Assignment report,

Advanced sequencing and genome assembly

MSc Applied Bioinformatics,

Marie Schmit

I) Assembly

1) Quality control and genome size

Illumina reads

A control quality is performed on the raw reads using FastQC. Short Illumina read1 file has 2 475 000 reads and a sequence length of 101. The mean quality of its reads is very poor, with a value of 17. However, in the general statistics, no sequence is flagged as poor quality.

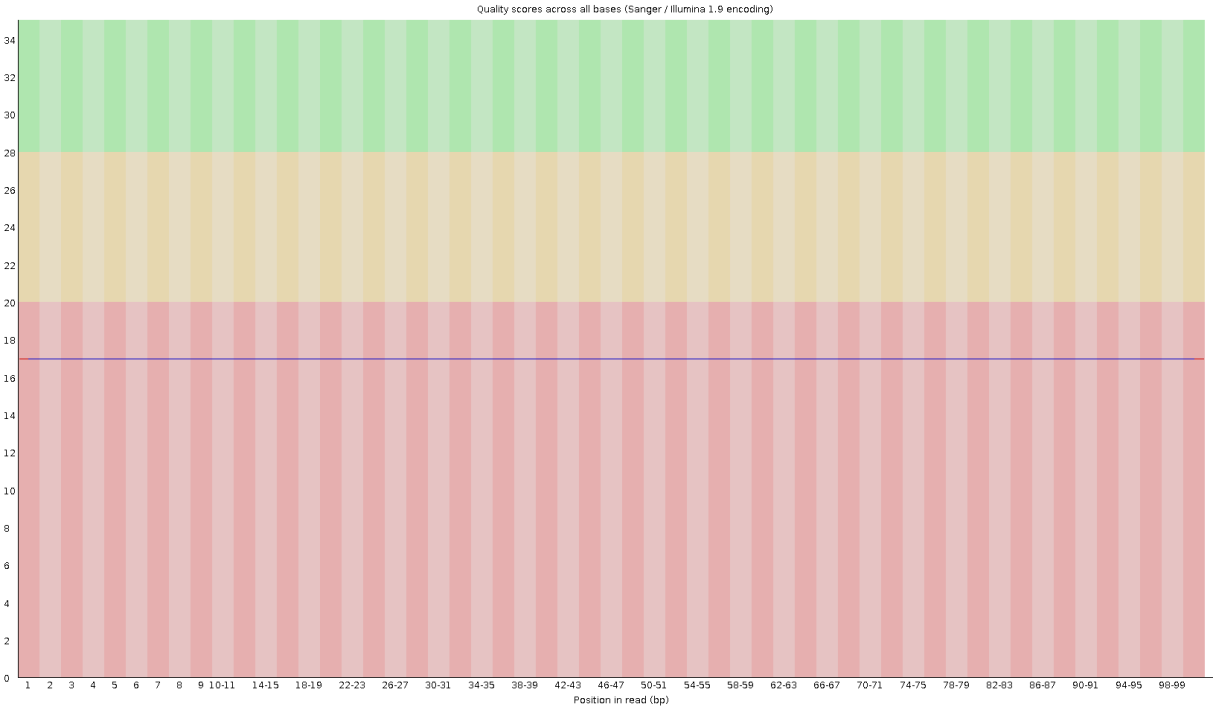


Figure 1 Quality score across bases of first Illumina reads

Its per sequence quality score is also too low, with a mean sequence quality of 17 (more than 1% error rate). The per base content of the first read is good, with little to no differences between bases: no bias were introduced in the reads.

The GC contents presents a warning as its distribution is slightly shifted, which could indicate a bias that does not depend on base position.

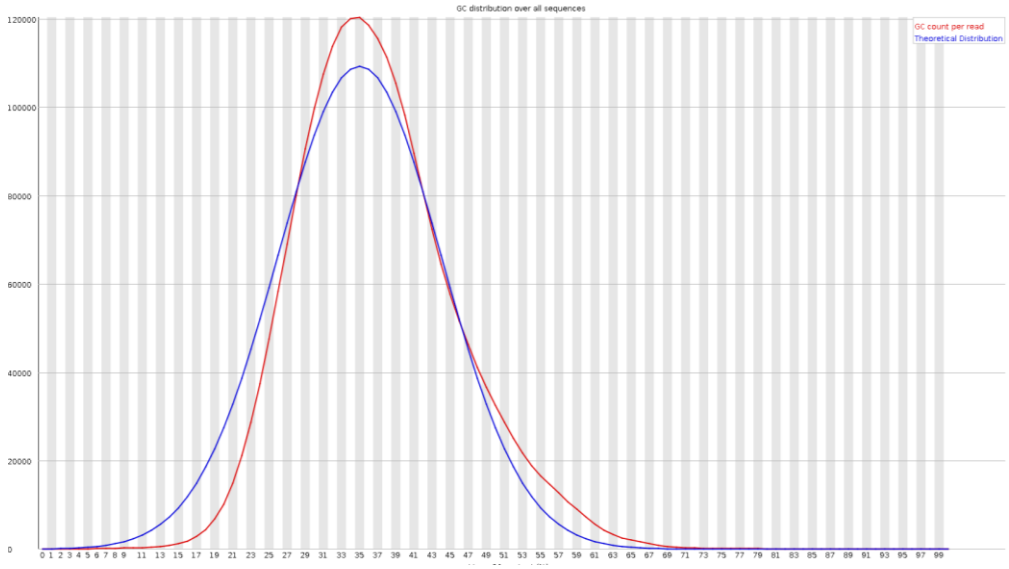
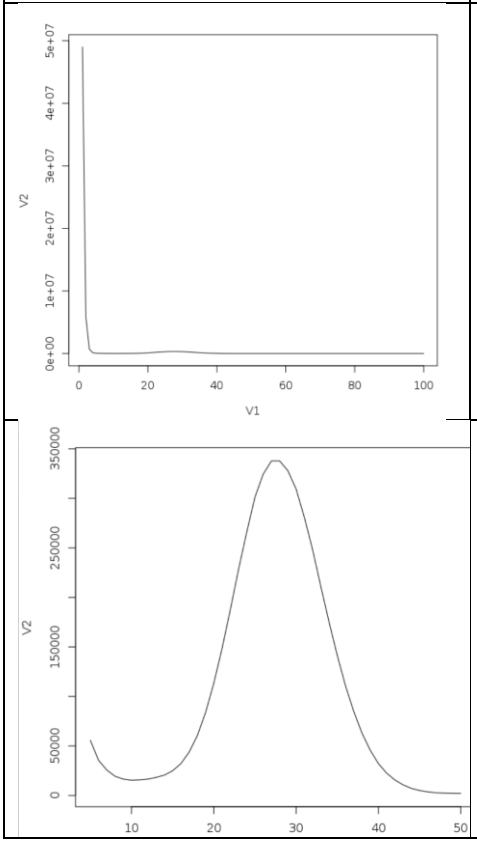
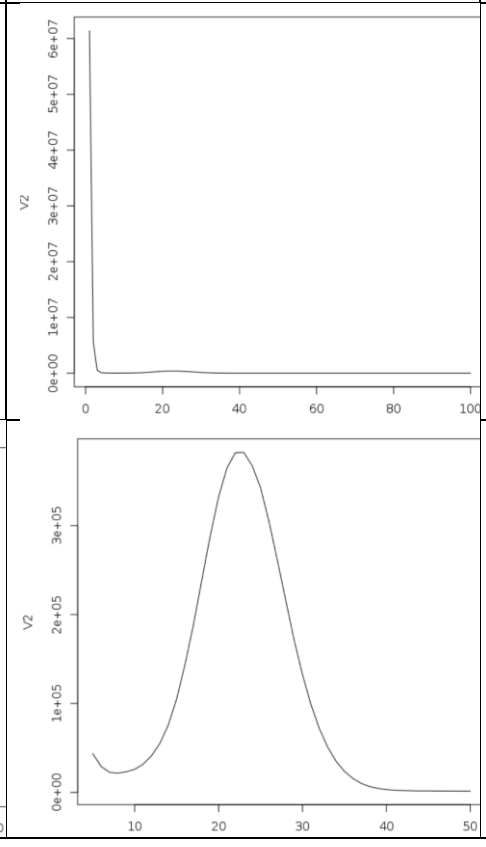
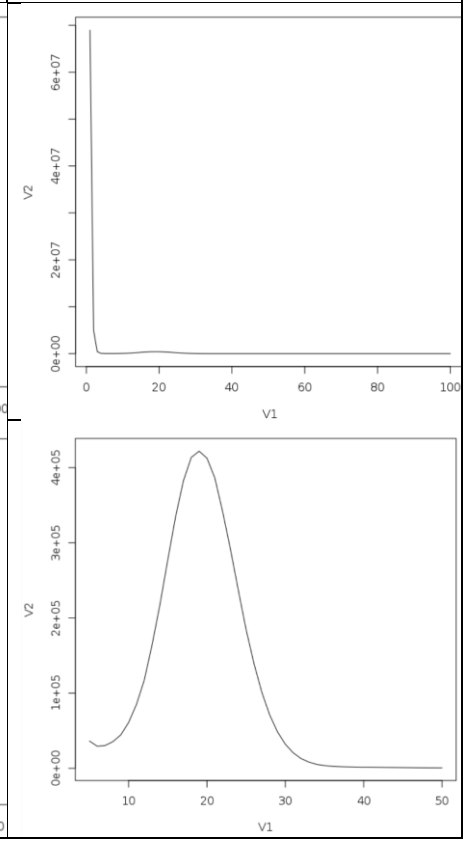


Figure 2 GC content of first Illumina reads

All the other metrics (per base N content, sequence length distribution, overrepresented sequence and adapter content) are good.

