OH SFS Handbook

Release v0.0.1

Sep 10, 2021

Main page

CHAPTER 1

About the OH SFS Handbook

The work found in these pages got its start in the ORION project and continued in the frame of the BeOne project. The ORION project, launched in 2018, is aimed at establishing and strengthening inter-institutional collaboration and transdisciplinary knowledge transfer in the area of surveillance data integration and interpretation, along the One Health (OH) objective of improving health and well-being. The BeOne project, launched in 2020, aims at developing an integrated surveillance dashboard in which molecular and epidemiological data for foodborne pathogens can be interactively analysed, visualised, and interpreted by the relevant experts across disciplines and sectors.

Through three main work packages (WP), **ORION's specific goals** can be summarized as the delivery of three main resources:

- a "OH Surveillance Codex" (WP1) - a high level framework for harmonised, cross-sectional description and categorisation of surveillance data covering all surveillance phases and all knowledge types;
- a "OHS Knowledge Hub" (WP2) - a cross-domain inventory of currently available data sources, methods / algorithms / tools, that support OH surveillance data generation, data analysis, modelling and decision support;
- "OHS Infrastructural Resources" (WP3) – that are practical, infrastructural resources forming the basis for successful harmonisation and integration of surveillance data and methods.

Through its WP1, the **BeOne project** targets the typing and nomenclature issues that exist within WGS-based pathogen surveillance and outbreak detection. The goals of this WP can be summarized as:

- establishing the current state of the art within genomics methods for WGS-based typing;
- providing a strain dataset to capture the genomic diversity within the populations of four main pathogens: *Salmonella enterica, Escherichia coli (STEC), Listeria monocytogenes* and *Campylobacter jejuni*;
- assessing cluster agreement between different WGS-based typing approaches to ensure the comparability between distinct methodologies to reinforce and promote a global surveillance and control of infectious diseases.

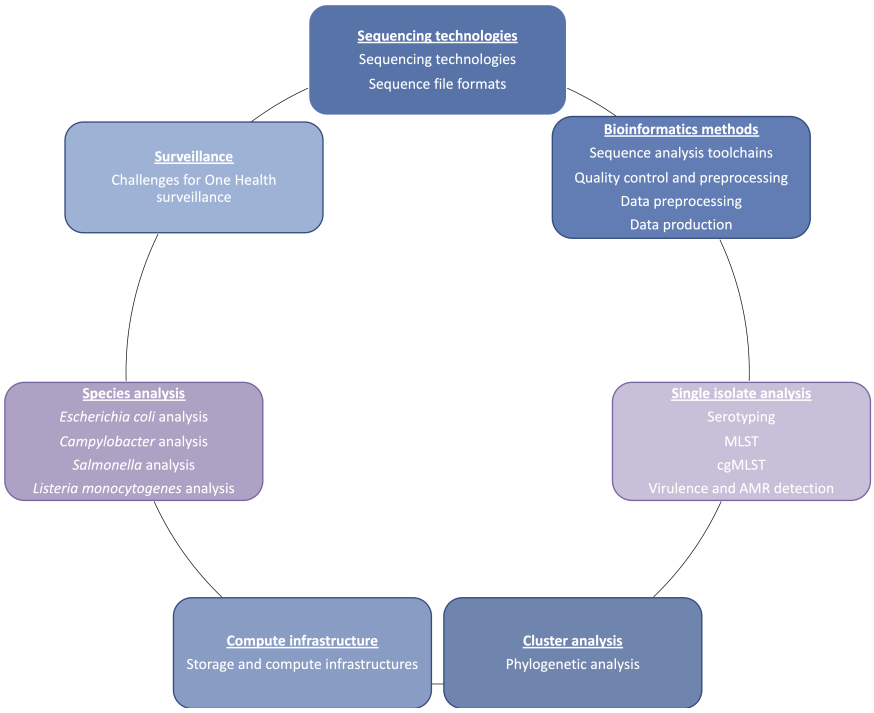
The work in these pages springs out ORION's WP2 and BeOne's WP1 and focuses on being an inventory over current practices regarding the use of sequencing data for surveillance purposes, with a special focus on the methodologies used foodborne diseases and the current state of One Health surveillance.

1.1 Updates and contributions

The underlying technologies within this field is a moving target. It is thus important to keep this handbook updated with new information. Contributions to this handbook are very welcome, please see the *Contributing* document for more information.

1.2 Overview of the handbook

This handbook consists of various sections.



Sections of the handbook - Blue boxes describe species-agnostic processes, while the purple ones depend on which biological agent is being analysed.

Sequencing Technologies - The focus for this section is on describing the available sequencing technologies, highlighting their differences and consequent impact on WGS data analysis.

Bioinformatics methods - This section aims to describe the basic bioinformatics methods and tools that can be used to analyze whole genome seqencing data. This includes how to do quality control, how assembly and mapping works, and how a bioinformatics pipeline might be stitched together.

Single isolate analysis - This section explores different bacterial typing pipelines, including cg/wgMLST and SNP-based pipelines, and pipelines for virulence and antimicrobial resistance (AMR) detection.

Cluster analysis - The focus for this component is to provide an overview of the different methods which can be used to perform an integrated analysis of several samples, obtaining clustering information.

Compute infrastructure - The focus for this section is on exploring the options and the requirements for establishing possible infrastructures for using NGS methods for surveillance purposes. This section spans from storage, compute infrastructures and data management to workflow managers and currently available platforms for automated analysis.

Species analysis - This section is focused on four bacterial pathogens: *S. enterica, E. coli (STEC), L. monocytogenes* and *C. jejuni*, providing a historical overview of their respective typing methods, and exploring their specific needs and

available pipelines/platforms for WGS surveillance. Their respective state of the art regarding WGS and One Health surveillance is also reviewed.

Surveillance - This section explores the relevance of WGS for One Health surveillance, and the challenges that we are currently facing for the implementation of an international and inter-sectoral surveillance.

CHAPTER 2

License

2.1 Instructional Material

CHAPTER 3

EFSA and ECDC

Both EFSA and ECDC have done extensive work on how to implement whole genome sequencing for surveillance and outbreak purposes. This page is meant to collect reports and opinions from these institutions.

These institutions also have reports that specifically advice on specific pathogens, information on these are found in the species specific sections.

3.1 ECDC resources

- 2019 - Collection and analysis of whole genome sequencing data from food-borne pathogens and other relevant microorganisms isolated from human, animal, food, feed and food/feed environmental samples in the joint ECDC–EFSA molecular typing database
- 2019 - ECDC strategic framework for the integration of molecular and genomic typing into European surveillance and multi-country outbreak investigations
- 2018 - ECDC public health microbiology strategy 2018–2022
- 2018 - Monitoring the use of whole-genome sequencing in infectious disease surveillance in Europe 2015–2017
- 2016 - ECDC roadmap for integration of molecular typing and genomic typing into European-level surveillance and epidemic preparedness – Version 2.1, 2016-19
- 2016 - Expert opinion on whole genome sequencing for public health surveillance
- 2015 - Expert Opinion on the introduction of next-generation typing methods for food- and waterborne diseases in the EU and EEA

3.2 EFSA resources

- 2019 - EFSA and ECDC technical report on the collection and analysis of whole genome sequencing data from food-borne pathogens and other relevant microorganisms isolated from human, animal, food, feed and food/feed environmental samples in the joint ECDC-EFSA molecular typing database

- 2019 - Technical specifications on harmonised monitoring of antimicrobial resistance in zoonotic and indicator bacteria from food-producing animals and food
- 2019 - Whole genome sequencing and metagenomics for outbreak investigation, source attribution and risk assessment of food-borne microorganisms
- 2018 - INNUENDO: A cross-sectoral platform for the integration of genomics in the surveillance of food-borne pathogens
- 2018 - Final report of ENGAGE - Establishing Next Generation sequencing Ability for Genomic analysis in Europe
- 2018 - Outcome of EC/EFSA questionnaire (2016) on use of Whole Genome Sequencing (WGS) for food- and waterborne pathogens isolated from animals, food, feed and related environmental samples in EU/EFTA countries
- 2018 - Use of next-generation sequencing in microbial risk assessment
- 2014 - Scientific Opinion on the evaluation of molecular typing methods for major food-borne microbiological hazards and their use for attribution modelling, outbreak investigation and scanning surveillance: Part 2 (surveillance and data management activities)
- 2014 - Scientific Opinion on the evaluation of molecular typing methods for major food-borne microbiological hazards and their use for attribution modelling, outbreak investigation and scanning surveillance: Part 1 (evaluation of methods and applications)

CHAPTER 4

International efforts

CHAPTER 5

Sequencing Technologies

5.1 Shotgun sequencing

Most sequencing projects use shotgun sequencing of DNA, cDNA or RNA fragments to produce sequence data that can be used for various analysis types such as whole genome assembly, transcriptomics or metagenomics. To do shotgun sequencing the genetic molecule is randomly fragmented using mechanical or chemical shearing. The random fragments are then size selected and analyzed using either short (25-500 bp) or long (> 1000 bp) read sequencing platforms. The intermediate size (500-1000 bp) fragments are not used in large scale sequencing projects, but they are typical for Sanger sequencing platforms, such as the ABI3100. Such fragments are now typically used for the closing of genomes, or the sequencing of cloned PCR products.

5.2 Short read sequencing

Short read sequencing is the analysis of short DNA / cDNA fragments (25-500 bp) using second or next generation sequencing platforms. The second generation sequencing platforms are characterized by a much higher throughput than the original first generation or Sanger sequencing platforms. The dominant methods can be divided into two groups: 1) sequencing by synthesis or 2) sequencing by ligation.

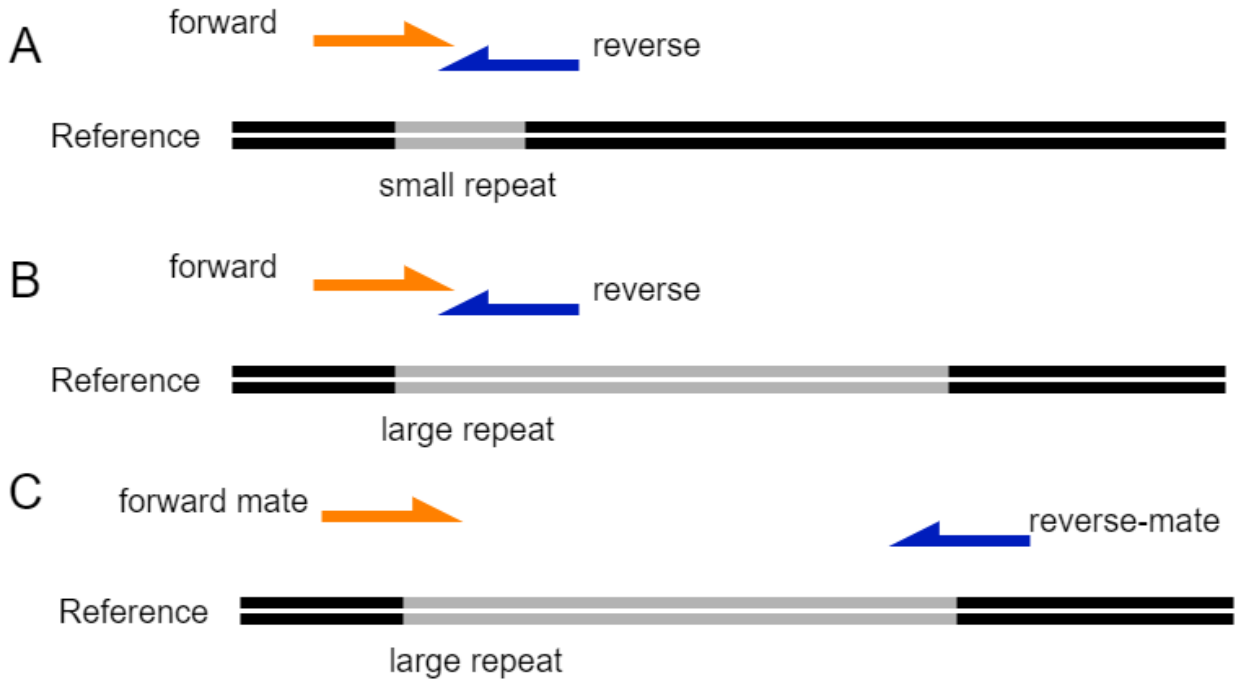
Sequencing by synthesis is performed using a polymerase that incorporates nucleotides. These nucleotides can be fluorescently labeled and then the incorporation of nucleotides is registered optically with a camera (Illumina or 454). Another detection method is by using ion sensors that monitor the release of hydrogen ions when a nucleotide is incorporated. Upon incorporation an electronic signal is produced (Ion Torrent semiconductor sequencing).

Sequencing by ligation is a technique where the sequencing reaction is performed by a ligase and not a polymerase. The ligase anneals short one or two base encoded probes to a sequence. After non-ligated probes are washed away, a camera measures which probes have been ligated and the color of the probe gives the nucleotide composition. This technique is represented by SOLID or DNA nanoball sequencing.

5.3 Paired-end sequencing

Shotgun sequencing of short DNA fragments (200-800 bp) is done by sequencing from either one end of the fragments (single-end sequencing) or from both ends of the molecule. In the later case it is called short fragment paired-end sequencing, because it generates two sequencing reads (one from each end) which correspond to the extremities of the same fragment. If the insert size (i.e. the genomic distance separating the two reads) is short enough compared to the read length, it is possible that the sequencing from both ends of the fragment generates reads that can be combined into one fragment encoding the original fragment. In this regard, it is important to note that the read size directly depends on the sequencing platform that is used. For instance, contrary to other Illumina platforms, Illumina MiSeq, one of the most used platforms for microbiology research, allows the sequencing of 2 x 300bp reads.

The advantages of paired-end sequencing over single-end sequencing are several. In the case when both reads align it is possible to correct sequencing errors in one of the reads by using a base with a higher Phred score (higher sequencing quality) located on the other read. This is widely used in amplicon sequencing. Another advantage is that paired-end sequencing improves genome assembly as the distance between reads pairs provides additional information on the relative position of the reads in the genome. In addition, it can solve structural rearrangements, such as gene deletion However, short fragment paired-end sequencing will not resolve large repeats found in most genomes such as rRNA operons as the length of those repeated regions is superior to the distance between reads. There long-read sequencing or mate-pair sequencing is needed to assist the assembly process.



This figure shows how short-read sequencing can be used to solve repetitive regions during genome assembly. A) A small repeat with a size smaller than the insert size of a paired set of reads. Here the location of the repeat is solved because both reads contain information on the repeat and on the sequence outside the repeat. B) The location of a large repeat (size larger than insert size of paired reads) can not be solved by a normal paired-end sequencing, since it is unclear to which part of the genome the reverse read belongs. In addition, there will be many reads mapping inside the repetitive regions. C) Mate-pair sequencing reads that have large insert sizes can be used to identify the correct location of a repetitive area because they match both the repeat and the area outside the repeat.

5.4 Mate-pair sequencing

In the case where the insert size of a fragment is much longer than the read length it is possible to do mate-pair sequencing. In mate-pair sequencing a library is created of large fragments (for example 10 Kbp) and only the ends of these fragments are sequenced. In order to create mate-pair sequencing libraries, the ends of the fragments are biotinylated and then the fragments are circularized by joining the biotinylated ends. The circularized fragments are then digested and the smaller biotinylated fragments are captured. These are then prepared in the same way as short fragment paired-end sequences and sequenced from both ends (see this figure). Because the original location of the mate-pair sequences was much further apart, it is possible to use mate pair sequences to bridge contig caps that were generated after paired-end sequencing. This technique is slowly becoming obsolete due to the improvements in long-read sequencing

5.5 Long read sequencing

Third generation sequencing, or "true" long read sequencing, is a method where long DNA fragments are analyzed individually (reviewed in: Amarasinghe, S.L. et al. 2020 & Mantere, T. and Hoischen, A. 2019). This is different from "synthetic" long read sequencing where synthetic long reads are produced using a variety of methods that link short read sequences using barcoded adapters, proximity ligation, or via optical mapping (section 8 in: Amarasinghe, S.L. et al. 2020). With true long read sequencing fragmented DNA does not need to be amplified as is the case for most next/second generation sequencing techniques. Long read sequencing is dominated by two companies with different methods: Oxford Nanopore sequencers (minION, gridION, promethION) and Pacific biosciences with their SMRT (Single Molecule, Real-Time) sequences (Pacbio RSII / Sequel Sequel II) (Amarasinghe, S.L. et al. 2020). These companies provide platforms that allow for true long read sequencing.

True long read sequencing as performed by the Oxford Nanopore machines depend on measuring a change in the ionic current when a base on a DNA strand is pulled through a nanopore (section 8 in: Amarasinghe, S.L. et al. 2020). The Nanopore is a protein that is embedded in an electrically-resistant polymer membrane. The Nanopore and the membrane are together integrated into a microscaffold, which is part of sensorchip. Each microscaffold is connected to its own electrode which is attached to its own channel of the sensory array chip (The ASIC) https://nanoporetech.com/how-it-works). By measuring the change in ionic current it is possible to determine what kind of a nucleotide is passing the pore. Nucleotides have different properties which causes a specific change in the ionic current, and thus allows for the detection of methylated bases as well. The read length for Nanopore sequencing is dependent on being able to load high molecular weight onto the flowcell.

The SMRT technology developed by Pacific Biosciences uses polymerases that are attached to the bottom of picoliter-sized wells. Incorporation of a nucleotide is detected in real-time via the emission of a fluorescence signal. The read length of this technique is limited by the resilience of the polymerase to stay active. In addition, the error rate with SMRT sequencing is depending on how often a DNA molecule is read by the polymerase. The SMRT sequencing therefore comes as two variants: Continuous Long Read (CLR) sequencing and Circular Consensus Sequencing (CCS). For the later the molecules are only sequenced once and this allows for very long sequences. With CCS sequencing the DNA molecules are circularized, so called circular SMRTbell DNA molecules. These molecules are read by a polymerase. With every pass of the molecule all bases are read, and since errors are randomly introduced, it is possible to identify which basecalles are incorrect. Thus the repeated reading of the same molecule allows for error correction, and improves the error rate (see figure 3 in D'Amore et al., 2016). Nonetheless, indels in homopolymers are still a problem in SMRT sequencing (Amarasinghe, S.L. et al. 2020).

The error rate with nanopore sequencing is higher than with SMRT sequencing due to the fact that not one but five bases affect the ionic current over the membrane. With SMRT sequencing a single signal is emitted for each base, while with Nanopore five bases affect the signal. Indels and substitutions are partly randomly distributed but not in a uniform manner (Amarasinghe, S.L. et al. 2020). The error rate in Nanopore sequencing is dependent on a uniform translocation speed along the pore, which can be affected by a variety of factors such as temperature, modified bases, 3D-structure of the DNA, etc. In addition, the structural and chemical characteristics of the pore play a role in error

rate. Therefore a lot of development goes into improving this by developing basecalling algorithms that can identify the correct base even though the signal from the nanopore sequencer is very noisy.

5.6 Short vs long read sequencing

The main advantage of long read sequencing is that the sequences are often longer than 10 Kbp, which improves a lot the assembly quality known from short read sequencing. For instance, the assembly of microbial genomes using short reads is often hampered by repeats such as the rRNA operon that prevent the closing of the genomes. Long reads can span such large regions thus allowing the completion of genomes. A disadvantage of the long reads is that the error rate of long sequences are much higher than with short read sequencing. A lot of study goes into the improvement of the error rate such that downstream issues with this high error rate are resolved. There is a wide variety of methods available that can help to reduce the sequencing error found in long reads. These approaches are divided in error-correction methods that either use only the long reads (non-hybrid) and methods that include short read sequencing information (hybrid) to remove errors (reviewed in detail in Amarasinghe, S.L. et al. 2020). This can be done before assembly, by aligning the reads and generate consensus sequencing. In addition, errors can also be resolved after genome assembly is performed by mapping either only the long reads or in combination with short reads. This is an iterative process and is called polishing. Last, but not least, a major difference between long and short read sequencing is the cost. Illumina sequencing is per isolate around 70-100 Euros depending on what machine that is used, while Nanopore is about 50% higher than that, and Pacbio is about 4 times as high. Note, these numbers are current as of spring 2021, so these are likely to change.

CHAPTER 6

Sequence file formats

Sequencers output their data in different formats according to what technology they are. Illumina sequencers output fastq-files, while Nanopore output their files in the Fast5 format. Pacbio however outputs their results in the BAM format.

6.1 Illumina - Fastq files

Fastq files are text files containing the biological sequence data produced by a sequencing machine. The "FASTQ format" is used to represent sequence data: a nucleotide sequence together with associated quality scores (for each base). The fastq file format is a standard format, consisting of 4 lines for each sequence:

- Line 1 starts with @, and is followed by an identifier. The identifier can be designed as the user wished, but in most cases it contains the sequencing machine ID, the sequencing run ID, the sequencing date, and a unique ID for the sequence.
- Line 2 is the raw sequence data. This section can be wrapped over multiple lines.
- Line 3 begins with a +, and can optionally contain the same information as line 1. This line is to indicate when the raw sequence in line 2 has stopped.
- Line 4 contains the Phred quality score information which is encoded using the ASCII characters 33 to 126.

Fastq files are usually compressed allowing storage of the same information while using less storage space. Compression softwares such as: gzip, bzip are standardly used. The compressed format is indicated by the following extensions: ".fastq.gz" or ".fq.gz". Most modern applications are able to extract the data from gzip compressed datasets and process the data for further use.

For paired-end data the fastq files come in pairs: "SAMPLE_R1.fastq.gz" and "SAMPLE_R2.fastq.gz" are the forward and the reverse reads files.

6.2 Illumina sequence architecture

The sequences produced by illumina sequencers consist of several parts, which is dependent on the experimental design. For a simple shotgun sequencing experiment, the sequence contains the shredded DNA fragment to which sequencing adapters are ligated. In contrast an amplicon sequencing experiment with multiple samples, contains the amplicon sequence as well as the amplification primers, a heterogeneity spacer (needed to make the library more complex), and a barcode or index to separate the samples. At the end it contains the sequencing adapters (see for instance Figure 1 in this article).

The experimental design therefore dictates how the sequences are handled bioinformatically after delivery of sequencing data. The Illumina machine software will use the indexes to group sequences in samples and it can be set up to remove the adapter sequences. For most sequences this means that one adapter sequence is removed. But the other adapter might be present as well: depending on the length of the sequenced fragment and how many bases were sequenced, therefore the later needs to be removed by the user of the data.

6.3 Nanopore - Fast5 files

Fast5 files are used for storing the output from Nanopore sequencing machines. These files are based on HDF5 file format. A fast5 file contains three sources of information: raw sequence data in picoampere, the event-level data, and base-level data. In addition, the fast5 files can also contain configuration data, based on the pipelines used to process the fast5 file, and summary data.

The event-level data is an aggregate of the raw data that on average describes the signal that belongs to one nucleotide (for a more detailed explanation see here).

The base-level data is the fastq information of each base with the corresponding quality value and stored in a compressed manner. The quality scores inside the fast5 format are capped at a maximum score of 31. That means that quality scores higher than 31 are set to 31.

6.4 Nanopore sequence architecture

Nanopore sequencing produces sequences that contain the DNA fragment with ligated to its ends sequencing adapters. In the case of multiple samples, barcodes are first ligated to the DNA fragments before sequencing adapters are ligated to the barcodes.

6.5 Pacbio - BAM files

The sequencer from Pacific biosciences will output the sequence data in a Binary Alignment Map file, or BAM file. Such files are binary, compressed, and they are a record-oriented container format for raw or aligned sequence reads Pacbio BAM file format. The uncompressed BAM files are called Sequence alignment map (SAM) files, which are text files that contain alignment information of sequences against a reference sequence. A difference with "regular" BAM files, is that the Pacbio BAM files can contain information in the header on the sequencing machine used, the sequencing kit, barcodes etc. Since the Pacbio sequencer can produce different kinds of reads, the BAM files are named accordingly. e.g. a file with unaligned ccs reads: "FileName.ccs.bam", and aligned ccs reads are : "FileName.aligned_ccs.bam". The sequences in the BAM files can be exported to FASTQ or FASTA formatted files.

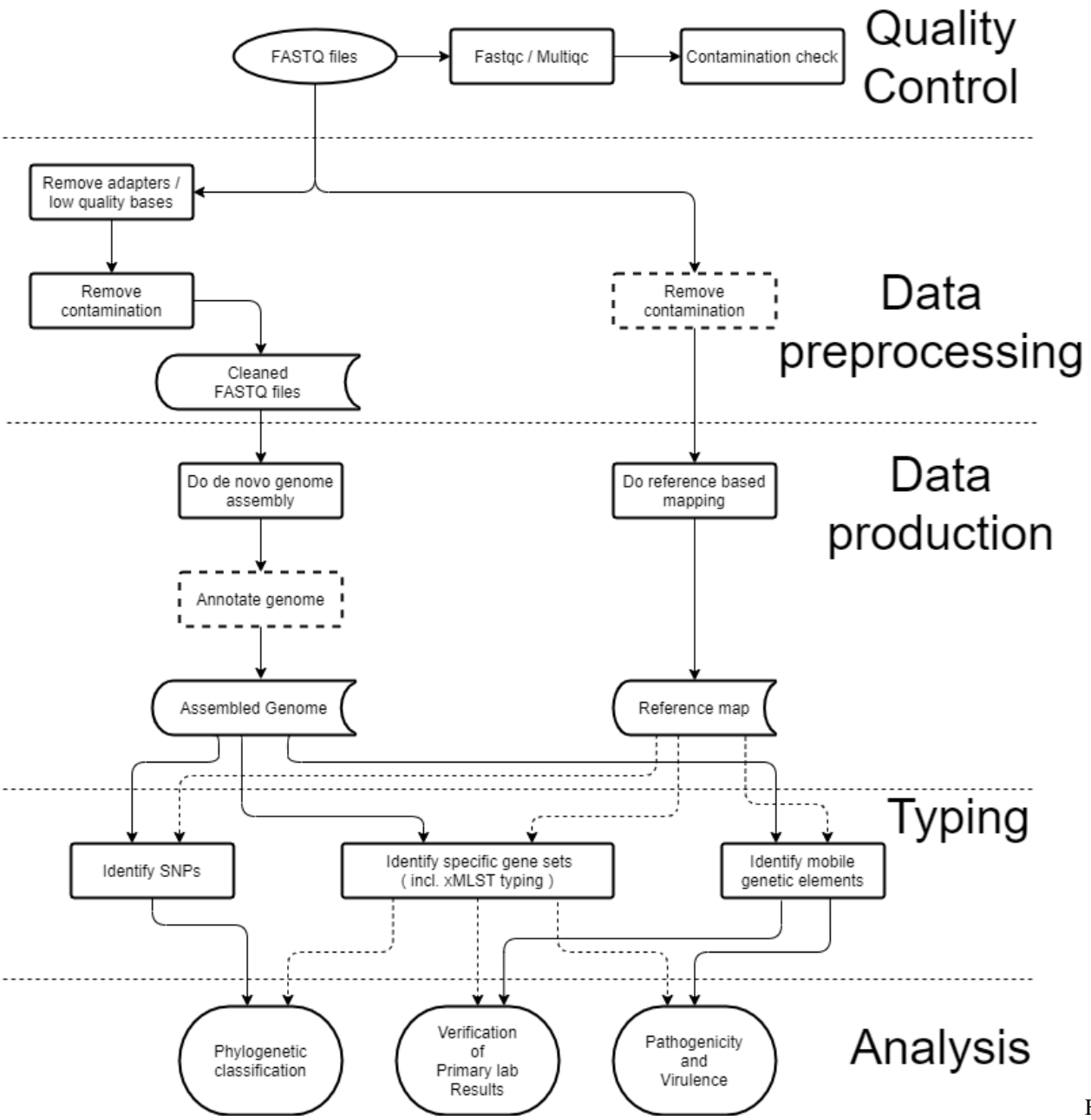
6.6 Pacbio sequence architecture

With Pacbio sequencing DNA fragments are end-repaired and special sequencing adapters are ligated to the ends (see Figure 1 in this paper). The structure of the sequencing adapters allows for repeated sequencing of both strands since a circular template molecule is created. The sequencing of such fragments then generates sequences that consist of both strands of the original fragment separated by the adapter sequence. The reads generated using such a strategy are called circular consensus sequences (CCS) and using dedicated tools these sequences are then transformed into a high quality sequence. The amount of times that a single sequence passes the polymerase, determines the sequencing quality after data processing. For instance, four passes will give a read with a Phred Quality score of 20 (99 % accuracy), while nine passes will give a quality score of 30 (99,9 % accuracy) (Amarasinghe et al., 2020).

