Sveriges lantbruksuniversitet
Swedish University of Agricultural Sciences

Faculty of Veterinary Medicine and Animal Science
Department of Animal Breeding and Genetics

Design of an in-silico workflow to trace the spread of zoonotic bacteria using NGS data

Adrien Janssens

Examensarbete / Swedish University of Agricultural Sciences
Department of Animal Breeding and Genetics

Master's Thesis, 30 hp

475

Uppsala 2015

Sveriges lantbruksuniversitet
Swedish University of Agricultural Sciences

Faculty of Veterinary Medicine and Animal Science
Department of Animal Breeding and Genetics

Design of an in-silico workflow to trace the spread of zoonotic bacteria using NGS data

Adrien Janssens

Supervisors:
Erik Bongcam-Rudloff, SLU, Department of Animal Breeding and Genetics
Robert Söderlund, SVA, Department of Bacteriology

Examiner:
Göran Andersson, SLU, Department of Animal Breeding and Genetics

Credits: 30 hp
Course title: Degree project in Biology
Course code: EX0578
Level: Advanced, A2E

Place of publication: Uppsala
Year of publication: 2015
Name of series: Examensarbete / Swedish University of Agricultural Sciences, Department of Animal Breeding and Genetics, 475
On-line publicering: http://epsilon.slu.se

Key words: Salmonella enterica, next generation sequencing, genotyping, GATK, single nucleotide polymorphism, traceability, clustering

LEXICON: basic abbreviations and definitions.

SNP : Single **N**ucleotide **P**olymorphism. It can be defined as a genetic variation of one single base pair between chromosomes of individuals of the same species.

Antigen : natural macromolecules which can be recognized by the immune system and induce an immune response.

Serotype : synonymous with serovar. Antigenic properties that allow the identification of a cell or virus using serologic methods.

PCR. : stand for **P**olymerase **C**hain **R**eaction, an extensively used method for DNA amplification.

Primers : short single-stranded DNA sequences used as the starting point for all lab experiments requiring extension of DNA .

Transposome : Macromolecular complex composed of a transposase (protein) and a transposon (DNA transposable elements).

Scaffold : structure encompassing several oriented and ordered contigs.

SAM : Sequence **A**lignment **M**ap

BAM : Binary **A**lignment **M**ap

Heterozygous : is used to describe a diploid organism presenting two different alleles at the same locus.

NCBI : National Center for Biotechnology information.

NGS: Next **G**eneration **S**equencing

Typhoid Fever : The most dangerous disease of all the foodborne illnesses induced by *Salmonella*. Often lethal, it has been almost eradicated in industrialized countries but continue to be a major sanitary issue in other parts of the world. Tourists from aforementioned developed countries are particularly at risks.

ABSTRACT

Tracing the spread of foodborne illness to analyze and prevent future outbreaks has become a major subject of the food-producing industry and the national food agency these past decades, and one of the main worries of the veterinarian sector. Methods for comparing bacterial strains for this purpose have generally required substantial manual effort, and have provided insufficient information. With the rise of New Generation Sequencing technologies, and the phenomenal cost decrease for complete genome sequencing, new opportunities have arisen. The possibility to completely and simultaneously sequence dozens of bacterial samples at the same time for a reasonable cost is now available, and with it the need to create tools to exploit the data produced.
In this work, we will discuss how bioinformatic tools, and in particular SNV calling and tree-building softwares, can be used to create a single-command, easy to use, fast-processing workflow using genomic sequencing data to analyze trace of bacteria from different locations and times; and how these results can be interpreted and exploited.
For this purpose, We will explain the entire analyzing process, from sample isolation to genotyping, passing by library preparation and sequencing, using two batches of samples from two different serotypes of *Salmonella enterica* that were involved in outbreaks in Sweden these past years.
One of these batch presenting a case of cross-contamination, its case will be used as a template for future analysis.

BACKGROUND

Salmonella enterica

Generalities

Salmonella enterica is one of the two species composing the *Salmonella* genus, the other being *Salmonella bongori*, for a long time classified as a subspecies of *S. enterica*, before being recognized as a full-fledged species in 2011[1].

Salmonella enterica are rod shaped, gram negative, non spore-forming and often pathogenic enterobacteria that are divided into 6 subspecies: *“Salmonella enterica* subsp. *Enterica”, “Salmonella enterica* subsp. *Arizonae”, “Salmonella enterica* subsp. *Diarizonae”, “Salmonella enterica* subsp. *Houtonea”, “Salmonella enterica* subsp. *Indica”, “Salmonella enterica* subsp. *Salamae”*.

Each of these are then subdivided into serovars using the characteristics of three surface antigens: the flagellar “H” antigen, the oligosaccharide “O” antigen and the polysaccharide “Vi” antigen.To date, 2610 different serovars of *Salmonella enterica* subsp. e*nterica* alone have been described.[2][3][4]

Out of these 2610 serovars, 3 are worth a special mention here, even while not being a part of this project:
- *Salmonella enterica* subsp. *enterica* serovar Enteriditis: according to Salmonella.org, a reference website about *salmonella*, *S.* enteriditis is the single most common cause of food poisoning in the united states at this hour.

- *Salmonella enterica* subsp. *enterica* servoar Typhy: *S.* Typhy is the pathogenic agent responsible for the deadly Typhoid fever[lexicon]. Unlike most of the other serovars of *salmonella*, this one solely infect humans, and no other host has ever been identified.
- *Salmonella enterica* subsp. *enterica* serovar Typhymurium: The *S.* Typhy of mouses. Another particularly common serotype of *salmonella* which, while being less harmful to human than *S.* Typhy, can still cause some severe cases of salmonellosis, especially in infants and elders.

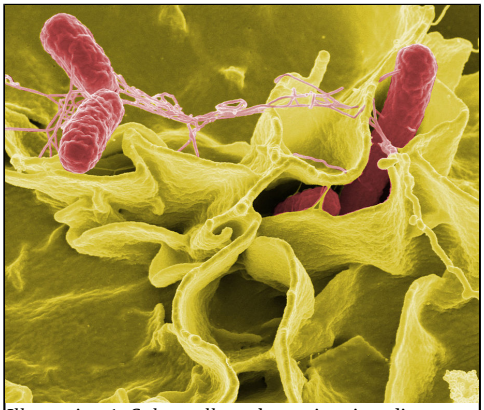


Illustration 1: Salmonella typhymurium invading human cells, picture taken from the microbewiki, originating from Rocky Mountain Laboratories, NIAID, NIH

While the vast majority of *Salmonella* strains are pathogenic, the risk induced by an infection varies widely depending on the strain and the infected host. Some serotypes are specialized on a single species, while other are generalists.

Economic and sanitary impact

According to a report of the World Health Association, Salmonellosis (the most usual disease resulting from a *Salmonella* infection) is one of the most common and widespread foodborne illness on earth, with tens of millions of people infected every year and over a hundred thousand casualties reported[5]. In addition to that, the United States department of Agriculture estimated in 2013 the yearly cost of non-typhoidal *Salmonella* illness to be around 3.7 billion dollars in the USA alone (and this statistic did not take into account the economical loss of farmers in case of animal-specific outbreaks)[6].

This project will mainly revolve around two specific strains of *Salmonella*: *S.* Dublin and *S.* Mbandaka

***Salmonella enterica* subsp. *enterica* serovar Dublin**

Salmonella Dublin rarely infects human, but these rare cases are often severe. *S.* Dublin serotype of *Salmonella* is cattle-hosted and particularly dangerous to calves, often resulting in significant economic losses to farmers.
From totally asymptomatic infection to death, response from *S.* Dublin infected individuals display a wide range of intensities and symptoms. Milk production reduction, miscarriages and the well-known gastroenteritis are also commonly observed[7].

***Salmonella enterica* subsp. *enterica* serovar Mbandaka**

The Mbandaka serotype, on the other hand, does not cause any symptoms in the carrier animals, but is highly pathogenic to human and, while not presenting any life-threatening risk to healthy adult individuals, can generate severe complications for young children, people harboring chronic immuno-deficiency and the elderly[8].

Genome sequencing

Generalities

The sequencing of a genome is a multi-step experiment during which the genetic material of an organism is isolated and then processed in order to obtain its nucleotide sequence. In order to achieve this, most sequencing technologies proceed using the same fundamental steps:

- Extraction and isolation of the genomic DNA
- Fragmentation of the DNA molecule in smaller pieces, the length of which are dependent on the sequencing technology used (current machines being unable to process the entire DNA molecule in one go)
- Depending on the technology, these fragments can be tagged with small specific sequences or not
- Amplification of the fragments
- Introduction into the sequencer and determination of the sequences of the fragments. The output of this step is one or several files containing the sequences of the fragments (from this point on, called “reads”) and information regarding the quality of the sequencing for each of those.
- Bioinformatics work, further discussed in the “genome assembly part”.