The results are the same for Illumina second reads. The quality per base sequence is problematic. Indeed, they are all the same and very low. Thought, no sequence is flagged as poor quality. A problem might have occurred with the sequencing. I will neither trim nor filter those data to improve their general quality, since all the values are the same and are poor. No data would be left after trimming or filtering.

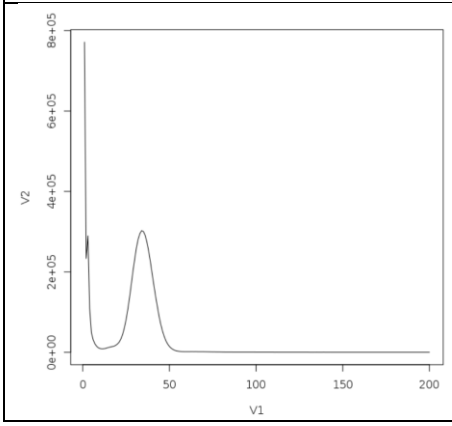
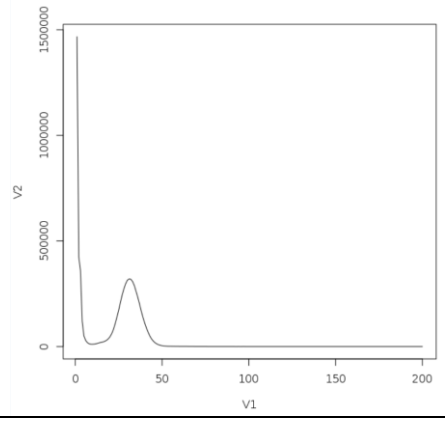
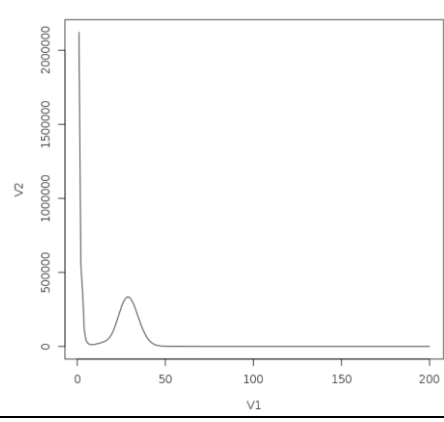
A k-mer analysis is then performed with Jellyfisch, from MaSURCA on both short reads. 2 threads, 1G of elements hash are used. Different values of k-mer are tested: 19; 25; 30. The number of kmer is calculation with “count” function and a histogram is computed using the function “histo”. The results are plot with R.

19mer, R1	25mer, R1	30mer, R1
		

The results are the same for the second reads.

The first major peak of the histogram, at count 1, is the error component. Errors boost the number of kmers appearing once. Apart from this error peak, the distribution presents one major peak, which is characteristic of a rather good data quality.

A correction is applied to try to decrease the error peak, with KmerFreq_AR and Corrector_AR functions (parameters: 1 thread, 33 ASCII shift). The k parameter for the correction is 15 for lighter computation.

19mer, R1 after correction	25mer, R1 after correction	30mer, R1 after correction
		

The number of reads went from 2 475 000 to after 2 378 197 reads correction. 100 000 reads were lost during the correction, but 96% of the reads were kept: the data was not too messy, and a large number of reads could be corrected.

For 19 kmer, the error peak is still very high. Its density value has decreased, going from 5e7 before correction to 8e5 after correction, but the amelioration is small (only a difference of e2). The improvement is poorer with 25mer: the density went from 6e7 to 1e7. The same goes for 30mer, with a density going from 6e7 to 2e7 after correction.

The rest of the analysis will still be done with the correction, since the control quality with FastQC gave poor quality results. The correction process is repeated for R2. The number of reads goes from 2 475 000 before correction to 2 377 838 after.

The resulting k-spectrum has two peaks, one at half the abscise of the other. The first, the hetero-peak, has a frequency of almost 15 000, while the homo-peak has a frequency of around 250 000. It indicates a low level of heterozygosity. This might correspond to heterozygotic alleles in diploid cells.

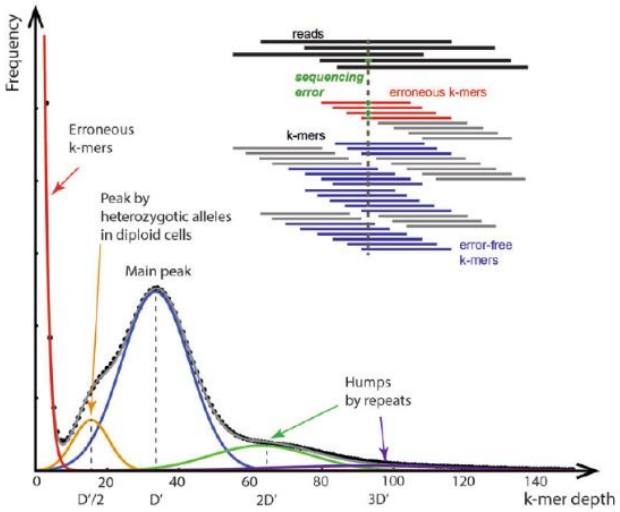
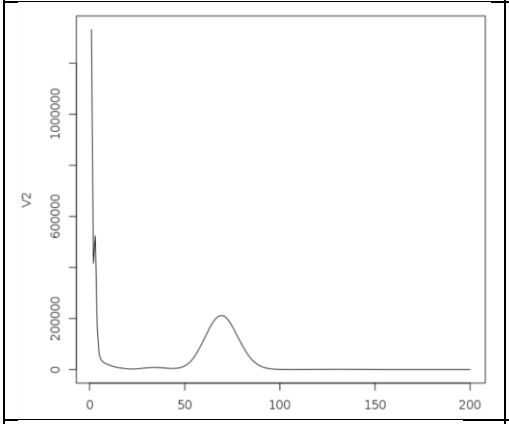
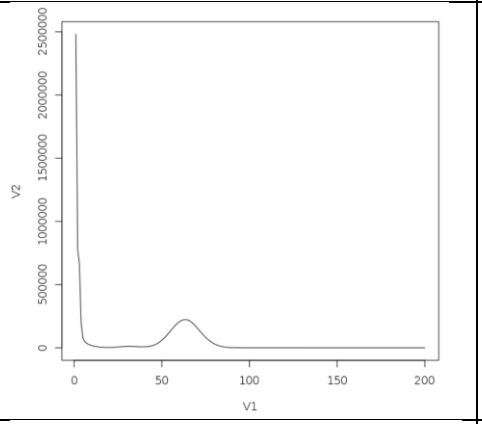
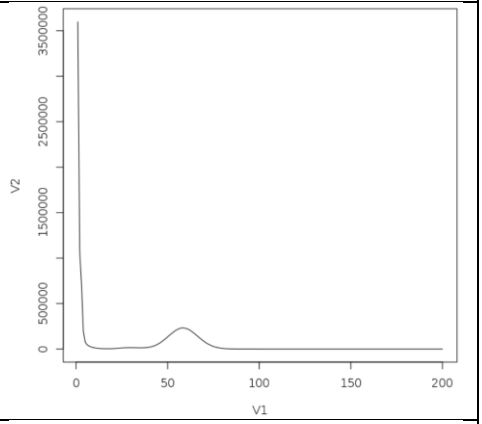
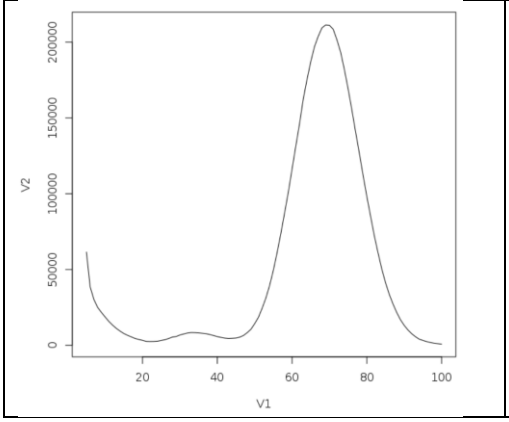
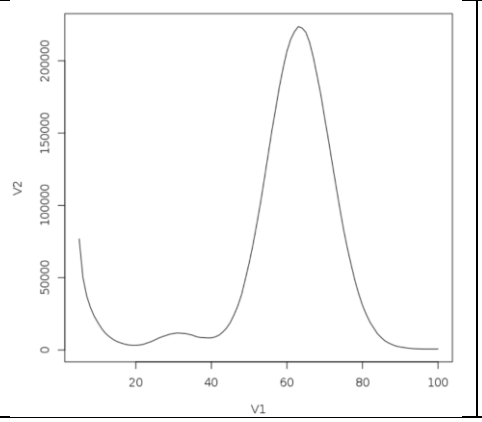
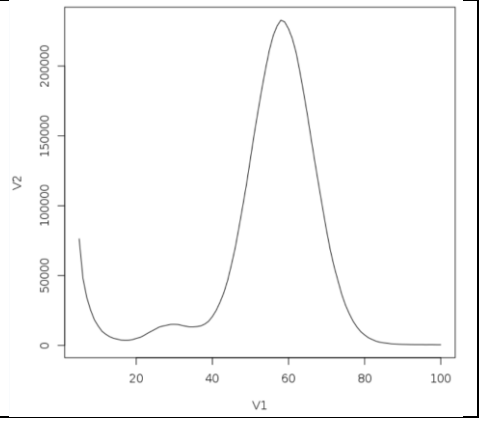