CHAPTER 7

Sequence analysis tool chains

In recent years next generation sequencing has matured and with that a solidification on the required methods for WGS projects has occurred. Nonetheless, the current practice of microbial WGS is mostly confined to the academic world, and has not yet resulted in large scale introduction of WGS in clinical settings, with the exception of a few countries. However, while it is clear that as the technological developments in bioinformatics proceed at a rapid pace, a consensus has formed on which steps are required to produce WGS datasets that can be used for surveillance and outbreak detection and investigation. These steps involve quality control of the raw sequence data and a subsequent usage of clean data to produce an annotated genome or reference map. Such "products" can then be used to obtain typing data, which is needed for the final analysis under surveillance or outbreak scenarios. These steps are illustrated in the figure below.



Flowchart showing a standardized WGS bioinformatic analysis workflow. The standardized workflow can be divided into four sections: Quality control, Data preprocessing, Data production, and Data analysis. Boxes with dotted lines are optional in the workflows presented. Solid and dotted lines are used for clarity and both indicate possible analysis pathways.

The first part of the process is to ensure that the quality of the reads is sufficient to proceed. Then, reads are 'trimmed', i.e. low quality parts and adapters are removed, and they are checked for any contamination. Once the sequence data is clean, it can then be used in two different downstream processes.

- Genome assembly, which is the computational process of combining all the shotgun DNA sequences in order to recreate the original genome sequence of an organism.
- Reference mapping, which is the alignment of the shotgun sequence fragments to a chosen reference genome, in order to identify differences between the reference and the investigated isolate.

Based on either a mapping or a genome assembly, the isolates can be further characterized through various typing and clustering tools. Different tools need different inputs - in most cases they either take an assembly or reads, but not both. It should be noted that the read mapping pathway is usually a lot less computationally expensive than the assembly pathway.

As is evident from the processes described here, the tool chain needed for analysis can get quite complicated. For these purposes it can often be useful to either use a workflow language, like Nextflow and Snakemake, or to use an analysis platform, like those described under Data management and analysis platforms. Without such systems it can get quite complicated to keep account of what analysis has been done, and the version of tools used.

CHAPTER 8

Quality control and preprocessing

8.1 Fastqc / Multiqc analysis

After sequence data is delivered to the analyst / researcher it first needs to be checked to assure that the data is of good enough quality to work with. This is mostly done by performing an analysis with the program Fastqc. The software Fastqc summarizes the fastq file data, and displays information about read length, average quality score along the read, GC-content, number of ambiguous bases, the presence of sequencing adapters, and various other parameters useful to determine the quality of the raw sequence data. The program Multiqc can be used to combine the output of multiple fastqc output files, or many other programs, so that many datasets can be inspected simultaneously.

8.2 Controlling contamination

Besides general inspection of the data, it is also wise to check for possible contaminants present in the sequence data files. Sequence data can contain exogenous sequences (generally at low frequency) derived from contaminants introduced during either the DNA extraction or the library prep phases. This is mostly of serious concern when working with small amounts of input material and when using PCR amplification (Salter et al., 2014; Drengenes et al., 2019). In all cases, it is advisable to remove contaminating sequences from the sequence data.

The origins of the microbial contaminants can be diverse and they are found in ultrapure water systems, molecular biology kits or laboratory reagents (Salter et al., 2014). In addition, some experimental procedures deliberately add control DNA to improve throughput or for normalization purposes so different samples can be compared. For example, the Illumina sequencing platform uses the genome of the phage PhiX (Sanger et al., 1977), as a control in the sequencing run. It is included for quality and calibrations purposes, but when not detected in the sequence data it can contaminate such data with far reaching consequences (Mukherjee et al., 2015). Additional sources for contaminating data can be human sequences due to laboratory contamination, or because the sample was associated with host tissue (as in the case of metagenomic experiments). In labs where DNA extraction of other species is performed on a large scale, it is possible that DNA fragments of other species can contaminate the samples when DNA extraction is done without measures to prevent such contamination.

In addition to contamination due to laboratory methods, contaminants can also be present due to entirely natural causes. In the case of whole genome sequencing, contaminants are often due to pure cultures containing an additional bacterial species or the culture is a different species than expected.

Contamination is commonly detected by either:

- comparing the sequence data to a reference genome and calculating a distance measure as done by the program Mash,
- by mapping the reads to the human or other relevant genomes,
- or by classifying the reads against a database containing reference genomes and identifying if there are reads that do not belong to the target species. This is often done with tools from metagenomics research such as Kraken2, Centrifuge, etc.

Contamination can be removed via mapping or classification of reads to either a reference genome (e.g. PhiX or Homo sapiens) or a dedicated reference database (see for instance this publication: Bush et al., 2020). Reads that match are then removed and it is assumed that the remaining reads are clean from contamination. Assembled microbial genomes can / should also be screened for the presence of contaminating sequences, for instance the genome of the Phage phiX readily assembles as one contig when the Phix sequences were not removed before assembly. This can also be done by screening the contig sequences using blast or other approaches (Mukherjee et al., 2015).

CHAPTER 9

Data preprocessing

Nucleotide sequences in sequencing files can be of low quality. Thus, the sequences needs to be processed such that the overall quality in the sequence file is improved before it is used in any kind of data analysis. Adapters might be attached to the sequenced fragments, these are also often removed before further processing. Also, while Illumina data these days come already basecalled (i.e. the signal has been translated into DNA letters), this might not be the case for Pacbio and Nanopore data. This step might thus have to be performed before adapter and quality trimming.

9.1 Quality and adapter trimming

Before further analysis, it is common to evaluate the quality of the data, and to remove any adapters found in the reads and also low quality regions. Commonly used tools frequently do both of these things.

Quality is denoted on a per base level, via the PHRED score, which denotes the likelihood of the base being wrong. For Illumina data, the quality of a read will commonly be quite high in the beginning (Q30-40), but then fall along the read, dipping towards the end. Commonly anything below Q15-Q20 is regarded as bad, and portions of the reads where the average quality is getting too low are generally trimmed, i.e. removed from the read. The first read (R1) in a pair commonly has better quality than the second (R1) read.

9.1.1 Nanopore basecalling and trimming

Nanopore sequence data is delivered in the fast5 file format which contains the raw signal data. That data can be translated into fastq files using dedicated basecallers such as Guppy / Bonito. Guppy comes with two different models for basecalling, a fast basecalling model and a high accuracy model. As the names indicate the high accuracy model gives more accurate basecalling and with better detection and binning of barcoded reads than the fast model. The average quality scores of sequences generated by Oxford Nanopore instruments are between 7 and 14 with quality being variable along the reads. Any sequences having a Q-value below 7 are usually discarded. In addition, trimming of the first group of bases (10-50) improves the overall quality score of the reads. Trimming of adapters and low-quality bases at the end of the sequences is also performed.

9.1.2 Pacific biosciences data

Pacbio sequences are delivered as BAM-files, where the bases do not have meaningful quality scores. Pacbio sequences do however have highly variable qualities for the bases. Depending on the sequencing technique used (Continues Long Reads (CLR) or Circular Consensus Sequencing (CCS)) the Pacbio reads can be corrected or not. The raw pacbio sequences can be converted into fastq or fasta files. When converted to fastq, the quality scores are marked with the exclamation mark: "!", which is similar to "0". CLR reads can easily be converted to fastq using the program bam2fastx, but with low quality scores. These reads can best be used in combination with Illumina reads to generate a hybrid assembly. CCS reads are demultiplexed and can be filtered using the number of passes using the SMRT portal software. More passes gives a better sequences afterwards. The CCS reads then can be converted to fastq reads with ccs, which uses each of the subreads in an alignment to polish the reads and generate high quality bases. It also removes the hairpin sequences from the CCS reads. At that point only limited or no trimming is needed of the reads.

9.1.3 Software availability

There are many tools available for doing QC and adapter trimming. This paper, although not quite new, contains a good overview of the process and the effect of some commonly used tools for Illumina data (Fabro et al., 2013). Important to note, all these tools can be used for paired-end and mate-pair sequencing data. Nevertheless, they usually do not account for the particularities of mate-pair sequencing protocol, often discarding more data than necessary. NxTrim is a trimming software optimized for mate-pair sequencing. For Nanopore and Pacbio data there fewer options available. Good starting points are tools such as NanoPack and Pauvre that give information about the quality of the sequence data.

Note: it might not be necessary to do quality trimming and adaptor removal in cases where mapping is the primary approach. The adapters are unlikely to match anything in the reference sequence, and mapping tools commonly take the quality score of the base into account and may leave low quality regions out.

CHAPTER 10

Data Production

10.1 Reference-based vs *de novo* genome assembly

Once data has been preprocessed, the reads can be analysed further. To this end, we need to "solve the puzzle" and understand to which genomic region a read may correspond. This usually proceeds via one of two pathways: via mapping to a reference genome or via *de novo* genome assembly. In cases where the desired output can be acquired via mapping in some way or shape, that is often preferred since genome assembly is a computationally demanding process. However, that should be balanced against the subsequent use. Anecdotal results suggest that mapping methods might give more variants for cgMLST, and thus result in artificial inflation of allele differences between isolates.

10.2 Assembly and annotation

10.2.1 What is an assembly

After sequencing, the genome is available in the form of sequenced reads, and these can be used to create an assembly of the genome. An assembly is a reconstruction of the genome, in that the actual genome being sequenced is in itself an unobservable entity. In the assembly process, the reads stemming from the sequencing process are aligned and merged with the aim of producing a consensus sequence of the genome.

The end result of an assembly is a fasta file containing the reconstructed genome sequence. The assembly is likely to be fragmented, at least if the input data was Illumina reads due to their short insert size, which influences the ability to solve genomic repeats (see also the *Sequencing technologies* section). The individual parts of the genome that could be reconstructed without any issue will be present as contigs - contiguous sequences. Some assembly tools analyze the read data further and try to organize contigs to make scaffolds. In a scaffold, the contigs are in the order that the assembler thinks they should be in, and there will be Ns present as place holders between the contigs to show how far apart the assembler thinks the contigs are. The assemblers create scaffolds by examining the read data and use the pairing information to order the contigs into scaffolds and to estimate what the distance between the contigs might be. There would then be scaffolds present in the fasta file. Please note, some assemblers have a default setting for the minimum gap in a scaffold length.

10.2.2 Estimating genome coverage

For assembly, it is important to have reads covering all bases, and in sufficient quantity. In order to assess if this is the case, it is possible to calculate the approximate coverage of the sequenced genome before an assembly is created. For this calculation, an estimate of how long the "real" genome is likely to be is needed. Frequently, the length of a closely related reference genome can be used.

Coverage = (number of reads * read length) / estimated genome length

Example:*Listeria monocytogenes* genome length: 2.944 Mbp\Number of reads: 1 400 000 (paired-end)\Read length: 150 bp\Coverage = (1 400 000 * 2 *150) / 2 944 000 = 143\

I.e. the expected coverage for this genome is 143. This is frequently written as 143x coverage.

For assemblies from Illumina data, it is commonly recommended to have a coverage between 50x and 100x. Assembly software commonly does better with higher coverage, but there can be deterioration in the results with coverage above 100x or so since for most de Bruijn graph assemblers (for instance SPAdes), higher coverage can complicate and break apart the assembly graph.

10.2.3 How does the assembly process work

There are two main types of methods in use today for assembly. The two types are Overlap-Layout-Consensus (OLC), and de Bruijn-based methods (check this paper). OLC methods are more frequently used with longer reads, and was the commonly used method in the first wave of sequencing, when reads were from 400-1000 bp long. With the advent of short-read sequencing (i.e. 36-300 bp long reads), de Bruijn based methods came to the forefront. OLC methods are again becoming more widespread due to long-read sequencing becoming available.

Both methods use graphs to create the assembly. The two main features of a graph are nodes and edges. Nodes are often depicted as circles, with the edges being lines between the circles. The main difference between the two methods is how reads are used to build the graph. In OLC methods, the reads are assigned to the nodes in the graph. All reads are then compared all-against-all to discover overlap. If there is sufficient overlap between two reads, an edge will be created between the two nodes. This represents the overlap between the two reads. The assembly is then created by walking along this graph and in the process creating contigs. An OLC graph will contain as many nodes as there are reads in the read set.

In de Bruijn-based methods, the reads are chopped up into subsequences of length k - these shorter pieces are called k-mers. This cutting into k-mers happens by shifting a window of size k base by base along the reads, so that each k-mer from a read will have an overlap of length k-1 with the preceding and the following k-mer. The graph is built by assigning each k-mer to a node, and if two k-mers have an overlap of k-1, an edge will be created between the two nodes. Contigs are again created by walking along the graph and finding paths in it that become contigs. A de Bruijn graph will have as many nodes as there are unique k-mers in the read set. Therefore, a critical step is the choice of the value of k. If k is small, there will be many connections between the nodes in the graph, i.e. many overlaps between k-mers will be found. However, this increases the chance of nodes and edges being introduced in the graph due to overlap between random k-mers, and this can contribute to the assembly fragmentation due to the graph becoming very entangled. If k is too long, the chances of detecting overlapping sequences decreases, and this can also lead to assembly fragmentation. As an attempt to get a good compromise between these factors, several assembly programs (e.g. SPAdes) have implemented a system in which different k-mer sizes are used iteratively and in the end a combined genome assembly is generated.

The main difference between the two methods lies then in how the reads are being used to build the graph. These differences have as their main consequence how the complexity of the graphs increases. While OLC graphs do not depend on the k-mers but on the number of reads, they tend to increase complexity with increased read coverage. By opposite, de Bruijn graphs deal very well with the high coverage of NGS data (because it does not imply an increase in the number of nodes), but the size and complexity of the target genome (e.g. presence of repetitive regions) may have a great impact on the performance, as will also sequencing errors.

10.2.4 Assumptions made in the assembly process

There are certain assumptions that are made when performing assemblies. The main assumptions are:

All bases are sequenced: it is assumed that all bases in a genome are present in the reads from that genome. If not, that region will naturally not be present in the assembly, and it will lead to breaks in the assembly. Several studies have reported some bias in the sequencing data, which can be related, for example, to the read starting position, the GC content, or the protocol of library preparation (check this paper).

Sufficient depth of coverage: it is assumed that there are a sufficient number of reads that cover each base in the genome, this reflects the minimum number of times that a base has been sequenced. Many of the assembly methods actively use evenness of coverage when trying to resolve repeats.

Errors are random: it is assumed that any sequencing errors appear randomly in the reads. If they are, it is likely that these errors will be eliminated in the assembly process, either by error correction processes, or simply by the other reads from that region forming a consensus that eliminates the error. If they are not random, they can form an alternate contig, thus giving rise to two contigs from the same region. Errors may have great impact in de Bruijn graphs because they contribute to the increase in the number of nodes and edges. These issues can be alleviated by good QC analysis and trimming during data preprocessing.

10.2.5 The influence of read length

Read length is a determining factor on the outcome of an assembly process. This is due to the fact that in an assembly process it is not possible to resolve repeats that are longer than the read length. This is exemplified in the section about paired-end sequencing. This is also known as Ukkonen's condition, and for a deep dive into that, please read this blogpost. This is the main reason why long read sequencing has been gaining ground the last years, longer reads give assemblies with fewer contigs. For microbial genomes it is not uncommon for long read data to result in fully closed genomes.

10.2.6 Some commonly used assembly programs

The amount of sequencing assembly tools increased drastically with the advent of second generation sequencing instruments. The first assemblers that were available were primarily Overlap-Layout-Consensus tools, such as Newbler and Celera, both now discontinued. These were used for Sanger sequencing and for sequences from 454 machines, which were generally 400-1000 bp long. With the advent of Illumina sequencing, whose first reads were 36 bp long, de Bruijn based methods came to the forefront. This development was started by the velvet software package, which uses a fixed k-mer size for assembly. After the release of velvet in 2008, many different assemblers were created and got varying levels of use (for an early review, see this paper. In 2012 the program SPAdes was released, this tool uses different k-mer sizes (and other tricks too), which means that it frequently produces more contiguous assemblies than velvet does. This program gradually took over as the main assembler for microbial data.

For short read data, there are today four tools in common use, three of which involve SPAdes in some way. The first is naturally SPAdes itself. Then there is the software package shovill that uses SPAdes as its assembly component. Shovill does downsampling of the data to avoid overloading the assembly graph, and also includes trimming, so it is a good choice for a one-stop-shop assembler. It is also known to be fast, which increases its usefulness for bulk analysis. The tool Unicycler also uses SPAdes in its internals, and works as a SPAdes optimizer when only used on Illumina data. Unicycler is also a hybrid assembler, and can thus be used in cases where both long and short read data is available. Last but not least, there is SKESA which was created by the NCBI. This assembler has as its goal to be a bit conservative and rather break up at repeats to avoid mis-assemblies, and to be fast.

PacBio was the first of the 3rd generation platforms that came on the market with long single cell reads. With long reads, OLC methods again came to the forefront. The tools that came into common use were frequently modifications and adaptations of previous OLC software, such as canu which is a fork of Celera. Today, for PacBio the most commonly used assemblers are HGAP, which was developed by Pacbio, and also canu. Oxford Nanopore, the most commonly used tools are Unicycler and canu.

For more on what tools are in use, see this paper.

10.2.7 Assembly quality evaluation

Once a genome has been assembled, the assembly has to be evaluated to see how good it is. It is common to evaluate this on three different features:

Completeness: Completeness shows to what extent the entirety of the genome has been captured in the assembly. This can be difficult to quantify, since each new genome is a novel entity unto itself. However, this can be estimated by examining to what extent genes that so far seem universally present in a specific type of genome can be recovered from the assembly. This can be done with tools such as CheckM or BUSCO. The latter gives an estimate of how many of a set of expected genes are either found in a complete form, a fragmented form, or not found at all.

Another measure of completeness is how the reconstructed genome length compares to the expected genome length. This can be challenging for genomes of species like *Escherichia coli*, where the known span of genome lengths vary with approximately 1M bases.

Another alternative is the comparison of the k-mers present in the final genome assembly with the k-mers present in the trimmed and cleaned fastq files with programs like KAT. Such an approach can give an idea not only of the missing portions of that particular genome, but also of the presence of non-collapsed repetitive regions in the assembly.

Contiguity: Contiguity shows to what extent the assembly process is capable of knitting together the reads into a contiguous sequence. The aim is to have as few and as long contigs as possible, while avoiding mis-assemblies. Contiguity is frequently measured as N50. This value is the length of the longest contig where the contigs longer than this contig in total contains 50% or more of the assembled bases.

Example of what N50 is:Let's say an assembly has 7 contigs, and they have these lengths:120, 170, 320, 550, 750, 760, 850

The total length is 3520, and the length halfway point here is 1760. Thus, the N50 becomes the contig length which, when counting from the longest to the shortest, tips over 1760 bases. In this case that is 750 - the contigs with a length of 750 and longer in total contain more than 50% of the bases of the assembly.

A variant of this is the LG50 or L50. In this case the length of the assembly in the calculation above is replaced by an estimate of how long the genome should be. More on these measures, and other similar measures are available here.

Correctness: Correctness shows to what extent the bases and their order in the assembly reflect what is in the sequenced genome. This is per definition impossible to truly measure since the actual sequenced genome is not a directly observable entity. Common errors that can be looked for are mis-joins, repeat compression or expansions as well as indels. Such errors can either be detected by comparing to a reference (with for instance QUAST), or by doing an internal comparison between the reads and the genome assembled from those reads (as done by REAPR or the nuc-suite of tools).

10.2.8 Genome annotation

DNA or genome annotation is the process of identifying the location of functional regions in DNA / genomes sequences (Wikipedia) and subsequently designating a function to this region. Functional regions can consist of both coding regions and non-coding regions, and they can be identified using a variety of tools that are trained to detect rRNA, tRNA, non-coding RNA, protein coding genes, CRISPR regions and more. The input for this process is a fasta file containing the DNA sequence and which the annotation tools use to identify the location of various functional regions. That information can be stored inside a General Feature Format (GFF) file or in a Genbank / EMBL / DDBJ format file. The former only contains information on the location of functional regions, but does not contain the DNA sequence nor the translation of protein coding genes. A Genbank file (or EMBL / DDBJ) contains the complete genome sequence, as well as the location of the coding regions with the translation of those regions when applicable (e.g. Protein coding genes). Specialized algorithms exist to predict the presence of each type of functional region. This can happen in one of two main ways: either through *de novo* gene prediction, or through homology searches. Both of these methods

have one thing in common: they both presuppose the availability of already annotated genes that may be present in the genome that is being examined. With the amount of sequencing done today, such a data set is often available. However, in situations when new species, or possibly even new genuses are being examined, this can be an issue.

In either case, the data that is available is then used to predict the functional regions in the genome at hand. If prediction is done directly via homology searches, this commonly proceeds through blast searches where the available genes are searched for in the genome. If prediction is done *de novo*, the available data set is instead used to train a model that is subsequently used to search the genome for genes. This is the tactic employed by Glimmer and Prodigal, which are two of the more commonly used prokaryotic gene finding tools. These tools extract a wide range of information from their training sets to determine protein coding regions such as: The presence of start and stop codons usage, ribosomal binding site (RBS) motif, GC content bias, hexamer coding bias etc. *De novo* gene finding tools frequently come with default training models. However, since these factors can differ from species to species, a species specific training file can be created from other annotated genomes from that species and used for prediction for new genomes from that species.

Once the various types of genomic features have been identified, they have to be assigned a function. For some types of regions the functional assignment is baked into the finding tool. For instance, for finding tRNAs, the fact that the found regions are tRNAs are a given since that is what was searched for. Functional assignment is predominantly an issue when it comes to assigning a function to proteins. If homology searches were used for gene prediction, the gene function is frequently then "lifted" from the search onto the found gene. If *de novo* methods have been used, this functional assignment is then a separate step, commonly involving homology searches using blast. In either case, this highlights the need for a well curated and a well selected database for these searches.

For command line prokaryotic genome annotation, the most commonly used tool today is likely PROKKA. This tool uses prodigal for gene prediction, along with other specialist tools for finding rRNAs, tRNAs and other genomic features. For functional assignment, this tool does several different kinds of searches in a hierarchical manner. First a core set of curated and included with the program databases is searched using blast. If a match is found, then the function is lifted over. If the user wants to, it is possible to add their own core database to this set of annotated databases. Next, for the genes who did not gain a function through this step, a different kind of search using profiles is done, using HMMER3. This program allows for more distant searches than blast. In addition, the PROKKA program allows a user to supply their own annotated proteins via the "-proteins" option.

Another commonly used tool is the web based RAST system. RAST has as its core a set of subsystems, which are functionally related proteins, and these proteins have a "FIGfam" associated with them, which is a gene family which have been curated by human experts in the FIG group. When doing genome annotation, RAST starts out by annotating some specialist types of genes first. Then, it uses GLIMMER do do ab initio gene prediction, these are then used to find the 30 most closely phylogenetically related genomes. In addition, k-mer searches are done towards all of the subsystems in RAST to find genes that seem to be present in the genomes. These subsystems, together with the subsystems in the 30 most related genomes are then used to train a GLIMMER model, and the genome is then searched with this model. Genes that are not annotated in this process are blast-ed against the 30 most related genomes and annotated that way.

Genome annotation is a complex business, and many methods and tools exist. Only two of the available tools have been mentioned here. This paper goes into depth about gene prediction and annotation, including how these processes work for eukaryotic genomes. That paper also includes information on genome visualization, and tools for assigning more high level functions.

10.3 Sequence read mapping

10.3.1 How mapping works

Mapping is used for many different purposes, such as contamination removal, SNP calling, and for finding specific genes such as through MLST typing and serotyping. Mapping is the process by which reads are placed onto a reference sequence. This sequence can represent the whole-genome of one or multiple organisms (*de novo* assembled or retrieved

from public databases), multiple genes, or any other DNA sequence of interest. During mapping a read is matched up with a location on a genome provided the read is identical or nearly identical to the sequence in that location. Consequently, mapping is only appropriate when it is presumed that the reads are very similar to the reference. Due to this, mapping is rarely done with long read data due to the error profile of long reads. For long reads, alignment processes such as blast are often used instead.

The results of mapping are output in a SAM or a BAM (Binary SAM) format, which specifies the coordinates in the reference sequence where the similarity between the read and the reference sequence is the highest, together with other relevant information about the read mapping. For instance, mapped reads commonly receive a mapping score detailing how well it mapped. However, the interpretation of that score varies widely with the tool used for mapping. It should be noted that since Illumina reads are quite short, it is possible that a read might have two or more equally well fitting locations on the genome. These reads are then called "multimapping" reads. The fact that they are multimapping should be evident in the output. The "CIGAR string" is also included, which is a shorthand description of how much of the read mapped, and how many mismatches there were. For paired-end data it also includes information about the mapping characteristics of the other read in the pair.

During mapping the reads might end up being "clipped", either hard or soft clipped. This will be evident in the output and can be seen in the CIGAR string. With hard clipping, the parts of the read that did not match are removed from the read, while in soft clipping the clipped region will be present, but will be masked so that other downstream tools don't use that part. As read mapping can be done using any DNA sequence as a reference, this strategy can be used for contamination checking and filtering, in that it can be used to figure out if a certain contaminant (e.g. PhiX genome) is present (for instance, any remaining PhiX reads will map to the PhiX reference sequence). If so, the reads associated with the contaminant can be removed. In cases such as MLST finding and serotype finding, where the main focus is a specific set of target genes (see *MLST section*), the reads are commonly mapped to a database of sequences comprising the genes of interest and not the whole genome sequence. The reads that then map to a reference sequence (a MLST sequence, a serotype gene, an AMR gene etc), will show what gene is present in the data. After mapping, SNP calling (see SNP calling section) can highlight any nucleotidic differences between the reads and the reference, being useful to determine SNPs (and INDELs) across the genome when a whole-genome reference is used, or to determine allele-specific variation in the MLST, AMR or serotyping analysis.

It is important to note that read mapping can be performed for multiple samples towards the same reference (single sequence or a set of sequences, often referred to as a database). Therefore, for downstream analysis all the regions of interest will be indirectly aligned between the different samples. I.e., position X in sample A corresponds to position X in sample B, and any information relative to that position can be compared without the need of a multi-sequence alignment.

There are many mapping tools available today. However, among the ones more commonly used are BWA and Bowtie. In addition, BBMap from the BBTools package is a popular choice. Read mapping results (BAM files) can be visualized with so-called genome browsers, e.g. IGV, where the user can have a visual idea of the genomic features of a given sample or even multiple samples as long as their reads were mapped to the same reference.

10.3.2 How to choose a reference genome?

A "reference" allows for creating a coordinate system (the reference map) that can be used to compare samples at each position. This reference map can be used for producing multiple-alignments and variants files required for downstream analyses. A reference genome is usually chosen because the genome is complete, annotated and the sequence has been validated. The reference sequence is usually chosen as a representative of the species/or taxa under study. This choice strongly influences the precision of the SNP calling tools, and should be chosen to be "as similar as possible" (closely related) to the isolates under study (Bush et al. 2020). A good source are genomes from refseq. Be also aware of whether the genome contains plasmids.

If a reference from a specific strain or sequence type is needed, Enterobase offers options for searching for specific genomes. Note some options are only available for logged in users. In Enterobase, organisms of interest can be selected, such as *Salmonella, E. coli, Streptococcus*, and others. After selecting a species, specific strains can be searched for using the search fuctions "search strains" or "find STs". For sequence type searchers, for some species the schema then has to be selected. Data can subsequently be filtered in the resulting table to get isolates from specific

continents, matrices, year, etc. The data can also be downloaded (see the Data dropdown menu) if there is a wish to filter the data in separate programs such as R or excel. Note: Make sure that the experimental data, on the right of the separator, is presenting the wanted information, these will be downloaded with the other info. Several types of experimental data, including MLST and cgMLST data is available.

As an alternative (or if a good genome assembly is not yet available for the species that is being studied), genome assembly of the samples at hand can be done (see de novo *genome assembly section*), and one of these can be selected as a reference. This genome should preferably be the one that appears as the most complete, i.e high N50, few contigs, etc. Fast clustering, for instance using popPUNK can be done to figure out which ones of the samples are at hand would be "equally related", i.e. a centroid in the similarity space of the samples at hand.

10.4 Sequence searches

10.4.1 BLAST

Basic Local Alignment Search Tool (BLAST) [https://en.wikipedia.org/wiki/BLAST (biotechnology)] is a method (and a program) that allows searching in sequence databases. These databases can contain whole-genome sequences, a set of genes or proteins, or any other DNA sequences of interest. The method takes in one or several query sequences (e.g. the predicted protein-coding genes of a genome assembly, check de novo *genome assembly section*), and tries to align them in the sequences of the provided database (e.g. the nucleotide sequence database of NCBI). When it finds corresponding matches (similar sequences), it reports the name of the match, the respective percentage of identity, length of the alignment, query coverage, e-value, and other important scores. It is worth noting that these matches may be partial, both on the query side and the database side. That is, if searching with a 100 bp query sequence, it is not given that the matching region will cover the entire 100 bp query sequence. This is important when using it to search for genes. Therefore, it is important to post-process the results to be able to infer homology relationships from the alignment. The BLAST tool enables the user to set criteria on the results, such as lower values for e-value, query coverage, etc, which will reduce the number of reported hits. BLAST can be used to align nucleotide to nucleotide sequences (BLASTn), protein to protein (BLASTp), protein to nucleotide (tBLASTn) or nucleotide to protein (BLASTx).