Illumina/solexa technology

The Illumina/Solexa dye sequencing technology, developed by the Solexa company, now part of the larger Illumina Inc., in the late 90‘s is a particular NGS method using the principle of sequencing by synthesis[9].

Basically the sequencer here is an amplification machine, which will use dye-marked terminal nucleotides (each type being marked with a different color) to amplify the fragments of the library one base at a time. Then it will excite the dye using a laser and record the color produced thus deducing the type of nucleotides incorporated. Afterward, it proceed by washing the remnants and remove the terminal block and the dye before starting a new cycle.[10]

It is to be noted that this kind of sequencing technology is faster and cheaper than most of the others available on the market and allow the sequencing of several samples in one run by the addition of a specific “primer bar-code” during the library preparation.

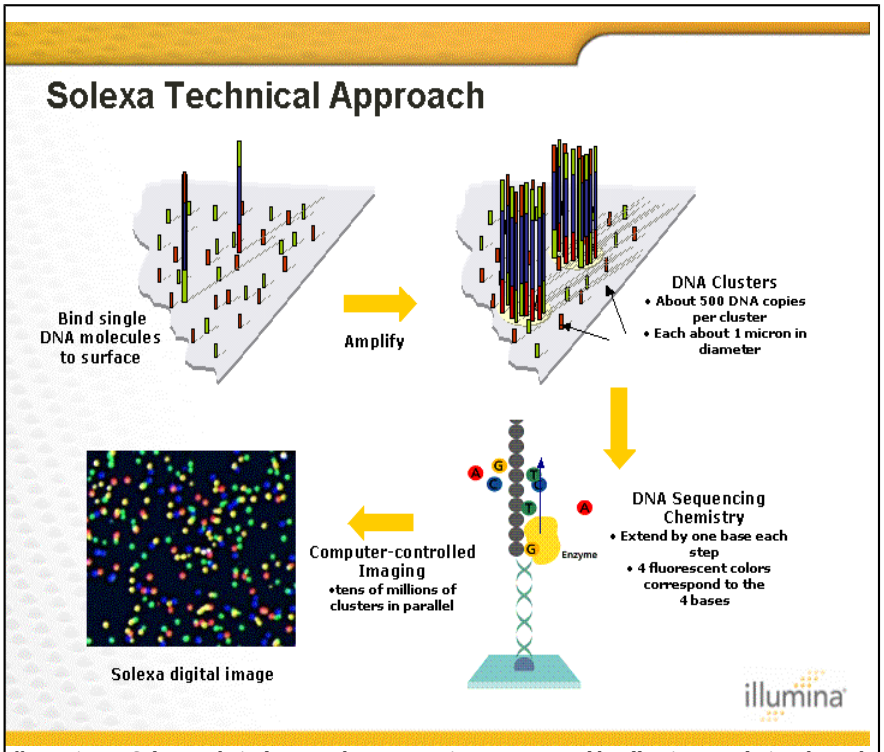


Illustration 2: Solexa technical approach to sequencing, as presented by Illumina Inc. during the 25th JPMorgan Healthcare Conference, 2007

Library preparation

Preceding the sequencing step is the library preparation of the samples. In this step, the extracted DNA is purified and fragmented using a wide variety of possible methods (enzymes, sonication, metallic beads, ...) before being amplified using a common PCR.
It is also here that the bar-coding is realized in the case of a multi-sample sequencing (multiplexing), like Illumina sequencing. The principle behind bar-coding is really simple: by using different, specific-sequence primers for each sample (which, in the case of the Illumina Nextera XT library praparation kit is possible due the an earlier incorporation of a tag sequence by specialy engineered transposomes during the fragmentation), one can easily separate the reads later by looking at these primers sequences[11].

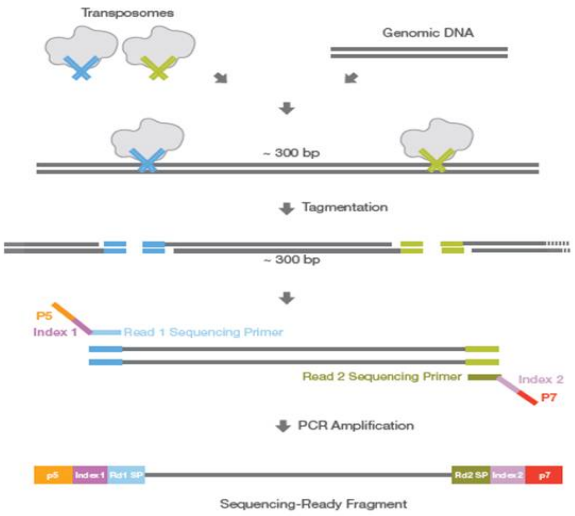


Illustration 3: Nextera XT tagmentation. Picture taken from the user manual.

Genome assembly and mapping

After the sequencer has performed its run, it will output short sequences called reads that need to be reassembled into a complete genome.

Raw reads

“Raw reads” is the name given to the output of a sequencer after its run is finished. It is usually given as files (one in the case of single end sequencing, two for paired end) containing all the information regarding the run and the data produced. While it might be possible to use these data as is (usually with the help of software developed by the same company that sell the sequencing machine); converting them to the FASTQ format is the way-to-go for future exploitation.
This format is the actual de-facto standard when it come to handling sequences with quality scores. A FASTQ entry is composed of four lines:

- The first, starting with a “@” character, contain information regarding the sequencing (from sequence name to complete description on the sequencing run during which it was created)
- The second is simply the sequence itself
- The third line start with the symbol “+” and may or may not contain a set of information in the same fashion as the first line. Its main purpose is to mark the separation between line 2 and 4.
- The fourth and last line display a succession of ASCII character, each one used to describe the quality of the associated base[12].

Quality check

Checking the quality, with the help of software like FastQC, and applying corrections if necessary is the first important step to perform in the exploitation of sequencing output. In reality, a lot of our current sequencing technologies have recurring issues (difficulty in long repetitive regions, drop of the quality at the end of reads, etc.) that need to be addressed before starting anything else.
FastQC is a small program that perform basic statistical analysis on reads and displays a report highlighting the quality (or lack of) of certain aspects of the data-set.

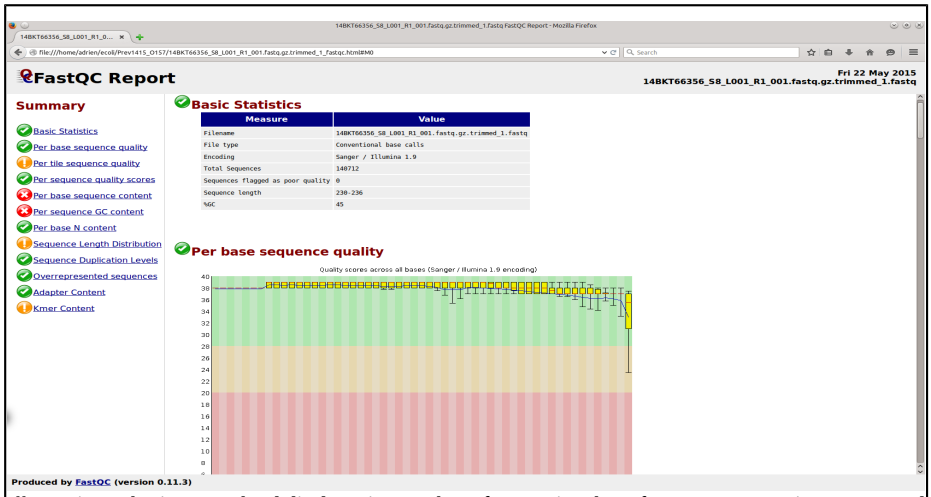


Illustration 4: basic Fastqc html display, picture taken after running the software on some Miseq-sequenced EHEC fastq reads from another project

The information provided by FastQC can then be used by Prinseq to correct and improve the quality of the data set. The two most common ways to improve the general quality of a read set are the complete removal of low-quality reads (when the entire read is compromised) and the trimming of bad-quality bases (when a general drop in quality can be seen at the beginning or end of reads).

Note that a really bad quality report is often pointing at a problem upstream (during the sequencing or even before) and that excess filtering/trimming can lead to a non-exploitable data-set.

De-novo assembly

A *de-novo* assembly is an *in-silico* operation during which the computer and the operator try to assemble the genome by solely using the reads produced by the sequencer, the reason for using this method usually being the non-existence or unreliability of previously closely-related-species genomic data. It is a really long, difficult and resource consuming operation relying on probabilistic algorithms, most commonly derivatives of the De Bruijn graph[13].
By overlapping reads possessing partial sequence identity the software (called an assembler, ex: MIRA, velvet, abyss, etc.) produces consensus regions of DNA called contigs. These contigs can be long or short, with good or bad assembly quality, but will extremely rarely encompass the entire genome: it is not a rare thing for an assembler to output thousands of contigs). The task of completing the assembly thus requires manual effort by the operator. A quantity of tools are at his disposition to continue the job, like going back to the lab and trying to extend the contigs by sequencing from their end using specifically designed primers or scaffolding them if the reads are paired-end.