Figure 4. *K*-mer histogram. The *x*-axis refers to the k-mer depth $D(k)$, which indicates 'k-multiplet'; the *y*-axis refers to the frequency of the k-multiplet, $f(D(k))$ [64]. For example, if a set of k-mers is given by $K = \{ATT, ATA, GTG, GCA, GCA, CAT, CAT, TAT, TAT, TAT, TAT\}$. the frequency is calculated as $f(1) = 3$ because there are three unique k-mers, {ATT}, {ATA} and {GTG}; $f(2) = 4$ because there are two twins, {GCA, GCA} and {CAT, CAT}; $f(3) = 0$ because there is no triplet; $f(4) = 4$ because there is one quadruplet, {TAT, TAT, TAT, TAT}. The curve of the k-mer histogram shows a normal distribution in ideal cases, provided that the depth of the read is sufficient. If there are sequencing errors in the reads, an exponentially decreasing curve is produced. The humps beyond the normal distribution peak are generated due to the repetitive structures and copy-gained regions. In the plot, a small peak resulting from heterozygous alleles appears below the main peak. The black dots are obtained by using ERR244145, and the exponential (erroneous; red) and Gaussian (error-free; orange, blue, green and purple) functions are ideal case curves. The gray line (sum of ideal cases) is similar to the real data (black dots).

Figure 3 In orange, a similar case of hetero-peak due to heterozygotic alleles in diploid cells (Nam Jin-Wu, 2016)

The genome will be considered as highly homozygote, so the second peak is used to calculate the genome size. Statistics on the peak are calculated with R, like the height of the peak or the number of kmers in it. The genome size was calculated with the following formula: G = N/C1 = R*L*((L-k+1)/(L*C2)) where N is the number of base sequenced, L is the length of the reads, C1 is the base coverage and C2 is the kmer coverage.

19mer spectrum for both reads (corrected)	25mer spectrum for both reads (corrected)	30mer spectrum for both reads (corrected)
		
		

Total of kmers in the peak: 338 767 481 Maximum value: 211 332 at position 69 Genome size = 338 767 481 / 69 = 4.91M Single copy portion of the genome: 4 562 644[1] (92.93%)	Total of kmers in the peak: 307 463 847 Maximum value: 223 641 at position 63 Genome size = 4.88M Single copy portion of the genome: 4 678 372[2] (95.86%)	Total of kmers in the peak: 308 190 218 Maximum value: 232 832 at position 58 Genome size = 5.3M Single copy portion of the genome: 4 738 324[3] (89. 17%)

The k parameter providing the best percentage of estimation of the genome is 25. This value will be kept for the assembly: 25mers seems a good compromise between long and short k values. Indeed, long kmers help remove repetitions in the reads but with a decrease of coverage, while small kmers can remove errors and provide a better assembly when the coverage is low (Luo R, 2012).

The number of bases in reads 1, before correction, is 249.975.000. For a genome of 5M, the coverage is approximately 50, which is a very high coverage value. After correction, the number of reads is 218 052 444, so the coverage is approximatively 43, still a very high value, which seems too high in comparison with the poor quality of the data.

[1] sum(as.numeric(df19[45:95,1]*df19[45:95,2]))/69
[2] sum(as.numeric(df25[40:90,1]*df25[40:90,2]))/63
[3] sum(as.numeric(df30[35:85,1]*df30[35:85,2]))/58

For this file, the number of sequences is 33 413. Their minimal length is 167 and the maximal is 246451. No sequences are flagged as poor quality and the percentage of GC content is 37.

The per base sequence quality is very poor, with a mean of 8.5. The quality is the same for almost all the reads, with a very light drop of 0.5 at the end. Since the quality is certainly bad but constant, no filtering nor trimming will be applied.

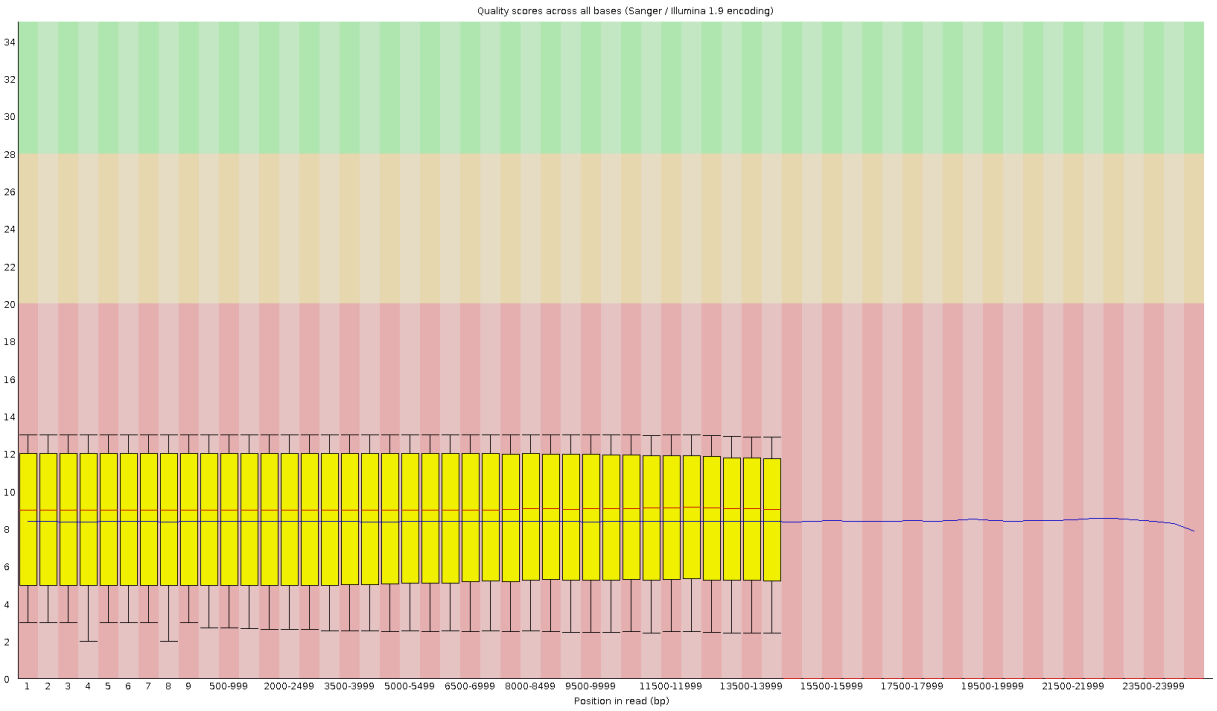


Figure 4 Per base sequence quality, PacBio reads

The per sequence quality score is also flagged as failed. The mean quality value is around 8, and the maximal value is 10. It confirms the per base sequence quality: both are problematic.