BLAST is commonly available online as a tool on sites offering access to sequence data. The most well known site is the NCBI website. BLAST can also be downloaded and installed as a command line tool on Windows, Linux and Mac. For local BLAST a database has to be generated beforehand. This database can be any set of sequences/genome(s) of interest. Public databases (e.g. non-redundant database or uniprot database), are available for download in this ftp.

Reciprocal Best Hits (RBH) are a common proxy for orthology in comparative genomics. A RBH is defined as when two sequences, each from its own genome, find the other as the best scoring match when searched for in the other genome. That is, if a BLAST search is done between the set of genes from genome A towards a database comprising the genes from genome B, and vice-versa, and genes A1 and B1 are the best hit of each other in both cases, they are RBHs.

10.4.2 ePCR or insilico PCR

ePRC aims at emulating the process of PCR using WGS data. This is done by searching for a set of specific primers in a genome, and then examining the lengths and directionality to ensure that the result would be a valid PCR product. This can then be used as a method for detecting the presence and absence of a gene. This is primarily done today as a means of emulating serotype finding in some species.

CHAPTER 11

MLST

MLST (multi-locus sequence typing) was first developed in 1998 for *N. meningitidis*. The idea is to select a set of loci in the genome, and get the alleles in each genome for these loci. Typing is then done by seeing what isolates have which combination of alleles for each locus.

11.1 Schema

With MLST, the focus is on a subset of genes or loci that are known to be present in most, if not all strains of that species, but which are also known to have some sequence variation. The loci selected for use in MLST constitute the schema for that species. Commonly, for the original type of MLST, the number of genes chosen was 7. These are commonly housekeeping genes. There are several extensions of this that have come in later years, among them core genome MLST (cgMLST) and ribosomal MLST (rMLST). The main difference between the methods is in the criteria for how loci are included in the schema.

11.2 Allele and nomenclature

For each gene or locus in the schema, it is possible to have different sequence variants, or alleles. The different variants that have been observed can be collected in a database. Each observed allele for a gene can in this database be given a label, commonly a number. It is important to realize that these labels or numbers are attributed sequentially when detected, but do NOT provide any indication of relatedness or similarity between the alleles.

For the original "7-gene" type of MLST, several international nomenclature databases with alleles have been established. With access to such nomenclature databases, it is possible to ensure that all alleles have a unique number. This means that if using the same schema and the same database, two institutions can exchange information and know that they are referring to the same alleles.

11.3 Profile and sequence type

Each isolate can be typed by finding the sequence for each of the loci the schema contains. Once the sequence is found, it is compared to known alleles for that locus. This is usually done by comparing to alleles downloaded or otherwise accessed from the nomenclature server. If it is identical to an allele from the nomenclature database, the allele number for that locus is assigned to that isolate. If not, the allele is commonly uploaded to the server, which then assigns it a new number. This process happens automatically in some tools, in other cases it is done manually. The set of identifiers for an isolate is called its profile. The profile is commonly collapsed into a sequence type number, where each ST represents a unique combination of alleles for each locus.

Provided that the allele numbers and sequence type numbers are assigned by the same central nomenclature server, sequence types can be compared between institutions. Some institutions, like EFSA, can under certain circumstances allow organizations to submit sequence types, thus avoiding the need for uploading the full sequence.

11.4 MLST resources

11.4.1 Typing/Nomenclature databases

There are three main MLST nomenclature/database servers.

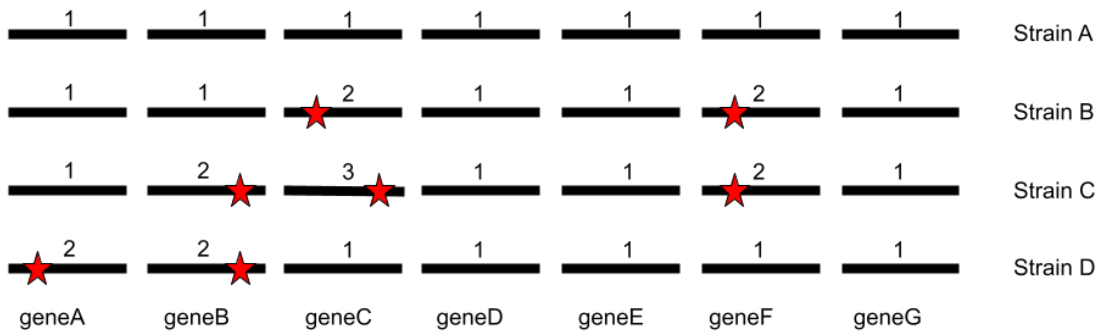
- https://pubmlst.org/
- https://bigsdb.pasteur.fr/
- https://enterobase.warwick.ac.uk/

It is important to realize that these work independently from each other and may contain different schemas, and even if they have the same schema there is no guarantee that the allele numbering is the same. Thus a sequence type only makes sense provided it is known which system has been used.

11.4.2 MLST finding methods

There are two main strategies for finding MLST genes.

- Assembly strategy: in this case the genomes are first assembled. Next, the allele set included in the typing database is used to search the genome, commonly using BLAST. The results are subsequently post-processed to get the allele and then compared to a profile database to assign ST numbers.
- Mapping strategy: in this case, the reads for each isolate are mapped against the allele typing database. Next, those reads are collected and assembled into genes. These are then compared to the alleles in the database, and allele and ST numbers are assigned.



This figure shows how MLST works. In this case there are seven loci or genes being used, genes A to G. Strain A is the first strain that is classified, thus each version or allele of these genes are given the number 1. Subsequent new alleles found in other strains are then given new numbers as they are found. Note, the numbers are specific to each locus or gene, thus two different alleles within the same strain can have the same number.

11.4.3 MLST software

There are many tools for doing MLST from both reads and assemblies available. It is important to realize that even though they might use the same schema, they might not give the same results. A review of various MLST tools and their results can be found in this paper. Due to this review no extensive examination of tools is given here. Some communities lean towards using certain tools, if so, these are mentioned in the Species specific sections.

CHAPTER 12

cgMLST

Core genome multi-locus sequence typing is an extension of the original seven-loci MLST method (Maiden et al. 2013, Lüth and al. 2018). Conceptually speaking cgMLST works in the same way as MLST, however, many more genes/loci are being used. For instance, one popular set of genes that is used as a schema for *Escherichia coli* consists of 2,513 genes. As for 7-gene MLST, there are different sources for the schemas. Schemas are defined as a set of loci/genes and their respective alleles. There are also different tools in use for finding the alleles which might not always produce perfectly comparable results. One notable difference between 7-gene MLST and cgMLST is that there is frequently no sequence type. Since so many genes are used, and since there are so many alleles possible for each gene, translating that into a sequence type label becomes difficult. cgMLST is thus primarily used as basis to compute pairwise distance measures matrices based on the number of shared alleles (eg. hamming distance), distance measures can then be used for clustering. However, recently methods have appeared that do allow for assigning a sequence type, this is described below.

12.1 Schemas and nomenclature servers

As with 7-gene MLST, a schema is needed for doing cgMLST. Schemes provide a list of loci (including an identifier), the set of alleles (including sequence for the allele) that are defined for each locus as well as an allele identifier. Schemes are designed for each bacterial pathogen and therefore cannot be used for a species it has not been designed for. Several schemes might have been designed for specific species. In such cases, the allele identifier of the different schemes will not be directly comparable even for alleles that might be included in both schemes (Uelze et al. 2020).

The process of typing with cgMLST works similarly as it does for MLST. The alleles in the schema are searched for in the genome. If an allele is found, its locus id ID and allele number is logged for the genome. However, the two processes differ when it comes to handling novel allele variants. For MLST, it is common to submit the allele to a nomenclature server, where it will be given a centrally assigned number. For cgMLST this process may happen locally, i.e. within the system of whomever is using the tool.

Database servers give access to schemas and allow for synchronisation of alleles designations when new alleles are added to the schemes, and synchronisation or submitting of newly discovered alleles to the scheme. The more commonly used servers are:

- Enterobase

- PubMLST
- ChewBBACA online: Chewie-NS
- Ridom SeqSphere
- BIGSdb Pasteur Institute

When new alleles for a locus in a schema are discovered, the allele sequence is added to the scheme and is given a numerical ID. This number is commonly attributed sequentially. Analyzing isolates that contain new alleles in a different order will lead to differences in attribution of allele ID numbers for the same allele. Therefore, the numbering of cgMLST typing might not be comparable between laboratories that are keeping local typing databases, even for laboratories that use the same scheme, due to possible divergence of allele numbering after local database initiation (Deneke et al.2021), unless local databases are in constant synchronisation with nomenclature servers. Therefore there is a growing interest in developing hash-based cgMLST. Hash-based cgMLST transforms the typed alleles sequences into a hashID (a compressed representation of the sequence in form of a string). Due to the way hashes work, this means that each allele will have an unique identifier This allows direct sharing of typing results that are independent of allele numbering and thus directly comparable between laboratories without relying on centralised and curated nomenclature servers (Eyre et al. 2019, Deneke et al.2021). Thus hash-based cgMLST typing is as such not a new sequence typing method but a new way to allow optimization of data exchange.

12.2 cgMLST resources

cgMLST software generally speaking works in the same way as MLST methods do: the software compares the sequencing data for an isolate, either in the form of reads or an assembly, to a set of alleles. If an allele is located in the sequencing data, it is recorded as found in that isolate. Similarly to MLST, there are two main approaches to finding cgMLST alleles, either based on using BLAST on an assembly, or using the reads directly somehow against an allele database. Another variable is whether the tool is available as a stand-alone installable tool, or whether it is web-based. The xMLST tools page contains information about several commonly used tools.

12.3 Factors to consider

12.3.1 How is allele calling performed?

The nature of the starting material used for cgMLST typing, ie. reads or draft assemblies influences how the calling process and therefore which algorithms for calling can be used.

12.3.2 Assembly based methods

Loci defined by a cgMLST scheme usually are delimited by a start and stop codons, eg. coding units. The size coding genes and thus of alleles might exceed the read lengths obtained by NGS. This is most likely why most cgMLST typing tools are assembly based.

Verifying that start-stop codons are recovered prior to allele typing ensures that single gene units are compared to the scheme. However, failure to detect some loci may occur if there is a lack of a proper start-stop codon due to genome mis-assembly or assembly fragmentation, where eg. the start and stop codons might occur on different contigs or be frame shifted. Most software used for cgMLST typing flag loci for which there are potential loci matches but from which compliance with the start-stop codon condition failed. This quality assurance control might provide an indication of poor typing quality due to insufficient assembly quality, particularly if many loci are flagged as such. Those flags are usually visible in the summary table or can be output as a separate file. Note that new alleles in ChewBBACA are flagged as NEW-xx the first time it is encountered, the "NEW-" part must be removed as well as other indicative flags prior to computing distances matrices.

BLAST based calling methods are usually used for assembly based cgMLST typing. Allele calling using BLAST methods employs the similarity of the sequences both in alignment and size (length), with a predefined threshold criteria for allele calling. Performance is generally optimized for example by using a hierarchical blast search strategy, in several passes. It can for instance allow for rapidly selecting CDS that perfectly match scheme alleles, which can be followed by a similarity search for the remaining CDS, thereby determining if new alleles should be called or if loci must be considered as missing (Silva et al. 2018). Alleles are usually considered as missing if they differ from a given length and similarity/homology threshold to the previously identified alleles, eg. within 98% of identity and 98% of the total length of a known allele. This implies that if not all the allelic diversity has been recovered during scheme creation, some loci might be flagged as missing while they would have been found during reanalysis at a later stage when the database had been augmented. This, because the nomenclature allele database has been populated with increasingly divergent alleles, as illustrated in Deneke et al. (2021).

12.3.3 Read based methods

MLST callers (eg. reviewed in Page and al. 2017) that employ reads as input might be used for cgMLST based typing. MLST callers that use this approach map the reads to a set of reference alleles, and evaluate the "stack" or pileup of reads that map to each allele and use this to determine the type (eg. MOST: Tewolde et al 2016, SRT2: Inouye et al. 2014). The frequency of variants in the pileup are often used to control for potential contamination and/or ambiguous allele matches. Read mapping with targeted local assembly of reads mapped to the alleles of the schemes prior to typing allows for verifying if alleles in the isolate are complete. This is an approach that has been developed in ARIBA: Hunt et al. 2017). However, MLST callers that use read mapping approaches usually require substantial compute resources when large cgMLST schemes are employed. Therefore, alternative tools, designed to specifically handle the large cgMLST schemes might be better suited to the task (Feijao et al. 2018). Kmer based methods allow rapid and low compute resource demand in comparison to mapping based typing methods. They allow for comparing the kmer composition of the alleles in the schemes that are transformed into a hash database, to the kmer composition of the reads. The general idea is that an allele is typed when a representative allele of scheme at a given locus maximizes the number of kmers that is also recovered in the reads. There are different variations of kmer typing algorithm , eg. kmer indexing and counting of kmers at the middle of the reads (Gupta et al. 2016) and kmer voting (Feijao et al. 2018)

12.3.4 Nomenclature storage

Nomenclature is hosted on web servers. It is possible to download schemes for local usage, however, synchronisation of eventual new discovered alleles that are called locally is not always straightforward. Some nomenclature servers allow for directly synchronizing schemes (eg. Chewie-NS) while other offer the possibility to submit new alleles via API (eg. BIGsdb), while other appear to have disabled the possibility to synchronize schemes or submit new allele calls via API, in which case the only alternative to update those schemes would be by running the analyses directly through their platform (eg. Enterobase). Consequently, choice of analysis tool might also be influenced by local constraints and aim, eg. If the aim is to analyse data locally, using a nomenclature server that allows continuous synchronisation could be considered, or optionally hash-cgMLST solutions to share data with other labs if needed could be adopted.

CHAPTER 13

Serotyping

13.1 Methods

Serotyping has traditionally played an essential role in determining species and subspecies, and has been used for epidemiologic classification down to subspecies level. The method is based on serological typing of the cell surface antigens of the bacteria. A serotype (syn. serovar: International Code of Nomenclature of Prokaryotes, 2019, p134) corresponds to the combination of surface structures or antigens. This has been proved to be an effective way of discriminating groups of bacteria, with some serotypes being host specific and others associated with virulence intensity (high/mild) (CDC). Moreover, further research experiments have associated serotypes to particular features such as pathological properties, susceptibility to antimicrobials and niche distribution (eg. Ørskov, F. and Ørskov, I. 1992, Yang, X. et al. 2015, Lee, S. et al. 2018, Zoz, F. et al. 2017).

Traditional serotyping (wet lab) relies on the detection of cell surface antigens by agglutination assays using species-specific antibodies (see the Species specific sections for which ones). Serotyping has been used in epidemiology since 1960 in the wet-lab to detect *Salmonella* outbreaks, and as per today more than 2,500 have been described in this species. As the expression of surface structures of each pathogen are coded in their genome, molecular typing methods based on the detection/amplification of certain alleles or DNA sequences have been developed (eg. Beaubrun et al. 2012, Borucki et al. 2003, Iguchi et al. 2015). The advent of WGS technologies brought a higher discriminatory power for genetic clustering without losing the ability to link genomic information to previously available knowledge. This has led to the implementation of a gradual technological transition and to the need of using WGS data for serotyping. Some studies have compared traditional serotyping with WGS-based serotyping (eg: *Salmonella*, *Escherichia coli*, *Listeria monocytogenes*).

Whole-genome based serotyping can be done in different ways:

- By emulating laboratory methods such as presence/absence DNA-fragments detection PCR (see ePCR section)
- By detecting alleles within a set of antigen-genes loci (serotyping performed similarly as MLST finding, see *MLST method description*)
- By specific genes identification, by either mapping or BLAST searches towards serotype determinants (similarly to AMR and virulence finding.)

In many cases, the tools used for MLST and AMR finding can also be used for serotype finding. When marker genes are associated to at set specific antigens are defined, it is possible to determine serogroups based on the presence/

absence of the PCR amplification pattern of a combination of specific gene markers (eg. Doumith et al. 2004). PCR based serotyping can be bioinformatically emulated (ePCR, in silico PCR), and serotypes can thus be inferred from whole genome sequencing data. In such cases presence / absence of genes can be detected using blast in conjunction with a set of rules determining when to consider a match presence or not (Eg. as performed on assemblies in LisSero for ePCR serotyping of *Listeria monocytogenes*, or *In Silico* PCR for fliC and filB alleles of the H antigens for *Salmonella* serotyping with assemblies in SeqSero1.

CHAPTER 14

Virulence and AMR Detection

14.1 Finding methods

Disease surveillance has as its ultimate goal the decrease of human (and animal) illness. This is done not only by the rapid detection and control of disease outbreaks, for which WGS typing methods represent a relevant technological advance (see Serotyping section), but also by the identification of phenotypically-relevant markers (and their changes at the population level), such as virulence- or antimicrobial resistance-associated genes. Therefore, the analysis of the virulome (complete set of virulence genes) and the resistome (complete set of antimicrobial resistance genes) is of extreme relevance in the context of surveillance and outbreak control.

Differences at the genome level involving point mutations or presence/absence of certain loci may have great impact on a pathogen's behavior, and consequently on the disease. For example, presence of *tetO* and point mutations in *gyrA* have been associated with increased resistance to tetracyclines and fluoroquinolones in *Campylobacter jejuni* (Fiedoruk et al. 2019). Moreover, specific genes, such as those involved in the adhesion to human cells or those related to the efflux of certain molecules, have been particularly associated in many microorganisms with virulence and antimicrobial resistance, respectively (Poimenidou et al. 2018, Anbazhagan et al. 2019, Vieira et al. 2017). These genomic features can be acquired not only by vertical evolution, but also by horizontal gene transfer. For instance, the existence of plasmids and transposable elements allows the interchange of genetic material between distantly related lineages (reviewed in Gyles and Boerlin, 2014), thus contributing to the introduction and expansion of new virulence- or resistance-related phenotypes in some lineages. This may contribute to the emergence of different phenotypes which may impact, for example, the pathogenic behavior and the host range. Therefore, there is a need for constant surveillance of the resistome and the virulome in bacterial pathogens.

Similarly to what occurs with molecular typing, virulome and resistome analysis relies on the assessment of the presence of genetic traits (specific genes or mutations) which have previously been associated with relevant phenotypes. In the pre-WGS era, this search could be performed, for example, by amplification and detection of specific target genes. Such an approach could be particularly challenging when encountering unexpected genomic changes which could prevent the amplification of the region of interest, or by the presence of horizontally transferred genes which would not be detected. By providing information at the whole-genome level, WGS can bypass these issues as information is expected to be provided independently of the genetic variability of the sample. With WGS data, the identification of genes/alleles of particular interest can be performed by comparison of the genome of the sample to a database comprising precisely the set of genes of interest. In the particular case of the virulome and the resistome, there are public databases where those sets of genes are already available and programs which automatically perform this search.

14.2 Database resources

Several databases exist for both antimicrobial resistance-associated genes and mutations and virulence genes. These differ in their content and curation procedures, and may therefore produce different outputs when used within the same tool. Some databases have species-specific subdatasets, such as the PointFinder and VirulenceFinder databases. Other databases have more comprehensive content, such as MEGARes, CARD, and VFDB. A user should be cautious when selecting a database, and have knowledge about their limitations and content, as it is only possible to identify the genes/mutations that are present in the database. Examples of predefined resistome databases for bacteria include:

- Bacterial Antimicrobial Resistance Reference Gene Database
- The Comprehensive Antibiotic Resistance database (CARD)
- ResFinder
- PointFinder (detection of chromosomal point mutations associated to resistance)
- MEGARes

Examples of predefined virulome databases for bacteria include:

- Virulence Factor Database (VFDB)
- Virulence Finder
- VirulenceDB
- PathogenFinder

14.3 Tools

ResFinder, PathogenFinder and VirulenceFinder are associated with web sites that have the same name as the database where searches for genes of interest can be done. ResFinder has the option to search for both acquired resistance genes, as well as resistance-associated chromosomal mutations. The above tools also exist as command-line tools, which can be implemented into workflows and pipelines. While these web- or command line- based programs rely on their respective databases, other programs have been developed to perform a broader search for the genes of interest by integrating several of the above-mentioned databases. Examples of such programs are

- ABRicate,
- ARIBA and
- AMRFinderPlus.

Moreover, ABRicate and ARIBA allows the user to create their own databases. database.

The programs mentioned above generally work in one of two modes, either via a sequence search using BLAST where the assembled genome is compared to a database (ABRicate, AMRFinderPlus, ResFinder), or via mapping reads to a database (ARIBA). In both cases the results from the search or from the mapping is evaluated and interpretation leads to conclusions regarding virulence/resistance. Exceptions to this are PathogenFinder, which applies prediction models to determine the pathogenic nature of the isolate, and the most recent versions of ResFinder and PointFinder, which have incorporated KMA allowing the k-mer based alignment of genomic reads (no need for genome assembly) on the database.

14.4 Interpretation of results

Regardless of the tool, the main output provided by this kind of analysis is the presence/absence of gene of interest or mutations (i.e., genotype) and their potential impact on the phenotype (e.g., increased virulence, antibiotic resistance).

This output is usually provided in tabular format (e.g., text files, which are useful for automated report generation and downstream applications), in combination with additional output files for better data visualization and interpretation (e.g., interactive QC color codes, graphics, etc). This is true for both the online and command-line versions of the tools, where the command-line version often has the option to produce additional output files. The tools can thus provide a lot of information, and it is important that the user spends time on understanding the results to ensure that there is no mixup with regards to what the results mean. For example, ARIBA may produce reports that scale several hundred rows of data for a single isolate. This is due to the extensive quality control of each gene and/or variant identified, where ARIBA supplies a "flag" to describe the success or failure of the process. Each flag has its own interpretation, which the user needs to be aware of to interpret the results correctly. Reports of such scale are therefore not meant for in-depth human reading, but rather for automatic handling and interpretation by set rules. This is already supplied with ARIBA, as it can interpret and summarise the results from several isolates with one line of code. The detailed output also allows the user to find and implement their own rules that can satisfy the needs for their situation.

Due to the differences in output reports, comparing results across tools may be a difficult task. The hAMRonization tool addresses the issue of comparing results from several AMR gene-finding tools. hAMRonization is a parser tool that combines the output of several AMR gene-finding tools and generates a standardised AMR gene report, easing interpretation and comparison of tools.

CHAPTER 15

Clustering of xMLST data

Once typing information from xMLST methods have been obtained, it is quite common to cluster isolates on their profiles. In order to be able to do clustering, a pairwise distance matrix of allelic differences has to be obtained. It is usually calculated from the hamming distances, the number of pairwise allelic differences between isolates. Missing loci in one or both isolates are commonly omitted from the distance calculation. Dissimilarity matrices are obtained from normalized distances matrices (scaling the values of the distance matrix to the interval [0-1]). Calculating distances matrices is often integrated in the workflow provided by commercial tools and analyses web platforms. The tools section below describes other tools that may be used.

15.1 Commonly used clustering methods for (cg)MLST data

Clustering of xMLST data is used to classify groups that are more similar to each other than to other groups. Because it is assumed that the isolates that group together are more closely related, it is expected that if a distance measure makes it possible to to separate between very similar isolates (i.e. it gives high resolution), it can be used to identify isolates belonging to a single (sub)lineage or to a single outbreak.

Agglomerative hierarchical clustering is used to group isolates based on their similarity, and by extension, it is assumed that highly similar isolates are closely related. Compute requirements of hierarchical clustering methods are generally low, when compared to phylogenetic inference, and therefore it is possible to analyze a large number of isolates. Hierarchical clustering algorithms work iteratively, combining groups that are more similar to each other at each step, starting by considering each isolate as a single group. The type of linkage indicates how the distances between groups are (re)calculated at each iteration. Good introductions to agglomerative hierarchical clustering and linkage types can be found in Chap1 in Machine Learning with R, the tidyverse and mlr and in the book Practical Guide to cluster analysis in R.

In short: **Single linkage** appears to be the most frequently used linkage type when doing hierarchical clustering with cgMLST. In this linkage method, the distance between groups is the smallest distance between two isolates in the two groups. Single linkage might have been chosen as the method of choice for surveillance, as it produces long clusters that may better reflect clonal expansion of bacterial pathogens. In **average linkage** (equivalent to UPGMA), the distance between groups is calculated as the mean distance of all members of each group to all members of the other group. In **centroid linkage**, the central point of each group is calculated, and this centre is used as distance between groups. **Complete linkage** produces more compact groups as the distance between clusters is defined as

the maximal distance between all pairs of elements belonging to each group. **Neighbor joining** is also a type of hierarchical clustering that is frequently used to reconstruct phylogenetic trees (unrooted tree with a topology that minimizes the total tree length, see phylogenetics section).

Clustering can also be based on Minimum-spanning trees (MST). MST is a type of tree graph that connects isolates by the shortest possible distance, pairwise distances representing the dissimilarity between isolates, that is frequently used to represent population structure in epidemiology (Salipante and Hall 2011). Distance thresholds are used to delineate between clusters. Recently, Zhou et al. (2020) developed a hierarchical clustering method for cgMLST data, that calculates and uses a MST to assign bacterial genomes in real-time to the different stable hierarchical levels of population structure of bacterial pathogens that have been previously identified using a seeding dataset of representative bacterial genomes.

It is important to consider that using different distance metrics and different linkage types or different clustering methods may produce different grouping hierarchies. Because it is assumed that the similarity was representative of the relatedness between individuals, this can therefore lead to different representation of the relationships between individuals, see eg. the different hierarchical structure when the same data used with different clustering methods are visualized with dendrograms: image 15 in a Hierarchical clustering presentation by Shawn Hopkins. Note that the uncertainty associated to this hierarchical clustering reconstruction is rarely evaluated, neither for linkage based clustering nor MST clustering, while this could (and probably should) be done by eg. using bootstrapping methods (see eg Salipante and Hall 2011, Yu et al. 2019)

Moreover, while the clustering provides an indication of the hierarchies of groups of individuals, it does not indicate at which level of the hierarchical structure individuals can be considered as belonging to a single outbreak cluster. Using a distance threshold or cutoff value is frequently employed for this purpose. The clustering threshold can be defined on data with known epidemiological links using a longitudinal study (as done for SNPs in Walker et al. 2013). Unfortunately it appears that might not possible to define an universal cutoff value that would allow to successfully discriminate between epidemiology related VS non related clusters in a single analysis (eg. Henri et al. 2017, Chen et al. 2016), and methods that allow comparison of workflows might be used to assess one's ability to produce concordant results with other institutions (see Coipan et al. 2020).

Note: Many other and/or less traditional clustering methods can be employed in surveillance and to define lineages. An example of such use is presented in PopPUNK (Lees et al. 2019). PopPUNK employs density-based clustering (DBSCAN/HDBSCAN) of core and accessory distances computed by Minhash sketching (Broder 1997).

15.2 Visualisation of clustering

The results of the hierarchical clustering (eg. exported as nexus file) can usually be visualised with MST or dendrograms/phylogenetic trees graphical viewers eg. FigTree or GrapeTree (Zhou et al. 2018 provide a list of software that are convenient for visualising results given a large amount of isolates). Commercial softwares such as Seqsphere+ or Bionumerics and web-platforms such as Enterobase provide integrated tools to visualize your results. Libraries available in programming languages such as R or python are also good alternatives, and can be conveniently used if you wish to produce non-standard graphs to better suit your needs.

15.3 Tools (non exhaustive list) - See also species specific tools

Allele calling - assembly based

- chewBBACA (Silva et al. 2018)
- chewieSnake (Deneke et al.2021) - hash-cgMLST (allele calling pipeline with Chewbbaca, allelic distance matrix computation, clustering and report)
- Bionumerics (commercial)
- Seqsphere (commercial)

- LOCUST (Brinkac et al. 2017) (BLAST based)

Allele calling - sort-read-based

- ARIBA (Hunt et al. 2017) (combined mapping/alignment with local assembly, although not specifically designed for cgMLST purpose)
- MentaLIST (Feijao2018) (kmer based)
- stringMLST (Gupta et al. 2017) (kmer based)
- MOST (Tewolde et al 2016) (mapping based)
- SRST2 (Inouye et al. 2014) (mapping based)

CHAPTER 16

SNPs and variant calling

16.1 What is SNPs and variant calling?

Genomic variants are polymorphic sites within segments that are "identical by descent", i.e. are genomic regions or positions which originate from a common ancestor, but differ between the different samples/isolates/species at study. Such differences can correspond to single-nucleotide polymorphisms (SNPs), insertions or deletions ([INDELs), copy number variations (CNVs), duplications, translocations or inversions. All these types of variants are extremely relevant for comparative genomics analyses. Nevertheless, given the higher rate of point mutations and their lower complexity, SNPs are the most commonly used variants in comparative genomics. SNP or variant calling corresponds to the identification of each of these polymorphisms in the dataset, which is normally coupled with additional downstream filtering steps to remove regions that can bias the clustering of the different samples, such as recombination-prone regions, specific homoplasic sites or SNPs associated with antibiotic resistance. The relevance and impact on the analysis of these additional filtering steps is contingent on the species (or lineages) under evaluation.

