length_264693_cov_5.233288	CP028915.1	100.000	540	0	0	1	540	2687286	2687825	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	CP024489.1	100.000	540	0	0	1	540	2238557	2239096	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	CP024482.1	100.000	540	0	0	1	540	2238556	2239095	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	CP021955.1	100.000	540	0	0	1	540	2988574	2988035	0.0	Klebsiella pneumoniae	K
_1_length_264693_cov_5.233288	CP013711.1	100.000	540	0	0	1	540	5053804	5053265	0.0	Klebsiella pneumoniae	K
NODE_1_length_264693_cov_5.233288	CP012883.1	100.000	540	0	0	1	540	409516	408977	0.0	Klebsiella pneumoniae	
NODE_1_length_264693_cov_5.233288	CP006722.1	100.000	540	0	0	1	540	2411370	2411909	0.0	Klebsiella pneumoniae	
NODE_1_length_264693_cov_5.233288	CP002910.1	100.000	540	0	0	1	540	2757654	2757115	0.0	Klebsiella pneumoniae	

Fig (7b): A representation of similarity match for patient B assembled genome.



NODE_1_length_438421_cov_119.223478	GQ900477.1	96.825	126	4	0	1	126	3392	3517	2.42e-50	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	GQ900430.1	96.825	126	4	0	1	126	2678	2803	2.42e-50	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	GQ900428.1	96.825	126	4	0	1	126	10754	10629	2.42e-50	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	GQ900427.1	96.825	126	4	0	1	126	6044	5919	2.42e-50	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	CP038268.1	83.645	214	32	3	125	336	441364	441152	1.88e-46	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	CP038268.1	92.063	126	9	1	1	126	1427332	1427456	8.83e-40	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	CP038268.1	93.976	83	4	1	459	540	1426825	1426907	3.24e-24	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	LR134084.1	83.645	214	32	3	125	336	412011	411799	1.88e-46	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	LR134084.1	92.063	126	9	1	1	126	1409790	1409914	8.83e-40	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	LR134084.1	93.976	83	4	1	459	540	1409283	1409365	3.24e-24	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	FR821779.1	83.645	214	32	3	125	336	412534	412322	1.88e-46	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	FR821779.1	92.063	126	9	1	1	126	1410461	1410585	8.83e-40	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	FR821779.1	93.976	83	4	1	459	540	1409954	1410036	3.24e-24	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	CP039848.1	84.264	197	29	2	122	317	2296718	2296913	3.15e-44	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	LR134139.1	84.264	197	29	2	122	317	398014	397819	3.15e-44	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	LR134091.1	84.264	197	29	2	122	317	391008	390813	3.15e-44	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	CP033505.1	84.264	197	29	2	122	317	423054	422859	3.15e-44	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	CP033506.1	84.264	197	29	2	122	317	425233	425038	3.15e-44	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	LS483314.1	84.264	197	29	2	122	317	392254	392059	3.15e-44	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	CP029629.1	84.264	197	29	2	122	317	440729	440534	3.15e-44	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	AP018376.1	84.264	197	29	2	122	317	393749	393554	3.15e-44	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	CP020020.1	84.264	197	29	2	122	317	398014	397819	3.15e-44	Staphylococcus aureus
NODE_1_length_438421_cov_119.223478	AP014942.1	84.264	197	29	2	122	317	398527	398332	3.15e-44	Staphylococcus aureus

Fig (7c): A representation of similarity match for patient C assembled genome.



NODE_1_length_205179_cov_141.895149	AP019853.1	100.000	540	0	0	1	540	252473	253012	0.0	Neisseria gonorrhoeae	Neisse
NODE_1_length_205179_cov_141.895149	AP019853.1	99.237	131	0	1	1	131	38867	38996	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	AP019853.1	100.000	127	0	0	1	127	155490	155364	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	AP019853.1	100.000	127	0	0	1	127	746987	747113	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	AP019853.1	100.000	127	0	0	1	127	1323061	1323187	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	AP019853.1	100.000	127	0	0	1	127	1460071	1460197	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	AP019853.1	97.638	127	3	0	1	127	1034871	1034997	1.45e-52	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	AP019853.1	99.160	119	1	0	1	119	948472	948590	1.87e-51	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041586.1	100.000	540	0	0	1	540	224482	225021	0.0	Neisseria gonorrhoeae	Neisse
NODE_1_length_205179_cov_141.895149	CP041586.1	99.237	131	0	1	1	131	10172	10301	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041586.1	100.000	127	0	0	1	127	127512	127386	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041586.1	100.000	127	0	0	1	127	719155	719281	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041586.1	100.000	127	0	0	1	127	1295187	1295313	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041586.1	100.000	127	0	0	1	127	1432182	1432308	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041586.1	99.213	127	1	0	1	127	1007031	1007157	6.68e-56	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041586.1	99.160	119	1	0	1	119	920635	920753	1.87e-51	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041585.1	100.000	540	0	0	1	540	1702860	1702321	0.0	Neisseria gonorrhoeae	Neisse
NODE_1_length_205179_cov_141.895149	CP041585.1	100.000	127	0	0	1	127	495335	495209	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041585.1	99.231	130	1	0	1	130	632339	632210	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041585.1	100.000	127	0	0	1	127	1208156	1208030	1.44e-57	Neisseria gonorrhoeae	
NODE_1_length_205179_cov_141.895149	CP041585.1	100.000	127	0	0	1	127	1799857	1799983	1.44e-57	Neisseria gonorrhoeae	