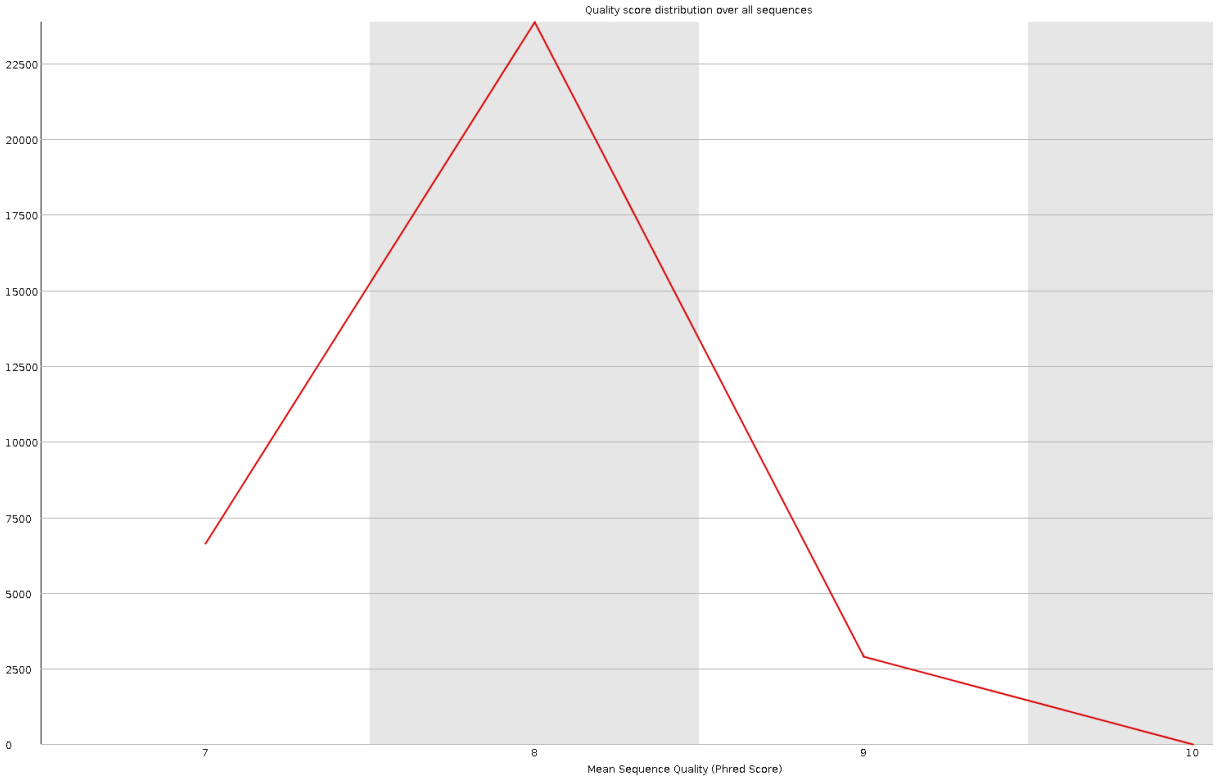


Figure 5 Average quality per read (Quality score distribution over all sequences), PacBio reads

The per base content is also flagged as a failure. The percentage of adenine is for instance 30% higher than the percentage of guanine for the last bases (between 23500 and 24000).

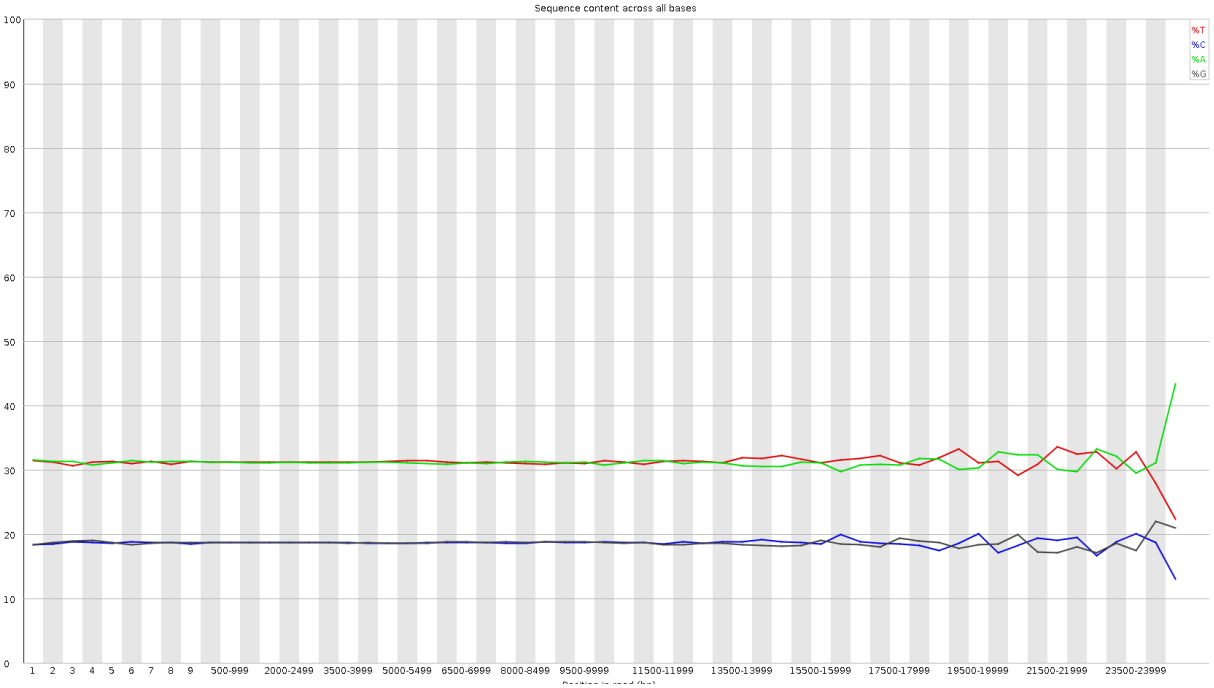


Figure 6 Sequence content across all bases

As for the short reads, the per sequence GC content has a warning. The same goes for the sequence length distribution. The length of the sequences indeed vary, with a majority of sequences (5000) having a length between 1500 and 2500 bp.

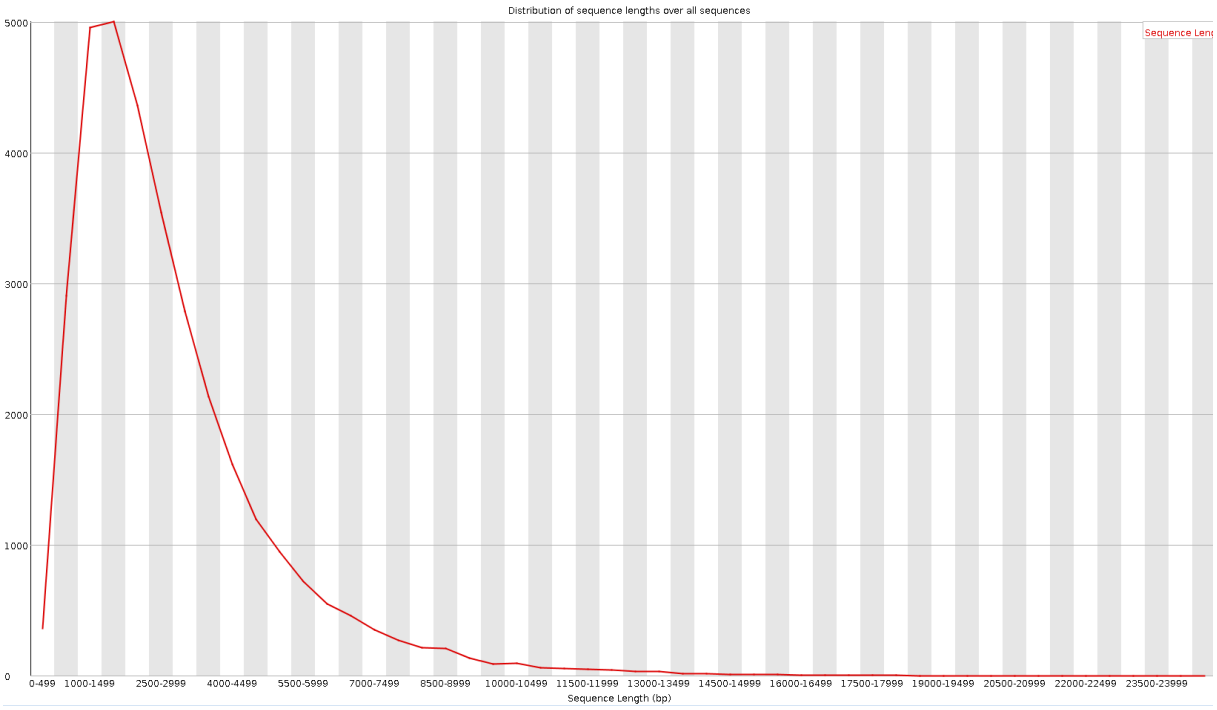
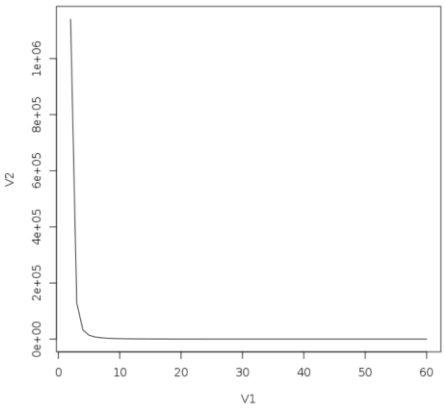
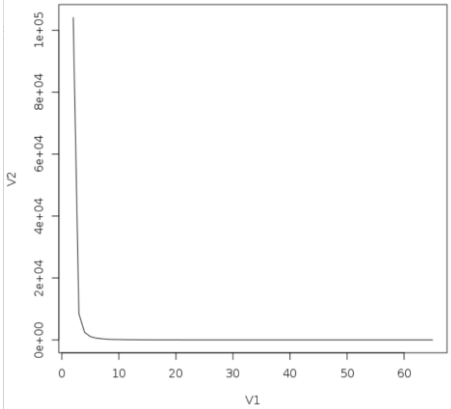
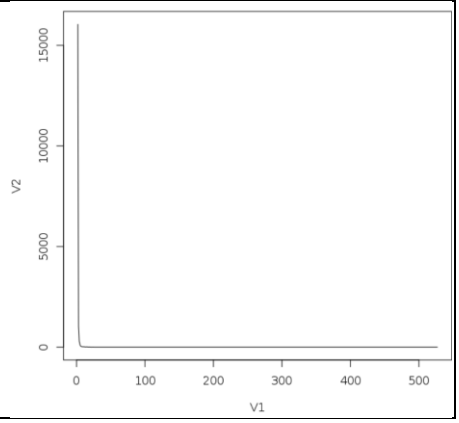