16.2 SNP calling by mapping

SNP-calling by mapping (i.e,. mapping the reads against a reference genome sequence) implies the choice of an adequate reference sequence (see species sections for details). This approach has the advantage of being able to use the read coverage depths, or the proportion of mixed alleles to calculate the confidence with which a given polymorphism is called (Olson et al. 2014). Variant calling results are usually output in VCF format, which indicates for each variable position the information relative to the respective coordinates in the reference, the reference allele, the alternative allele (single nucleotide for SNPs, multiple nucleotides for INDELs), and other parameters. A posterior variant filtration step is always recommended. For that, the minimum number of reads that mapped to the reference, the proportion of reads that differ from the reference, the base sequencing quality, as well the proportion of mixed alleles can be used (eg. as used in Snippy, and the underlying tool Freebayes). Although SNP-calling is performed independently for each sample, the ultimate goal of this step is to obtain a multi-sample SNP alignment/matrix. This can be achieved as far as all samples were compared against the same reference sequence. In this case, some pipelines, such as Snippy, offer the possibility to easily combine (and filter) SNP data from multiple samples, providing all the necessary files for subsequent clustering/phylogenetic analyses. Noteworthy, these tools can additionally provide useful functionalities, such as consensus sequence generation and SNP annotation, which may be relevant at several levels beyond the clustering (e.g., rapid detection of mutations of interest, etc).

NOTE: It is important to read the recommended guidelines of each variant caller before setting all the parameters. The values that work for one program may not be the best for another one.

16.3 SNP calling by multiple alignment

SNP calling by multiple alignment intends to produce core multiple alignments for subsequent clustering/phylogenetic analyses. This method takes assembled genomes of multiple samples as input, and generates a "core" alignment, in which the core is defined as the proportion of the genome that is common to all isolates included in the analysis. For this reason, the choice of the dataset will greatly influence the resolution of downstream phylogenetics, i.e., a highly diverse dataset tends to yield shorter core alignment, thus reducing the discriminatory power among closely related isolates, since there is less core genome to gather SNPs from. Although several programs are available for genome alignment (e.g., Mauve, Muscle), further steps of SNP filtering/masking are usually needed. As such, there are a few solutions that integrate alignment, SNP filtering and even phylogenetics. For instance, Harvest is a suite that allows the quick analysis of thousands of sequences, enabling variant calling, recombination detection, and phylogenetic tree visualization. It integrates Parsnp, which uses short sequences of the genome that are unique and shared in all the genomes (Maximal unique matches: MUMs) under study and thereafter extend the area of the alignment recursively across all the genomes (Locally collinear blocks LCNs). Aligned nucleotides (and gaps) which are not unique at position X across samples that are defined as variants. Exporting the multiplement (multi-fasta) or as variant VCF format requires a reference as coordinate system (by default it is chosen as the first reference in the alignment). The reference cannot contain gaps, so if some of the isolates have sequences that are not represented in the references, those will not be included in the output, therefore, according to which reference that is used, there might be small variation in what is given as output with an otherwise identical dataset.

Note: it is possible to create an MSA of the whole genome (ie. with progressiveMauve), however further processing is challenging as most softwares do not handle multiple-alignments formats where the number of sequences aligned is not identical. If you are working with an epidemiologic group, the samples are assumed to be very closely related and therefore even a core-alignment with Parsnp will represent a substantial fraction of the genome size.

16.4 SNP calling using kmers

K-mer-based SNP calling methods rely on the comparison of k-mers between different samples in order to detect and identify polymorphic positions and subsequently perform clustering/phylogenetic analysis. For this reason, these methods can be used on assembled genomes or directly on the genome sequencing reads. Programs able to perform all the analysis from k-mer definition to clustering of multiple samples have been developed, such as SKA and [Ksnp3[(https://academic.oup.com/bioinformatics/article/31/17/2877/183216). Briefly, each sequence is sliced into k-mers of a specified odd-length and each k-mer is then compared between the different samples searching for their differences. In both programs a table of common k-mers allowing one central difference is the basis to compare samples and reported SNPs result from the central differences of k-mers. Because of this definition it is not possible to detect variants composed of successive SNPs, and k-mer-based SNP calling loses power with increased sample variability. As these k-mer comparisons may be performed in the absence of an assembled genome, SNPs can be provided without coordinates. Nevertheless, both programs have the option of mapping back k-mers to an input reference genome which gives the position of the SNPs. Of note, Ksnp3 provides the possibility to download genome annotations from GenBank.

16.5 hqSNP vs non-hqSNPs

"All SNPs are equal, but some SNPs are more equal than others". hqSNPs (high quality) are SNPs usually detected by a combination of methods, usually two, and where methods reported congruent SNPs, that were not flagged as problematic by the quality filters of each method (robust SNPs). There might be some variations in the variants found by

different methods, according to the detection criteria of each particular method. This can e.g. arise because multiple alignment is not perfect, alternative local alignments may be equally possible and therefore different alternative variants can be detected by the different methods. Therefore, MSA-based and mapping-based methods are likely to detect a set of variants that is not totally overlapping. For further explanation, a good overview of the different methods can be found in this paper (Olson N.D et al. 2015). Algorithms underlying those methods are explained here, and (Canzar, S. and Salzberg S.L. 2015) and some tutorials online from the Broad Institute can be found here.

CHAPTER 17

Phylogenetic analysis

17.1 Introduction: History and purpose

Phylogenetics is the study of relationships between organisms based on common ancestry (vertical descendance). Historically, phylogenetics was developed to contribute to taxonomical classification, as it was expected that species classification into genera, families or superior order would be made on the basis of shared common ancestry. Therefore the first phylogenetic studies attempted to reconstruct relatedness relationships based on matrices of shared morphological characters. The field of phylogenetics thereafter evolved to molecular methods, ie. with the first evolutionary studies of allozymes. Phylogenetics methods have evolved in parallel with the development of sanger sequencing and next generation sequencing, with the wishful/idealistic idea that being able to use a larger amount of character states (ie. nucleotides present at each position of the genome) would provide the utmost possible resolution when determining relationships between organisms. The ability of delineating and identifying isolates belonging to groups of highly related pathogens using NGS molecular phylogenetics has become an invaluable tool in epidemiological surveillance (Rife et al. 2017) and outbreak investigations and can contribute to better understanding of the factors driving the emergence and spread of infectious diseases, either at macro or micro temporal and geographic scales. In this regard, phylogenomics is nowadays frequently employed in bacterial epidemiology. However, it is not employed as first intended: to evaluate the relationships between species. Most of the focus of phylogenetics in epidemiology is on reconstructing "genealogies" within populations, lines of descent from most recent common ancestor (eg. determining the source of a food-borne pathogen outbreak).

17.2 Terms

In molecular phylogenetics, **homology** is defined as a similarity between sequences due to the sharing of an ancestral sequence. Sequence homology can thus arise by duplication events of an ancestral sequence, or by vertical descent (eg. divergence from an ancestor during speciation), also called **orthology**. In phylogenetics reconstruction methods, the hierarchical relationships between taxa are derived by comparison of sequence/loci that are assumed to be orthologous.

The positions in a **multiple sequence alignment (MSA)** used for phylogenetics analysis are thus assumed to be orthologous. **Gaps** (or indels: insertions and deletions) are artificially introduced markers to allow alignment of sequences or sequence portions that differ in length.

Multiple sequence alignments can be extracted from a **whole genome alignment (WGA)**, however, note that due to genome reorganisation, and the complexity of the alignment, the resulting MSA is a concatenated set of several alignment of supposedly orthologous sequences. (See SNP and variant calling section).

Single nucleotide polymorphisms (SNPs) are the representation of the variants loci/positions in MSA.

Taxon (pl. taxa): term derived from taxonomy to represent a group of organisms sharing an identical taxonomic ranking (eg. species, genus, family, etc . . .)

Operational taxonomic unit (OTU): a group of organisms under study, (ie. species, population, clonal lineage, family, . . .). The concept of OTU is most convenient in phylogenetics, as it allows for defining groups of organisms we are referring to without prior knowledge of their taxonomic level.

Clusters in phylogenetics: Clusters are aggregates of similar isolates. The concept of cluster allows to define a group of things with similar properties. In molecular epidemiology, clusters can be defined eg. based on pairwise distance measures), where pairwise distance thresholds (cutoffs) often are used to determine which isolates belong to a specific cluster. In phylogenetics, defined clusters MUST be **monophyletic**.

A **monophyletic group (or clade)** is a group where all OTUs that form this group are linked by a single ancestor (decipited by a single node), and all the descendants of this ancestor belong to this singular group. The monophyletic concept is often used to describe the relations of the OTU to each other in phylogenetics. It contrasts with the concept of **paraphyletic group**, which allows describing OTU groups sharing a common ancestor but where not all descendants are included in the group. See this figure for descriptive terms of OTU grouping.

A **lineage** is a line of descent, including all the ancestors and all the descendants (leaves/tips), from the most ancient ancestor that defines the lineage.

Phylogenetic trees are tree-like graphs (non-cyclic) that are used to visualise the reconstructed relationships between organisms. The **topology of the tree** (ie. the branching pattern of the tree) is expected to reflect how OTUs are related to each other. Reading and interpreting phylogenetic trees is not always as straightforward as it seems (Baum 2008, Gregory 2008). Here we only present some few of the several terms and concepts that are available to interpret phylogenetic trees and describe relatedness among OTUs. In short, each ancestor (depicted at nodes of the tree) are expected to have two, and only two (bifurcating trees) descendants represented at the tips (also called terminal nodes or leaves). The **most recent common ancestor (MRCA)** is the term used to describe the ancestor that is most recent to all OTU we are referring to. When phylogenetic reconstruction methods allow to evaluate the support of the splits (one ancestor to two descendants), and when the support of the split is low, the branching pattern of the tree is not well resolved. This could be visualised as polytomies (or **multifurcation**: when one ancestor diverged into more than two descendants). Note that in most commonly used phylogenetics inference, tips always are the isolates under study, and ancestors are always reconstructed (inferred), ie. ancestors can never be sampled, they are purely a reconstruction of ancestry of your isolates.

Note that when we are speaking about the succession of ancestors, we assume a direction of evolution in time, this implies that the phylogenetic tree is **rooted**. The root of the tree allows for providing a direction (the order of splitting events) on the tree. However, when the tree is **unrooted** (not rooted) the order of the branching pattern of the nodes is not defined (ie. we do not know which node is the ancestor of which node).

A specific terminology is used to describe the different types of trees. A **dendrogram** is a bifurcating graph tree that represents a hierarchical structure but does not necessarily represent evolutionary relationships, it is basically a generic term for tree-graphs. A **cladogram** is a dendrogram where the hierarchical structure indicates a common ancestry. Branch lengths do not provide indications about evolutionary distance between OTUs. A **phylogram** is a phylogenetic tree where the topology indicates the ancestry history and the branch length represents the amount of evolution (or evolutionary distances) between OTUs. **Ultrametric** trees are trees where all branches have equal length to the root. Beware that those ultrametric trees must be reconstructed under the assumption of a **molecular clock**, and that all tips are contemporary isolates. Ultrametric trees are very sensitive to deviations of those assumptions, in which case the branch lengths may not indicate the real amount of evolution of OTUs since their last MRCA. Ultrametric trees are generally poorly adapted to represent evolutionary relationships of rapidly evolving organisms with short generation times, where lineages may evolve at different rates, most particularly if you mix contemporary isolates to isolates previously stored in the freezer for many years. **Additive trees** are trees where the branch length represents the

amount of evolution, but do not require the assumption of molecular clock (note that ultrametric trees are a particular case of additive trees).

The **molecular clock hypothesis** is the assumption that sequence divergence occurs at the same pace among the studied lineages, and therefore that the genetic distance is proportional to time elapsed since divergence from MRCA.

Phylogenetics and phylogenomics derived terms: The joint study of phylogeny and population genetics has given rise to the term of **Phylodynamics**. In epidemiology, this can be defined as "the study of the interaction between epidemiological and evolutionary processes within and among pathogen populations" (Rife et al. 2017). **Phylogeography** is the study of spatial arrangements of genealogical lineages within and among conspecific populations and closely related species." (Avise 2010 in Avise 2016).

17.3 Examples of usages of phylogenies in molecular epidemiology

17.3.1 Detecting and investigating outbreaks

Finding the source of some food-born pathogen causing an outbreak can be challenging. A patient might have consumed food originating from diverse origin, and the usage of preprocessed ingredients entering in each individual meal composition might be nearly impossible to trace back. Hoffmann et al. (2016) provided an example where Maximum likelihood (ML) phylogenetic analysis of WGS data of a preselected set of isolates (geolocated isolates: outbreak isolates, environmental isolates and historical isolates) that were identified as similar by PFGE, allowed to pitpoint toward the most probable origin of Salmonella Bareilly outbreak. The pathogen responsible for the USA outbreak, was associated with tuna imported from a Indian fishery. They also demonstrated that the method had a high resolution to distinguish between geographically distinct lineages, which showed the high potential of those methods to monitor evolution and transmission routes of either food or environmental pathogens to new niches/hosts.

Chen et al. (2016) presented an interesting case of distance based phylogeny (using cgMLST data) of Listeria monocytogenes outbreaks. They demonstrated that in somes cases, it is necessary to investigate the hierarchical levels of population structure sequentially, first by identifying the main lineages, and thereafter refining analysis by working within lineages. This to successfully discriminate between isolates belonging to closely related outbreaks.

17.3.2 Tracking transmission routes of pathogens at different geographic scales

Pathogen transmission can be studied at different geographic scales: from worldwide patterns of strain dissemination to direct transmission events eg. between hosts (see Croucher and Didelot 2015). At intercontinental scales, the focus is mainly on detecting changes in the global population structure of the pathogen, which can affect the epidemiology of the disease associated to the pathogen, or can eg. help pinpoint possible routes of transmission (eg. through food import) and can contribute to narrowing the area of where epidemiological investigation should be directed. At a microgeographic scale, transmission routes can help identify reservoirs of persistence in the environment (eg. food processing) and contribute to improvement of routines and disinfection practices.

Understanding evolution, spread and transmission routes of antimicrobial resistance

AMR strains have been associated with increasing usage of antimicrobials worldwide. Usage of antimicrobial induces a selection pressure on the bacterial pathogens, which can lead to change in strain population structure (eg. population replacement of some lineages by others) and transform the epidemiology of the disease. Phylogenetic analysis can contribute to the understanding of the evolution and spread of resistant phenotypes (Klemm and Dougan 2016). Wong et al. (2015) identified inter and intracontinental transmission of multidrug resistant (MDR) Salmonella typhi H58 clades. The H58 clade emergence was identified as recent (~30 years ago). They identified multiple transmission events between Asia and Africa and demonstrated that mutations in a region of GyrA, involved in quinolone resistance, evolved independently on several occasions. An indication that selection at this gene was driven by the geographical pattern of antibiotic usage. Moreover, the expansion of the H58 lineage leads to replacement of other S.typhi strains,

and therefore is likely to transform the epidemiology of disease, which justifies the necessity for long term surveillance of this pathogen. This was further supported by the increasing number of reported MDR resistant cases, which coincided with the identification of previously unrecognized ongoing outbreak.

Evaluating effectiveness of interventions and actions to fight persistence or reintroduction of pathogens from local reservoirs at a microgeographic scale

Phylogenetic analysis at a microgeographic scale has been used in combination with contact tracing of patients to support hospital infection control of nosocomial strains of Carbapenems resistant Klebsiella pneumonia strains. This allowed to identify potential reservoirs of infection (Cella et al. 2017). The authors reconstructed a time-scaled phylogeny and inferred the geographical location of ancestral lineages through Bayesian phylogeography. Their results indicated that the ancestor of the strain could have been introduced at the hospital as early as 6 years previously to their study, indicating that the bacteria was likely transported in the different parts of the hospital by employees, and that some strains might have been maintained at reservoirs related to ie. endoscopic procedures. This type of study can help detect which procedures and routines need to be improved, and where intervention measures lack effectiveness.

Fagerlund and al. (2020) studied the genetic diversity of Listeria monocytogenes ST9 lineage, and showed that gradually refining their analysis (cgMLST, wgMLST and SNP data), first focusing on the whole ST9 lineage and then on Norwegian clones allowed to increase the discriminatory power of the analyses while providing a background for interpretation of the analyses. Using time-scaled analysis, they were able to estimate the timing of emergence of the two Norwegian subclones that belonged to several meat processing plants. This showed that the emergence of the two Norwegian clades occurred relatively recently (95% HPD 1970-1991, and 1992-2004). Their study highlights that deep sampling is critical for studies aiming at finding the origin of bacterial contamination in food products. Moreover they provide a rich example of the challenges related to the interpretation of contamination routes, cases of multiple introduction events and persistence within production facilities when isolates are closely related.

17.3.3 Improving risk assessment, risk management and assessment of risk prevention measures

The risk associated with exposure to pathogens from a single species might vary between lineages, i.e., a subset of strains from a bacterial pathogen might be more likely to induce disease than other strains (Rantisou et al. 2018). Some phenotypes might confer a higher risk, eg. resistance to antimicrobials, ability to synthetize toxins, ability to grow in specific niches such as determined by pH or specific hosts. Those specific phenotypes might be shared within lineages but some might also be acquired through horizontal transfer, gene gain and loss. The risk of invasive disease might also be associated to host factors (eg. [Klemm and Dougan 2016) and to the evolution mechanisms of the pathogens.

Here three examples are provided on the possible contribution of phylogenetics analysis for these situations.

Risk prevention

As mentioned previously,phylogenetic analyses at a microgeographic scale Cella et al. 2017 example), can be used to assess whether intervention measures against a pathogen are effective and allow adjustment those measures.

Understanding host-pathogen interaction, pathogen evolution Within-host

diversity, within-host evolution and within-host niche adaptation has been observed in bacterial pathogens (Ailloud et al. 2019). Within-host evolution is particularly relevant in regard to the development of antimicrobial resistance, modifications of virulence, host-niche adaptation and for host to host transmission studies (reviewed in Didelot et al. 2016). Phylogenetics can contribute to elucidating the evolutionary mechanisms allowing host shifts in pathogens. Such host shifts are a threat to food safety and public heaths, and have been associated with acquisition of genetic material required for survival into the new host-species in Staphylococcus aureus (Ridchardson et al. 2018).

Risk prevention by flagging phenotypes associated to specific lineages, contributions of phylogenetics to gene-phenotype association

Phylogenetic comparative methods can help study the pattern of phenotypic traits that are epidemiologically relevant. Indeed, phylogenetics can contribute to the ability to discriminate between phenotypes that are associated to specific lineages from those under control of genes that are shared among several lineages. This can eg. be used to flag some lineages as presenting a higher risk than others. Moreover, phylogenetic based methods can be eg. employed to evaluate the heritability of virulence traits (Hassler et al. 2020), which can provide an estimate of the strength of the risk associated with a trait.

Phylogenetics also contribute to the identification of gene-phenotype associations. Genome-Wide Association Studies (GWAS) have been employed to evaluate the association between phenotype and the presence of specific SNPs or genes in bacterial pathogens and pathogen lineages, in the hope that candidate genes discovered might be responsible for the observed phenotype. To disentangle candidates identified through lineage effects (due to linkage-disequilibrium: LD) from the candidates with likely phenotypic effect, population structure needs to be accounted for. This is particularly important as LD between genes is particularly strong in bacteria, due to their clonal population structure as well as to the physical linkage of genes. The population structure can be deduced from phylogenetic trees and incorporated into bacterial GWAS analyses (eg. Lees et al. 2020), or directly incorporated within the GWAS analysis (eg. Collins and Didelot 2018, Saund and Snitkin 2020).

17.3.4 Providing population dynamics estimates to support epidemiological modeling

Phylodynamics (field combining phylogenetics and population genetics a common analysis framework has been used to estimate epidemiological parameters (eg. effective reproduction numbers) of Mycobacterium tuberculosis (Kühnert et al. 2018), (eg. duration parameters) to assist modeling of demographic and epidemiological processes (Saunier et al. 2016, Saulnier et al. 2017, Volz and Silveroni 2018, Baele and al. 2018).

17.4 A short overview of phylogenetic reconstruction methods and considerations

17.4.1 Phylogenetic reconstruction methods

Distance based methods

In **Distance based methods**, any distance measure (eg. similarity between sequences, hamming distances between shared alleles at cgMLST, computed from a MSA/WGA employing an evolutionary model, ANI: Average nucleotide identity distances, distances based on shared k-mer approaches as used in alignment free phylogenetic methods) that is assumed to be minimal when organisms are closely related, can be used as the basis for phylogenetic tree reconstruction. The distance matrix of pairwise dissimilarities is used to perform a hierarchical clustering (e.g. UPGMA: Unweighted pair group method with arithmetic means, NJ: Neighbor joining) which leads to obtaining a single tree.

Character state phylogenetic methods

Character state phylogenetic methods require a multiple alignment of orthologous DNA sequences/genomes. "Nucleotide states" (A,C,G,T and depending on the model used, indels: "-") at each position (column) of the multiple alignment might either be ignored (ie. considered as missing data) or analyzed independently.

Maximum parsimony

Maximum parsimony method aims at reconstructing a phylogenetic tree whose topology minimizes the number of character state changes in the tree, since the last common ancestor. This method does not allow explicit substitution modeling to compute the parsimony criteria (neither allow for multiple substitutions). Those implicit assumptions might be unrealistic (Kapli et al. 2020). No further details concerning this method will be provided as it is not frequently employed in bacterial phylogenetics, but it is described in many introduction-level phylogenetics books (eg. chap 8 in Lemey et al. 2009), for those interested.

Statistical phylogenetics methods

Statistical phylogenetics methods, ie. **Maximum likelihood methods (ML)** and *Bayesian methods** (or approximation of those), use nucleotide states at each position of the multiple sequence/genome alignment to reconstruct phylogenetic trees. A genetic distance is estimated according to an evolutionary model during phylogenetic inference. A set of trees within possible trees (the tree space) is evaluated.

In **ML methods** the likelihood optimality criterion allows for assessing which tree in the tree space is an "optimal" tree. Because the set of all possible trees is often very large given the number of isolates under study, it is not computationally possible to evaluate all possible trees. Algorithms are designed to explore the tree space in search of the best tree: the tree topology that has the highest likelihood to have given rise to the data (MSA/WGA) given the chosen evolutionary model. However there is no guarantee that they will find the absolute best tree among all the possible trees.

Bayesian methods allow estimating the probability distribution of trees and model parameters, and provide estimates of the confidence of inferred relationships (clades), estimates of the evolutionary hypotheses (the evolutionary model distribution) and the data through the posterior (posterior probability distribution). Bayesian methods require specification of prior belief on the evolutionary model parameters and are the most computationally intensive, due to the modalities of the exploration of the "tree space" through sampling and updating of the model parameters with MCMC (Markov chain Monte Carlo). They are more complex to implement than ML methods, computationally heavier and are so far rarely used in molecular epidemiology for routine surveillance of bacterial outbreaks. Bayesian phylogenetics provides a statistical framework for hypothesis testing and allow to incorporate a variety of data types into the phylogenetic modelling (eg. sampling locations, sampling dates that will allow calibrating an evolutionary timeline) and are therefore quite powerful analysis methods.

17.4.2 Some major assumptions to be aware of and to take into consideration

One major assumption of phylogenetics methods is that it describes the vertical evolution of the OTU under study. Genomes evolve through different mechanisms, and this implies that the different sections of the genome, following the specificities of their evolutionary mechanisms (and when those evolution events occurred) might not have the same evolutionary history (ie. do not point to the same ancestral lineage: eg. case of recombinant sequences, nor to the same timeline of evolution to the rest of the genome: eg. case of evolution by gene duplication). Therefore, it is important that phylogenetic reconstruction originates from the comparison of orthologous sequences. cgMLST schemes appear to be designed such as the phylogenetic noise introduced by recombination is minimal, if the recombinant signal is only counted once (eg. in Neumann et al. 2019) but his might still be rather problematic for species with high recombination rates. Detecting and discarding recombinant sequences for methods employing MSA as input data (eg. Whaley et al. 2018) is the most commonly use approach, also this is not without consequences and recombination-aware methods are being developed (see eg. Vaughan et al. 2017).

Genetic distances can be obtained from MSA by explicitly specifying sequence evolutionary models, as eg. done for statistical phylogenetics methods. Evolutionary models allow to make different assumptions about the evolutionary process of the sequences, eg. the rates of substitutions from one nucleotide to another, difference of rates of evolutions between different sites in the genome, and can allow to include different evolution rates for the different lineages (Lemey et al. 2009).

More complex models (with more parameters eg. different rates of substitutions for each nucleotide change) than simpler models are not always necessary to provide better phylogenetic reconstruction (see eg. Kelchner and Thomas 2007).

Interpreting phylogenetic trees requires having an idea of how confident we are in the different lineages obtained during phylogenetic analysis. Support attributed to the groups by resampling techniques (eg. bootstrapping) are frequently used in ML methods. Confidence is assessed directly from the posterior for Bayesian phylogenetic methods. Moreover, robustness of trees inferred by different methods, might be an additional indicator of the strength of the phylogenetic signal.

17.4.3 Which method should I choose?

Choosing a method depends on the biological question you want an answer to, and to the level of similarity between your sequences.

Distance based phylogenetic methods are less computationally intensive than statistical phylogenetic methods, and might be a good choice if you have a large amount of samples to analyse. This is particularly important since the number of possible tree topologies increase rapidly with increasing number of studied OTU, in which case statistical phylogenetic methods might be too demanding of compute resources. Distance based methods produce a single tree. It is possible to access the confidence in the inferred clades by calculating support values using eg. resampling methods such as bootstrapping (see eg. Efron et al. 1996), on the character or loci used to calculate the distance matrix.

Maximum likelihood (ML) are with distance based methods the most commonly used phylogenetic reconstruction methods used in bacterial molecular epidemiology. For distantly related isolates, MSA from concatenated genes are better suited than MSA obtained from WGA of collinear sequences or from SNP typing using a closely related sequence as coordinate system. This is due to the complexity of WGA alignment when organisms are too dissimilar due to the. abundance of gene duplications, genome reorganisation events, and if sequences are too divergent. See e.g., Kapli et al. 2020 for a comprehensive overview of phylogenomic methods for distantly related OTUs, as well as considerations for method choice. Distantly related isolates usually belong to different serogroups, relatedness is higher for isolates belonging to identical sequence type or clonal complexes. Closely related isolates are eg. outbreak isolates.

Note that MSA only represents a fraction of size of the aligned genomes. Therefore the ability to discriminate between clusters of related individuals will not only depend on the quality of the MSA, but also on the fraction of the genome that is compared (that is common to all isolates: core). Different methods can be used sequentially, to gain additional resolution (new MSA) for subsets of closely related isolates.

Note that software that provide different MSA or SNP typing approaches might be optimized for different types of analyses, and different amounts of divergence between isolates under study.

17.5 What do you need to be able to reconstruct phylogenetic trees with WGS data?

For distance based methods a pairwise distance matrix is needed, eg. from alignment-free methods that allow computation of distances based on kmer/word frequencies, or from a MSA/WGA.

For statistical phylogenetics a MSA/WGA/MultipleSNP alignment is needed, eg. concatenated from a set of gene alignments or a whole genome alignment.

17.5.1 Workflow for phylogenetic reconstruction with distance based methods

Obtaining a pairwise distance matrix

A pairwise distance matrix can be obtained from a multiple alignment, or by any other measure where the distance is assumed to represent the amount of evolution between isolates, eg. as perform for alignment free phylogenetic reconstruction where the distance is estimated by kmer difference counts or spaced words matches (see eg. Röhling et al. 2020, Morgenstern 2019)

Phylogenetic tree building

UPGMA (unweighted pair group method with arithmetic mean) produces ultrametric trees, where all OTUs are equally distant to the root. UPGMA assumes that all lineages evolve at the same rate: molecular clock and that all OTU have been sampled at the same time. While this method is still relatively frequently used, this is likely not the best choice for pathogen surveillance datasets. It is therefore not presented here. (For an overview of principle see any phylogenetic textbook or review such as eg. at Sharma et 2018).