Mapping

In the case when the species, or a really closely related one (like a different serovar from the same bacteria) has already been previously sequenced and assembled, one can instead perform a mapping using this previous sequence as a reference. In the same fashion as *de-novo* assembly, there is a huge

amount of software available to perform this operation, each more effective than the others in a particular field. The most commonly used are bowtie, bwa or MIRA.[14]
Performing a mapping is less demanding and faster than a *de-novo* assembly, only consisting of aligning the reads against the reference genome sequence, thus obtaining a consensus sequence that is similar to the previous one while still conserving its own specificity (SNPs). The output of a mapping is a voluminous SAM file (which can be compressed into a lighter binary BAM file), containing the entirety of the read sequences with their position on the reference.

Genotyping

The genotype can be defined as the entirety of an individuals genetic characteristics. As such, genotyping can then be considered as the discipline, in biology, which aim to determine the characteristics and identity of an individual by looking directly at its DNA sequence (and possibly comparing the results with other individuals afterward).[15]

SNP genotyping

SNP genotyping (which we will call from now on SNP typing) is a particular form of genotyping, based on variations between selected SNPs and mostly used to measure the genetic distances between individuals of a same species. This method is particularly adapted to intra-species analysis.

SNV calling

Single Nucleotide Variant calling is the name given to computational SNP typing using next generation sequencing data. This method is particularly effective and, with the constant improvement of sequencing technologies, is becoming really affordable. The principle behind SNVcalling is simple: using an already sequenced genome from the species as a reference sequence, aligning the raw reads on this sequence and listing all the SNPs in an output file. These SNPs are later filtered for quality and consistency.

It is also of note that, although whole genome sequencing may seems to be overdoing it just for SNP typing, at the difference of other genotyping methods, the data obtained can be used to perform other kind of analysis afterward, where genotyping-specific methods are limited to just that.[16]

SNP typing in bacteria

Due to the usually small size of bacterial genome, computational methods are particularly effective when it come to SNP typing. In fact, the most time consuming step in the process (except for the library preparation and the sequencing itself, of course) is the mapping of the reads against the reference sequence (and obviously the de-novo assembly, in the case when the specific strain has never been assembled before) . While this can become extremely problematic in the case of organisms like mammals or plants, bacterial genomes are usually so small that the mapping rarely exceed one hour on a simple personal computer, making it a viable option even for large amount of samples.

It should be emphasized, however, that the vast majority of bacteria only contain a single chromosome, and that this fact is not taken into consideration by the SNV calling software themselves, so a particular filtering step is usually required to dismiss heterozygous variant call.

VCF files

A VCF (Variant Call Format) file is the standard output of most SNV calling software (and in our particular case the Genome Analysis Tool Kit, abbreviated GATK). It has been created by the 1000 genome project because, after the rise of NGS, a need arose to preserve disk space when sequencing closely related samples, thus the creation of a format that does not store all the information but only the variants, compared to a reference sequence (which is not stored in the vcf itself). Tt is constituted of two parts:

- a header, containing all the basic information about the file and its creation (and the organism studied/method of sequencing if the user took the trouble to input them before starting the SNV calling process)

- a table, containing the actual informations about the SNPs

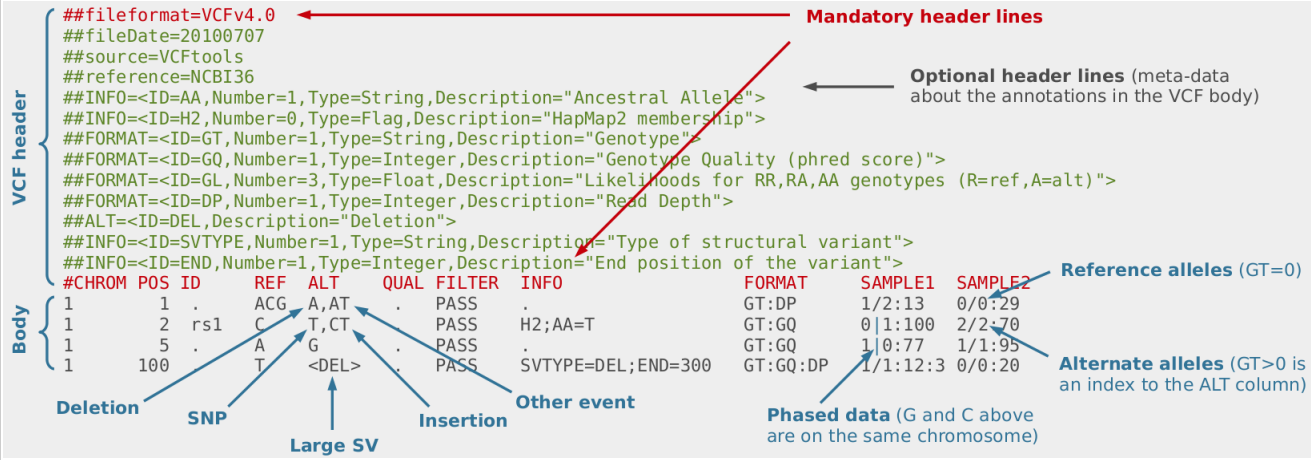


Illustration 5: Overview of the beginning of vcf file. Bioinf.comav.upv.es/courses/sequence_analysis/snp_calling.html, COMAV, universitat politecnica de valencia.

for the complete information regarding all of these, please refer to the 1000 genome wiki[17]

MATERIALS AND METHODS

Biological data

The bacterial samples of *S.* Mbandaka were provided already isolated by the National Veterinary Institute (SVA). The isolates represented primarily *S.* Mbandaka from imported animal feed raw materials such as soy products.

For *S.* Dublin, sequencing data in the form of fastq files were provided by the same institution. The samples were collected from infected cattle feces at different locations in Sweden, representing five geographic regions. The bacteria were isolated using a four successive steps method consisting of a pre-enrichment in non-selective medium (buffered pepton water), an enrichment in selective medium (rappaport-vassiliadis broth), a colony selection based on color/morphology on selective media (brilliant green agar and xylose lysine deoxycholate agar) and a PCR confirmation of the colonies. The serotypes were then determined using an serological analysis.[18]

All samples had their DNA extracted using the EZ1 DNA tissue kit on a EZ1 advanced XL machine at the SVA.

The *S.* Mbandaka libraries were prepared using the illumina Nextera XT kit, and the quality was assessed using the Agilent technology 2100 bioanalyser (summary informations will be displayed on a picture bellow and in an annex). Once diluted to a common concentration (2nM) and pooled, these libraries were then denatured, diluted and finally sequenced following the standard Illumina Nextseq 500 protocol[19], using the Nextseq 500 v2 300 cycles Mid-output kit.

The *S.* Dublin libraries were prepared using the same kit, their qualities assessed using the bioanalyzer as well, but the sequencing was done on the Illumina Miseq instead, using V2/V3 Miseq kits.

In total, reads from 3 batches of 10 samples of *S.* Dublin and one batch of 33 *S.* Mbandaka should have been produced and subsequently analyzed.