Figure 7 Sequence length distribution over all sequences for PacBio reads

The adapter and per base N contents, the sequence duplication level are good and there are no overrepresented sequences.

The same kmer analysis and spectrum are obtained for the long reads, with the different values of k. No other peak than the error one is visible.

19mer, PacBio reads	25mer, PacBio reads	30mer, PacBio reads
		

A correction is applied as previously on the Illumina short reads. The results are much better, with the apparition of, again, two peaks. Again, no peak appears. This might be due to the large number of errors that often occurs with a PacBio analysis. The distribution that is mostly represented would be the error one. This hypothesis seems confirmed, since "in PacBio, assuming a typical kmer size of 16, only ~5% of the distinct kmers from the reads are error free" (Carvalho AB, 2016). The error peak thus contains more than 90% of the kmers, explaining the observed k spectrum.

2) De-novo assemblies

Multiples assemblies will be tested, with three different values of kmers. Short and long reads assembler will be used, then hybrid assemblers. The best results are expected from the hybrid assemblies, since they combine the pros of both the short (more precise) and long reads.

a) Short reads assembly using SOAP

Assembly

For the first assembly, only the short reads are used. The de novo assembler used is SOAPdenovo2, which uses De Bruijin graphs, and the performed steps are the following: pregraph and contig, since no long jump library is provided for the assembly. Pregraph calculates for instance the frequency of kmers, the number of vertexes, of edges, etc. Contig uses that information to create the graph, by calculating for instance information about contigs.

SOAPdenovo-63mer is used to avoid using too much memory. In the configuration file, the insert size is 350bp, the reverse_seq is 0 because the sequence does not need to be complementarily reversed, the asm_flags is 1 for a contig assembly, a rank of 1 since only one library is used. 2 CPU are allocated. The corrected reads are used. The genome size that is used is 4.9M for 19 and 25mers, and 5.3M for 30mers.

Quality statistics

Gnx-tool is computed to display statistics about the assembly.

Contigs	19mers	25mers	30mer
Total number of sequences	125.392	95.827	67.916
Total length of sequences	7.978.233 bp	7.974.697 bp	7.424.920 bp
Shortest sequence length	20 bp	26 bp	30 bp
Longest sequence length	11.346 bp	17.573 bp	21.479 bp
Total number of Ns in sequences	0	0	0
N50	481	950	1.764

For the three kmers, the total length of the sequences is higher than the estimated size of the genome (approximately 7 or 8M bp). Also, the N50 is very low: there is probably a large amount of little fragmented contig. The total number of sequences is also rather high. 30mers is better because it has the highest N50 and the smallest number of sequences. The length of its sequences is also smaller, but still closer to the estimated size of the genome. However, those statistics are in overall poor.

b) Short reads assembly using velvet

Velvet is also a de novo assembler that uses De Bruijin graphs. The function “velveth” is used for graph creation, “velvetg” for graph assembly. The parameters are “shortPaired” and “separate”, since short paired Illumina reads in two separate files are used. The short reads files used are non-corrected, because the corrected are indicated as comporting too few sequences by velvet. After correction, the number of reads in read2 is indeed smaller than in read1.

Contigs	19mers	25mers	30mer
Total number of sequences	354.059	235.638	200.766
Total length of sequences	14.707.282 bp	14.044.733 bp	14.223.431 bp
Shortest sequence length	37 bp	49 bp	57 bp
Longest sequence length	128 bp	205 bp	282 bp
Total number of Ns in sequences	0	0	0

N50	37	57	70

For this analysis, the N50 is again extremely small, while the length of the sequences is very high. The short read assembly has poor statistics for all the kmer.

c) Short read using platanus

Platanus de novo assembly is ideal for highly heterozygous diploid genomes (Rei Kajitani, 2014). This assembler might help to deal with the hetero-peak present in the kmer spectrum, since it presented very good results for various levels of heterozygosity in de novo assembly contest Assemblathon2 (Rei Kajitani, 2014).

First, the option "assemble" of platanus is used, to create contigs. The parameters are 19 for 19mers, and the corrected fastq pair end short reads files. Then, the contigs are assembled into scaffhold with the function "assemble". The average insert size parameter is 350bp. The reads are entered as inward, since they are paired-end reads (Kajitani). The gap closing is performed with the function gap_close and the same parameters.

For this analysis, only 25mers, chosen at the first after the kmer analysis, will be tested (see other k values: 1). Also, 30mers have a better N50 (around 3000 after scaffolding) but it comes with 300 NS. Thus, 25mers are kept for this assembly.

Contigs	25mers, assembly
Total number of sequences	2.980
Total length of sequences	5.141.223bp
Shortest sequence length	64 bp
Longest sequence length	77.759 bp
Total number of Ns in sequences	0
N50	19.664

Contigs without correction	25mers, assembly	25mers, scaffolding	25mers, gap filling
Total number of sequences	7.326	4.851	4.851
Total length of sequences	5.348.653	5.069.939	5.069.939
Shortest sequence length	58 bp	100 bp	100 bp
Longest sequence length	9.906 bp	9.856 bp	9.856 bp
Total number of Ns in sequences	0	0	0
N50	2.093	2.133	2.133

The N50 is here better than with the other two assemblies. The total length of the sequences is also closer to the estimated genome size, and the total number of sequences is lower than with soap and velvet. The longest sequence length is approximatively five times longer, which is better. There is not improvement after gap closing, but scaffolding makes the sequences longer and the N50 larger. The correction applied to the file has considerably improved the statistics: without it, the N50 value was 2.093, the longest sequence 9.906 bp and the total number of sequences 7.326. However, scaffolding could not be done after it, since the corrected number of reads were not equals. Platanus has the best results for short reads assembly. Nonetheless, it's N50 is still very low.

Platanus is the best assembly tool for short reads, with 25mers. This tool is the best candidate to be used later for hybrid assemblies.

d) Long reads assembly using canu

The assembly is then done with PacBio long reads. Cd The first tested assembler is canu. Canu is tested with 25mers, also with the non-corrected reads, because the read coverage of the corrected pacbio reads is too low. The coverage with canu is too low to run canu, which may be due to an incorrect genomeSize or a poor quality of reads that could not be enough corrected.

e) Hybrid assembly with DBG2OLC

First, both the short and long reads are computed with this assembler. It builds a De-Bruijn graph for the short reads with the command "SparseAssembler". The assembly is computed on corrected genome, with 25mer. The genome size is 5000000, the false kmer threshold is 1, as well as the false edge threshold and the skip size.

For long reads assembly, the overlap layout consensus is performed with an adaptive k-mer matching threshold to filter low quality reads (AdaptiveTh) of 0.0001, a fixed kmer matching threshold of 2 (KmerCovTh) and a minimum overlap score (MinOverlap), the minimal overlap length required to make a De Bruijin graph, of 20. The PacBio long read file is converted in fasta format.

The consensus is then called to combine data from both short and long reads. The bundled split_and_run_sparc.sh script used is the one given in class. Blasr, a long-read aligner is used to perform the consensus.

Contigs with correction	25mers
Total number of sequences	79
Total length of sequences	905.952 bp
Shortest sequence length	1.804 bp
Longest sequence length	35.122 bp
Total number of Ns in sequences	0
N50	12.486

For this hybrid assembly, the N50 is still small. The length of the contigs are however larger than with the previous assembly, and the total number of sequences is smaller. However, the length of the sequence is very small, it makes only one fifth of the genome size.