The most frequently used distance method to reconstruct phylogenetic trees is the neighbor-joining method (NJ). NJ does not assume a molecular clock and produces a single unrooted tree (Saitou and Nei 1987). The NJ principle is to build the phylogenetic tree iteratively, starting from a star topology and by creating nodes joining OTU with the smallest pairwise branch length and by minimizing the sum of branch length in the tree. At each node created, the distance between the taxa in the node and other OTU is calculated. This is repeated until all OTU are joined by bifurcating nodes (see eg. for a detailed process using 4 OTU [DeSalle and Rosenfeld 2013])(https://books.google.no/books/about/Phylogenomics.html?id=IAcPBAAAQBAJ&redir_esc=y)).

Evaluating clade confidence

Clades can be evaluated using Bootstrapping, as mentioned elsewhere in this handbook.

17.5.2 Workflow for MSA/WGA/MSA-SNP methods

Obtaining a multiple alignment (MA: MSA/WGA/MA-SNP)

One critical assumption is that at each position of the multiple alignment (MA), the characters are assumed to have evolved from a common ancestor (homology) and have diverged from the same ancestral sequence (ie. vertically: orthology). Depending on the nature (gene, SNPs, collinear regions) of the MA, different preprocessing such as removal of recombination between homologous segments methods might be employed prior to phylogenetic reconstruction (see eg. Kapli et al. 2020). There are different strategies to obtain a multiple alignment from WGS data based on either

- The set of presumed orthologous genes that are present in all isolates are extracted from genome annotations, and specialized databases (eg. described in Kapli et al. 2020). A multiple sequence alignment is performed for each gene individually. In bacterial surveillance, alignments of all the genes are generally concatenated in a single MSA/WGA for phylogenetic reconstruction, eg. by exporting the core gene alignment using Roary (Page et al. 2015). Alternatively, it is possible to reconstruct one phylogenetic tree per gene and "reconcile" the obtained set of phylogenies in a single tree using supertree methods (see eg. Boussau and Scornavacca 2020, and Kapli et al. 2020).
- Whole genome multiple alignment is a complex computing problem, this due to eg. genome rearrangement (Henning and Nielselt 2019, Dewey 2019). A strategy that is frequently employed is the "hierarchical" multiple alignment of collinear blocks (segments of the compared genomes that do not contain rearrangements) or a "local" alignment of parts of the genomes that are later merged as multiple alignments (Darling et al. 2004, Darling et al. 2010). Whole genome alignment contains information about genome synteny, and typically includes pangenomes. The core MA must be extracted from pangenome WGA for phylogenetic analysis. MSA/WGA can be either directly obtained (eg. ParSNP, Treangen et al. 2014) or extracted by concatenating the several

collinear blocks that were common (core) in all genomes under study (eg. from extracting the core from a WGA alignment file output from Mauve/progressiveMauve aligner (Darling et al. 2004, Darling et al. 2010) with biopython script or with eg., with harvesttools (Treangen et al. 2014).

- Multiple SNP alignment can be obtained when an identical reference is used to type SNPs for all isolates under study. The reference serves as a coordinate system, therefore a multiple SNP alignment can be reconstructed by concatenating all the SNPs at all the different coordinates of the reference. This can eg. be done with Snippy.

Ensuring that aligned sequences are orthologous

During WGA building, recombinants from homologous regions might be included in the alignment. Those regions must be either deleted or masked (hidden) from the alignment used in phylogenetic reconstruction because those sites are not inherited by vertical descent (non orthologous). The methods developed to detect recombinant loci (see eg, Lai and Loerger 2018) are not universally applicable to all multiple alignments. The methods that use sliding window where positional information of SNPs is crucial to detect hotspots of SNPs that are then considered as recombinants (eg. Gubbins (Croucher et al. 2015), ClonalFrameML (Didelot and Wilson 2015)) are not applicable for detection of recombinants in MA from concatenated genes because the order of genes is arbitrary and intergenic regions are not present. Moreover some methods have been developed to work with relatively closely related isolates (within lineages), eg. ClonalFrameML and Gubbins. Other methods might be better adapted when working with more distantly related isolates within a species (eg. fastGear Mostowy etal. 2017).

Defining an evolutionary model of the sequences (Modeling evolution)

Statistical phylogenetic methods aim at modeling the process of evolution that is evidenced by the SNPs seen between taxa. Therefore it is necessary to provide a evolutionary model, which is a representation of the sequence evolutionary process: the change in character/nucleotide over time in orthologous sequences belonging to two taxa that diverged from a most recent common ancestor (MRCA), to perform phylogenetic inference with statistical phylogenetic methods.

Each evolutionary model is composed of a set of sub-models that represent the different components or biological characteristics that are relevant to describe the characteristics of the sequence evolution. A "minimum evolutionary model" that might be realistic enough to represent sequence evolution for a set of OTUs can be composed of a nucleotide substitution model and a model of heterogeneity rate among sites. Below these are described, as are also several additional models which when used in conjunction with the "minimum evolutionary model" can provide a means to increase the realisms of the evolutionary process modeling.

Which evolutionary model should I choose?

Choosing an evolutionary model might be a daunting task. Therefore, a strategy for "choosing" the model, is actually to perform phylogenetic inference with several models and then evaluate the fit of the model to the data. Fortunately, model testing is implemented and automated in several phylogenetic softwares, thus this does not require you to manually perform several analyses.

Generally, when reconstructing phylogenies of isolates that are very closely related, you can expect that the best fit evolutionary model will be more simple (less parameters) than when you reconstruct phylogeny of distantly related OTU (Lemey et al. 2009). This can eg. be explained because you are less likely to have to model multiple substitutions events and because the sequence nucleotide frequencies are likely to be nearly identical among the isolates unders study.

Nucleotide substitution models

What is a substitution model

Substitutions models are one of the major components of the sequence evolutionary model. Substitution models are a statistical description of how nucleotides evolve from the MRCA to another during sequence evolution; they provide the probability of change of each base into another base (eg. describe the probability of A becoming T over time). For amino-acid substitution models please see phylogenetic books).

Nucleotide substitutions are usually modeled as a random event (ie. occurring randomly at one nucleotide location within your MA). Nucleotide sites are assumed to evolve independently of each other. Through phylogenetic inference, evolution can be modelled step wise (discrete time) or continuously (continuous time, eg. Bayesian methods). Herein we only use "time step" to present the general idea of those methods. At each time step, each site evolves or not, independently from other sites. At each time step, the nucleotide substitution probabilities remain unchanged (time homogeneity assumption). A very good introduction on substitution models can be found here: (Lecture primer by Paul Lewis - Part 1).

The relation between an observed nucleotide change and the "true" evolutionary distance is not one to one, because some types of base changes may be more likely than others. This can be due to a bias in nucleotide composition, or a bias in base biochemical properties that favors some types of nucleotide changes being more probable than others, thus having less evolutionary weight than other changes. Other reasons of the absence of one to one relationship between evolutionary distance and observed nucleotide changes are the occurrence of multiple substitutions events, where only the final state is observed (eg: A -> T -> G), or the occurrence of back mutations (eg: A -> T -> A) or convergence (eg. A -> T and C->T) where sequence evolution did not leave an observable footprint in the sequences under comparison. Reversal and multiple substitutions are more likely if organisms have diverged a long time ago.

Frequently employed substitution models

The most frequently employed substitution models belong to a family of models called time reversible models (REV). The general time reversible model (GTR) can be seen as the most complex (most parameters) of the REV model family.

The most simple REV model is called the **Jukes and Cantor model (JC69)**. JC69 assumes equal probability of change of each nucleotide to one-another and equal equilibrium frequencies of each nucleotide, i.e. 0.25 of each. The slightly more complex model **Kimura two-parameter mode (K2P or K80)** introduced different rates of changes for transitions and transversions, as transitions are more frequent than transversion during evolution. The **GTR model** is composed of different rates of changes for the different nucleotides and different equilibrium nucleotides frequencies (see eg. review in Arenas 2015 and a figure here for the relations between different REV models.) All GTR models, and submodels (nested models) assume that the relative frequencies of each nucleotide are at equilibrium. This means that relative nucleotide frequencies do not change in the course of evolution (also called stationary).

Note that if a MA-SNP is used it might be necessary to provide the number of invariant positions for each nucleotide. This is required to compute the relative nucleotide frequencies of the dataset.

Note: that non-REV models exist. They allow providing a direction of evolution, but those are so far not frequently encountered (see eg. Williams et al. 2015, Woodhams et al., 2015) in molecular epidemiology phylogenetic inference of bacterial pathogens.

Modeling the heterogeneity of substitution rate among sites (Rate variation)

The model of the heterogeneity of substitution rate among sites is the second major component of the evolutionary model.

Not all loci/positions in a MA evolve at the same peace. The rate of nucleotide change can be different (heterogenous), for eg. different types of genes, for different codon positions in protein sequences (rate 3d > 1st > 2d position) and for different parts of proteins such as enzyme active sites. This can be implemented with a rate heterogeneity among sites model.

To specifically define which sites can evolve at different rates, it is possible to define groups of sites, for instance genes, codons, or parts of codons that may evolve at a different rate, and use those for phylogenetic analysis. This is called partitioning Kapli et al. 2020. Partitioning has been subject to controversy, particularly as it is a challenge to identify appropriate partitions *a priori*. See Kainer and Lanfear (2015) for a review of the effects of partitioning to accommodate variation in substitution rates among sites.

Most often, the heterogeneity of substitution rates among sites is modelled without defining a-priory partitions. This is done by assuming that the sites can be categorized and attributed to discrete groups (classes) that evolve at the same rate. The different classes corresponding to each discrete rate are drawn from a gamma distribution (eg. GTR+G4, +G6 +G10), where 4, 6 and 10 indicate the number of rate classes for a GTR model. Six to 10 classes are generally a good approximation for the variability of the rate among sites for intraspecific studies (Jia et al 2014).

However, because assuming that the substitution rate variation follows a gamma distribution has no biological basis, but is rather a convenient implementation means, the question of how to define those rate categories has been subject to discussion in the scientific community, eg. in Jia et al. 2014). Therefore alternative methods of modeling the heterogeneity rate of substitution among sites have been implemented (eg. Free-rate model).

From Strict molecular clock models to models that take into account the variation of the evolutionary rate among lineages and time

The molecular clock hypothesis

The molecular clock (strict molecular clock) is an assumption that all the lineages under study evolve at an identical rate. The evolutionary rate is then proportional to elapsed time since divergence from each MRCA across all lineages, although this does not exclude that different parts of the genome evolve at different rates, as long as the rate for each part is identical for the isolates under study.

Assuming a molecular clock, allow for example for estimation of evolutionary rates, estimating the timing of emergence of a lineage (eg. associated with the origin of an outbreak). This can also help eg. to discriminate between persistent strains and reintroduction events from a common source, and eg. identify practices associated with the appearance of a new strain in a food processing environment (eg. Fagerlund et al. 2020), which in turn can help improve biosecurity measures.

The molecular clock assumption is usually implicit unless other clock models are used.

Correlating evolution rate and time requires calibration of the molecular clock. Tip-dating allows calibration of the molecular clock, it allows rescaling of the nodes into calendar time by linking calendar time units to evolutionary rate and thus easing epidemiological interpretation (eg. Volz and Fost 2017, Baele et al. 2018). Tip-dating is increasingly used for virus outbreak investigation and monitoring (Baele et al. 2018). Time-resolved phylogenies are currently to a lesser extent used for bacterial phylogenetics (but see eg. Fagerlund et al. 2020), but will likely be increasingly common.

Tip-dating to calibrate a molecular clock model during phylogenetic inference (eg. Bayesian inference) is conceptually and methodologically different to calibrating/translating evolutionary time into calendar dates posterior to phylogenetic reconstruction, from the finished tree. Calibrating methods posterior to tree inference include eg. root to tip regression, least-square dating. Those two types of methods that produce“time-trees” differ in regard to uncertainty treatment (Duchêne et al. 2016, reviewed in Duchêne and Duchêne 2021).

Note: Data that contain sampling time information are also referred as “Time-stamped” and “heterochronous” data.

Moreover, providing estimates of the nodes’ age (time) depend on the position of the root (Duchêne and Duchêne 2021 in Ho 2020).

Heterotachy: variation of the evolutionary rate over time and among lineages

Assuming an homogeneous evolutionary rate among lineages when this is not true, may alter the tree topology reconstruction. Heterotachy appears to be more likely the more the lineages are distantly related to each other and can eg. result in a phenomena called long-branch attraction. In this case, lineages that evolve more rapidly than other will appear as to have diverged since a longer evolutionary time than other lineages (overview in eg. Kapli et al. 2020).

Heterotachy is accounted for by using relaxed clock-models. Bayesian inference software may offer s choice between several relaxed clock models: the (auto)correlated relaxed clock (assumption of molecular-clock more similar the more closely related lineages are), uncorrelated relaxed clock and flexible local clock (Ho 2009, Fourment and Darling 2018). Some models allowing accounting for heterotachy have recently been developed for ML phylogenetic inference (eg. see Crotty et al. 2020, heterotachy model in IQTree2, Minh et al 2020).

Character based statistical phylogenetic methods.

Maximum likelihood methods principle

In **ML methods** the likelihood optimality criterion allows for assessing which among the trees in the tree space is an "optimal" tree, the tree that is considered to best represent the data. The likelihood of the possible trees given the data (here, the multiple alignment, MA) and the evolutionary model is computed. For each possible tree topology, the likelihood is computed backwards: starting from the tips, then through the successive ancestral nodes. Because ancestral character states at each position of the MA are unknown, the probability of having a specific nucleotide in the MRCA sequence at each site is given by the probabilities of change of each nucleotide (REV model) which allow to compute the likelihood of the tree. Branch lengths, representing the evolutionary distance of each OTU to MRCA, are found by maximizing the log-likelihood function that is used to compute the probability of an MA for a given tree topology. However, because the set of all possible trees is often very large [(2n – 5)! / ((n-3)!2n-3) if n>2, possible unrooted trees] given the number of isolates under study, it is not computationally possible to compute the likelihood for all possible tree topologies. Therefore, heuristics methods have been developed to search the set of all possible trees (the tree space), in order to find a reasonably good tree (with maximum likelihood). Note that there is no guarantee that the best tree will be found. Heuristics can eg. rely on the rapid building of a first tree (NJ distance based tree, parsimony tree) followed by tree-rearrangements such as SPR: subtree pruning and grafting or simulated annealing, a method apparented to Markov Chain Monte Carlo which allows walking through the tree space.

Bayesian inference principle

Bayesian phylogenetics provides a statistical framework for hypothesis testing and allow to incorporate a variety of data types into the phylogenetic modelling (eg. sampling locations, sampling dates that will allow calibrating an evolutionary timeline) and are therefore quite powerful analysis methods. Bayesian methods are also often used because they can allow for a more realistic modeling of the evolutionary process than ML, allow joint estimation of the model parameters and tree topologies (see eg. Holder and Lewis 2003). They are seen as more complex to implement than ML methods, are frequently computationally heavier and are so far rarely used in molecular epidemiology for routine surveillance of bacterial outbreaks.

Bayesian methods allow estimating the probability distribution of trees and model parameters, and provide estimates of the confidence of inferred relationships (clades), estimates of the evolutionary hypotheses (the evolutionary model distribution) and the data through the posterior (posterior probability distribution). Bayesian phylogenetic inference requires specification of prior belief on the evolutionary model parameters. Prior parameters of the evolutionary models are given in a form of probability distributions (see eg. Ronquist et al. (chap 7) in Lemey et al. 2009) of the model components eg. topology, branch lengths, substitutions model. The components of the evolutionary model can be complexified by using priors on heterogeneity rate among sites, by specifying the distribution of the evolutionary rates among lineages, and by specifying distributions around sampling time to account for uncertainties. Flat priors, i.e. that do not influence too much the posterior are often specified when reconstructing phylogenetic trees.

The posterior probabilities distributions are estimated by searching the model "parameter space" (incl. "tree space") through sampling and updating of the model parameters with Markov Chain Monte Carlo (MCMC). The idea of exploring the parameter space with MCMC is similar to drawing a high resolution map, exploring the map step by step, near a zone of interest while minimizing resolution of areas that are not of interest. Zones of interest are represented by hills. Mapping a hill with a high resolution requires the number of steps that represent the length of the MCMC chain, to be sufficient. When mapping resolution cannot further be improved, this corresponds to a stationary posterior distribution of the model parameters. Because there might be several hills in the map, and that it is difficult to explore other hills by going through valleys that represent zones of lower interest, the exploration is achieved by exploring the landscape several times (different runs) using random starting points (seeds). When the different exploration converges, ie. is congruent, the analysis provides similar major clades frequencies and the results must be summarized for interpretation. When the evolutionary signal is sufficiently strong in the MA, it is possible that the 95% of the posterior probability distribution of trees will be represented by one or a limited set of trees. Accessing the strength of evidence eg. for clades support or node ages is generally provided in the form of a 95% confidence interval. You can look at eg. the BEAST documentation to see how to summarize the different trees.

Note that ML and Bayesian inference differ in how to evaluate clade support, which has lead to several discussions on interpreting clade support (see eg. Douady et al. 2003, Erikson et al. 2003, Svennblad et al. 2006). Moreover, it is possible to explore the "model space", and find the model parameters using a special type of MCMC (reversible-jump MCMC, Huelsenbeck et al. 2004, Bouchkaert and Drummond 2017).

A simple principle introduction to Bayesian theorem can be found here , a comprehensive introduction to Bayesian phylogenetic inference can be found in Ronquist et al. (chap 7) in Lemey et al. 2009. A set of very good introductory videos can be found at phyloseminar.org: Introduction to Bayesian phylogenetics by Paul Lewis (3a, 3b)).

Model testing for statistical phylogenetic methods

Model selection affects phylogenetics inference (Posada and Buckley 2004). Model testing allows one to choose the model that best fits the data while avoiding under or overparameterization. Model testing can be used for **hypothesis testing** of alternative scenarios. Evaluating the support of alternative phylogenetic models can eg. be used to evaluate which evolutionary mechanism is more likely eg. for a specific gene, to examine patterns of trait evolution in phylogenies, or to test if the topologies of two alternative phylogenetic trees are equally supported (Lemey et al. 2009, and see review in Irisarri and Zardoya 2017). To go further please have a look at the section: Phylodynamic methods and molecular epidemiology.

ML methods

The Likelihood ratio test (LRT) allow to compare 2 models that belong to the same family (nested models, eg. submodels of GTR family) models. It is possible to use a hierarchical approach (hLRT) to successively compare nested models of different complexity levels (see eg. Posada and Buckley 2004, Posada 2008). The **Akaike Information Criterion (AIC)** is a measure that estimates how much the model differs from the true evolutionary process. It is most generally employed to compare several substitution models at once and does not require models to be nested. The best model is the model with lowest AIC. Methods that allow allows testing models that include heterogeneity rates across sites have been developed (eg. ModelFinder, Kalyaanamoorthy et al. 2017, implemented in IQ-TREE).

Bayesian methods

Bayesian factor comparison is a ratio test of the model likelihood which is estimated with the posterior probabilities of each tested model given the data while setting equal priors on the models: eg. both models are equally probable. **Bayesian Information Criterion** (BIC) is a test closely related to AIC that uses the estimates of the marginal likelihood of the substitution models. BIC can be used to compare all kinds of models, including models used in ML methods. The best model is the model with the smallest BIC. The advantages of traditional AIC and BIC testing are described in Posada and Buckley (2004). Model testing is not limited to substitution models, and can eg. also be

performed for molecular clock models, see Baele and al. (2012) and eg. allow testing of models that include invariant sites (Bouchkaert and Drummond 2017).

Other tests

If you reconstructed a time-tree, evaluating evidence of temporal signal can eg. be performed using a clustered-date randomization test (see Duchêne et al. 2015 and reviewed in Duchêne and Duchêne 2020)

Evaluating confidence of the different clades

Algorithms used in tree reconstruction assume that relationships between organisms are bifurcating (ie. no polytomies - no radiation). Therefore, it is important to consider a consensus tree of good possible trees, and evaluate branch support (see Simon 2020 for historical review).

Bootstrapping: ML and distances methods

Branch support (confidence in the clades) can be assessed by resampling methods (eg. bootstrapping, jackknifing).

Bootstrapping is random resampling with replacement of the MA sites followed by repeated tree reconstruction (Felsenstein 1985, Lemoine et al. 2018). It is the most commonly used method in ML. Note: it can also be used for estimating confidence in clades reconstructed with distance methods.

Bayesian methods

Clade confidence in Bayesian inference can be obtained from the summaries in the form of a consensus tree. Bayesian MCMC tree samples are used to derive approximate probabilities for each split/clade.

Robustness and the strength of the phylogenetic signal

Comparing phylogenetic trees reconstructed with different methods (eg. distance and ML) can also provide an indication of the strength of the signal, and help interpretation regarding which clades are not consistent between methods.

Direction of evolution: rooted tree vs non rooted tree

To provide a direction of evolution, it is possible to estimate the position of the root using a method employing a molecular clock model and using time-stamped (tip-dated data, see in Duchêne and Duchêne 2020). It is also possible to use a phylogenetic reconstruction method that allows constructing rooted trees (eg. ML using non-reversible substitution models, see eg. Williams et al. 2015 and Bayesian phylogenetics Huelsenbeck et al. 2002). See Kinene et al. (2016) for a review of the most common tree rooting methods.

When using phylogenetic methods without a clock model, such as ML methods using time reversible models, unrooted trees are inferred. To improve interpretability of unrooted trees in terms of direction of evolution, it is possible to root trees. The methods can eg, be outgroup rooting, or midpoint rooting.

When **rooting using an outgroup**, the root is placed at the midpoint of the branch that links the outgroup to the rest of the OTU. The outgroup is composed of one or several OTU that are not very closely related to the rest of the isolates under study. It is usually preferable to choose an outgroup that is not too distantly related, because you would encounter difficulties to align the sequences, and probably lose informative sites, which might lead to inference of topological errors due to the presence of saturated sites (multiple mutations, and reversals at the same site) (Lemey et al. 2009). For **midpoint rooting**, the root is placed at the middle of the longest branch (the longest evolutionary

distance between two OTU). Midpoint rooting require assumptions that the lineages evolve at identical rates (molecular clock hypothesis) and that the tree has a balanced shaped topology (Kinene et al. 2016, see eg. Anonymous 2011, University of California for tree shapes).

Rooting an unrooted tree with outgroup or midpoint rooting can usually be done using phylogenetic tree visualisation software, or programming softwares languages with libraries that allow tree manipulating (eg. ape package in R). Be aware that some software might incorrectly place the bootstrap support of the clades after rooting/re-rooting (Czech et al. 2017).

Phylogenetic tree interpreting

Be-aware that the way phylogenetic trees are displayed and annotated can trick our mind into misinterpretation (eg. branch rotations, ladderization of unrooted trees, Novick et al. 2012). It can be good to explore the different views with a visualisation software when starting phylogenetic tree interpretation. Please see Baum 2008, Gregory 2008 and McLennan 2010 for an introduction of how to read phylogenetic trees. You can find some example of cautious interpretation for bacterial epidemiology in eg. Pightling et al. (2018) and Fagerlund et al. (2020).

17.5.3 Going further

Phenotypic traits inference The use of phylogenetic methods is not limited to the study of relatedness between OTUs. The pathogen lineage structure inferred by phylogenetic analysis can be employed to extract the reference population structure that is required to correct for lineages effects in bacterial whole genome association studies (GWAS, eg. as in Pyseer, Lees et al. 2018). Alternatively, the phylogenetic information, in the form of phylogenetic trees, can be directly employed by GWAS methods as used in treeWAS (Collins and Didelot 2018). Alternatively to GWAS, comparative phylogenetics can be employed to study the evolution of some phenotypic traits across organisms that share a common evolutionary history (Hassler et al. 2020).

Phylodynamic methods and molecular epidemiology It is possible to extend the phylogenetic framework to phylodynamic inference, which combines the recovery of evolutionary processes through phylogenetic inference and joint modeling of population dynamics. This is highly suited to study the transmission and the spread of rapidly evolving pathogens (Baele et al, 2017, Ingle et al. 2021). Phylodynamics can also be used to estimate population dynamics parameters of pathogens such as effective reproduction rates (Ingle et al. 2021).

By incorporating non-genetic data eg. environmental, spatial data in phylodynamic analysis, it is possible to test alternative epidemiological hypotheses regarding the impact of ecological factors on pathogen evolution and spread. This has eg be used to test how dispersal and pathogen demography is impacted by temperature (Baele et al, 2017 , Dellicour et al. 2020a, Dellicour et al. 2020b) The inference of different population dynamics within a phylogenetic framework has so far been mostly the focus of virus epidemiology, however, this is likely to become a powerful tool also in bacterial epidemiology (Ingle et al. 2021).

17.5.4 Some limitations of phylogenetic and phylogenomic methods

Statistical phylogenetics methods rely heavily on evolutionary models, models that may be far from adequate for reflecting the reality of the evolutionary mechanisms of the OTU under study (see eg. Simion et al. (Chap 2.1 in "Phylogenetics in the Genomic Era: Scornavacca et al. 2020). Assumptions underlying the major evolutionary models are presented in a comprehensive overview in Kapli et al. (2020). Not being aware that improper use of phylogenetic reconstruction tools and violation of hypotheses assumptions can lead to erroneous interpretation of the reconstructed trees.

One major assumption, implicit with phylogenetic inference, is that the split between OTU are bifurcating: one ancestor gives rise to two and only two lineages (this excludes radiation events) and that lineages do not interact after emergence (Lemey 2009). Phylogenetic analysis also forces a tree structure even if the relationships supported by the data might better be reflected by a network (eg. in case of frequent recombination events in the species you study).

Indeed, phylogenetic network might be better suited for analyses when non-vertical evolution events might be predominant (eg. horizontal gene transfer, recombination, gene duplications) (see Huson and Bryant 2006 and Wen et al. 2018).

In most commonly used phylogenetic inference models, isolates are considered as tips. Given the nature of bacterial pathogens, it is possible eg. that frozen isolates analysed jointly with contemporary samples, might be representative of the ancestral genome. Some authors (eg. Gavryushkina et al. 2014) have developed "ancestral trees" reconstruction methods under this assumption.

Lastly, until recently, one of the limitations of phylogenetic inference for surveillance/epidemiology was the need to restart the analyses from scratch, when new data become available. This is highly problematic when analyses are computationally intensive (eg. Bayesian) and when detection of potential outbreak must occur rapidly. Methods to circumvent this problem have recently been developed (eg. Hu et al. 2020). It is also now possible to incorporate new data when available to update the posterior distribution in Bayesian analyses (Gill et al. 2020), implemented in Beast V1).

17.6 Common tools

Tools are diverse and often implemented into species specific pipelines. Please see section species specific tools. This list is far from an exhaustive list, but it might help you to start with phylogenetic analyses. Note that numerous packages for phylogenetic analysis, including phylogenetic inference, preparation of input files to eg. BEAST, tree manipulation, estimating transmission trees, dating, visualisation tools are also available in R . Not all packages are deposited in R packages list, some might be available through Bioconductor or can be fetched on github or other repositories.

17.6.1 Multiple sequence alignment and whole genome alignment

Depending on the method used, it might be required to extract the genes/colinear regions that are common to all genomes under-study)

Concatenated core gene alignment

- Roary (Page et al. 2015)
- Panaroo (Tonkin-Hill et al. 2020)

Whole genome alignment

- Mauve /progressiveMauve (Darling et al. 2004, Darling et al. 2010)
- GPA (based on progressiveMauve), (Henning et al. 2019)
- ParSNP (Treangen et al. 2014)
- Mugsy (Angiuoli and Salzberg, 2011)
- TBA (Blanchette et al. 2004)

SNPs/variant calling by mapping to reference

- BWA + SAMtools
- GATK (Broad institute)
- Snippy (Torsten Seemann)
- MUMmer4 (Marçais et al. 2018)

17.6.2 Distance matrices from multiple sequence alignments

- Snp-dists (Torsten Seemann)
- dnadist
- EMBOSS distmat
- MEGA11 (Tamura et al. 2021)

17.6.3 Recombinant detection (masking in MSA) removal

- PhiPack (Phi test, Bruen et al. 2006)
- Gubbins (Croucher et al. 2015)
- ClonalFrame (Didelot and Falush 2007)
- clonalFrameML (Didelot and Wilson 2015)
- fastGEAR (Mostowy et al. 2017)

17.6.4 Phylogenetic inference softwares

Abbreviations: Maximum Likelihood (ML), Bayesian (B).

Note: identical abbreviations used in different softwares might actually refer to different algorithms, including algorithms with the same purpose, eg. ASC and fconst that are different in iqtree and RAxML. Please always refer to the software manual

- IQTREE (ML)
- MEGA11 (Tamura et al. 2021) (ML)
- RAxML (Stamatakis 2014) (ML)
- PALM (ML)
- BEAST exists in two versions that evolve somewhat separately to each other. BEASTv1 and BEAST2
- MrBayes
- RevBayes

For more, there is a lot of programs to explore, see eg. the wikipedia list of phylogenetic softwares and the wikipedia list of bayesian phylogenetic softwares, the most common softwares that are currently used are also provided in table 10.1 (Challa and Neelapu 2019).

