Third-year progress report for thesis advisory committee

This manuscript (permalink) was automatically generated from mbhall88/TAC3_Report@e175c71 on September 25, 2020.

Authors

- **Michael B. Hall**
  0000-0003-3683-6208 · mbhall88 · mbhall88
  EMBL-EBI; University of Cambridge · Funded by EMBL International PhD Programme (EIPP)

Thesis Advisory Committee

- Zamin Iqbal (Supervisor) - EMBL-EBI
- John Marioni (Chair) - EMBL-EBI
- Georg Zeller - EMBL Heidelberg
- Estée Török - University of Cambridge

Starting Date: 12/10/2017
Qualifying Assessment Date: 06/07/2018
Second TAC Meeting: 15/10/2019
Third TAC Meeting: 13/10/2020

Part A: Progress Report

Examining bacterial variation with genome graphs

Executive summary

Genomics is now ubiquitous in clinical and public health microbiology, at least in the developed world. However, many significant challenges remain.

- Bacterial genomes harbour huge amounts of diversity, even within a species, and traditional reference-based approaches are problematic.
- Much of the variation in bacteria is fundamentally inaccessible to short reads.
- Long Nanopore reads are noisy, and SNP calling with this data is not properly benchmarked or standardised.
- Since *Mycobacterium tuberculosis* (Mtb) infects so many people, there is potential for considerable impact for clinical applications.
- There is also much to be gained from a high-quality pan-genome of Mtb as well as a detailed map of its enigmatic *pe/ppe* gene repertoire.

These motivations drive the following PhD thesis structure:

1. Develop algorithms and software for variant discovery using bacterial genome graphs, building on work of a previous student in the lab (my first paper, second author).
2. Benchmark Nanopore versus Illumina SNP calling, showing our algorithms meet the needs of clinical and public health users, validate, and publish (second paper).
3. Improve upon current whole-genome sequencing-based drug resistance prediction for Mtb using genome graphs.
4. Curate a high-quality reference pan-genome for Mtb that includes a detailed map of the *pe/ppe* genes.

Motivation and Background

In 2018, 1.4 million people died of tuberculosis (TB) globally, and over 10 million people fell ill to the disease. There were also 484,000 new cases of rifampicin-resistant TB reported, of which 78% were multi-drug resistant (MDR) [1]. Drug-resistant TB is not a new problem; the first clinical trial using Streptomycin to treat TB reported resistance and this outcome influenced the creation of the four-drug first-line regimen used today. Standard of care requires phenotypic testing of the infecting organism against these four drugs to ensure that appropriate treatment is prescribed. However, *Mycobacterium tuberculosis* (Mtb), the causative agent of TB, is a slow-growing organism and the gold-standard phenotypic tests ("mix bug and drug") take around two months to complete. Thus, the traditional clinical diagnostic and treatment regimen is slow and expensive. Whole-genome sequencing offers a faster solution; recently it was shown that equivalent results are achievable by sequencing Mtb grown in "liquid culture" (also known as MGIT, Mycobacterial Growth Indicator Tube) after two weeks of culture in contrast to the two-month traditional (Lowenstein-Jensen) culture method [2]. A number of genes are implicated in drug resistance and predicting resistance from sequencing data based on a catalogue of resistance single-nucleotide polymorphisms (SNPs) and indels (insertions or deletions) works with high-specificity [3,4]. For the four first-line drugs, a study by the CRyPTIC consortium is the first of its kind to demonstrate that phenotyping is not required if genotype predicts susceptibility [5]. However, as the genetic basis for drug resistance is not entirely understood, there is still a sensitivity gap that differs drug-by-drug.

Public health requirements for TB diagnostics are **resistance prediction, species identification, and clustering** of genomes. The clusters are intended to contain potential transmission events, thus enabling more effective contact tracing. All of the main requirements are currently successfully achieved with Illumina sequencing technology. However, there are reasons to consider abandoning Illumina for Oxford Nanopore Technology's (ONT) sequencer. First, there is cost - Illumina has raised its reagent prices considerably. Second, the burden of TB lies primarily in places where there is neither capital nor infrastructure for purchasing or maintaining a large machine. The third is speed: Votintseva *et al.* showed that it was feasible to go from patient to result via a Nanopore sequencer in 12.5 hours [6]. Since this publication, the yield and quality of Nanopore has risen. Therefore, a diagnostic that delivers results while the patient is in the clinic is now imaginable.