After this first analysis, the parameters are tuned to find a better combination. The tuning is made with a grid search coded in a sub script (see script 3). The coverage for PacBio corrected long reads is: coverage = total number of bp / gene size = 99600120 / 5000000 = 19.2. The parameters to tune for a coverage between 10x and 20x for PacBio data are according to the manual of DBG2OLC: KmerCovTh 2-5, MinOverlap 10-30, AdaptiveTh 0.001~0.01. So, their tested values in the tune grid are: k = (19; 25), NodeCovTh=(1; 2), KmerCovTh=(2; 3; 4; 5), MinOverlap=(10; 20; 30), AdaptativeTh=(0.0001; 0.001; 0.01).

For 19mer, the best combination of parameters is: NodeCovTh = 1, KmerCovTh = 2, MinOverlap = 10, AdaptiveTh = 0.0001.

Contigs with correction	19mers with optimum parameters
Total number of sequences	185
Total length of sequences	5.128.354 bp
Shortest sequence length	576 bp
Longest sequence length	163.820 bp
Total number of Ns in sequences	0

N50	66.789

Here, the N50 is larger and the total length of the sequences matches the calculated length of the genome. The longest sequence length is also much larger than before the parameter's optimisation.

Polishing DBG2OLC assembly

Pilon is used to polish the results of DBG2OLC assembly, to make the results better. Thanks to bwa and samtools, a .bam is created, which contains all the alignments. It is then sorted and indexed before running pilon. Normally, iterations of Pilon (for instance 5 iterations, to correct enough sequences without introducing bias) would have been done. However, no SNP were corrected after the first iteration. Thus, only one is computed. This might be linked to the poor quality of the data.

DBG2OLC and platanus

Since platanus was the short-reads assembler having the best results with 25mer, a hybrid analysis is made between short reads with platanus and long reads with DBG2OLC. A tuned grid is again used to find the best parameters.

The assembly is tested with 19 and 25 kmers, with 10 or 20 MinOverlap. The best combination is 19mers, 0.0001 AdaptiveTh, 2 KmerCovTh, 10 MinOverlap, 1 RemoveChimera.

Corrected reads	
Total number of sequences	48
Total length of sequences	4.797.012 bp
Shortest sequence length	1.659 bp
Longest sequence length	536.674 bp
Total number of Ns in sequences	0
N50	171.235 (9 sequences)

The assembly is better than with DBG2OLC only: the N50 is larger, the 9th largest contigs cover 28% of the genome. However, the total length of the sequence is smaller, but still close to the genome size.

f) Hybrid assembly with MaSuRCA

A MaSuRCA configuration file is generated to set the corrected reads files path, the number of threads to 4 and the PE value (insert size of 350 and read length of 50).

Corrected reads	
Total number of sequences	16
Total length of sequences	4.957.066
Shortest sequence length	15.151 bp
Longest sequence length	1.434.334 bp
Total number of Ns in sequences	0
N50	1.349.311

Compared to the other assemblies, the results with MaSuRCA are excellent: the total length of the sequence is almost equal to the length of the genome, the sequences have a large size, even the smallest one. The N50 covers more than a fifth of the genome, which means that the continuity of the assembly is good.

Polishing MaSuRCA assembly

Pilon is used to polish the results of MaSuRCA assembly, in order to find the best possible. Like before, no SNPs were corrected.

g) Hybrid assembly with spades

Spades is run with 4 cpus and 9GB of ram (a first try was made with 4GB, which was not enough for the assembly). For spade, assemblies are computed for uncorrected reads, since the corrected one have an unequal number of sequences. Spades statistics are worse than the other hybrid assemblies' results. Indeed, the N50 is smaller: 34 contigs are necessary to cover half of the genome, and they have a small length. Ns were introduced, so this assembly presents numerous ambiguous bases, which was not the case with MaSuRCA or DBG2OLP. The total length of the sequence is close to the genome size, but some very small sequences (the smallest is 56 bp) are present.

Uncorrected reads	
Total number of sequences	999
Total length of sequences	4.955.502 bp
Shortest sequence length	56 bp
Longest sequence length	134.311 bp
Total number of Ns in sequences	1100
N50	44.504 (34 sequences)

3) Quality control

The three best assemblies were made with MaSURCA, with DBG2OLC and platanus (19mers, 10 MinOverlap) and with DBG2OLC (19mer, 10MinOverlap) (see fasta files here: Best Fasta\Three best assemblies). Only the basic statistics very exanimated: those values can be misleading without a deeper quality check. For each resulting fasta file, a quality check will be made with ALE and with KAT.

a) MaSURCA reads

200 000 errors were introduced, visible in dark. Those errors are in majority located in the error peak, which is usual, but some are also present in the small first peak at 30 kmers. The goal is to minimise the number of kmer present in the assembly but not in the reads: here, those kmers are quite numerous. The largest peak at 65 kmers is red, which mean that no copy was created for those kmers: they are only present once in the assembly and the reads.

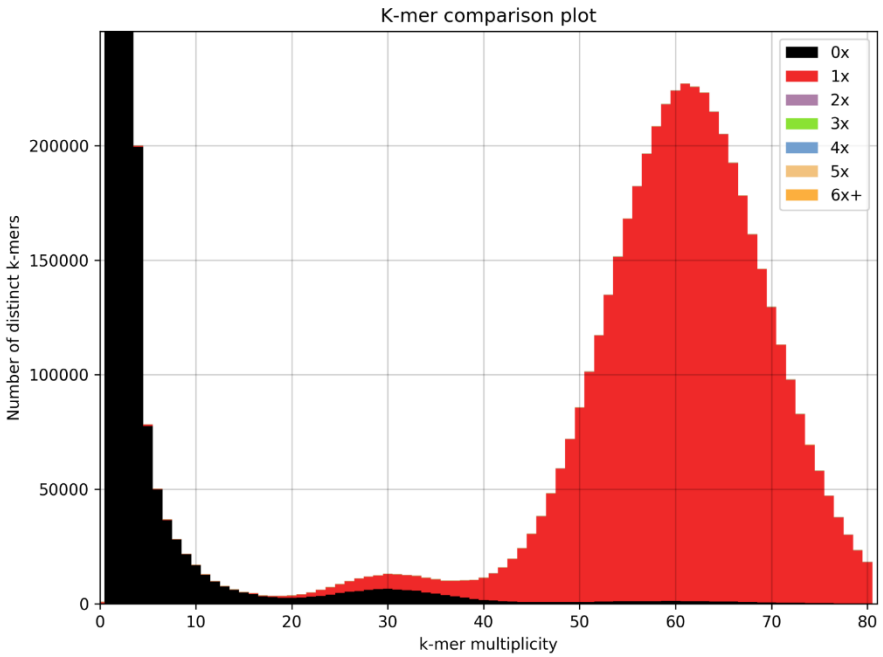


Figure 8 kmer comparison plot (MaSuRCA assembly, corrected short reads)

b) DBG2OLC and platanus

The previous assembly with MaSuRCA has better quality. Here, the second peak also has errors (in black at 60 kmers), that are quite numerous: above 50 000 distinct kmers are present only in the assembly. A small amount of duplicate was introduced (in purple at the top of the peak). This assembly is mediocre: many errors were introduced across all kmers multiplicity, when the aim is to keep it as small as possible.

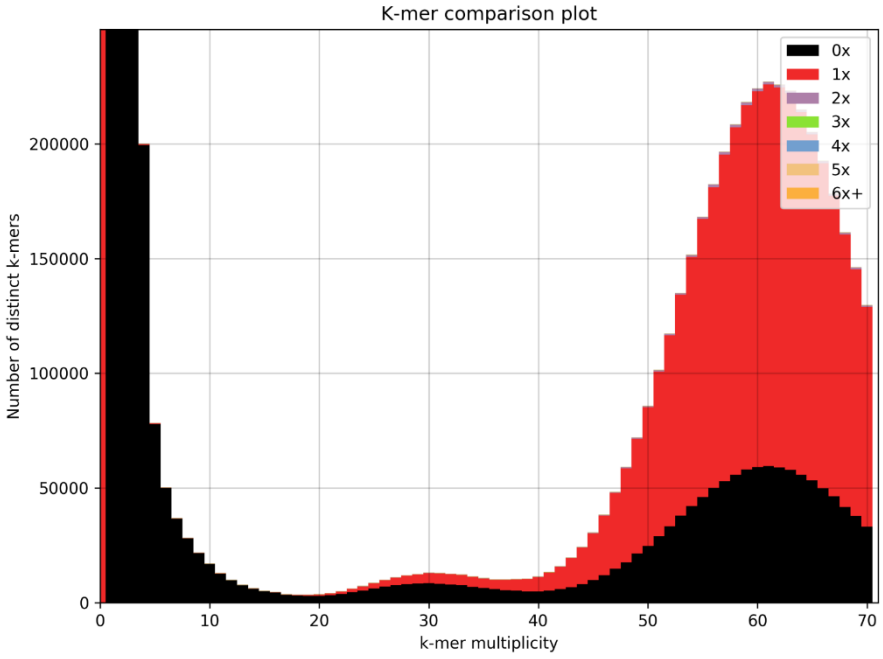


Figure 9 kmer comparison plot (DBG2OLC and platanus assembly, corrected short reads)

c) DBG2OLC

For this assembly, the number of errors is still high, and also impacts all the kmers multiplicity. Its height is now lesser than 50.000, which is better than the platanus analysis. However, more duplicates were introduced: 12.500 distinct kmers are duplicates.