17.6.5 Model testing

Model testing is often implemented in phylogenetic softwares. Please refer to the documentation.

- jModelTest (Posada 2008)
- ModelFinder (Kalyaanamoorthy et al. 2017) (implemented in IQTREE)
- bModelTest (Bouchkaert and Drummond 2017) (BEASTv2 package)

17.6.6 Time-scaling

Correlation methods

- TempEst (root-to-tip regression) (Rambaut et al. 2016)
- treeDater (R package, Likelihood, Volz and Frost 2017)
- TreeTime (Likelihood, Sagulenko et al. 2018)
- Physher (Likelihood, Fourment and Holmes 2014)
- LSD (least-square dating, To et al. 2016, implemented in IQTree 2.0.3)
- MEGA11 (Several methods: least-square, Tamura et al. 2012, Miura et al. 2020, see also Mello 2018)
- R package BactDating (Partial bayesian, Didelot et al. 2018)
- Older softwares
  - r8s (Likelihood, Sanderson 2003)
  - TipDate (Likelihood, Rambaut 2000), might be implemented in PALM

17.6.7 Identification of clusters from phylogenetic trees

There has been some research to automatically define clusters based on phylogenetic trees, mostly for viruses, but such methods might well be transposable to determine clusters for bacteria (see eg. TreeCluster).

17.6.8 Visualisation tools

Visualisation software come in many flavours (desktop, web-interface . . .). Wikipedia presents a good list here.

- FigTree (has all the requirements to visualize and annotate, and is easy to use)
- Diverse libraries in R and python (eg. ggtree in R) . . .

17.6.9 Online platforms for bacterial phylogenetics and surveillance

There are several online platforms, aiming at facilitating phylogenetic data analysis and/or visualisation, metadata integration (epi-data, geography . . .) of diverse pathogens. Those have proliferated those last years (eg. nextstrain, microreact, microbetrace) The functionality of the platforms are diverse, and can range from simple phylogenetic analysis to contact tracing.

17.6.10 Detection of trait association and phylogeny

- R package treeWAS (Collins and Didelot 2018)
- Hogwash (Saund and Snitkin 2020)
- [Pyseer])(https://pyseer.readthedocs.io/en/master/index.html) (assuming that phylogeny is used as a basis to define clusters)
- Comparative phylogenetics (eg. Hassler et al. 2020, implemented in Beast v1)
- In ape R package: phylogenetic convergence test test for selection & detect resistance (Farhat et al. 2013)

17.7 Additional resources

While we covered a limited set of phylogenetics inference methods, those most frequently used in molecular epidemiology, a large amount of alternative approaches to workflows, or methods variants has not been treated in the present document.

Here are some resources that might be useful to explore further the potential of use of phylogenetics methods in epidemiology.

A series of very good series of introductory lectures to statistical phylogenetics, by Paul. Lewis, can be found in phyloseminar.org website. Moreover, phyloseminar.org also gives access to a diverse range of topics about the latest research developments in phylogeny, including molecular epidemiology.

A list of few selected books:

- Lemey, P., Salemi, M., & Vandamme, A. (Eds.). (2009). The Phylogenetic Handbook: A Practical Approach to Phylogenetic Analysis and Hypothesis Testing (2nd ed.). Cambridge: Cambridge University Press. doi:10.1017/CBO9780511819049
- Robinson, D. Ashley, Edward J. Feil, and Daniel Falush. Bacterial Population Genetics in Infectious Disease. John Wiley & Sons, 2010.
- (Online Book). Celine Scornavacca, Frédéric Delsuc, Nicolas Galtier. Phylogenetics in the Genomic Era.
- DeSalle, Rob., and Jeffrey. Rosenfeld. Phylogenomics: A Primer. New York: Garland Science, Taylor & Francis Group, 2013 (first edition) and 2020 (second edition)

CHAPTER 18

Storage and Compute Infrastructures

18.1 Storage

18.1.1 Package format

Commonly, when data is delivered by the sequencing facility, they will come in what is called a "tar" format. This is a way of grouping files into one package. Note, as opposed to the "zip" format, files in a tar archive are only packaged into one file. Files that are in the archive may or may not be compressed, but that is done independently of the packaging. Files inside of such an archive are commonly compressed with a tool called "gzip".

Note: commonly the tools to unpack a tar archive are not available on a windows machine. These tools are available on linux and in the terminal on Macs (Mac systems are linux based). However, on windows the terminal program Git Bash does contain these tools, in case there is a need for unpacking on a windows machine.

To unpack a tar file the following command can be used:

```
tar -xvf mypackage.tar
```

The `-x` means extract, the `-v` means show me the progress on the screen and `-f` means the filename of the archive is given on the command line.

Once an archive file has been unpackaged, there will likely be a folder or a directory present on your computer filled with files ending in ".gz". This indicates that your sequencing files are compressed, most likely using the tool mentioned above. Many sequence analysis tools are able to work with compressed files, thus these can be left as is. If not, the "gzip" tool can be used to unpack the files. In true linux fashion, however, the command for unpacking these files is not "gzip", it is "gunzip".

18.1.2 Space

Sequencing data can consume quite a bit of space. Generally speaking, one set of Illumina paired end read files for one isolate will take up about 0.5-1GB of space. However, in the course of processing this is commonly likely to expand somewhere between 5-10 times the original space of the raw data. As is described later, the analysis of the data mostly consists of processing the files, and then storing the results as a new set of files. Hence, it is likely that this process

will produce somewhere between 5-10 different sets of derived data. The space consumed by these derived files are likely to shrink by each step, but many of these processing steps will not reduce the file size drastically.

Here are some size estimates from an assembly pipeline consisting of commonly used tools. Reads here means the raw untrimmed data from the sequencer. Work folder here indicates the size of the files produced by the pipeline, i.e. trimmed files, bam files used for polishing etc. These do not necessarily need to be kept. Results indicates the size of the output that would be used further on, i.e. assemblies, annotation files, antibiotic resistance results, etc. These data are likely to be kept and used onwards.

CHAPTER 19

Workflow managers

19.1 What is a workflow manager

As is described in [Data Production], a bioinformatics analysis pipeline commonly consists of many tools chained after each other, each tool providing input to the next step in the process. This means that an analysis can consist of many steps, each with their own inputs and options. Running these analyses by hand can be time consuming and error prone. To help with this, several so-called workflow managers or engines have been developed. These are tools that allow for creating a script for your analysis, detailing all of the steps. Many people write this up in bash the first time they try to do this. However, with bash the person writing it will have to figure out where all of the files are, and also take care of the error handling. In addition, in many cases analyses will have to be done on a compute cluster, and that adds another layer of complexity that workflow managers can sort out on their own.

There are several tools to manage workflows/pipelines available. In this section some of the major players will be described.

19.2 Nextflow

Nextflow is a domain specific language DSL created specifically for handing bioinformatics analyses. This is a programming language that allows the user to create pipelines connecting many tools together.

The central entity in Nextflow is a process. A process is akin to a function, in that it takes in a set of input data, and produces output. Output is produced by the process executing something on the input, for instance running SPAdes on a read set to produce an assembly. Data input and output to a process is handled via non-blocking unidirectional FIFO queues. A data set is put into the input queue of a process, and the process will then pick that data set out of the queue and process it. The output will then be put into a separate output channel, which then can be used as an input channel by a different process. In this way processes can be chained together to create a pipeline.

The channel system also takes care of file handling. Nextflow operates with the concept of a "work" directory. For each run of a process on a specific data set, a specific directory is set up in the work directory to deal with the data needed and produced by that process. The input data to that process is "softlinked" in from its current location into that directory. Softlinking is a way of making "shortcuts" from one location in the file system to another. In this case, the input data to that process is softlinked into the process directory in such a way that it looks like it is in that directory.

The output data from that process is stored in the processis directory. Any processes working on the output data further downstream in the pipeline will then softlink to the output data in its current work directory location. Thus, most of the data that is produced by a pipeline will be found in the "work" directory. Data that should be viewable by the user at the end of the run can be tagged in the process by the "publishdir" directive. These data will then be available in the directory specified by the "publishdir" directive.

Due to the way processes are set up, it is also possible to have conditional executions of parts of the pipeline. This allows for evaluating results before proceeding with the analyses. In addition, it allows the user to have one complete pipeline, and depending on user input and user options only parts of the pipeline will be run.

Another benefit of Nextflow is that it is able to run on High Performance Computing systems. In practice this means that Nextflow will produce a batch file for each run of a process, and that file will then be submitted to the queueing system. Nextflow will then check in with the system and see if processes are running well or not. If a process does not complete well, it can be auto-restarted. If a process still won't produce satisfactory results, the rest of the pipeline will either not be run and the pipeline will terminate, or the particular step is ignored and there is no further processing of the dataset that failed.

Since a lot of scientific software tools can be difficult to install on local or HPC systems, it might be challenging to build a pipeline in nextflow using several tools, each with their own installation requirements. To circumvent this Nextflow can works with with a variety of systems such as both conda environments or with container systemss (Docker, Singularity, etc) both singularity and docker). These systems can make installation of software easier since the installed software is isolated and independent from the software installed locally or on a cluster. In addition, containers can make pipelines more reproducible, since the used software version can easily be installed on a different computing system. Nextflow can be pointed to software, available online as a conda environment or a container, required for a single or multiple processes and nextflow will automatically install the required software needed for a process before starting the analysis.

19.3 Snakemake

Snakemake is another workflow manager that functions in a similar manner as nextflow. A main difference to nextflow is that Snakemake uses a Python based syntax for programming pipelines. Other than that Snakemake uses rules that describe each of the steps in a pipeline. These rules also contain which input files to use and what to do with the output files. One difference with nextflow is that Snakemake is set-up via conda, and to run snakemake it is required to start a conda environment.

An interesting part of Snakemake is that workflow creation can be checked with a code quality checker which helps the creater to improve the readability and stimulates best practices when writting code. In a similar way to nextflow, Snakemake can be run on a local computer as well as on a HPC cluster or the Amazon cloud.

19.4 Galaxy

Galaxy is a web based, user friendly, scientific workflow platform for analyses especially for researchers who want to analyse their data using bioinformatics tools within a graphical interface. Programming knowledge is not needed to upload data, run analyses and export the results. However, it is also possible to use Galaxy as a pure workflow manager, without the graphical interface.

Most of the known bioinformatics tools can be installed through Galaxy toolshed, tools repository for galaxy. New tools also can be added without any complex technical steps. Each new tool needs a tool definition file (xml) where input data, parameters, output and tool location are defined. Galaxy uses this file to produce the user interface for the tool, execute the tool and display the results. Galaxy also comes with visualization tools to visualize data and the results. Galaxy recommends to use conda package manager as the best practice to manage the tool dependencies for each tool which can be configured in galaxy.config file.

Each user can have username and password. Using a galaxy without a username and password is also possible. Galaxy history section keeps track of all the analyses done by each user. Users can not see other users' history if they are not shared with them. These analyses can be easily re-run and also exported to another galaxy to reproduce.

Galaxy allows users to create workflows easily using a simple user interface. Workflows can use high performance computing to analyse big/high throughput data. Workflows can be exported from one instance of Galaxy and imported to another instance manually. Thus, Galaxy makes the reproducibility of analyses easier.

Galaxy is an open source software implemented in the Python programming language. Galaxy has a very active developers' community which actively adds new features a few times a year. Galaxy can be installed in a user laptop for a single user use or in a high performance computing server for a multiuser purpose. Docker containers and Ansible playbooks are also used to deploy galaxy easily. Popular database management systems such as MySQL, PostGres can be used by Galaxy to store the user, data and analysis details.

Galaxy can be configured to use slurm to make use of high performance clusters. Galaxy comes with a remote job running system called Pulsar. Using pulsar Galaxy jobs can be sent to remote computing resources and get back the results. Transport of data, tool information and other metadata can be done using RabbitMQ.

CHAPTER 20

Escherichia coli analysis

Escherichia coli are gram-negative bacteria which may reside in the intestinal tract of most warm-blooded animals contributing to a healthy microbiota. However, some of these bacteria have a pathogenic behavior, and may be transmitted by contaminated water or food. *E. coli* are divided into six different pathotypes, from which phage-encoded Shiga toxin-producing *E. coli* (STEC) (also known as Verocytotoxin-producing *E. coli* (VTEC)) are the ones most commonly associated with foodborne outbreaks (CDC). Indeed, Shiga toxins (Stx) are thought to be the key virulence factors for STEC infections (Gyles, 2007). STEC represents the third most relevant human foodborne bacterial pathogen, just behind *Campylobacter* and *Salmonella* (EFSA (2019)). Amesquita-Lopez et al. (2018) revises the possible routes of STEC transmission, classification, virulence factors and antimicrobial resistance.

Considering the relevance of STEC for human health, different methods have been applied in order to determine their diversity and associate these features to pathogenic traits. *E. coli* serotyping is based on somatic surface (O-antigens) and flagellum (H-antigens) antigens, and so far more than 400 STEC serotypes have been identified (Amesquita-Lopez 2018). Moreover, these serotypes are also divided into pathotypes (from A to E), according to their association to outbreaks and hemolytic-uremic syndrome (Karmali et al. 2003). STEC O157:H7 serotype belongs to the pathotype A and is responsible for the majority of outbreaks. For this reason, it is the main focus of many studies (Amesquita-Lopez 2018). However, in recent years the epidemiology of this disease has been shifting with the increasing number of cases of non-O157:H7 STEC infections (Shen et al. 2015, Lang et al. 2019). Similar to what happens with other species, STEC serotyping can be time-consuming and have limited discriminatory power for epidemiological studies. Therefore, molecular typing methods have been developed and are also used to assess STEC diversity.

20.1 Typing methods

STEC molecular typing is an evolving field, constantly seeking for the best typing method. A good typing method is not only highly discriminatory, but also reproducible and automated. STEC molecular typing can be performed through:

- **Pulsed Field Gel Electrophoresis (PFGE)** - PFGE is a fragment length restriction analysis that has long been considered the most discriminatory typing method for STEC in the pre-WGS era (Amesquita-Lopez 2018). This is currently the "gold-standard" for PulseNet network, and has been used by public health authorities and food regulators for outbreak investigations. Several studies have suggested that combination of PFGE with other typing methods may increase the discriminatory power and be useful to determine outbreak infection's sources (Amesquita-Lopez 2018).

- **MLVA (Multiple locus variable tandem repeat analysis)** - Multiple Locus Variable Number of Tandem Repeats Analysis is a PCR-based typing method, which is the second major typing tool used by the PulseNet network (before WGS). This method is fast and might also be able to differentiate fast-evolving bacteria with a similar PFGE profile. Therefore, MLVA has been used to complement PFGE results, thus providing a useful resource during outbreaks (Parsons et al. 2016).
- **MLST (Multi-Locus Sequence Typing)** - As for other bacteria, MLST methods based on 7 locus have been developed for *E. coli*. Two protocols have been established; one specifically developed for STEC (aspC, clpX, fadD, icdA, lysP, mdh, and uidA; STEC center) and one developed for a more general approach for *E. coli* (adk, fumC, gyrB, icd, mdh, recA and purA; Wirth et al. 2006). MLST can provide faster results when compared to PFGE, and it is highly reproducible.
- **WGS (Whole-Genome Sequencing)** - With the advent of NGS technologies, WGS was shown to be useful for STEC outbreak investigation (Parsons et al. 2016). By providing information at the genomic level, WGS allows not only a highly discriminatory typing (cgMLST, wgMLST and SNP-typing), but also to establish the backward compatibility with previously mentioned molecular typing methods, as the in silico serotyping and 7-loci MLST. For this reason, these methods will tend to continue to be used. Furthermore, it allows the analysis of specific genes, such as virulence factors and antimicrobial resistance genes. Genetic clustering using WGS can be performed on any distance measure (eg. issued from allelic differences detected using cgMLST typing) or evolutionary-model based clustering (ie. phylogenetics) relying on variants/SNPs detection. PulseNet network is making efforts to implement WGS as a routine tool to replace PFGE and MLVA.

20.2 "One Health" surveillance and WGS of STEC

The identification of infection sources is essential for outbreak monitoring. Hence, an integrated analysis of clinical, food and veterinary samples relying on the concept of One Health is the key to achieve a good surveillance system. As shown here by PulseNet network, the high discriminatory power of WGS increases the chances to find the bacterial source of infection, and possibly reduces the time that it takes. Indeed, WGS analysis has proven to be an effective way to determine the genetic clustering of STEC isolates, as well as the source of infections (Parsons et al. 2016, Jenkins et al. 2019, Nouws et al. 2020, Joensen et al. 2014, Chattaway et al. 2016). For instance, in England and Denmark WGS-based STEC surveillance has been implemented with success (Parsons et al. 2016, Dallman et al. 2021. However, this has mainly focused on STEC from patients. Nevertheless, WGS-based STEC surveillance at the EU level has been proposed to be delayed until the technological transition has been made for listeriosis (ECDC roadmap).

20.3 WGS lab protocol

20.3.1 DNA extraction

Before DNA extraction, STEC is cultured in the laboratory. Commonly used media for STEC include tryptic soy broth, *E. coli* broth and buffered peptone water (Amezquita-Lopes et al. 2018) as well as more specific growth media. Regarding DNA extraction, there is not a standard protocol or kit that is used, but a protocol directed towards Gram-negative bacteria will be recommended.

20.3.2 Sequencing technology

There is not a prefered WGS technology to sequence STEC. Similar to other fields, Illumina paired-end reads represent the most commonly used strategy. Due to the number of samples that can be handled at a single run and the possible higher read size, MiSeq sequencing machines seem to be the choice for the majority of the labs.

20.4 Bioinformatics protocol

20.4.1 Mapping or assembly

The first step to perform when receiving the sequencing data , is to evaluate the sequencing quality and perform trimming and cleaning of the reads (see *Data preprocessing*).

The cleaned sequence data can then be used for downstream analysis following one of two approaches (or both in parallel, check *Data production*):

- *De novo* genome assembly of the sample(s),
- Read mapping of each sample on a reference sequence (obtained from a database or by *de novo genome* assembly of one of your samples).

It is important to note that both approaches have advantages and disadvantages. The decision on which of them to follow should be made according to what is more appropriate for the data at hand, and the purpose of the analyses. De novo genome assembly of all sequenced isolates followed by their annotation seems to be a common approach in studies including STEC genomes. A commonly used *de novo* genome assembler for STEC is SPAdes (Iramiot et al. 2020, Reid et al. 2020, Sonda et al. 2018). It performs very well and is freely available. There are command-line pipelines, such as INNUca, which incorporate these programs and provide the opportunity to automatically perform all the analyses from quality control to genome assembly. If a platform with predefined pipelines (and that usually does not require bioinformatics skills) is preferred, Enterobase is available for *E. coli*. As for read mapping, BWA is a commonly used approach (Holmes et al. 2015, Iramiot et al. 2020, Parsons et al. 2016, Dallman et al. 2021). However, as mentioned before, these represent commonly used approaches, and not recommendations. Thus, other methodologies, pipelines or even platforms may be used.

20.4.2 Choosing a reference genome

Should an analysis require the use of a reference genome, the choice of the reference genome is a crucial step. Analyses relying on read-mapping approaches might be strongly influenced by reference choice, as the genetic distance between the reference and the sample may influence the performance of downstream steps, namely SNPs/INDELs calling (Pightling et al. 2014, Pightling et al. 2015). This reference can be picked from the samples (after genome assembly), or from a public database. Enterobase is a good site for choosing a reference for this species.

20.4.3 Serotyping

Besides the wet-lab approach for serotype determination of STEC samples, in silico approaches using WGS data can also be performed (Joensen et al. 2015, Ingle et al. 2016). SRST2 can be used to determine serotyping without the need of de novo genome assembly, by comparing the genomic reads directly to the database (Ingle et al. 2016). SeroTypeFinder is another alternative for in silico determination of *E. coli* serotype, requiring sequencing reads or genome assembly as input. Bionumerics (using the database from SeroTypeFinder), Enterobase is an exampoe of of a platform where this function is available.

20.4.4 Getting SNPs

Analysis of SNPs is a frequently used approach for the analysis of STEC samples (Parsons et al. 2016).

How to detect SNPs is described earlier. Briefly, there are three different approaches.

- Perform *de novo* genome assembly of each sample and then align their genomic sequences.

- Use a reference genome where the reads of all the samples will be mapped, and then use a variant-calling pipeline to determine the polymorphic positions. CFSAN SNP is a commonly used pipeline which performs both processes (read mapping and variant calling). Snippy and SNVPhyl are also commonly used alternatives for STEC analyses.
- Determine the polymorphic positions in the sample by analyzing the k-mer pattern using kSNP. For this approach either the genome assembly or the genomic reads must be provided. This is not a commonly used approach for STEC analyses.

20.4.5 Getting alleles and allele differences

The allele sequences of the samples can be retrieved by:

- Replacing the nucleotide of the reference genome by the observed alternative allele, and then retrieve the sequence of each gene of interest considering the genome annotation of the reference.
- Obtaining the *de novo* genome assembly of each sample, and performing the respective genome annotation. Prokka is acommonly used program for STEC.
- Some allele callers, such as chewBBACA, provide locus-specific alignments in an automated manner, being a good option to determine the allelic profile of samples.

It is important to note that nowadays there are several platforms which can automatically do all this analysis. One of the more commonly used for *E.coli* is Enterobase, and also Bionumerics. These platforms provide assembly, serotyping and allele calling. Several of these platforms are mentioned in the xMLST section.

20.4.6 Allele based typing

Allele-based typing consists of retrieving clustering information considering the different alleles present in a population for a given set of genes (e.g. the core genome). With the advent of WGS, the 7-loci based MLST approach was broadened to the use of a cgMLST or a wgMLST approach. In this context, there is a public cgMLST scheme which has been used in STEC analysis considering an allele-based approach. This scheme comprises 2,513 loci and is available in the most commonly used platforms, such as EnteroBase and Ridom SeqSphere+. Noteworthy, although the scheme used by the platforms is the same, their allele calling is independent, and therefore there may be some nomenclature incompatibilities between the different platforms.

20.4.7 SNP based typing

A SNP-based approach relies on the comparison of SNPs in a population. This strategy can be seen as an alternative to the allele-based approach, but many studies actually perform both of them and assess the overlap of the results. Although for the majority of important bacterial pathogens WGS-based typing is performed following an allele-based approach, in the case of STEC SNP-based typing is frequently used. For instance, Public Health England has performed WGS-based STEC surveillance for a long time following a well established pipeline (PHEnix) for surveillance and outbreak detection (Dallman et al. 2021, Dallman et al. 2015). This pipeline relies mostly on variant-calling with GATK after read-mapping with BWA-MEM, followed by clustering analysis with SnapperDB.

Examples of other available pipelines for SNP-based typing are:

- Center for Food Safety and Applied Nutrition (CFSAN) HqSNPs pipeline
- Lyve-SET pipeline for HqSNPs typing
- SNV-Phyl (Canadian Public Health Agency)
- PHEnix (The Public Health England SNP calling pipeline)

20.4.8 Outbreak definition

As defined by the World Health Organization , "a disease outbreak is the occurrence of cases of disease in excess of what would normally be expected in a defined community, geographical area or season" . WGS data provides a high discriminatory power allowing clustering of different isolates (from different geographical areas, and clinical, animal or environmental sources) according to their genomic similarity. This contributes not only to an earlier detection of outbreaks and determination of contamination sources, but also to the detection of more outbreaks, as has been reported by PulseNet network for *Listeria*. It is difficult to establish a clear cluster outbreak definition, a threshold at which we decide whether two isolates belong to the same genetic cluster, thus linking two cases of infection. Previous studies have shown that outbreak-related isolates differ in up to five SNPs in the whole genome, and therefore this is a commonly used threshold to determine an outbreak-related cluster (Dallman et al. 2021, Holmes et al. 2018, Dallman et al. 2015).

20.4.9 Virulence and AMR

Several genes are important for *E. coli* ability to cause infection and are medically relevant and many of these are associated to different pathogroups. Relevant virulence-associated genes for STEC are different stx subtypes (stx1a, stx2a, stx2d) and other virulence associated genes such as eae and aggR (ref) while Extra intestinal *E. coli* (ExPEC) other virulence genes such as pap, fimH, sfa, iha, hlyA, cnf1 or sat are of importance (eg. Hung et al. 2019, Wang et al. 2009, Rodríguez-Villodres et al. 2019). As stx subtypes might be highly similar a specific database has been created associated with VirulenceFinder. Natural evolution, horizontal transfer of antimicrobial resistant elements as well as the use of antibiotics have contributed to the emergence of multi-drug resistant isolates, and this has become a worrying issue that is increasingly observed (Poirel et al. 2018). Of particular concern is the acquisition of genes conferring resistance to broad-spectrum cephalosporins, carbapenems aminoglycosides, and (fluoro)quinolones (Poirel et al. 2018). For this reason, monitoring of virulence- and antimicrobial resistance-related genes is of great relevance to determine the best way of action in the presence of a case of infection or even an outbreak. As mentioned in the *Virulence and AMR detection section*, where more details can be found, this is performed by comparing the genome to a database comprising a set of genes of interest. Examples of predefined resistome databases are mentioned in the same section.

CHAPTER 21

Campylobacter analysis

Campylobacter are gram-negative bacteria responsible for the majority of the cases of foodborne bacterial infections (Kirk et al. 2015, The European Union One Health 2019 Zoonoses Report). Although poultry has been pointed as the major source of campylobacteriosis, several cases have been linked to other sources such as ruminants or environment (Cody et al. 2019). Campylobacteriosis causes gastroenteritis, with symptoms that involve diarrhea and fever, but it may also be responsible for a neurological disorder called Guillain-Barré syndrome. So far, 17 *Campylobacter* species and six subspecies have been described, from which *Campylobacter jejuni* and *Campylobacter coli* are the most commonly associated with human illness (ECDC 2018). Thermophilic *Campylobacter* species grow at temperatures between 37ºC and 42ºC (41.5ºC being the optimal temperature) (Silva et al. 2011). The thermophilic species are the ones that are of greatest concern for human illness.

In a WGS approach regarding *Campylobacter* spp. it might be of importance to identify the species. The similarity between different isolates (from clinical, animal or environmental sources), and their respective virulence and antimicrobial resistance markers is essential for a proper disease surveillance. *Campylobacter* serotyping is based on Penner serotyping scheme, which relies on a hemagglutination assay of lipooligosaccharides (LOS) and of a capsule polysaccharide (CPS), with CPS being the primary serodeterminant (Penner & Hennessy, 2000, Parkhill et al. 2000, Karlyshev et al. 2000, Pike et al. 2013). More than 40 *Campylobacter* serotypes have been described with this methodology. Nevertheless, similar to what happens with other species, molecular typing has a higher discriminatory power, which is useful for epidemiological purposes.

21.1 Typing methods

An ideal typing method presents not only a high discriminatory power, but also high reproducibility and the possibility of automation. For this reason, molecular typing is a constantly evolving field always seeking for better technologies. Nowadays, different techniques can be applied for *Campylobacter* molecular typing, namely:

- **Pulsed Field Gel Electrophoresis (PFGE)** - PFGE is a fragment length restriction analysis that has long been considered the most discriminatory typing method for *Campylobacter* in the pre-WGS era (Sabat et al. 2013, Frazão et al. 2020). This is currently the "gold-standard" for PulseNet network, and has been used by public health authorities and food regulators for outbreak investigations.
- **MLVA (Multiple locus variable tandem repeat analysis)** - Multiple Locus Variable Number of Tandem Repeats Analysis is a PCR-based typing method, which is another typing tool used by the PulseNet network (before

WGS). This method is able to differentiate fast-evolving bacteria even if they look similar with PFGE. Therefore, MLVA is usually performed as a complement to PFGE results, thus providing a useful resource during outbreaks (Techaruvichit et al. 2015).