Fig (7d): A representation of similarity match for patient D assembled genome.

Table 3.0: A Table showing a summary of disease or pathogens identified for each patient.

Patients	**Disease / Pathogen**
Patient A	Mycobaterium Tuberculosis
Patient B	Klebsiella Pneumonia
Patient C	Staphylococcus aureus
Patient D	Neisseria gonorrhoea

Conclusion

In this report, it has been demonstrated how relevant analysing the DNA sequence of a data can help identify organisms in an unknown sample. This analysis can help assist clinicians in prediction and overall clinical decisions making. This practical highlighted several procedures that should be followed or adopted in order to obtain optimal results. In addition, the report demonstrated the essence of checking quality scores of raw data acquired from DNA sequencing, why it is crucial to perform quality trimming and ways to assemble the data into genome sequence. The above procedures that were carried out justified the metric used in selecting the best assembler use to assembly the raw sequence data. Finally, it evaluated how organisms can be identified from an assembled genome. However, this report was limited to the following software (Velvet, SPAdes and Unicycler); therefore, further research is required in the advancement of high performing optimal software packages which can be used to improve errors correctness and accuracy.

1. He, B., et al., *Assessing the Impact of Data Preprocessing on Analyzing Next Generation Sequencing Data*. Frontiers in bioengineering and biotechnology, 2020. **8**: p. 817.
2. Schmieder, R. and R. Edwards, *Quality control and preprocessing of metagenomic datasets.* Bioinformatics, 2011. **27**(6): p. 863-864.
3. Bioinformatics, B. *FastQC for Quality Check*. [cited 2021 20 April 2021]; Available from: https://www.bioinformatics.babraham.ac.uk/projects/fastqc/.
4. Bolger, A.M., M. Lohse, and B. Usadel, *Trimmomatic: a flexible trimmer for Illumina sequence data.* Bioinformatics, 2014. **30**(15): p. 2114-2120.
5. Usadellab.org. *Trimmomatic*. [cited 2021 20 April 2021]; Available from: http://www.usadellab.org/cms/?page=trimmomatic.
6. MacManes, M.D., *On the optimal trimming of high-throughput mRNA sequence data*. Frontiers in genetics, 2014. **5**: p. 13.
7. Documentation, A. [cited 2021 21 April 2021]; De novo assembly of Illumina reads using velvet]. Available from: https://angus.readthedocs.io/en/2016/week3/LN_assembly.html.
8. Bioinformatics. *Quast 4.5 Manual*. [cited 2021 21 April 2021]; Available from: http://bioinformatics.se/tools/quast/manual.html.
9. Andrewprzh. [cited 2021 20 April, 2021]; SPAdes genome assembler]. Available from: https://github.com/ablab/spades.
10. Unicycler. *Unicycler*. [cited 2021 21 April 2021]; Unicycler]. Available from: https://github.com/rrwick/Unicycler.
11. Seemann, T., *Prokka: rapid prokaryotic genome annotation.* Bioinformatics, 2014. **30**(14): p. 2068-2069.
12. [cited 2021 21 April 2021]; Prokka Genome Annotation]. Available from: https://github.com/tseemann/prokka.
13. Medicine, U.S.N.L.o. *Basic Local Alignment Search Tool*. [cited 2021 21 April 2021]; Blast Software]. Available from: https://blast.ncbi.nlm.nih.gov/Blast.cgi.
14. Chang, J., et al., *Biopython tutorial and cookbook.* Update, 2010: p. 15-19.
15. Romiguier, J. and C. Roux, *Analytical biases associated with GC-content in molecular evolution.* Frontiers in Genetics, 2017. **8**: p. 16.
16. Ongenaert, M., *Epigenetic databases and computational methodologies in the analysis of epigenetic datasets.* Advances in genetics, 2010. **71**: p. 259-295.