Workflow

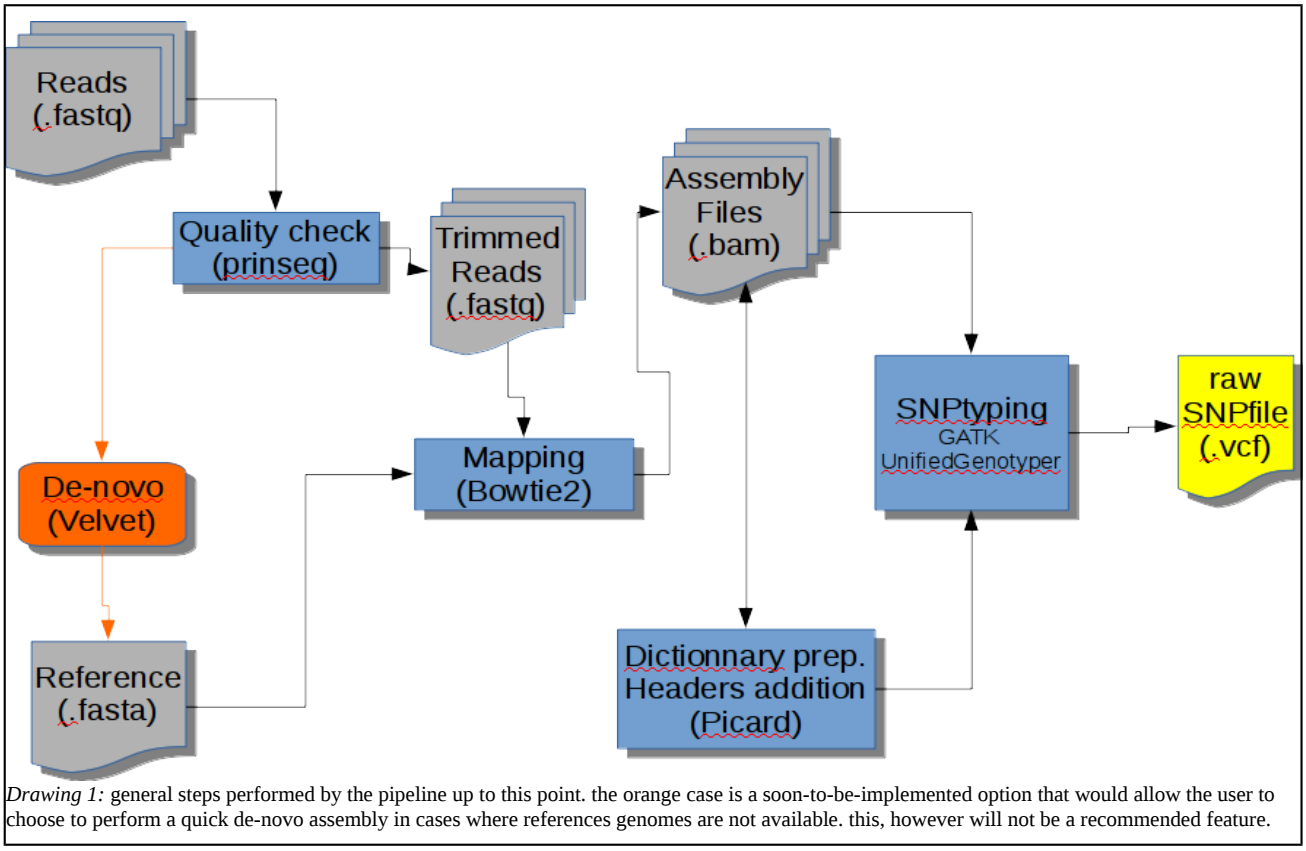
The workflow is a bash-coded script that takes a folder of gzipped paired-end fastq reads and a reference sequence as input, and outputs several BAM files, a fasta file and Variant Call Format file; together containing enough information to subsequently perform further analysis. It is specially intended to be used on multiple samples of a specific bacterial (or viral) serotype.

The workflow starts with an optional automatic quality step using the prinseq software[20] for which a certain set of hard-coded values has been assigned. These values were decided and should allow the output of decent-quality read sets, by eliminating the most common quality issues (see code in annex). In order to minimize the resources consumption here, the original raw reads data are deleted and only the filtered ones are kept for future analysis.

Following this, the pipeline will then proceed to perform a mapping of each of the read sets against the reference sequence using the bowtie2 assembler (slightly faster than its main rival bwa when it comes

to the mapping of small genomes)[21]. The resulting SAM files are then directly converted to the BAM format with the help of SAMtools[22] in order, once again, to limit the usage of disk space to a minimum. For this particular project, reference genomes were already available online. The *S.* Dublin genome was found on the NCBI web site (*Salmonella enterica* subsp. *enterica* serovar Dublin str. CT_02021853, accession number CP001144.1). A complete *S.* Mbandaka genome was also found, but as explained later, it was not used in the end.

After that, The Genome Analysis Toolkit (GATK)[23][24][25], and more particularly the UnifiedGenotyper software, is used to perform the genotyping on the assembly files. This step is the longest and the most resource consuming with the mapping and output a raw vcf files containing all the SNP discovered, regardless of their qualities. When you have a really high/low amount of SNPs compared to the "mean" SNP/samples, it can indicate a contamination or the presence of another serotype. For this reason, a little file is created just after the initial typing, using the vcf-tools[26] software vcf-stats, and displaying some basic statistics about the set.



Drawing 1: general steps performed by the pipeline up to this point. the orange case is a soon-to-be-implemented option that would allow the user to choose to perform a quick de-novo assembly in cases where references genomes are not available. this, however will not be a recommended feature.

What follow is a multiple step filtration of the SNPs, the objective being the removal false positive SNP calls. For that, two main tools are used: the GATK genotypeVariantFiltration, and grep. The first thing to be filtered out are SNPs that have been classified as heterozygous variants, since bacteria assayed here only possess a single chromosome. We then apply the GATK recommended filters for SNP typing (cf. code). The last step is the removal of all SNPs where no information is known in at least one sample: we want our markers to be universally present in the genome of each sample. For that, a simple grep -v "\/\.\/" is performed (the expression ./. being only present in a line when a simple does not have any information for that particular position). the final, filtered vcf file is then ready to be used. In the same fashion as for the raw vcf, vcf-stats is again used to get information about the file.

The final step, the one converting the vcf into a series of fasta entries soon-to-be submitted as input into the phylogenic-tree building software Splitstree[27], uses awk as its principal extraction tool (again, cf. code in annex).

For future statistical purpose (cf. chapter results hereafter), I need to mention that the analysis of the *salmonella* set was performed using my personal computer presenting the following characteristics:

- processor i7 second generation presenting 4 physical and 4 virtual cores cadenced at 2.0 GHz
- 6gb of RAM (DDRIII)
- 250 Go SSD, transfer speed of 6 Go/sec.

RESULTS

S. Mbandaka

The libraries were properly produced, and their concentrations and other information from the bioanalyzer can be obtained in the following table.

sample	Primer 1 (orange cap)	Primer 2 (white cap)	concentration (pg/µl)	molarity (nmol/l)	average size	
12-bkt000441	701	s503	3,8	9,6	1013	
12-bkt013432	702	s503	3,6	9;7	948	
12-fod003735	703	s503	2,8	5,9	1087	
12-fod004846	704	s503	3,45	7,4	1131	
12-bkt031496	705	s503	2,4	5,5	1108	
12-bkt070642	701	s504	1,6	3,9	1130	
13-bkt013495	702	s504	2,4	5,17	1141	Plate 1 (S I)
13-bkt016628	703	s504	3,2	7,5	1094	Plate 2 (S II)
13-bkt039936	704	s504	2,0	4,6	1157	Plate 3 (S III)
13-bkt097140	705	s504	2,6	6,27	1078	failure
11-fod012953	712	s506	1,7	4,28	1021	
11-bkt074210	709	s506	1,1	2,6	1053	
11-bkt035358	710	s506	1,88	4,67	1077	
11-fod005895	711	s506	1,9	4,28	1117	
11-fod005638	712	s506	1,74	3,85	1118	
11-bkt011025	712	s507	5,0	9,19	1453	
11-fod000160	709	s507	3,48	6,33	1274	
10-fod014354	710	s507	3,89	7,3	1347	
10-fod014202	711	s507	3,67	5,9	1485	
10-fod014127	712	s507	3,11	5,5	1423	
10-bkt077093	709	s501	3,73	7,59	1206	
10-fod009760	710	s501	1,79	3,65	1299	
10-fod009615	711	s501	2,68	5,0	1409	
10-fod001458	712	s501	2,97	5,0	1512	
10-fod000284	709	s502	3,43	6,3	1351	
09-bkt083408	710	s502	4,46	7,34	1423	
09-fod003872	711	s502	3,28	5,76	1403	
09-bkt024872	712	s502	4,26	7,73	1395	
09-fod001590	709	s505	3,29	5,93	1325	
09-fod001301	710	s505	4,99	8,33	1394	
08-bkt059122	711	s505	33,94	76,9	724	
08-bkt044928	712	s505	3,42	6,35	1363	
14-bkt050815	709	s508	1,97	4,36	1288	
14-bkt076764	710	s508	4,36	8,88	1294	
14-bkt076765	711	s508	4,42	8,34	1403	
14-bkt076766	712	s508	3,31	6,0	1436	

Table 1: summary of the informations regarding the Mbandaka samples, Nextera XT being a double primer index kit, keeping track of the primers used for each sample is particularly important.

However, the sequencing of the samples failed, during the first cycle of the run, right after the clustering phase, the log file indicating that the cameras couldn't detect a single cluster. This issue will be further addressed in the discussion section.

S. Dublin

The pipeline was successfully applied to the *S.* Dublin reads, properly creating the intended files. The raw vcf file comported a total of 71159 SNPs. The list of samples and related snp_count extracted from the vcf-stats with “grep” output can be found in the following table.