- **MLST (Multi-Locus Sequence Typing)** - As for other bacteria, a MLST method based on 7-locus (asp, gnl, glt, gly, pgm, tkt, and unc) has been developed for *Campylobacter* (Dingle et al. 2001). MLST can provide faster results compared to PFGE, and it is highly reproducible. However, it shows lower discriminatory power than PFGE and MLVA (Sabat et al. 2013, Techaruvichit et al. 2015, Frazão et al. 2020), and therefore it was suggested that it should be used as a complement to PFGE (Frazão et al. 2020). A big advantage of MLST analysis for *Campylobacter* in comparison for instance to PFGE, is the existence of a curated database with common nomenclature which allows the comparison of results between studies (PubMLST), which has made this technique being widely used in epidemiological studies.
- **Sequencing of the short variable region (SVR) of the flaA gene** - This method relies on the analysis of the genetic sequence of flaA in comparison with the alleles present in PubMLST, and has been described as a fast, discriminatory and reproducible tool to discriminate among *Campylobacter* isolates. This technique is useful in combination with PFGE or MLST to differentiate outbreak-related isolates (Niederer et al. 2012, Mohan & Habib, 2019, Frazão et al. 2020). Nevertheless, PFGE and MLVA are more discriminatory and used tools for *Campylobacter* typing in epidemiology (Frazão et al. 2020).
- **CRISPR** - High-resolution DNA melt curve analysis (HRMA) of the CRISPR region can be used to differentiate among *Campylobacter* isolates (Price et al. 2007). This technique has been shown to be less discriminatory than PFGE, MLST or even SVR of the flaA gene (Frazão et al. 2020). Nevertheless, when used in combination with another typing method such as MLST, this technique has proven to be useful for epidemiological studies (Kovanen et al. 2014).
- **WGS (Whole-Genome Sequencing)** - With the advent of NGS technologies, WGS was proven to be useful for *Campylobacter* outbreak investigation (Joensen et al. 2020). The *Campylobacter* genome size is approximately 1.8Mb with ~1,800 genes. By providing information at the genomic level, WGS allows not only a highly discriminatory typing (cgMLST, wgMLST and SNP-typing), but also to establish the backward compatibility with previously mentioned molecular typing methods, as 7-loci MLST, which, for this reason, will tend to continue to be used. Furthermore, it allows the analysis of specific genes, such as virulence factors and antimicrobial resistance genes, contributing to a better understanding of the different pathogenic populations. Genetic clustering using WGS can be performed on any distance measure (eg. issued from allelic differences detected using cgMLST typing) or evolutionary-model based clustering (ie. phylogenetics) relying on variants/SNPs detection. PulseNet network is making efforts to implement WGS as a routine tool to replace PFGE and MLVA. Nevertheless, this is still not the routine in the case of *Campylobacter*.

21.2 "One Health" surveillance and WGS of *Campylobacter*

The identification of sources of infection and the knowledge of pathogens' genomic features is essential for proper surveillance and outbreak monitorization. Hence, an integrated analysis of clinical, food and veterinary samples relying on the concept of One Health is the key to achieve a good surveillance system. As shown here by PulseNet network, the high discriminatory power of WGS increases the chances to find the bacterial source of infection, and possibly reduces the time that it takes. Indeed, WGS analysis has proven to be an effective way to determine the genetic clustering of *Campylobacter* isolates, as well as the source of infections (Joensen et al. 2020). According to the European Union One Health 2019 Zoonoses report, surveillance systems for infections by *Campylobacter* are present in almost all member states, with the notification of campylobacteriosis being mandatory in 21 countries. Moreover, *Campylobacter* is monitored along the food chain. Nevertheless, WGS is not yet being implemented in routine *Campylobacter* surveillance.

21.3 WGS lab protocol

21.3.1 DNA extraction

Before DNA extraction, *Campylobacter* is cultured in the laboratory. These bacteria are microaerophilic, and for this reason they should be cultured under an oxygen-reduced atmosphere (Buss et al. 2019). Moreover, *C. jejuni* is usually cultured at 41.5ºC, as it only grows at temperatures between 30ºC and 42ºC (Duffy & Dykes, 2006). Regarding DNA extraction, there is not a standard protocol or kit that is used, but many studies use QIAGEN DNeasy Blood or Tissue kit or DNA QIAamp Mini Kit (Qiagen, The Netherlands) (Meistere et al. 2019, Dunn et al. 2018, Dahl et al. 2020, Joensen et al. 2020).

21.3.2 Sequencing technology

There is not a prefered WGS technology to sequence *Campylobacter*. Similar to other fields, Illumina paired-end reads represent the most commonly used strategy. Due to the number of samples that can be handled at a single run and the possible higher read size, MiSeq sequencing machines seem to be the choice for the majority of the labs.

21.4 Bioinformatics protocol

21.4.1 Mapping or assembly

The first step to perform when receiving the sequencing data, is to evaluate the sequencing quality and perform trimming and cleaning of the reads (see *Data preprocessing*).

The cleaned sequence data can then be used for downstream analysis following one of two approaches (or both in parallel, check *Data production*):

- *De novo* genome assembly of the sample(s),
- Read mapping of each sample on a reference sequence (obtained from a database or by *de novo* genome assembly of one of your samples)

It is important to note that both approaches have advantages and disadvantages. The decision on which of them to follow should be made according to what is more appropriate for the data at hand, and the purpose of the analyses. *De novo* genome assembly of all sequenced isolates followed by their annotation seems to be a common approach in studies including *Campylobacter* genomes, which then perform a cgMLST analysis. A commonly used *de novo* genome assembler for *Campylobacter* is SPAdes (Dunn et al. 2018, Redondo et al. 2019, Kelley et al. 2020). It performs very well and is freely available. As for read mapping, when performed, it usually relies on the usage of Bowtie or BWA (Golz et al. 2020, Dunn et al. 2018, Mandal et al. 2017, Wallace et al. 2020, Chung et al. 2016). There are command-line pipelines, such as INNUca, which incorporate these programs and provide the opportunity to automatically perform all the analyses from quality control to genome assembly. If a platform with predefined pipelines (and that usually does not require bioinformatics skills) is preferred, Enterobase is available for *Campylobacter*. In addition, the IRIDA system and CLC Genomics Workbench is in common use.

21.4.2 Choosing a reference genome

Should an analysis require the use of a reference genome, the choice of the reference genome is a crucial step. Analyses relying on read-mapping approaches might be strongly influenced by reference choice, as the genetic distance between the reference and the sample may influence the performance of downstream steps, namely SNPs/INDELs calling (Pightling et al. 2014, Pightling et al. 2015). This reference can be picked from the samples (after genome assembly),

or from a public database. A read mapping approach is not commonly used in *Campylobacter* analysis, and for this reason there is not a specific reference genome in public databases that is in common use.

21.4.3 Getting SNPs

How to detect SNPs is described earlier.

Briefly, there are three different approaches.

- Perform *de novo* genome assembly of each sample and then align their genomic sequences (or gene sequences after annotation). *Campylobacter* analyses usually use MAUVE or PRANK to align the genomes (Clark et al. 2018, Weis et al. 2016, Fiedoruk et al. 2019, Parker et al. 2021). The last aligner is mostly used as part of the pan-genome pipeline Roary.
- Use a reference genome where the reads of all the samples will be mapped (check above), and then use a variant-calling pipeline to determine the polymorphic positions. CFSAN SNP is a commonly used pipeline which performs both processes (read mapping and variant calling). Snippy is also a commonly used alternative.
- Determine the polymorphic positions in the sample by analyzing the k-mer pattern using kSNP. For this approach you can either provide the genome assembly, or the cleaned genomic reads. This is the less frequently used approach for *Campylobacter*.

Each of these approaches provides you with information about the genetic variability of your dataset. This information can then be used to perform SNP-based clustering and phylogenetic analysis. Alternatively, if you follow a read mapping approach, you can replace the reference nucleotide by the observed allele, and consequently reconstruct the haplotype of each sample. This is the approach used by the CFSAN SNP pipeline.

21.4.4 Getting alleles and allele differences

The allele sequences of the samples can be retrieved by:

- Replacing the nucleotide of the reference genome by the observed alternative allele, and then retrieve the sequence of each gene of interest considering the genome annotation of the reference.
- Obtaining the *de novo* genome assembly of each sample, and performing the respective genome annotation. Prokka is acommonly used program for *Campylobacter*.
- Some allele callers, such as chewBBACA, provide locus-specific alignments in an automated manner, being a good option to determine the allelic profile of samples.

It is important to note that nowadays there are several platforms which can automatically do all this analysis. One of the more commonly used for *Campylobacter* is BIGSdb. These platforms provide assembly, serotyping and allele calling. Several of these platforms are mentioned in the xMLST section.

21.4.5 Allele based typing

Allele-based typing consists of retrieving clustering information considering the different alleles present in a population for a given set of genes (e.g. the core genome). With the advent of WGS, the 7-loci based MLST approach was broadened to the use of a cgMLST or a wgMLST approach. In this context, there is a public cgMLST scheme which has been used in *Campylobacter jejuni/coli* analysis considering an allele-based approach. This scheme comprises 1,343 loci (Cody et al. 2017).

Platforms available for cgMLST typing of *Campylobacter* include BIGSdb, BioNumerics, IRIDA, Pathogen Watch, and Ridom SeqSphere+. BIGSdb and BioNumerics seem to be commonly used by the community.

21.4.6 SNP based typing

A SNP-based approach relies on the comparison of SNPs in a population. This strategy can be seen as an alternative to the allele-based approach, but many studies actually perform both of them and assess the overlap of the results. For a SNP-based analysis all of the the SNPs that are present in the samples need to be acquired and used to obtain clustering information. Examples of publicly available pipelines for SNP-based typing are:

- Center for Food Safety and Applied Nutrition (CFSAN) HqSNPs pipeline
- Lyve-SET pipeline for HqSNPs typing
- SNV-Phyl (Canadian Public Health Agency)
- PHEnix (The Public Health England SNP calling pipeline)

21.4.7 Outbreak definition

As defined by the World Health Organization, "a disease outbreak is the occurrence of cases of disease in excess of what would normally be expected in a defined community, geographical area or season". WGS data provides a high discriminatory power allowing clustering of different isolates (from different geographical areas, and clinical, animal or environmental sources) according to their genomic similarity. This contributes not only to an earlier detection of outbreaks and determination of contamination sources, but also to the detection of more outbreaks, as has been reported by PulseNet network for *Listeria*. It is difficult to establish a clear cluster outbreak definition, a threshold at which we decide whether two isolates belong to the same genetic cluster, thus linking two cases of infection. Epidemiological related *Campylobacter* isolates can be distinguished from unrelated ones (Llarena et al. 2017). Nevertheless, the genomic variability within an outbreak-related clade varies depending not only on the dataset, but also on the methodology used (e.g. which MLST or cgMLST scheme is used). Furthermore, mixed infections may also influence the results (Llarena et al. 2017). In two independent outbreaks, a 3 SNPs variation has been found among the isolates (Revez et al. 2014a and Revez et al. 2014b). Using a 732-core-gene schema, Clark et al. (2016) found 4 allele differences between isolates. Lahti et al. (2017) described a maximum of 1 allele difference between clinical isolates, considering a reference-based cgMLST with 1,271 loci. Therefore, so far, there is no specific threshold used to define *Campylobacter* outbreaks, and more studies on the genetic variation within and between *Campylobacter* populations would provide a great contribution to the field.

21.4.8 Virulence and AMR

Similar to other pathogens, several genes are important for *Campylobacter* ability to cause infection, and therefore genes such as cadF and ciaB have been described as medically relevant (eg. Wu et al. 2016, Dasti et al. 2009, Fiedoruk et al. 2019, Chukwu et al. 2019). Moreover, despite the majority of infections not requiring the administration of antimicrobial drugs, in severe cases of disease antimicrobial therapy can be provided. In recent years, an increased resistance to these drugs has been observed in *Campylobacter* becoming a concern for public health authorities (CDC).

Several studies have determined genes and respective variations which are potentially related with increased virulence or specific antimicrobial resistance (e.g. Bravo et al. 2021, Lluque et al. 2017, Gahamanyi et al. 2021, Aksomaitiene et al. 2021). Moreover, the existence of several events of horizontal gene transfer may contribute to increase the list of relevant genes for this species (Aksomaitiene et al. 2021, Hull et al. 2021). For this reason, it is important to determine the presence of medically important genes/variations in the isolates. As mentioned in the *Virulence and AMR detection section*, where more details can be found, this is performed by comparing the genome to a database comprising a set of genes of interest. Examples of predefined resistome databases are mentioned in the same section.

CHAPTER 22

Salmonella analysis

Gram-negative bacteria of the genus *Salmonella* are a major cause of foodborne illness. Two *Salmonella* species have been identified, namely, *Salmonella enterica* and *Salmonella bongori*. Despite only harboring two species, this genus can be divided into several subspecies and then further to different serotypes. Isolates are often reported by the name of the genus followed by the name of the serotype, without mentioning the species or subspecies name (Eng et al. 2015). Moreover, *Salmonella* isolates are usually classified into typhoidal and non-typhoidal *Salmonella*, according to their role as causative agents of typhoid or paratyphoid fever and salmonellosis, respectively.

Salmonella serotyping is performed using the White-Kauffman-Le Minor scheme (Grimont and Weill 2007, Guibourdenche et al. 2010), which uses somatic (O), flagellar (H), and capsular (Vi) antigens. This is one of the "gold-standards" for *Salmonella* classification, being widely used for outbreak, surveillance and epidemiological studies. So far, more than 2,500 serotypes have been identified, and many of them seem to be particularly associated with certain niches (CDC). Thus, serotyping may guide public authorities during outbreak investigations. Nevertheless, a small number of serotypes which are globally distributed are responsible for the majority of outbreaks, and in these cases serotyping does not have high enough resolution. Moreover, the existence of so many serotypes obligates laboratories to keep a high amount of high-quality typing antisera and antigens for conventional serotyping of *Salmonella*. In this context, molecular typing methods acquired a key-role in Salmonella surveillance and outbreak investigation.

22.1 Typing methods

Salmonella molecular typing can be performed through:

- **Pulsed Field Gel Electrophoresis (PFGE)** - PFGE is a fragment length restriction analysis that has long been considered as one of the "gold-standards" for *Salmonella* typing, together with serotyping, due to its relatively high discriminatory power. This was until recently considered the "gold-standard" for PulseNet network, and has been used by public health authorities and food regulators for outbreak investigations.
- **MLVA (Multiple locus variable tandem repeat analysis)** - Multiple Locus Variable Number of Tandem Repeats Analysis is a PCR-based typing method, which is a major typing tool used by the PulseNet network. This method is able to differentiate fast-evolving bacteria even if they look similar with PFGE and is a faster, less laborious method. Therefore, MLVA is usually performed as a complement to PFGE results or instead of PFGE, thus providing a useful resource during outbreaks. As this analysis is specific for each serotype, different

Salmonella serotypes usually require different MLVA schemes. Therefore, isolates have to be serotyped before selecting the MLVA scheme.

- **MLST (Multi-locus Sequence Typing)** - As for other bacteria, a MLST method based on seven housekeeping genes (aroC, dnaN, hemD, hisD, thrA, sucA, and purE) has been developed for *Salmonella* (Achtman et al. 2012). MLST can provide faster and more reproducible results compared to PFGE. However, it shows lower discriminatory power than PFGE and MLVA, but at a similar level as serotyping.
- **Microarrays** - The *Salmonella* genoserotyping array (SGSA) is a microarray developed as an alternative to the usual serotyping method. This method presents very good results for the 57 most commonly reported serotypes, but fails for many others. Therefore, it is more useful for fast screening of those 57 serotypes, but not for the others. This method has been improved in SGSA v2.
- **CRISPR** - This method uses the diversity of spacers present at CRISPR loci to distinguish bacterial strains (Fabre et al. 2012). Amplified CRISPR loci PCR products are sequenced and analyzed to assign each locus to an allelic type in order to determine the allelic profile of each isolate, and their evolutionary relation. A CRISPR–multi-virulence-locus sequence typing (MVLST) approach using the genes sseL and fimH has also been developed (Shariat et al. 2013). A comparative analysis revealed that CRISPR–MVLST has a higher discriminatory power than the usual MLST, but lower discrimination than PFGE. This represents an expensive non-standardized protocol.
- **WGS (Whole-Genome Sequencing)** - With the advent of NGS technologies, WGS technology has led to the improvement of small salmonellosis outbreak investigation (Kubota et al. 2019). By providing information at the genomic level, WGS allows not only a highly discriminatory typing (cgMLST, wgMLST and SNP-typing), but also to establish the backward compatibility with previously mentioned molecular typing methods, as the molecular serotyping 7-genes MLST, which, for this reason, will tend to continue to be used. Furthermore, it allows the analysis of specific genes, such as virulence factors and antimicrobial resistance genes. Genetic clustering using WGS can be performed on any distance measure (eg. issued from allelic differences detected using cgMLST typing) or evolutionary-model based clustering (ie. phylogenetics) relying on variants/SNPs detection. PulseNet network, as well as ECDC and EFSA, are making efforts to implement WGS as a routine tool to replace PFGE and MLVA. Nevertheless, in the case of *Salmonella* this is still not a routine procedure.

22.2 "One Health" surveillance and WGS of *Salmonella*

The identification of infection sources is essential for outbreak resolution. Hence, an integrated analysis of clinical, food and veterinary samples relying on the concept of One Health is the key to achieve a good surveillance system. As shown here by PulseNet network for *Listeria*, the high discriminatory power of WGS increases the chances to find the bacterial source of infection, and possibly reduces the time that it takes. Indeed, as reported by the WHO, the use of WGS increased the resolution of *Salmonella* cluster analysis, and contributed to the identification of recurrent sources of infection. Furthermore, the integrated WGS analysis of food and human samples at international level during a multi-country *Salmonella* outbreak allowed the identification of the source of infection in Germany (Inns et al. 2015), reflecting the ease at which WGS data can be shared, analyzed and compared. However, several factors are hindering the implementation of a generalized WGS-based surveillance system. For instance, the resources for implementation of WGS differ between different sectors (human health, animal health and food safety), thus complicating the implementation of "One health" surveillance. For this reason, it has been decided that the technological transition to WGS-based surveillance at European level is performed first in *Listeria*, and only afterwards in other bacteria, such as *Salmonella* (ECDC roadmap).

22.3 WGS lab protocol

22.3.1 DNA extraction

Regarding DNA extraction, there is not a standard protocol or kit that is used, but a protocol directed towards Gram-negative bacteria will be recommended.

22.3.2 Sequencing technology

There is not a prefered WGS technology to sequence *Salmonella*. Similar to other fields, Illumina paired-end reads represent the most commonly used strategy. Due to the number of samples that can be handled at a single run and the possible higher read size, MiSeq sequencing machines seem to be the choice for the majority of the labs. Long-read sequencing technologies are now becoming more frequently used, and there is an apparent tendency to sequence *Salmonella* genomes using both short- and long-read technologies.

For Illumina sequencing, the choice of library preparation procedure may have adverse effects on in silico serotyping. For example, the Nextera XT library preparation kit seems to introduce a GC bias, which negatively affects O-antigen recognition due to increased fragmentation (Uelze, 2019). The new version, Nextera Flex, is therefore recommended over the XT kit.

22.4 Bioinformatics protocol

22.4.1 Mapping or assembly

The first step to perform when receiving the sequencing data of your samples, is to evaluate the sequencing quality and perform trimming and cleaning of the reads.

The cleaned sequence data can then be used for downstream analysis following one of two approaches (or both in parallel, check *Data production*):

- *De novo* genome assembly of the sample(s),
- Read mapping of each sample on a reference sequence (obtained from a database or by de novo genome assembly of one of your sample)

It is important to note that both approaches have advantages and disadvantages, and the decision on which of them to follow should be made according to what is more appropriate for the data you have at hand, and the purpose of your analyses. De novo genome assembly of all sequenced isolates followed by their annotation seems to be a common approach in studies including Salmonella genomes. Nevertheless, especially when a further SNP-based approach will be performed (see next questions), a parallel read mapping approach is also followed.

De novo genome assemblers that can be used for *Salmonella* include SPAdes and SKESA. Both of them perform very well and are freely available. A major difference between them is the fact that SKESA can not use longreads produced by Oxford Nanopore or Pacfic Biosciences machines. For this reason, it does not represent a good alternative for hybrid assemblies combining both short- and long-reads, which is the tendency in the field. In this context, Unicycler, a pipeline tailored to perform hybrid assemblies, combines SPAdes to other tools, is commonly used for *Salmonella* genomes. Other command-line pipelines, such as INNUca, also provide the opportunity to automatically perform all the analyses from quality control to genome assembly. If a platform with predefined pipelines is needed instead, INNUENDO, BioNumerics, Ridom SeqSphere+ and EnteroBase can be used for *Salmonella*.

As for read mapping, BWA is a common choice for *Salmonella*. Alternatively, CFSAN SNP pipeline, which is tailored to create high quality SNP matrices for sequences from closely-related pathogens, is a commonly used pipeline for *Salmonella*. This pipeline covers all the steps of the analysis from read mapping to calculation of SNP distances and reconstruction of the haplotypes. Therefore, despite requiring some bioinformatics skills, it may represent a good

alternative to the development of your own pipeline. Noteworthy, as mentioned before, these represent commonly used approaches, and not recommendations. Thus, other methodologies, pipelines or even platforms may be used for your analysis.

22.4.2 Choosing a reference genome

Should the analysis require the use of a reference genome, the choice of the reference genome is a crucial step. Analyses relying on read-mapping approaches might be strongly influenced by reference choice, as the genetic distance between the reference and the sample may influence the performance of downstream steps, namely SNPs/INDELs calling (Pightling et al. 2014, Pightling et al. 2015). This reference can be picked from the samples themselves (after genome assembly), or from a public database. In both cases the reference must be chosen according to the serotype of each isolate. For this reason, it is essential to determine the serotype before read mapping, and there is not a specific reference genome that is used from public databases. However, a closed bacterial genome will be preferable.

22.4.3 *Salmonella* serotyping

As mentioned before, determination of *Salmonella* serotype is an important step to be able to perform further analysis. For instance, a read-mapping approach and downstream analysis obtain better results if the reference genome corresponds to the same serotype as the sample. Serotype determination can be performed with the White-Kauffman-Le Minor scheme, or with an *in silico* pipeline. The most commonly used programs for *in silico* serotype determination in *Salmonella* are SISTR and SeqSero (and v2. SeqSero2). Although less commonly used, the "bacterial analysis pipeline" is also an option. The Salmonella Type Finder is a pipeline developed by the Center for Genomic Epidemiology which uses SRST2 and SeqSero. Enterobase is a complete pipeline combining several tools and includes both SISTR and SeqSero2.

22.4.4 Getting SNPs

How to detect SNPs is described earlier. Briefly, there are three different approaches.

- Perform *de novo* genome assembly of each sample (check above), and then align their genomic sequences. *Salmonella* analyses usually use MAUVE to align the genomes. This multi-sequence alignment can be then input to SNP-sites to get the number of variants.
- Use a reference genome where the reads of all the samples will be mapped, and then use a variant-calling pipeline to determine the polymorphic positions. CFSAN SNP is a commonly used pipeline which performs both processes (read mapping and variant calling). Snippy and SNVPhyl are also commonly used alternatives for *Salmonella* genomes.
- Determine the polymorphic positions in the sample by analyzing the k-mer pattern using kSNP. For this approach you can either provide the genome assembly, or the cleaned genomic reads. This is the less frequently used approach for *Salmonella*.

Each of these approaches provides you with information about the genetic variability in the dataset. This information can then be used to perform SNP-based clustering and phylogenetic analysis. Alternatively, if a read mapping approache is followed, the reference nucleotide can be replaced by the observed allele, and consequently reconstruct the haplotype of each sample. This is the approach used by the CFSAN SNP pipeline.

22.4.5 Getting alleles and allele differences

The allele sequences of the samples can be retrieved by:

- Replacing the nucleotide of the reference genome by the observed alternative allele (check previous question), and then retrieve the sequence of each gene of interest considering the genome annotation of the reference.

- Obtaining the *de novo* genome assembly of each sample, and performing the respective genome annotation. Prokka and NCBI Prokaryotic Genome Annotation Pipeline are commonly used programs for *Salmonella* genome annotation. GLIMMER and RASTk are also used.
- Some allele callers, such as chewBBACA, provide locus-specific alignments in an automated manner, being a good option to determine the allelic profile of samples.

It is important to note that nowadays there are several platforms which can automatically do all this analysis. One of the more commonly used for *Salmonella* is Enterobase, which provides assembly, serotyping and allele calling. Several of these platforms are mentioned in the xMLST section.

22.4.6 Allele based typing

Allele-based typing consists of retrieving clustering information considering the different alleles present in a population for a given set of genes (e.g. the core genome). With the developmentadvent of WGS, the 7-loci based MLST approach was broadened to the use of a cgMLST approach. In this context, there is a public cgMLST scheme which has been widely used in *Salmonella enterica* analysis. This scheme comprises 3,002 loci, and is available in the most commonly used platforms, such as EnteroBase and Ridom SeqSphere+. EnteroBase and BioNumerics also use a wgMLST scheme which, besides the previously mentioned cgMLST loci, include the accessory genes.

Platforms available for cgMLST typing of *Salmonella* include Enterobase, INNUENDO, BioNumerics, CGE, IRIDA, Pathogen Watch, and Ridom SeqSphere+.

22.4.7 SNP based typing

A SNP-based approach relies on the comparison of SNPs in a population. This strategy can be seen as an alternative to the allele-based approach, but many studies actually perform both of them and assess the overlap of the results. For a SNP-based analysis all of the the SNPs that are present in the samples need to be acquired and used to obtain clustering information. Examples of publicly available pipelines for SNP-based typing are:

- Center for Food Safety and Applied Nutrition (CFSAN) HqSNPs pipeline
- Lyve-SET pipeline for HqSNPs typing
- SNV-Phy (Canadian Public Health Agency)
- PHEnix (The Public Health England SNP calling pipeline)

22.4.8 Outbreak definition

As defined by the World Health Organization, "a disease outbreak is the occurrence of cases of disease in excess of what would normally be expected in a defined community, geographical area or season". WGS data provides a high discriminatory power allowing clustering of different isolates (from different geographical areas, and clinical, animal or environmental sources) according to their genomic similarity. This contributes not only to an earlier detection of outbreaks and determination of contamination sources, but also to the detection of more outbreaks, as has been reported by PulseNet network for *Listeria*. Nevertheless, it is still difficult to establish a clear cluster outbreak definition for *Salmonella*, a threshold at which we decide whether two isolates belong to the same genetic cluster, thus linking two cases of infection.

22.4.9 Virulence and AMR

Several genes are important for *Salmonella* ability to cause infection and are medically relevant, such as motility genes, fimbrial adhesins and metabolic genes (Ilyas et al. 2017, Eng et al. 2015). In the particular case of *Salmonella*, horizontally transferred genes strongly influence the course of infection, as they can lead to the emergence of new

phenotypes and favor the adaptation to new niches (Ilyas et al. 2017, Ochman et al. 2000). Such events are not only important for *Salmonella* ability to infect humans, but also for the acquisition of resistance to antimicrobial drugs (Wang et al. 2019). Chloramphenicol, ampicillin, and trimethoprim–sulfamethoxazole are the first-line antimicrobial drugs used to treat *Salmonella* infections. However, over the years resistance towards one or several of these drugs (leading to multidrug resistant isolates) has been emerging (Eng et al. 2015). Alternative antimicrobials have been used. However, resistance towards the alternatives is also appearing, and antimicrobial resistance in *Salmonella* is considered a global threat (Marchello et al. 2020, Eng et al. 2015). For this reason, monitoring of virulence- and antimicrobial resistance-related genes is of great relevance to determine the best way of action in the presence of a case of infection or even an outbreak. As mentioned in the *Virulence and AMR detection section*, where more details can be found, this is performed by comparing the genome to a database comprising a set of genes of interest. Examples of predefined resistome databases are mentioned in the same section.

CHAPTER 23

Listeria monocytogenes analysis

The gram-positive bacterium *Listeria monocytogenes* is the causative agent of Listeriosis, a foodborne disease (reviewed in Buchanan et al. 2017). Listeriosis can be particularly severe, potentially deadly, in elderly and immunocompromised patients. It can cause misscariage in pregnant women or stillbirth. Delay between exposure and illness, and the possibility of consumption of contaminated products spread over-time (eg. frozen products) represent a challenge to the identification of epidemiological links between infection cases (eg. Datta and Burall 2018).

Matle et al. (2020) reviewed the different aspects of Listeriosis. They provide an overview of known *L. monocytogenes* virulence factors, as well as diagnostics and treatment options. The state-of-the-art dry-lab approaches employed to the study of *L. monocytogenes* are described in Luth et al. (2018).

Four evolutionary lineages have been identified in *L. monocytogenes*. These bacteria can be found in a variety of hosts and in environmental samples (Buchanan et al. 2017). *L. monocytogenes* genome is approximately 3Mb, with approximately 2,900 genes (Den Bakker et al. 2010). At least 13 serotypes have been identified (see Figure 1 in Ragon et al. 2008). Early studies showed that despite this diversity, the majority of human infections are caused by isolates belonging to serotypes 1/2a, 1/2b and 4b (eg. Burall et al. 2017). Serotyping is based on an antigen-antibody reaction using somatic (O) and flagellar (H) antigens. Although serotyping has traditionally been used to characterize *L. monocytogenes* isolates, it has gradually been replaced by molecular typing methods that provide enhanced discriminatory power and therefore represent a more suitable approach for epidemiological studies (see eg. Datta and Burall 2018 - not open access - and Matle et al. 2020, for an overview of the methods that have been employed for *L. monocytogenes* analyses).