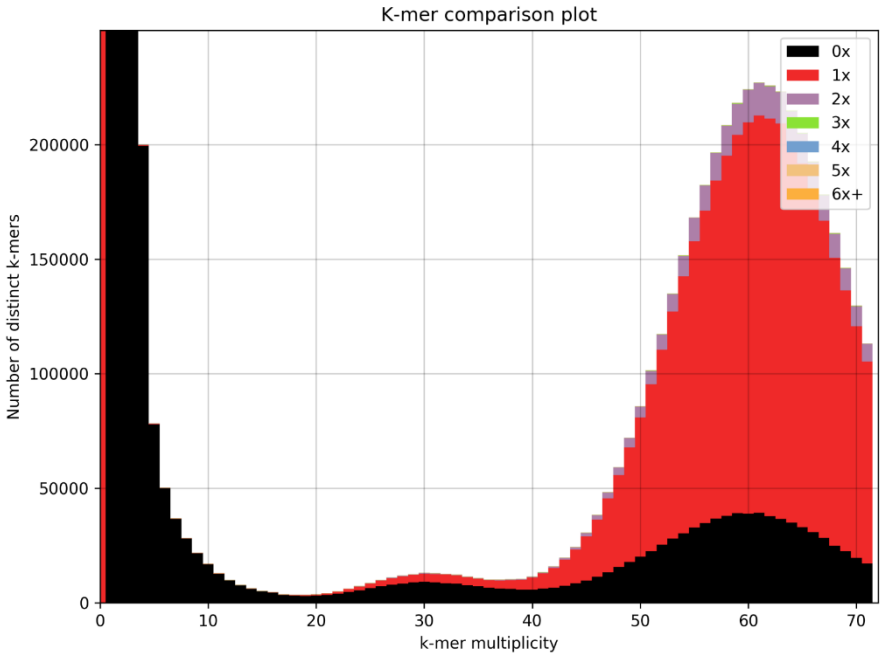


Figure 10 kmer comparison plot (DBG2OLC assembly, corrected short reads)

The quality of the three assembly is mediocre, but MaSuRCA's can be considered acceptable. The raw data also had bad quality, which was not highly improved by correction. MaSuRCA is the best assembly, with better statistics as well as fewer errors and no duplicates.

4) Gene prediction

Gene prediction is realised with Augustus to identify potential genes within the MaSuRCA assembly. Without supporting RNA sequence, the analysis is "Ab initio", relying on pre-computated trained models for probabilistic genes prediction. Augustus requires the selection of a closest specie, but no information was given on the provided reads.

A first analysis is made with Homo sapiens [Metazoa – Chordata – Mammalia] as closest specie. 26 genes were predicted with this specie. With closest specie: Staphylococcus aureus [Bacteria - Firmicutes - Bacilli], 534 genes are found. With Bathycoccus prasinos [Viridiplantae - Chlorophyta - Mamiellophyceae], 190 genes are found. One mammal, one bacterium and one plant were tested. Staphylococcus aureus will be kept for further analysis, since more genes were found for this specie.

For gene annotation, similarities are search between the sequenced assembly and BLAST database. No taxonomy filter is used, since the organism is unknown. The annotations will be less precise, but randomly choosing a taxonomy could introduce a bias. The studied specie is probably close to Homo sapiens, maybe a mammal, according to blast results. Both the number of hits, and the specie distribution, are higher for homo sapiens.

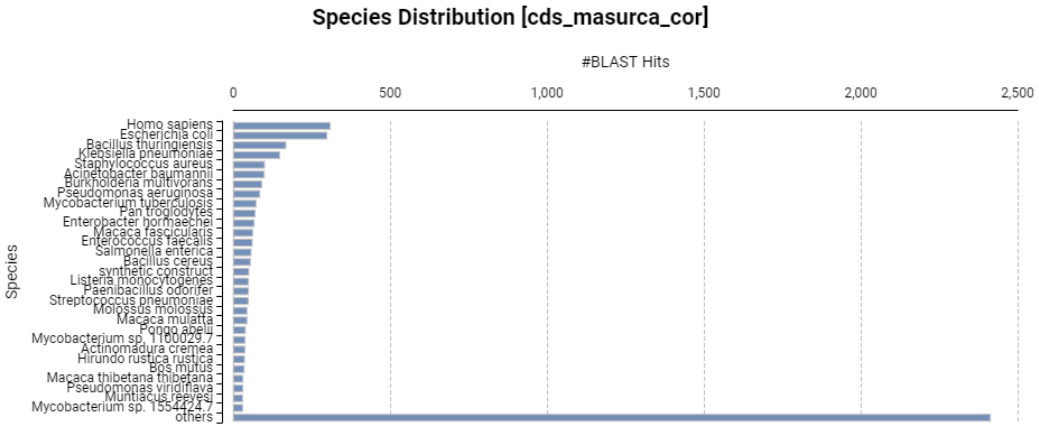


Figure 11 Species distribution after blast similarities search of MaSuRCA assembly

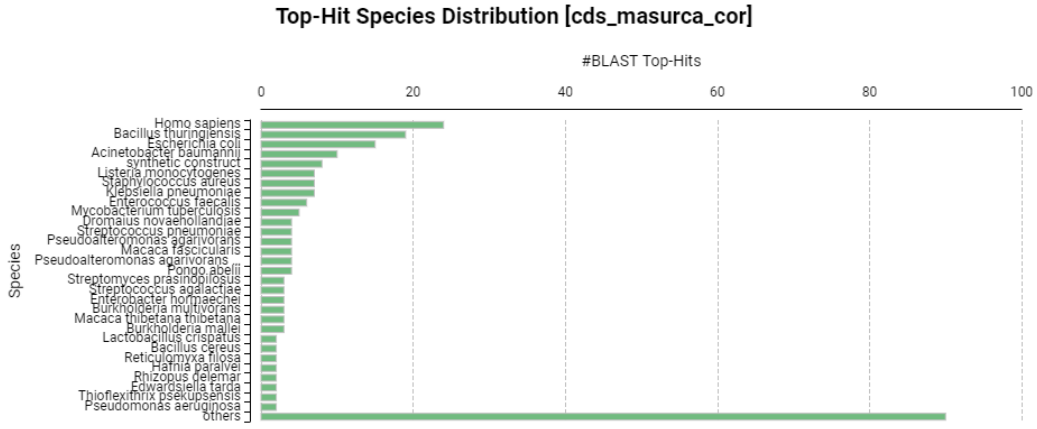


Figure 12 Top hit species distribution after blast similarities search of MaSuRCA assembly

The number of unknown specie is very high, with almost 2500 “other” species. Around 400 sequences were not mapped.

Next, the Gene Ontology is mapped. The annotations are computed with a cutoff of 50. The most annotated biological processes are cellular and metabolic processes. For molecular function, the binding and catalytic activity are predominant. The most represented cellular component is cellular anatomical entity.

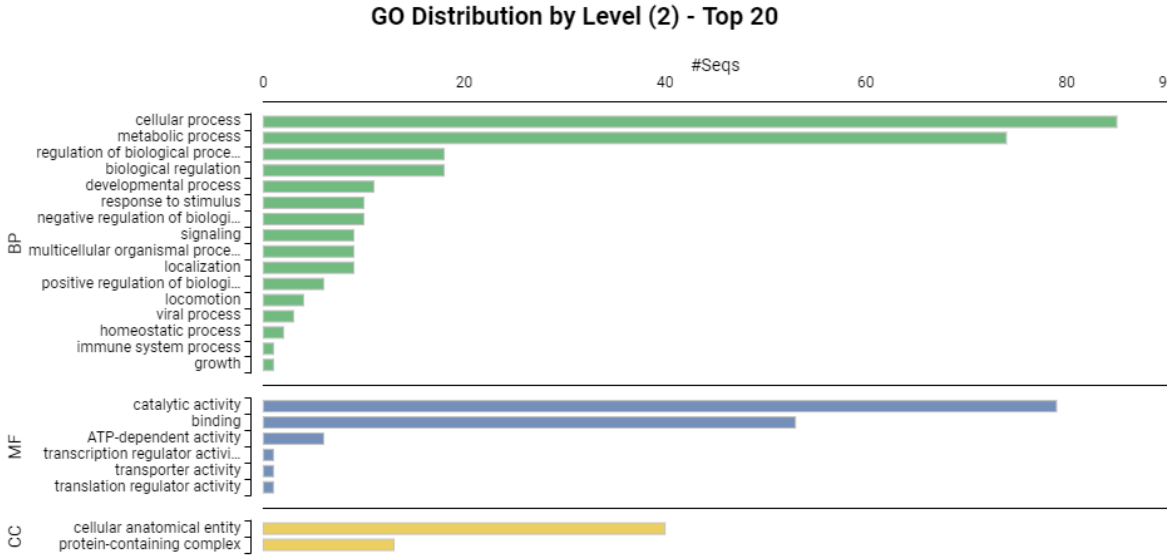


Figure 13 GO distribution by level, top 20 (MaSuRCA assembly)

A large part of the sequences could not be annotated. This might be linked to the lack of precision of the assembly.