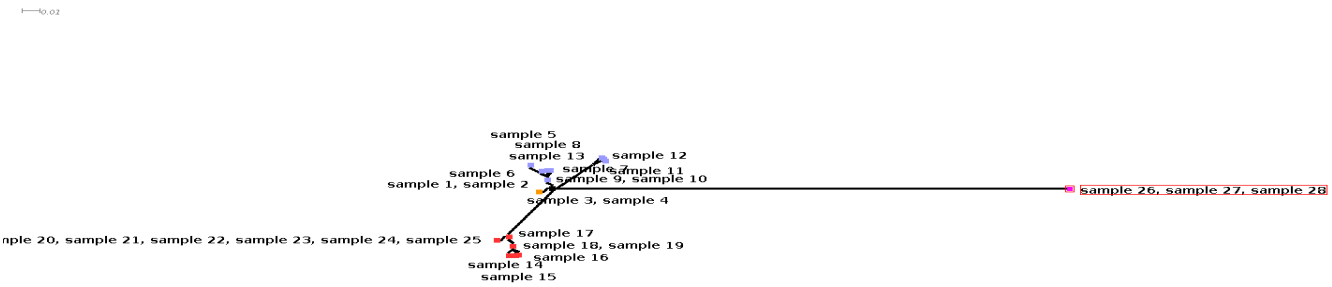
'samples'			
	'sample.data/66343o09_S3_L001_R1_001.fastq.bam'	{'snp_count'	2045,
	'sample.data/8734o14_S6_L001_R1_001.fastq.bam'	{'snp_count'	2201,
	'sample.data/27598o14_S4_L001_R1_001.fastq.bam'	{'snp_count'	1864,
	'sample.data/23034o14_S5_L001_R1_001.fastq.bam'	{'snp_count'	1546,
	'sample.data/08BKT48203_S6_L001_R1_001.fastq.bam'	{'snp_count'	1618,
	'sample.data/5257o14_S4_L001_R1_001.fastq.bam'	{'snp_count'	1830,
	'sample.data/53882o09_S1_L001_R1_001.fastq.bam'	{'snp_count'	2097,
	'sample.data/12BKT65575_S2_L001_R1_001.fastq.bam'	{'snp_count'	1867,
	'sample.data/76808o08_S8_L001_R1_001.fastq.bam'	{'snp_count'	1569,
	'sample.data/11BKT65504_S3_L001_R1_001.fastq.bam'	{'snp_count'	1497,
	'sample.data/09BKT36077_S5_L001_R1_001.fastq.bam'	{'snp_count'	1730,
	'sample.data/380o06_S8_L001_R1_001.fastq.bam'	{'snp_count'	45165,
	'sample.data/77890o09_S9_L001_R1_001.fastq.bam'	{'snp_count'	1710,
	'sample.data/40994o09_S10_L001_R1_001.fastq.bam'	{'snp_count'	1491,
	'sample.data/12BKT69449_S1_L001_R1_001.fastq.bam'	{'snp_count'	1721,
	'sample.data/13BKT92147_S7_L001_R1_001.fastq.bam'	{'snp_count'	1590,
	'sample.data/11756o14_S3_L001_R1_001.fastq.bam'	{'snp_count'	1838,
	'sample.data/13BKT92458_S9_L001_R1_001.fastq.bam'	{'snp_count'	1687,
	'sample.data/38205o08_S1_L001_R1_001.fastq.bam'	{'snp_count'	1662,
	'sample.data/51452_S6_L001_R1_001.fastq.bam'	{'snp_count'	1797,
	'sample.data/13BKT78060_S10_L001_R1_001.fastq.bam'	{'snp_count'	1703,
	'sample.data/11BKT8593_S4_L001_R1_001.fastq.bam'	{'snp_count'	1812,
	'sample.data/377o06_S5_L001_R1_001.fastq.bam'	{'snp_count'	45544,
	'sample.data/36322o08_S2_L001_R1_001.fastq.bam'	{'snp_count'	1584,
	'sample.data/23454o14_S2_L001_R1_001.fastq.bam'	{'snp_count'	1874,
	'sample.data/35544o09_S10_L001_R1_001.fastq.bam'	{'snp_count'	1936,
	'sample.data/35446o08_S7_L001_R1_001.fastq.bam'	{'snp_count'	2022,
	'sample.data/8125o11_S7_L001_R1_001.fastq.bam'	{'snp_count'	2165,
	'sample.data/13BKT83727_S8_L001_R1_001.fastq.bam'	{'snp_count'	1843,
	'sample.data/35000o09_S9_L001_R1_001.fastq.bam'	{'snp_count'	1898,
'total_snp_count'			71159,

Table 2: S. Dublin samples and the count of their unfiltered snps. The highligted lines indicates the two unusual samples.

As can be seen, two samples (380o06 and 377o06) had a really notable difference in their SNP count compared to the others. They were subsequently analyzed using the SeqSero online tool[28], and were determined

as serotype Virchow and not Dublin. All the resulting files were then discarded, and the pipeline was launched once more.

The resulting vcf was way smaller, reaching a total of 5179 SNPs unfiltered, and a small 135 SNPs once every filter had been applied. Once extracted, the tree was built and the samples clustered (simple neighbor net, equal angles, no correction applied):



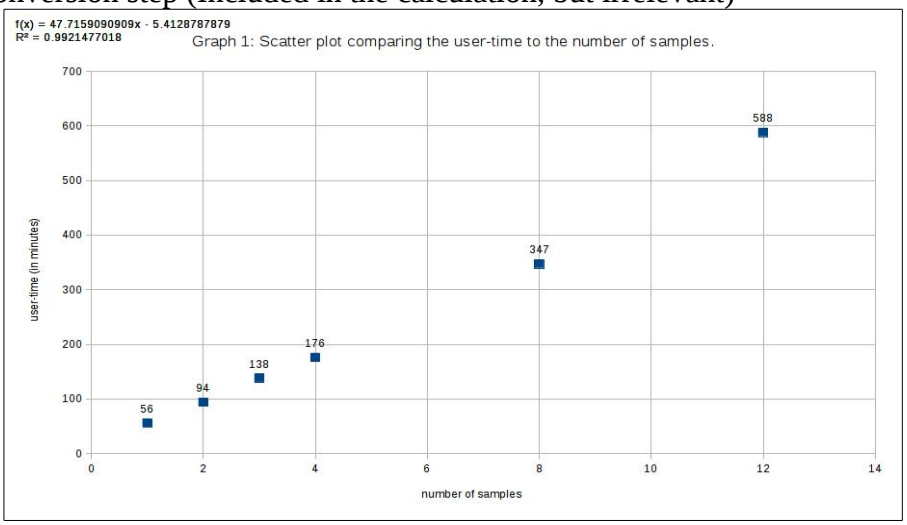
Tree 1: cluster-tree output by Splitstree. Each color represent a certain region, corresponding to the place the sample was collected. The designations of the samples are shown adjacent to each specific node. In the cases where multiple samples have the exact same genotype, they are placed one after another and separated by a comma (e.g.: sample 20 and sample 25).

Letter codes should be used to identify the region of origin of the samples, but to keep it anonymous, we will use a color-based identification instead. Red, orange, black, gray and blue dots represent the different locations.

Workflow performances

The workflow is divided into 6 principal parts (clearly delimited by ### in the code, allowing them to be easily run independently from each other), for which the total processing time has been calculated.

- Quality step (not included in the calculation. This step is time consuming but optional).
- Mapping step
- Picards and genotyping preparation step
- SNP typing step
- SNP filtration step (Included in the calculation, but irrelevant)
- Fasta conversion step (Included in the calculation, but irrelevant)



DISCUSSION

S. Mbandaka

An error during the sequencing run totally removed any hope to perform proper analysis on these strains for this thesis (there will be future attempt). As contacting Illumina did not give any decisive answers on what did happen, the only thing that could be done at this point was to design experiments in order to isolate the exact step that induced the failure, so that it would not append again in the future.