23.1 Typing methods

L. monocytogenes molecular typing is a constantly evolving field. An ideal typing method presents not only a high discriminatory power, but also high reproducibility and the possibility of automation. Nowadays, different techniques can be applied for *L. monocytogenes* molecular typing, namely:

- **Pulsed Field Gel Electrophoresis (PFGE)** - PFGE is a fragment length restriction analysis (Dalmasso et al. 2014) that has long been considered as the "gold-standard" for *L. monocytogenes* typing due to its high discriminatory power in the pre-WGS era. This method has been used by PulseNet to connect cases of disease through the comparison of their DNA fingerprints, and consequently identify potential outbreaks. Despite its robustness, PFGE is time-consuming, difficult to standardize (Van Walle 2018) and lacks discriminatory power for outbreak

delineation. Nevertheless, despite these disadvantages, PFGE still represented the best compromise between time and discriminatory power in *L. monocytogenes* molecular typing until the advent of NGS technologies. It was thus being considered as the "gold-standard"typing method for *L. monocytogenes* for many years, and played an important role in *L. monocytogenes* surveillance and in the resolution of multiple outbreaks.

- **MLVA (Multiple locus variable tandem repeat analysis)** - Given the drawbacks of PFGE, other typing methods started being considered as good alternatives or at least complements to PFGE analysis. MLVA represents another method of DNA fingerprinting. This method has the advantage of detecting fast-evolving bacterial strains among isolates which may look the same with PFGE. However, it requires highly trained technicians and does not have a standardized protocol for multiple pathogens. This is why it is not used as a routine typing method, but rather as a complementary method to PFGE by PulseNet for some microorganisms, but not *L. monocytogenes*. For this reason, it does not represent a standard method for surveillance of *Listeria*, but it is used by the scientific community to explore the diversity of these bacteria (e.g. Saleh-Lakha et al. 2013, Lunestad et al. 2013). Chenal-Francisque et al. (2013) compares MLVA performance to PFGE and MLST.
- **Multiplex-PCR for classifying 5 serogroups** - Consisting of the amplification of 5 different genes (lmo0737, lmo1118, ORF2110, ORF2819 and prs), this method was developed in order to facilitate serotyping discrimination by quickly classifying *L. monocytogenes* into 5 serogroups (Borucki and Call 2003, Doumith et al. 2004, Matle et al. 2020). Nevertheless, despite being a quick method to implement, it has low discriminatory power, which makes it less suitable for outbreak detection and investigation.
- **MLST (Multi-Locus Sequence Tying)** - DNA sequencing allows unambiguous identification of genetic differences by direct comparison of allele sequences between samples, and sequencing information can be easily shared between laboratories. Therefore, DNA sequencing provides a robust solution for molecular typing. In this context, a 7-gene MLST (housekeeping genes) method was developed for *L. monocytogenes* (Salcedo et al. 2003, Matle et al. 2020). Sequence types (ST) represent unique combinations of the MLST alleles. Clonal complexes (CC) are groups of ST differing by no more than one allele to another isolate belonging to the same CC (Ragon et al. 2008, Henri et al. 2016). A significant drawback of this method is that it requires multiple PCR reactions which cannot be multiplexed.
- **Ribosomal multi-locus typing (rMLST)** - rMLST has also been employed for strain characterisation (Jolley et al. 2012). This typing method has recently been employed for WGS data quality control of *L. monocytogenes* to detect potential intra-species contamination (admixture) of sequencing data (Low et al. 2019).
- **MVLST (Multi-virulence-locus sequence typing)** - Similar to MLST, but considering a set of virulence (prfA, inlB, and inlC) and virulence-related genes (dal, lisR, and clpP), which has been shown to accurately differentiate epidemic clones (see Lomonaco et al. 2013, Cantinelli et al. 2013, Burall et al. 2017b).
- **WGS (Whole-Genome Sequencing)** - With the advent of NGS technologies, WGS technology has led to the improvement of small listeriosis outbreak investigation and is currently being regarded as the new "gold-standard" in the analysis of *L. monocytogenes* (Nadon et al. 2017). By providing information at the genomic level, WGS allows not only a highly discriminatory typing (cgMLST, wgMLST and SNP-typing), but also to establish backward compatibility with previously mentioned molecular typing methods, such as the 7-genes MLST, Multiplex-PCR, rMLST and MVLST, which, for this reason, will tend to continue to be used. Furthermore, it allows the analysis of specific genes, such as virulence factors and antimicrobial resistance genes. Genetic clustering using WGS can be performed on any distance measure (eg. issued from allelic differences detected using cgMLST typing) or evolutionary-model based clustering (ie. phylogenetics) relying on variants/SNPs detection. PulseNet has been implementing WGS for *Listeria* surveillance and outbreak monitoring. Their results have shown that using WGS increases the number of outbreaks detected, and earlier outbreak detection facilitates timely action, thus limiting the extent of outbreaks. Similar studies in the EU have confirmed those findings (Nielsen et al. 2017, Van Walle et al. 2018, Moura et al. 2016). For this reason, efforts have been made in order to make WGS widely used for *Listeria* surveillance, replacing PFGE and serotyping methods. In a near future, WGS-based *Listeria* surveillance is expected to be implemented in most developed countries.

23.2 "One Health" surveillance and WGS of *L. monocytogenes*

The identification of infection sources is essential for outbreak resolution. Hence, an integrated analysis of clinical, food and veterinary samples relying on the concept of One Health is the key to achieve a good surveillance system. As shown here by PulseNet network, the high discriminatory power of WGS increases the chances to find the source of infection, and possibly reduces the time that it takes to identify the source. Indeed, as reported by the WHO, the use of WGS on *Listeria* strains has resulted in more accurate detection of clusters and allowed more outbreaks to be successfully resolved. However, several factors are hindering the implementation of a generalized WGS-based surveillance system. For instance, while the notification of clinical cases of listeriosis is mandatory in most EU members, most of the monitoring data on *L. monocytogenes* in animals and food are generated by non-harmonised monitoring schemes across member states and for which mandatory reporting requirements do not exist (ECDC 2019). Moreover, WGS requires laboratories to be equipped with expensive technologies, and highly skilled technicians able to analyze the data, which is not affordable for many countries. Furthermore, the resources for implementation of WGS differ between different sectors (human health, animal health and food safety), thus complicating the implementation of "One health" surveillance. Despite all these issues in the implementation of a proper WGS-based system, as *L. monocytogenes* was the selected bacteria to start implementing WGS at an European level, it is already some steps ahead from other pathogenic agents regarding WGS-based surveillance. Indeed, *L. monocytogenes* is currently the most frequently WGS-based typed pathogen for surveillance and outbreak investigation in all sectors in Europe (ECDC et al. 2019), and according to the ECDC roadmap, the capacity of the member states for use of WGS as a complement or replacement technology for PFGE is already significant and progressing fast.

23.3 WGS lab protocol

23.3.1 DNA extraction

Before DNA extraction, *L. monocytogenes* is cultured in the laboratory (usually liquid media). For proper growth, these bacteria need a medium containing the seven amino acids for which they are auxotrophic (arginine, cysteine, glutamine, isoleucine, leucine, methionine, and valine) and four additional vitamins (biotin, riboflavin, thiamine, and thioctic acid). Brain Heart Infusion (BHI) is a nutrient-rich medium harboring all these ingredients, thus being the most commonly used medium for *Listeria* culture (check Jones and D'Orazio 2013 for more details). An overnight incubation at 37 ºC in BHI is usually performed before DNA extraction. Regarding DNA extraction, there is no standard methodology or kit used for *L. monocytogenes*. However, commonly used kits include DNeasy Blood and Tissue kit (Qiagen) or Wizard Genomic purification kit (Promega).

23.3.2 Sequencing technology

There is not a prefered WGS technology to sequence *L. monocytogenes*. Similar to other fields, Illumina paired-end reads represent the most commonly used strategy. Due to the number of samples that can be handled at a single run and the possible higher read size, MiSeq sequencing machines seem to be the choice for many labs. Long-read sequencing technologies are now becoming more frequently used (alone and/or in combination with short-read sequencing). This because of the improvements in the error rates and price and a chance to improve or complete genome assemblies.

23.4 Bioinformatics protocol

23.4.1 Mapping or assembly

The first step to perform when receiving the sequencing data of your samples, is to evaluate the sequencing quality and perform trimming and cleaning of the reads (see *Data preprocessing*).

The cleaned sequence data can then be used for downstream analysis following one of two approaches (or both in parallel, check [Data production][../Pipelines/data_production.md]):

- *de novo* genome assembly of the sample(s),
- Read mapping of each sample on a reference sequence (obtained from a database or by *de novo* genome assembly of one of your samples).

Both approaches are commonly used for *L. monocytogenes*.

De novo genome assemblers that can be used for *L. monocytogenes* include SPAdes and Velvet. Both of them perform very well and are freely available. There are command-line pipelines, such as INNUca, which incorporate these programs and provide the opportunity to automatically perform all the analyses from quality control to genome assembly. If a platform with predefined pipelines (and that usually does not require bioinformatics skills) is preferred, CLC Genomic Workbench and Ridom SeqSphere+ can be used for *L. monocytogenes*.

As for read mapping, BWA and Bowtie are often used in *L. monocytogenes* analysis. The Center for Food Safety and Applied Nutrition (CFSAN) of USA developed the CFSAN SNP pipeline, which is tailored to create high quality SNP matrices for sequences from closely-related pathogens. This pipeline covers all the steps of the analysis from read mapping to calculation of SNP distances and reconstruction of the haplotypes. Therefore, despite requiring some bioinformatics skills, it may represent an alternative to the development a new own pipeline and it is commonly used for *L. monocytogenes* analysis (eg. Hurley et al. 2019, Scaltriti et al. 2020, Portmann et al. 2018). Similar to the genome assembly step, a platform with predefined pipelines for read mapping can be used. In this case, Ridom SeqSphere+ is the most commonly used one.

23.4.2 Choosing a reference genome

Should an analysis require the use of a reference genome, the choice of the reference genome is a crucial step. Analyses relying on read-mapping approaches might be strongly influenced by reference choice, as the genetic distance between the reference and the sample may influence the performance of downstream steps, namely SNPs/INDELs calling (Pightling et al. 2014, Pightling et al. 2015). This reference can be picked from the samples at hand (after genome assembly), or from a public database. If a sample is used as the reference genome, studies on *L. monocytogenes* usually perform preliminary analysis (e.g. hierarchical clustering based on some distance, such as mash, ANI or allelic differences - eg. MLST analysis) and then select a strain of each cluster to use as reference. If instead a publicly available genome is used, the most widely used ones are *L. monocytogenes* strain EGD-e, which is the reference genome assembly of NCBI database, strain CFSAN029793 (e.g. Ottesen et al. 2020, Chen et al. 2017) or strain 08-5578, strain HPB5622 (Pightling et al. 2014, Pightling et al. 2015). EGD-e is the reference strain for Lineage II and F2365 for reference strain for Lineage I (Knudsen et al. 2017). If the objective is to discriminate between highly related samples that may/may not belong to a single outbreak, using one outbreak isolate as reference and combining multiple analyses approaches might maximize the resolution of your analyses (eg. Chen et al. 2017).

23.4.3 Getting SNPs

How to detect SNPs is described earlier.

Briefly, there are three different approaches.

- Perform *de novo* genome assembly of each sample, and then align their genomic sequences. Studies involved in *L. monocytogenes* analysis use very often SPAdes, CLC Genomic Workbench or Ridom SeqSphere+ to obtain the assembly, and progressiveMAUVE, BLAST or MUSCLE to align the genomes. This multi-sequence alignment is the input for phylogenetic and clustering analysis (see sections on phylogeny and clustering). If instead of genome analysis, only the SNPs in the genes are of interest, alignments can be performed with eg. Roary or Panaroo for pangenome.
- Use a reference genome where the reads of all the samples will be mapped, and then use a variant-calling pipeline to determine the polymorphic positions. Studies involved in *L. monocytogenes* analysis use mostly

BWA and Bowtie for read mapping, and GATK and VarScan for variant-calling. Of note, many of them also use the CFSAN SNP pipeline for both processes.

- Determine the polymorphic positions in the sample by analyzing the k-mer pattern using kSNP. For this approach either the genome assembly, or the cleaned genomic reads are needed. This is the less frequently used approach for *L. monocytogenes*.

Each of these approaches will provide information about the genetic variability in the dataset. This information can then be used to perform SNP-based clustering and phylogenetic analysis.

23.4.4 Getting alleles and allele differences

The allele sequences in the samples at hand can be retrieved by:

- Replacing the nucleotide of the reference genome by the observed alternative allele (check previous question), and then retrieve the sequence of each gene of interest considering the genome annotation of the reference.
- Obtaining the *de novo* genome assembly of each sample, and:
  - Perform the respective genome annotation. Prokka is a commonly used program for *L. monocytogenes* genome annotation. BLAST or MUSCLE can be used to align the predicted genes to the set of genes of interest and identify homology relations. Alternatively, a less commonly used approach in the study of *L. monocytogenes* genomes, is the use of a program like eggNOG mapper to perform functional annotation.
  - Use BLAST or MUSCLE to align a set of genes of interest on the genome assembly and identify the respective homologs.
- Some allele callers, such as ChewBBACA, provide locus-specific alignments in an automated manner, being a good option to determine the allelic profile of samples.

It is important to note that nowadays there are several platforms which can automatically do all this analysis. Several of these are mentioned in the xMLST section.

23.4.5 Allele-based typing

Allele-based typing consists of retrieving clustering information considering the different alleles present in a population for a given set of genes (e.g. the core genome). With the advent of WGS, the 7-loci based MLST approach was broadened to the use of a cgMLST or wgMLST approach. In this context, there are two public cgMLST schemes which have been widely used in *L. monocytogenes* analysis considering an allele-based approach, namely, a 1,701-loci scheme proposed by Ruppitsch et al. (2015), and a 1,748-loci scheme proposed by Moura et al. (2016). Although a standardized approach for cgMLST analysis would be ideal, in reality both schemes seem to work well and provide similar results (Van Walle et al. 2018), and no preference is given to any of them. Nevertheless, when using automated platforms, usually only a single scheme is available. For example, BIGSdb uses the cgMLST scheme proposed by Moura et al. (2016), while Ridom SeqSphere+ uses the scheme proposed by Ruppitsch et al. (2015), (the scheme can be found here: https://www.cgmlst.org/ncs). Besides cgMLST, some studies also perform a MVLST analysis, considering a set of virulence-related genes, which has been shown to accurately differentiate epidemic clones (check Lomonaco et al. 2013).

Platforms available for cgMLST typing of *L. monocytogenes* include BIGSdb (Jolley & Maiden 2010), BioNumerics, CGE, IRIDA, Pathogen Watch, and Ridom SeqSphere+. However, cgMLST analysis can also be done outside a platform with software such as ChewBBACA and MentaLiST.

23.4.6 SNP-based typing

A SNP-based approach relies on the comparison of SNPs in a population. This strategy can be seen as an alternative to the allele-based approach, but many studies actually perform both of them and assess the overlap of the results.

For a SNP-based analysis all of the the SNPs that are present in the samples need to be acquired and used to obtain clustering information. Examples of publicly available pipelines for SNP-based typing are:

- Center for Food Safety and Applied Nutrition (CFSAN) HqSNPs pipeline
- Lyve-SET pipeline for HqSNPs typing
- SNV-Phy (Canadian Public Health Agency)
- PHEnix (The Public Health England SNP calling pipeline)

23.4.7 Outbreak definition

As defined by the World Health Organization, "a disease outbreak is the occurrence of cases of disease in excess of what would normally be expected in a defined community, geographical area or season". For foodborne diseases, outbreaks can be defined as two or more cases linked to the same food source (Hoezler et al. 2018). Henri et al. (2017) showed that clustering of a diverse dataset of *L. monocytogenes* isolates from food origin, using three different approaches (cgMLST, wgMLST and SNP based phylogeny) was highly concordant.

Regarding the interpretation of WGS data and the respective genetic clusters, in the particular case of *L. monocytogenes*, some thresholds appear to be more commonly used to define a cluster/outbreak. They are:

- 4 or 7 cgMLST allelic differences (AD) considering any of the above-mentioned cgMLST schemes (Van Walle et al. 2018). Cabal et al. (2019) also considered clusters 10 AD.
- Using PHE SnapperDB pipeline, Nielsen et al. (2017) found that the maximum HqSNPs pairwise distance between outbreak isolates were < 5 SNPs (5/9 outbreaks) while the pairwise distances of the remaining 4/9 outbreaks studied was between 8-21 HqSNPs.

However, despite cutoff-thresholds (eg., allelic differences for cgMLST, or SNPs pairwise differences between isolates belonging to a single cluster) being commonly used, those thresholds are dependent on your workflow. For example, Chen et al (2017) demonstrated that pairwise differences in SNP/allele count was not necessary and sufficient to distinguish between cheese outbreak isolates from related non-outbreak isolates. Moreover, thresholds are not directly transposable between studies, therefore it is good practice to access the sensitivity and specificity of the workflow when evaluating which threshold could be appropriate.

23.4.8 Viruence and AMR

Several genes are important for *Listeria* ability to cause infection and are medically relevant, such as internalins (inlA, inlB, inlF, or inlJ, essential for adhesion and invasion) or the prfA-regulated virulence gene cluster (pVGC) (Vázquez-Boland et al. 2001, Ward et al. 2004, Poimenidou et al. 2018). *Listeria* are naturally susceptible to penicillin, ampicillin, amoxicillin, gentamicin, erythromycin, tetracycline, rifampicin, co-trimoxazole, vancomycin and imipenem (Goméz et al. 2014, Byrne et al. 2016). Nevertheless, reports of antimicrobial resistance towards one or several of these compounds has been reported (eg. Boháčová et al. 2018, Escolar et al. 2017, Kevenk et al. 2015). For this reason, monitorization of virulence- and antimicrobial resistance-related genes is of extreme relevance to determine the best way of action in presence of a case of infection or even an outbreak. As mentioned in the *Virulence and AMR detection section*, where more details can be found, this is performed by comparing the genome to a database comprising a set of genes of interest. Examples of predefined resistome databases are mentioned in the same section.

CHAPTER 24

Challenges for One Health surveillance

Foodborne diseases (FBDs) represent an epidemiological scenario in which pathogens and humans contact through contaminated food or water. FBDs affect 600 million people every year, representing an important burden for human health (WHO). Proper management of FBDs requires a good understanding of the disease, a fast detection and a proper response (WHO 2018). Therefore, good surveillance systems able to track the circulation of pathogens and monitor their medically relevant features, such as the resistome or the virulome, are essential for disease control. Such systems could, for example, allow the early detection of multidrug-resistant pathogenic populations and lead to the implementation of specific control strategies reducing future human or even animal illness. Moreover, information regarding virulence potential or drug resistance may be crucial for public health decision-making on the allocation of resources during outbreaks.

24.1 WGS and One Health surveillance

As described in this handbook, WGS technologies represent the most discriminatory pathogen typing method (thus far), thus providing the highest surveillance resolution. For example, the application of WGS for listeriosis surveillance has increased the number of detected outbreaks, but decreased the number of cases per outbreak, thus contributing to a reduction of human illness (PulseNet). A major factor leading to these results is WGS highest ability to identify the sources of infection. For this reason, an integrated system where the clinical and the food and water sectors not only surveil for the presence of pathogens and their characteristics, but also interchange their data is a key to the proper FBDs management. In this regard, WGS represents an important advance as WGS data can be easily shared and compared between laboratories at national and international levels. Nevertheless, WGS implementation is challenging and many things need to be taken into consideration. In this section we do a brief overview of the challenges associated to WGS implementation in the surveillance of foodborne diseases which were raised by the WHO in 2018.

24.1.1 Organizational perspective

The existence of an integrated system for FBDs surveillance requires the implementation of different strategies which may be hampered at different stages. For instance, as reported by WHO (2018), from an organizational perspective, public authorities need to work on the implementation of adequate legislation and regulation that encourages the clinical and the food and water sectors to perform continuous FBDs surveillance and to report their results. However, to demand such a protocol, investment in proper surveilling infrastructures needs to be made. For example, the

implementation of WGS-based surveillance requires a technological transition which is not affordable in many countries. Also, resources and funding opportunities are unequal between different sectors (Human health, Animal Health, Environmental Health and Food Safety).

24.1.2 Scientific and technical perspective

From a scientific perspective, WGS-based surveillance systems require multi-tasked teams with different scientific backgrounds for proper data analysis, including specialized microbiologists and bioinformaticians (WHO 2018). Hence, investment (not only financial but also of time) in training highly skilled technicians to integrate the different teams is crucial for proper data analysis and interpretation of the results. Furthermore, there is a need to guarantee inter-laboratory data integration and comparability of results. For example, efforts should be made in the harmonization of genetic nomenclatures, because, as mentioned before in this handbook, different platforms use different allele nomenclatures hampering the comparison of their data. Such efforts in training human resources, standardizing protocols and in the harmonization of the results would definitely contribute to ease the comparison between different labs, sectors and even countries. In this regard, some technical aspects may also influence the implementation of such systems, such as the need for high computing power to handle the data, or servers where data can be stored and shared between partners. Good political strategies for investment have the power to overcome these issues. Noteworthy, data protection should always be guaranteed, and robust anonymization strategies implemented.

24.1.3 Cultural barriers

Ultimately, even if all the above-mentioned barriers for the implementation of a proper WGS-based surveillance system are overcome, cultural barriers may also be an issue (WHO 2018). Besides the language barrier in international cooperation, and the usual resistance to change, cross-sectoral collaboration may be difficult to implement. As stressed by several international entities during the last years, actions promoting the concept of One Health, and showing the relevance of this integrated perspective where human, animal and environmental health are connected, may contribute to raise the awareness of politicians, scientists, health practitioners, and ultimately all the citizens and surpass some of the above-mentioned obstacles (ECDC 2017).

24.2 Future perspectives

WGS is gradually being integrated in the surveillance system of developed countries. The possibility of connecting WGS data with data obtained from previous technologies allows a gradual transition, where data from the past is not lost because data from different technologies are still comparable with WGS. An agreement has been made that *Listeria* would be the first bacterial pathogen where such a system would be implemented, and this has been progressing fast. Meanwhile, WGS is gradually expanding to other FBDs causative agents, such as *Salmonella*, *Escherichia* or *Campylobacter*, but there is still a long way to go. This gradual technological transition is important for a proper allocation of resources and management of the different obstacles. In the end, it is expected that in a few years WGS-based surveillance systems are fully operational for several FBDs monitorization from a One Health perspective.

CHAPTER 25

How to contribute to this project

The OH sequencing for surveillance handbook project is an open documentation project, and we welcome all contributions to this handbook.

25.1 Contributor Agreement

By contributing, you agree that we may redistribute your work under the license that this project uses. In return, you will be recognized as a contributor to this project, which will be reflected in the AUTHORS document in this repo. We expect all contributors to abide by the project's code of conduct.

25.2 How to contribute

The easiest way to get started is to file an issue to tell us about a spelling mistake, some awkward wording, or a factual error. If you do not have a GitHub account, you can send us comments by email to ngs-handbook@groups.io. However, we will be able to respond more quickly if you use one of the other methods described below. If you have a GitHub account, or are willing to create one, but do not know how to use Git, you can report problems or suggest improvements by creating an issue. This allows us to assign the item to someone and to respond to it in a threaded discussion. If you are comfortable with Git, and would like to add or change material, you can submit a pull request (PR). Instructions for doing this are included below.

25.3 What to contribute

We are very happy to receive any and all contributions regarding the subject matter of this handbook in whichever manner you are able to submit them. Many different things can be contributed, you can send us an email about relevant reports, you can submit an issue informing about an error you found, or you can submit a pull request contributing text on a new method or a procedure.

25.4 Using Github.

If you choose to contribute via GitHub, you may want to look at How to Contribute to an Open Source Project on GitHub.The current version of the handbook is to be found in the master branch. Contributions to this will be checked by the editors before being added to the documentation.

How to proceed:

- Fork (i.e. copy) the project to your account
- Go to your fork.
- Then add/modify the contents you want. This can be done directly in your fork on github. We use mostly markdown format.
- Commit the change.
- Send your pull request.
- Your pull request will manually checked, and merged into the documentation
- The documentation will then be auto-rebuilt, and your change will be available.

CHAPTER 26

Contributors to this handbook

26.1 Editors

- Karin Lagesen (NVI)
- Thomas H. A. Haverkamp (NVI)
- Eve Zeyl Fiskebeck (NVI)
- Håkon Kaspersen (NVI)
- Jeevan Karloss (NVI/NIPH)
- Mohammed Umaer Naseer (NIPH)
- Vitor Borges (INSA)
- Verónica de Pinho Mixão (INSA)
- Miguel Pinto (INSA)

26.2 Contributors

- Wonhee Cha (SVA)
- Olov Svartström (FOHM)
- Eva Moller Nielsen (SSI)
- Mia Torphdahl (SSI)
- Eva Litrup (SSI)
- Katrine Grimstrup Joensen (SSI)
- Lesley Larkin (PHE)
- Taran Skjerdal (NVI),

- Camilla Sekse (NVI)
- Tim Dallman (PHE)
- Tasja Buschhardt (BfR)
- Matthias Filter (BfR)

26.3 Institutions

- NVI - Norwegian Veterinary Institute (Norway)
- NIPH - Norwegian Institute of Public Health (Norway)
- SVA - National Veterinary Institute (Sweden)
- FOHM - Public Health Agency of Sweden
- SSI - Statens Serum Institut (Denmark)
- PHE - Public Health England (UK)
- RIVM - National Institute for Public Health and the Environment (Netherlands)
- BfR - German Federal Institute for Risk Assessment (Germany)
- INSA - Instituto Nacional de Saúde Dr. Ricardo Jorge (Portugal)

CHAPTER 27

Contributor Covenant Code of Conduct

27.1 Our Pledge

In the interest of fostering an open and welcoming environment, we as contributors and maintainers pledge to making participation in our project and our community a harassment-free experience for everyone, regardless of age, body size, disability, ethnicity, sex characteristics, gender identity and expression, level of experience, education, socio-economic status, nationality, personal appearance, race, religion, or sexual identity and orientation.

27.2 Our Standards

Examples of behavior that contributes to creating a positive environment include:

- Using welcoming and inclusive language
- Being respectful of differing viewpoints and experiences
- Gracefully accepting constructive criticism
- Focusing on what is best for the community
- Showing empathy towards other community members

Examples of unacceptable behavior by participants include:

- The use of sexualized language or imagery and unwelcome sexual attention or advances
- Trolling, insulting/derogatory comments, and personal or political attacks
- Public or private harassment
- Publishing others' private information, such as a physical or electronic address, without explicit permission
- Other conduct which could reasonably be considered inappropriate in a professional setting

27.3 Our Responsibilities

Project maintainers are responsible for clarifying the standards of acceptable behavior and are expected to take appropriate and fair corrective action in response to any instances of unacceptable behavior.

Project maintainers have the right and responsibility to remove, edit, or reject comments, commits, code, wiki edits, issues, and other contributions that are not aligned to this Code of Conduct, or to ban temporarily or permanently any contributor for other behaviors that they deem inappropriate, threatening, offensive, or harmful.

27.4 Scope

This Code of Conduct applies both within project spaces and in public spaces when an individual is representing the project or its community. Examples of representing a project or community include using an official project e-mail address, posting via an official social media account, or acting as an appointed representative at an online or offline event. Representation of a project may be further defined and clarified by project maintainers.

27.5 Enforcement

Instances of abusive, harassing, or otherwise unacceptable behavior may be reported by contacting the project team at ngs-handbook@groups.io. All complaints will be reviewed and investigated and will result in a response that is deemed necessary and appropriate to the circumstances. The project team is obligated to maintain confidentiality with regard to the reporter of an incident. Further details of specific enforcement policies may be posted separately.

Project maintainers who do not follow or enforce the Code of Conduct in good faith may face temporary or permanent repercussions as determined by other members of the project's leadership.

27.6 Attribution

This Code of Conduct is adapted from the Contributor Covenant, version 1.4, available at https://www.contributor-covenant.org/version/1/4/code-of-conduct.html

For answers to common questions about this code of conduct, see https://www.contributor-covenant.org/faq

This website aims to be a handbook that can help national and local labs to build capacity and competence on the use of NGS methods for surveillance purposes. This work was started as part of the "One health surRveillance Initiative On harmonization of data collection and interpretatioN (ORION)" One Health EJP project, and continued as part of the "BeOne: Building Integrative Tools for One Health Surveillance" project.

The ORION project, launched in 2018, aimed at establishing and strengthening inter-institutional collaboration and transdisciplinary knowledge transfer in the area of surveillance data integration and interpretation, along the One Health (OH) objective of improving health and well-being. The BeOne project, launched in 2020, aims at developing an integrated surveillance dashboard in which molecular and epidemiological data for foodborne pathogens can be interactively analysed, visualised and interpreted by the relevant experts across disciplines and sectors.

This handbook consists of several parts, to understand more about how this handbook works, please read more about it in the About page.

New contributions to this handbook are very welcome. For instructions on how to contribute, please see our Contributing page.