The KEGG pathway is also computed, allowing to visualise the interaction between the enzymes, the genes, some cellular processes, etc. 58 pathways were found for KEGG, 106 for Reactome, 6 for plant Reactome database.

Conclusion

All in all, the best assembly was obtained with MaSuRCA on corrected reads. However, this assembly is still poor and could be improved. Other assemblers might be tested, and a better correction could be applied on the reads. A choice was made to not trim or filter them, but it could have improved their poor quality. The studied organism is probably a mammal with high catalytic, cellular and metabolic activities.

Annexes

1) *Results of short reads platanus analysis*

Contigs without correction	19mers, assembly	19mers, scaffolding	19mers, gap filling
Total number of sequences	6.362	6.362	
Total length of sequences	4.856.045	4.856.045 bp	
Shortest sequence length	100 bp	100 bp	
Longest sequence length	5.783 bp	5.783 bp	
Total number of Ns in sequences	0	0	
N50	1.178	1.178	

Contigs without correction	25mers, assembly	25mers, scaffolding	25mers, gap filling
Total number of sequences	7.326	4.851	4.851
Total length of sequences	5.348.653	5.069.939	5.069.939
Shortest sequence length	58 bp	100 bp	100 bp
Longest sequence length	9.906 bp	9.856 bp	9.856 bp
Total number of Ns in sequences	0	0	0
N50	2.093	2.133	2.133

Contigs without correction	30mers, assembly	30mers, scaffolding	30mers, gap filling
Total number of sequences	6.848	3.702	4.851
Total length of sequences	5.274.590 bp	5.014.987 bp	5.069.939
Shortest sequence length	52 bp	100 bp	100 bp
Longest sequence length	18.345 bp	18.308 bp	9.856 bp
Total number of Ns in sequences	0	320	320
N50	3.229	3.335	2.133

2) *Results of DBG2OLC hybrid assembly tuning*

Per default values are:

- NodeCovTh = 1
- EdgeCovTh = 1
- K = 25
- AdaptiveTh = 0.0001
- KmerCovTh = 2
- MinOverlap = 20
- RemoveChimera = 1

Tuning 1: k = 30

Impossible to tun with 30mers.

Tuning 2: k=19, corrected reads and corrected pacBio

Total number of sequences: 178

Total length of sequences: 4641005 bp

Shortest sequence length : 2942 bp

Longest sequence length : 110739 bp

Total number of Ns in sequences: 0

N50: 43423 (38 sequences) (2324879 bp combined)

3) Sub script for DBGOLC grid search

```
##Define the values for each parameters

k=(19)

NodeCovTh=(1 2)

KmerCovTh=(2 3 4 5)

MinOverlap=(10 20 30)

AdaptativeTh=(0.0001 0.001 0.01)

for k_param in "${k[@]}"; do

        for nodeCov_param in "${NodeCovTh[@]}"; do

                for kmerCov_param in "${KmerCovTh[@]}"; do

                        for min_param in "${MinOverlap[@]}"; do

                                for adapt_param in "${AdaptativeTh[@]}"; do

                                  ##Create a new folder for the current set of parameters

                                  echo "params_$k_param-$nodeCov_param-$kmerCov_param-$min_param-$adapt_param"

                                  mkdir
/scratch/s388143/gamod/Assignment/Assemblies/hybrid/DBG2OLC/grid_search_tuning/"params_$k_param-$nodeCov_param-
$kmerCov_param-$min_param-$adapt_param"

                                  cd
/scratch/s388143/gamod/Assignment/Assemblies/hybrid/DBG2OLC/grid_search_tuning/"params_$k_param-$nodeCov_param-
$kmerCov_param-$min_param-$adapt_param"

                                  echo "Execution of sparse assembler"

                                  singularity exec /scratch/s388143/gamod/gamod.simg
/DBG2OLC/compiled/SparseAssembler GS 5000000 NodeCovTh $nodeCov_param EdgeCovTh 1 k $k_param g 1 f
/scratch/s388143/gamod/Assignment/Data_for_Assignment/HS7_R1.fastq.gz.cor.single.fq
/scratch/s388143/gamod/Assignment/Data_for_Assignment/HS7_R2.fastq.gz.cor.single.fq

                                  echo "Call pitchfork"

                                  singularity exec /scratch/s388143/gamod/gamod.simg singularity shell
/scratch/s388143/gamod/gamod.simg source /pitchfork/setup-env.sh

                                  echo "Execution of DBG2OLC"

                                  #singularity exec /scratch/s388143/gamod/gamod.simg /DBG2OLC/compiled/DBG2OLC
k $k_param AdaptiveTh $adapt_param KmerCovTh $kmerCov_param MinOverlap $min_param RemoveChimera 1 Contigs
Contigs.txt f /scratch/s388143/gamod/Assignment/Data_for_Assignment/HS7_pacbioData_cor.fasta
```

```
                                cat Contigs.txt
/scratch/s388143/gamod/Assignment/Data_for_Assignment/HS7_pacbioData_cor.fasta > ctg_pb.fasta

                                echo "final consensus"

                                #/scratch/s388143/gamod/gamod.simg
/scratch/s388143/gamod/Assignment/Assemblies/hybrid/DBG2OLC/my_split_nrun_sparc.sh
/scratch/s388143/gamod/Assignment/Assemblies/hybrid/DBG2OLC/grid_search_tuning/"params_$k_param-$nodeCov_param-
$kmerCov_param-$min_param-$adapt_param"/backbone_raw.fasta
/scratch/s388143/gamod/Assignment/Assemblies/hybrid/DBG2OLC/grid_search_tuning/"params_$k_param-$nodeCov_param-
$kmerCov_param-$min_param-$adapt_param"/DBG2OLC_Consensus_info.txt
/scratch/s388143/gamod/Assignment/Assemblies/hybrid/DBG2OLC/grid_search_tuning/"params_$k_param-$nodeCov_param-
$kmerCov_param-$min_param-$adapt_param"/ctg_pb.fasta
/scratch/s388143/gamod/Assignment/Assemblies/hybrid/DBG2OLC/grid_search_tuning/"params_$k_param-$nodeCov_param-
$kmerCov_param-$min_param-$adapt_param"/consensusOUT 2

        done

      done

    done

  done

done
```

4) DBG2OLC with platanus short reads

Parameters: 19mer, 0.0001 AdaptiveTh, 2KmerCovTh, 10 MinOverlap, 1 RemoveChimera

---

Total number of sequences: 48

Total length of sequences: 4.797.012 bp

Shortest sequence length : 1659 bp

Longest sequence length : 536674 bp

Total number of Ns in sequences: 0

N50: 171.235 (9 sequences) (2543859 bp combined)

Parameters: 19mer, 0.0001 AdaptiveTh, 2KmerCovTh, 20 MinOverlap, 1 RemoveChimera

---

Total number of sequences: 79

Total length of sequences: 4322840 bp

Shortest sequence length : 10 bp

Longest sequence length : 291955 bp

Total number of Ns in sequences: 0

N50: 83.044 (17 sequences) (2218536 bp combined)

Parameters: 25mer, 0.0001 AdaptiveTh, 2KmerCovTh, 10 MinOverlap, 1 RemoveChimera

Total number of sequences: 55

Total length of sequences: 2.982.051 bp

Shortest sequence length : 3160 bp

Longest sequence length : 186467 bp

Total number of Ns in sequences: 0

N50: 83.436 (14 sequences) (1491911 bp combined)

Parameters: 25mer, 0.0001 AdaptiveTh, 2KmerCovTh, 20 MinOverlap, 1 RemoveChimera

Total number of sequences: 39

Total length of sequences: 1384920 bp

Shortest sequence length : 1275 bp

Longest sequence length : 128808 bp

Total number of Ns in sequences: 0

N50: 48056 (10 sequences) (718436 bp combined)