I saw at the time four “critical” possibilities:

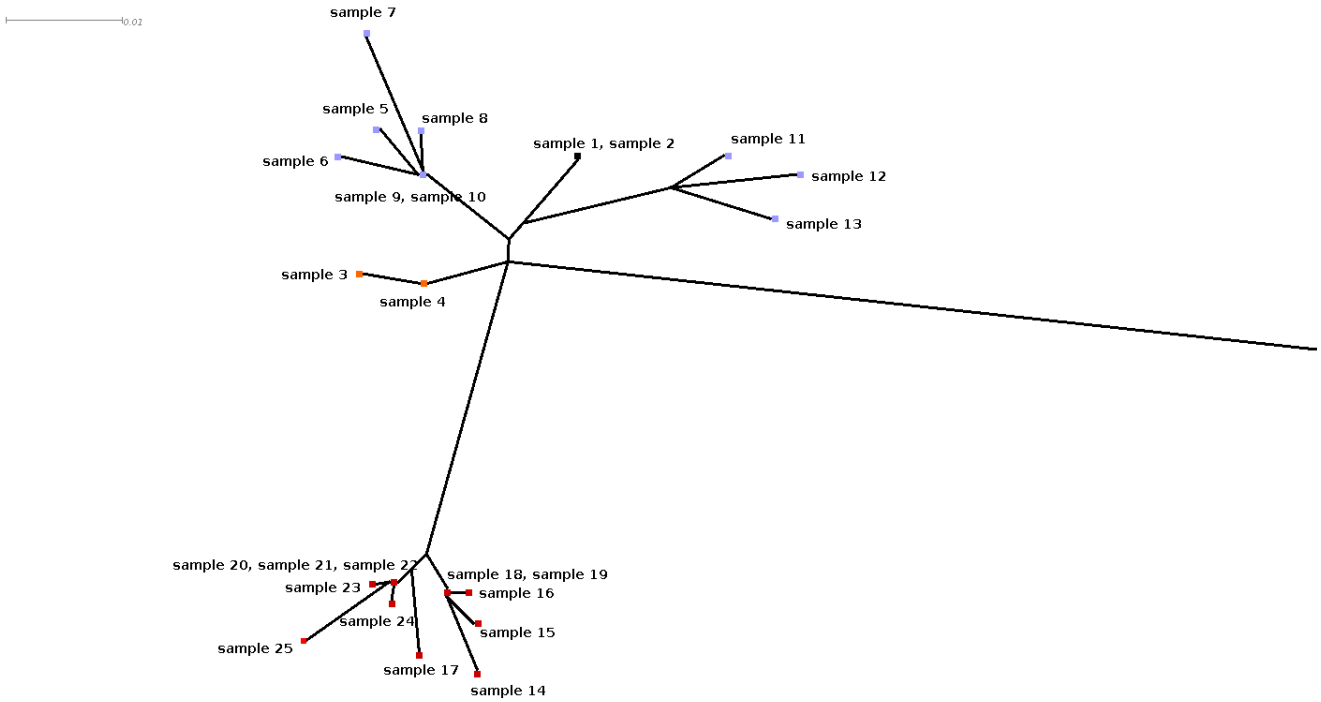
- The concentrations of the initial libraries were not properly calculated by the bioanalyzer OR were drastically diminished by their time in the freezer. This can easily be verified by recalculating the concentrations of the leftover of the libraries using another method, like the Qubit.
- The normalization of the libraries to 2nM was not done properly and the pooled libraries concentrations were way higher/lower than anticipated. Related to the first one, the leftover of these normalized libraries can also be quantified using the previous method.
- The denaturation and dilution of the pooled libraries were improperly done. This one is quite difficult to ascertain, as the protocol was followed to the letter and all the chemicals employed were freshly prepared/acquired.
- The problem is mechanical and comes from the kit or the sequencer itself and not from the previous steps. This would be likely if the testing of the other three possibilities are not concluding (A machine diagnostic will also be performed, as per Illumina customer support suggestion).

At the beginning, the first assumption seemed to be the correct one: After recalculating the concentrations of some of the libraries using the QuBit instead of the Bioanalyzer, it appeared that these concentrations were actually way higher than what was expected: from 2 to 10 times. That could have provoked an insufficient dilution during the next phase of the sequencing preparation and then induced an overload of the machine: The input material concentration being significantly higher than recommended, it is absolutely possible that the cameras could not focus at all.

However, a machine check-up realized just afterward also highlighted a mechanical issue from the sequencer itself, rendering our first assumption doubtful. Illumina then decided to send us some engineers and a new Nextseq 500 v2 kit, that we will use to retry the sequencing, this time using the QuBit values for our dilutions.

S. Dublin

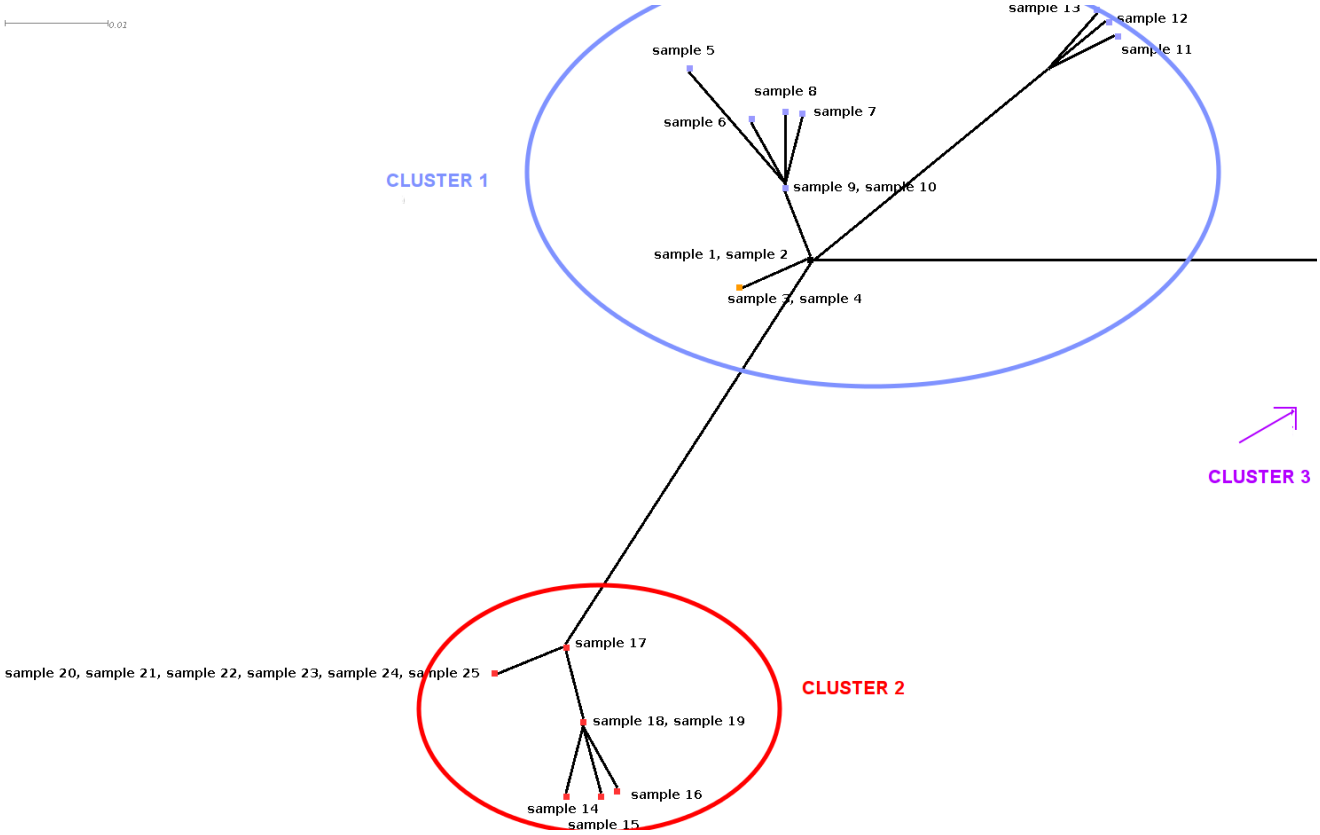
As a comparison to prove the reliability of the new pipeline, the *S.* Dublin data were also processed using the old workflow (based on MIRA[29] and MUMmer[30]), and the resulting was the following tree:



Tree 2: Zoom on the highly dense region of the MUMmer-based genotyping tree. 3 samples are too far away from this region to be displayed here. They correspond to the 3 samples at the extreme right of Tree 1.

As can be seen, the two trees are significantly similar, the major difference being the lengths of the terminal branches (shorter in the GATK-based pipeline), but as the requirement for this output is simple clustering, and not true phylogenetic analysis, this does not impact the results in a major way. This differences are probably coming from an excessive filtering from the GATK-based pipeline, or the opposite from the MUMmer based pipeline.
The trees were compared to the epidemiological background and were judged consistent. I would, however, recommend the use of a few more “already-completed” dataset to definitely assess the efficiency of the new workflow.

To be more manageable in our further analysis we will, instead of the full tree, use a zoom on the high density left part of it.



Tree 3: zoom in on the higher density left-part of the tree. In this picture, two clearly distinctive genotype clusters can be identified

As can be seen in the previous picture, 2 distinctive genotype clusters are observed (the third one is too far away to be displayed correctly here, but its analysis is exactly similar to the one of cluster 2).

Let us examine the simplest first: cluster 2. This cluster is composed of 12 extremely-closely-related samples, the largest distance between 2 samples being 3 SNPs. All of them were isolated in the same farm (red color). This kind of organization is characteristic of a purely regional outbreak, which probably happened due to local factors, such as a contaminated local feed supply or equipment.

On the other hand, cluster 1 displays samples from 3 different regions (orange, black and blue), indicating a more global outbreak. This greatly hint the existence of a common source for the pathogen spreading. A plausible scenario would be, for example, that blue supplied some of its feed to orange and black; and that the feed in question was contaminated. As *Salmonella* take some time to display symptoms, this would have probably passed as 3 different outbreaks without this analysis. The cause of this bacterial spreading cannot, obviously, be directly identified with just this tree; but combining it with different other data, such as the date when the samplings were done and the traces of exchanges between the farms at that given time could quickly isolate the source of the issue.

Hopefully, this tool will be able to allow veterinarians to quickly identify the origin of an outbreak.

Pipeline

The first thing that should be mentioned is that, even if an automatic quality control step is performed by the pipeline, I would recommend to entirely drop this option when analyzing data and to perform the quality check and possible filtration manually. Indeed, when it comes to sequencing, no data set is ever the same as another, and streamlining a quality software to act automatically on any kind of data is a task that would require an entire project on its own and is completely outside the scope of this project.

As the previous graphic indicate, the workflow process time follows a simple linear progression directly dependent on the number of samples analyzed. However, a considerable amount of factors can influence the initial processing time, the most obvious and important ones being the size of the reference genome and the number of Fastq files and their respective sizes. In a similar fashion, a wrong choice of reference genome (like using a different serovar) might still output usable data but can, and will, make the preparation and SNP typing steps time explode.

By taking a closer look at the code, we can easily deduce that the steps that consumes the vast majority of the time are the mapping and genotyping (preparation included), and the others are almost irrelevant:

- The quality step is a time consuming operation, but was not taken into consideration when calculating the computational time for it is an optional step and the processing time is entirely dependent on a huge amount of non-controllable factors, such as individual reads quality.
- The filtrations, file conversion and tree-building steps are simple file manipulations and thus their execution times ranges from a few seconds to a minute, making them completely anecdotic compared to the other steps.

Thus, one thing that could significantly improve the performance of the pipeline would be to multi-process the quality and mapping loops. Indeed, if the genotyping steps requires the samples to be processed one after another (and thus cannot really be engineered to work faster), it is not the case for these two steps, meaning that with a powerful enough CPU, and a large amount of RAM, the running time for those can easily be divided by 3 or more. It is, however, particularly difficult thing to achieve, as a bash script can easily be multi-treaded, but multi-processing a single loop is a complete different story.

The use of vcf-stats just after the creation of the raw vcf file might seem trivial and unhelpful, but as seen in the results part of this thesis, performing it can highlight the presence of some unexpected results that would not be visible once the filtration steps are done. Thus, it is of primary importance to always take a look at the basic statistics file created at this point.

The pipeline is currently working properly, and give the intended results. It is, however actually really simple and hard-coded. The most important critical improvement must focus on elasticity and user interactions, as the input from the users are, for now, particularly limited without going directly into the code. This is a really crucial point that will be fully addressed in the few days following the delivery of this thesis, as people without much command line knowledge should be able to use it properly without requesting the help of a bioinformatician.

After that, the next upgrade should be the addition of a few major SNP related options, like the possibility to run SnpEFF[31], a software designed to predict the consequences of the nucleotide

polymorphism at their given locations (This feature will soon be implemented, but is not a part of this thesis).

Lastly, other softwares, unrelated to direct SNP typing can, and probably will, be added as well. As mentioned in the background part, having genomic data at our disposal allow us far more secondary analysis than with conventional genotyping. I am particularly thinking about adding automatic CRISPR -typing[32], MLST[33], and virulence factor detection[34].

CONCLUSION

Tracing the spread of bacteria is a critical issue in food safety. Until recently, biological approaches focused on extensive and expensive lab-work to find similarities between bacterial samples. With the rise of NGS, a new panel of options have appeared that will allow us to perform faster, cheaper, precise and more versatile analysis. In this project, we demonstrated one of the possible uses of sequencing data for the field of traceability of food-born illness.

We created and optimized a workflow that use a number of FastQ reads and a related reference sequence to output a series of files that can be used for a wide array of analysis, notably the building of genotype cluster-trees allowing the user to identify cross-infections amongst different regions during an outbreak.

We subsequently used this pipeline on a set of *Salmonella* Dublin data, and successfully validated our results by comparing them with previously used and approved methods.

We then developed the possibility of a cross-contamination during one of the outbreaks, by highlighting the presence of samples from different regions in the same genotype-cluster.

We finally proposed a series of software and analysis to be added to the pipeline that would provide a significant amount of additional information concerning the studied strains, related or not to pure traceability.

AKNOWLEDGMENT

I would like to thanks the following persons, that helped and supported me during my stay in Sweden, and during this thesis:

- Erik Bongcam-Rudloff, associate-professor at SLU, my supervisor, without who nothing would have been possible.

- Robert Söderlund, PhD at SLU/SVA, my co-supervisor, who proposed the project in the first place and who managed to support me during the duration of it.

- Göran Andersson, professor at SLU, my examiner, for agreeing to be the examiner for this project and for his multiple advices.

- Marta Godia and Associate-Professor Tomas Bergstrom, for their help with the nextseq sequencing, even when things were falling apart.

- Hadrien gourle, for his help with multiple IT stuffs.

- All the people of the bongcam group in general.

- David Coornaert, Maître-assistant at HEH, For the opportunity he offered me several years ago to be an Erasmus student at SLU, and more generally for making me discover the world of bioinformatics.

- My family, for their moral and financial support during my stay

- My friends, wherever they are, for their unconditional support.

REFERENCES

[1] Evolution of *salmonella* nomenclature: a critical note, Folia Microbiologia, November 2011, M. Agbaje, R.H. Begum

[2] *Salmonella*, the host and disease: a brief review, Bryan Coburn, Guntram A Grass, B B Finlay

[3] Pathogen safety data sheet, *salmonella enterica* spp., public health agency of canada

[4] antigenic Formulae of the *salmonella* serovars, WHO and Institut Pasteur, 2007, 9th edition, Patrick A.D. Grimont & François-Xavier Weill

[5] W.H.O. fact sheet N°139

[6] Economic Research Service (ERS), U.S. Department of Agriculture (USDA). Cost Estimates of Foodborne Illnesses. http://ers.usda.gov/data-products/cost-estimates-of-foodborne-illnesses.aspx

[7] Review of pathogenesis and diagnostic methods of immediate relevance for epidemiology and control of *Salmonella* Dublin in cattle, Liza Rosenbaum Nielsen

[8] Multistate Outbreak of *Salmonella* Montevideo and *Salmonella* Mbandaka Infections Linked to Tahini Sesame Paste, Signs & symptoms, CDC website

[9] http://www.illumina.com/technology/next-generation-sequencing/sequencing-technology.html

[10] A Very Large Scale, High Throughput and Low CostDNA Sequencing Method based on a New2-Dimensional DNA Auto-Patterning Process P. Mayer, (L. Farinelli), G. Matton, C. Adessi, G. Turcatti, J.J. Mermod, E. Kawashima. Genomic Technology Department Serono Pharmaceutical Research Institute 28/10/98 Mayer et al

[11] http://support.illumina.com/content/dam/illumina-support/documents/documentation/chemistry_documentation/samplepreps_nextera/nextera-xt/nextera-xt-library-prep-guide-15031942-e.pdf, page 29

[12] The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants, oxford journal issue 38

[13] Velvet: algorithms for de novo short read assembly using de Bruijn graphs, Daniel R Zerbino & Ewan Birney, Genome research 18, 821-829 (2008)

[14] How to map billions of short reads onto genomes, Cole Trapnell & Steven L Salzberg, Nature biotechnology 27, 455 – 457 (2009)

[15] GeneReviews, Pagon RA, Adam MP, Ardinger HH, et al., editors. Seattle (WA):university of washington, Seattle; 1993-2015

[16] Genotype and SNP calling form next-generation sequencing data, Rasmus Nielsen, Joshua S. Paul, Anders Albrechtsen, Yun S. Sons, Nature reviews genetics 12, June 2011

[17] http://www.1000genomes.org/wiki/Analysis/Variant%20Call%20Format/vcf-variant-call-format-version-40

[18] http://www.antimicrobialresistance.dk/data/images/salmonella1_pdf.pdf

[19] http://supportres.illumina.com/documents/documentation/system_documentation/nextseq/nextseq-500-denaturing-and-diluting-libraries-15048776-a.pdf

[20] Schmieder R and Edwards R: Quality control and preprocessing of metagenomic datasets. Bioinformatics 2011, 27:863-864.

[21] Langmead B., Salzberg S.: Fast gapped-read alignment with bowtie2. Nature methods. 2012, 9:357-359

[22] Li H.*, Handsaker B.*, Wysoker A., Fennell T., Ruan J., Homer N., Marth G., Abecasis G., Durbin R. and 1000 Genome Project Data Processing Subgroup (2009) The Sequence alignment/map (SAM) format and SAMtools. Bioinformatics, 25, 2078-9

[23] The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M, DePristo MA, 2010 GENOME RESEARCH 20:1297-303

[24] A framework for variation discovery and genotyping using next-generation DNA sequencing data DePristo M, Banks E, Poplin R, Garimella K, Maguire J, Hartl C, Philippakis A, del Angel G, Rivas MA, Hanna M, McKenna A, Fennell T, Kernytsky A, Sivachenko A, Cibulskis K, Gabriel S, Altshuler D, Daly M, 2011 NATURE GENETICS 43:491-498

[25] From FastQ Data to High-Confidence Variant Calls: The Genome Analysis Toolkit Best Practices Pipeline Van der Auwera GA, Carneiro M, Hartl C, Poplin R, del Angel G, Levy-Moonshine A, Jordan T, Shakir K, Roazen D, Thibault J, Banks E, Garimella K, Altshuler D, Gabriel S, DePristo M, 2013 CURRENT PROTOCOLS IN BIOINFORMATICS 43:11.10.1-11.10.33

[26] The Variant Call Format and VCFtools, Petr Danecek, Adam Auton, Goncalo Abecasis, Cornelis A. Albers, Eric Banks, Mark A. DePristo, Robert Handsaker, Gerton Lunter, Gabor Marth, Stephen T. Sherry, Gilean McVean, Richard Durbin and 1000 Genomes Project Analysis Group, Bioinformatics, 2011

[27] D.H Huson and D. Bryant, Application of Phylogenetic Networks in Evolutionary Studies, Mol. Biol. Evol., 23(2):254-267, 2006

[28] Salmonella serotype determination utilizing high-throughput genome sequencing data. J. clin. Microbiol. May 2015, Zhan S, Yin Y, Jones MB

[29] Chevreux, B., Wetter, T. and Suhai, S. (1999): Genome Sequence Assembly Using Trace Signals and Additional Sequence Information. Computer Science and Biology: Proceedings of the German Conference on Bioinformatics (GCB) 99, pp. 45-56

[30] “Versatile and open software for comparing large genomes.” S. Kurtz, A. Phillippy, A.L. Delcher, M. Smoot, M. Shumway, C. Antonescu, and S.L. Salzberg, Genome Biology (2004), 5:R12.

[31] A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w1118; iso-2; iso-3.", Cingolani P, Platts A, Wang le L, Coon M, Nguyen T, Wang L, Land SJ, Lu X, Ruden DM. Fly (Austin). 2012 Apr-Jun;6(2):80-92. PMID: 22728672

[32] Horvath, P.; Barrangou, R. (2010). "CRISPR/Cas, the Immune System of Bacteria and Archaea". Science 327 (5962): 167–170. doi:10.1126/science.1179555

[33] Maiden, MC.; Bygraves, JA.; Feil, E.; Morelli, G.; Russell, JE.; Urwin, R.; Zhang, Q.; Zhou, J. et al. (Mar 1998). "Multilocus sequence typing: a portable approach to the identification of clones within populations of pathogenic microorganisms". Proc Natl Acad Sci U S A 95 (6): 3140–5

[34] Virulence factors and their mechanisms of action: the view from a damage-response framework; Arturo Casadevall and Liise-anne Pirofski, 2009, Journal of Water and Health.

ANNEXE: pipeline.

This code is not the streamlined one, which contain more user options, but correspond to the one that was used for the *S.* Dublin data

```
#!/bin/bash

### quality step, optional and not recommended.

if [ -z "$1" ]
then
	path='.'
else
	path=$1
fi

list=($path/*.fastq.gz)
((lenlist=${#list[@]}, num=lenlist - 2))
for (( i = 0; i <= num; i+=2)); do
	echo "unziping files ${list[i]} and ${list[i+1]}"
	zcat ${list[i]} > $path/output1.fastq
	zcat ${list[i+1]} > $path/output2.fastq
	echo "done unziping, starting prinseq"
	/home/adrien/bioinfotools/prinseq-lite-0.20.4/prinseq-lite.pl -fastq $path/output1.fastq -fastq2 $path/output2.fastq -out_bad null
-out_good "${list[i]}.trimmed"  -min_qual_mean 28 -trim_to_len 290 -min_len 230 -trim_left 15
	rm $path/*singletons.fastq
	echo "done"
	rm $path/output1.fastq $path/output2.fastq ${list[i]} ${list[i+1]}
done


### mapping and samtobam steps .... can't do much about this one.

trimmedlist=($path/*.fastq)
echo "indexing reference"
bowtie2-build referencesequence.fasta $path/indexref
samtools faidx referencesequence.fasta
((trimlenlist=${#trimmedlist[@]}, trimnum=trimlenlist - 2))
for ((i = 0; i <= trimnum; i+=2)); do
	echo "mapping ${trimmedlist[i]} and ${trimmedlist[i+1]}"
	bowtie2 -x $path/indexref -1 ${trimmedlist[i]} -2 ${trimmedlist[i+1]} -S ${trimmedlist[i]}.sam
	echo "creating bam file"
	samtools view -hSb ${trimmedlist[i]}.sam >  ${trimmedlist[i]}.bam
	rm ${trimmedlist[i]}.sam
	echo "done"
done

###Picards and GATK preparation steps.

java -jar /home/adrien/bioinfotools/broadinstitute-picard-b2a94f7/dist/picard.jar CreateSequenceDictionary
R=./referencesequence.fasta O=./referencesequence.dict
echo "dictionnary created, launching preparations for snptyping"
assemblies=($path/*.bam)
for i in ${assemblies[@]}
do
	java -jar /home/adrien/bioinfotools/broadinstitute-picard-b2a94f7/dist/picard.jar AddOrReplaceReadGroups I=$i
O=$i.sorted.bam SORT_ORDER=coordinate ID=$i LB=S.Dublin PU=flowcell PL=illumina SM=sample.$i CREATE_INDEX=True
VALIDATION_STRINGENCY=LENIENT
	java -Xmx4g -jar GenomeAnalysisTK.jar -T RealignerTargetCreator -R referencesequence.fasta -I $i.sorted.bam -o
$i.realign.intervals
	java -Xmx4g -jar GenomeAnalysisTK.jar -T IndelRealigner -R referencesequence.fasta -I $i.sorted.bam -targetIntervals
$i.realign.intervals -o $i.realigned.bam

done

echo "preparation finished, starting snptyping"
```

```
###SNPtyping

finalocation=($path/*.realigned.bam)
variable=`for i in ${finalocation[@]}
do
	echo "-I $i"
done`

java -jar GenomeAnalysisTK.jar -T UnifiedGenotyper -R referencesequence.fasta $variable -o $path/SNPs.raw.vcf -stand_call_conf
50.0 -stand_emit_conf 10.0 -dcov 500

###Filtering

cat $path/SNPs.raw.vcf |grep -v "LowQual" > firstfiltration.vcf
java -jar GenomeAnalysisTK.jar -T VariantFiltration -R referencesequence.fasta -V firstfiltration.vcf -o hetfilter.vcf
--genotypeFilterExpression "isHet==1" --genotypeFilterName "LowQual"
rm firstfiltration.vcf
cat hetfilter.vcf| grep -v "LowQual" > nohet.vcf
rm hetfilter.vcf
java -jar GenomeAnalysisTK.jar -T VariantFiltration -R referencesequence.fasta -V nohet.vcf -o secondfilter.vcf --filterExpression
"QD<2.0 || MQ<40.0 || FS>60.0 || Haplotypescore>13.0 || MQRankSum<-12.5 || ReadPosRankSum<-8.0" --filterName "LowQual"
cat secondfilter.vcf| grep -v "LowQual" > beforelast.vcf
rm nohet.vcf secondfilter.vcf
cat beforelast.vcf| grep -v "\.\/\." > filteredSNP.vcf
rm beforelast.vcf

### conversion to fasta.

for ((i=10; i<=38; i+=1)); do
	j="$"$i
	filename="$i.raw.csv"
	awk -v name="$filename" -F"\t" '{if("$j" ~ /^1/) print $2,$5 >> name; else if ("$j" ~ /^\.\//) print $2,$4>> name; else print
$2,$4 >> name}' filteredSNP.vcf
	printf ">" >> SNP.fasta
	awk 'FNR == 32 {print $"$i"}' test.vcf | cut -f2 -d"/" | cut -f1 -d"_" >> SNP.fasta
	awk '{ORS=""; if ($2 !~ /^\.\// && $2 !~ /^[A-Z]\,+/) print $2>>"SNP.fasta"}' "$filename"
	echo "" >> SNP.fasta
done

sed -e 's:REF::g' SNP.fasta > final.fasta
tr '[:upper:]' '[:lower:]' < final.fasta > readytouse.fasta
rm SNP.fasta
rm final.fasta
rm *.csv
```