As an aside, it is worth noting that in high prevalence areas, there is no interest in transmission analysis, and therefore all of the information that they need (species and resistance) could be attained via deep amplicon sequencing. We accept this but ignore it and focus on a whole-genome solution nevertheless; both because of the better phylogenetic resolution, and because we do not yet know the relevant genetic mutations associated with new drugs.

The idea of using a single, linear reference genome to represent a population of individuals is not ideal. Even more so in the context of a bacterium such as *Salmonella enterica*, where two individuals can differ to such a degree that they only share 16% of their genes [7,8]. This fluidity of genetic content leads to the concept of a pan-genome; defined as the set of all genes observed in a species. Some genes are more common than others within this pan-genome, leading to the distinction of the 'core' and 'accessory' genome with the core being those shared across (most) all individuals and accessory being everything else. The effect of using a single-reference genome to describe variation within samples from the same species is best illustrated in Figure 1. In this toy example, the maximum number of SNPs we can hope to discover is only 56%. Clearly, comparing many samples requires better methods than single-reference based ones.

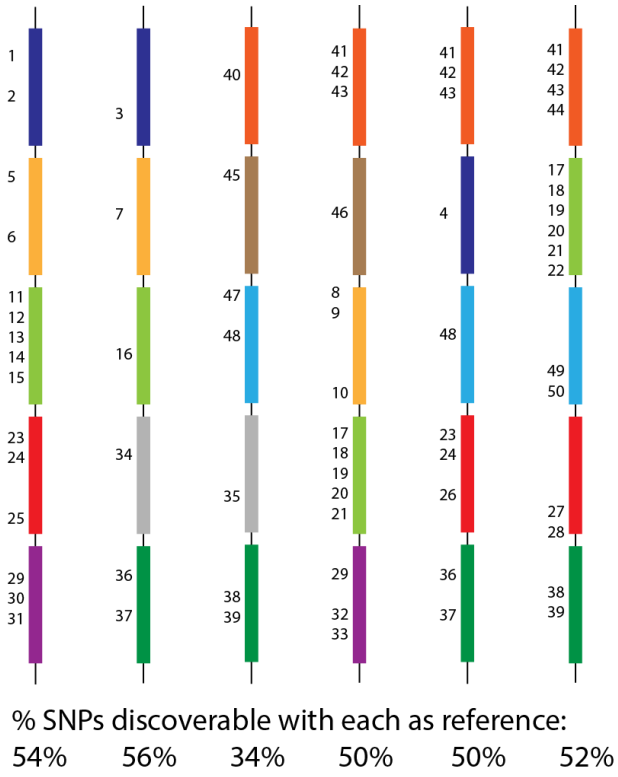

Figure 1: *An illustration of how reference bias can impact variant-calling. Each vertical column signifies an individual genome, with the coloured blocks representing genes. Numbers label 50 segregating SNPs. The percentages at the bottom express the proportion of SNPs which can be detected using each of the 6 'genomes' as a reference by mapping perfect reads from the remaining 5 to this reference.*

One of the first questions that arise from a computational point-of-view is how to apply current methods, which assume a single reference, to a pan-genome? An approach to answering this question that has been gaining considerable traction in recent years is that of genome graphs. This new paradigm uses a population reference graph (PRG) as its equivalent of a single, linear, reference genome. A PRG is built to represent variation seen within a population, conceding the fact that no single genome can accurately represent an entire species. To construct such a PRG, one takes a multiple sequence alignment (MSA) and collapses shared sequence and creates "bubbles", or branch points, where they do not (see Figure 2 for an illustration of this process). We define a collection of PRGs as a pan-genome reference graph (PanRG), which we will interchangeably refer to as a pan-genome. Thus far, genome graph methods have focused mainly on eukaryotes. Given the rich diversity of genetic content in the prokaryote world, it would seem genome graphs are more suited to utilisation there.

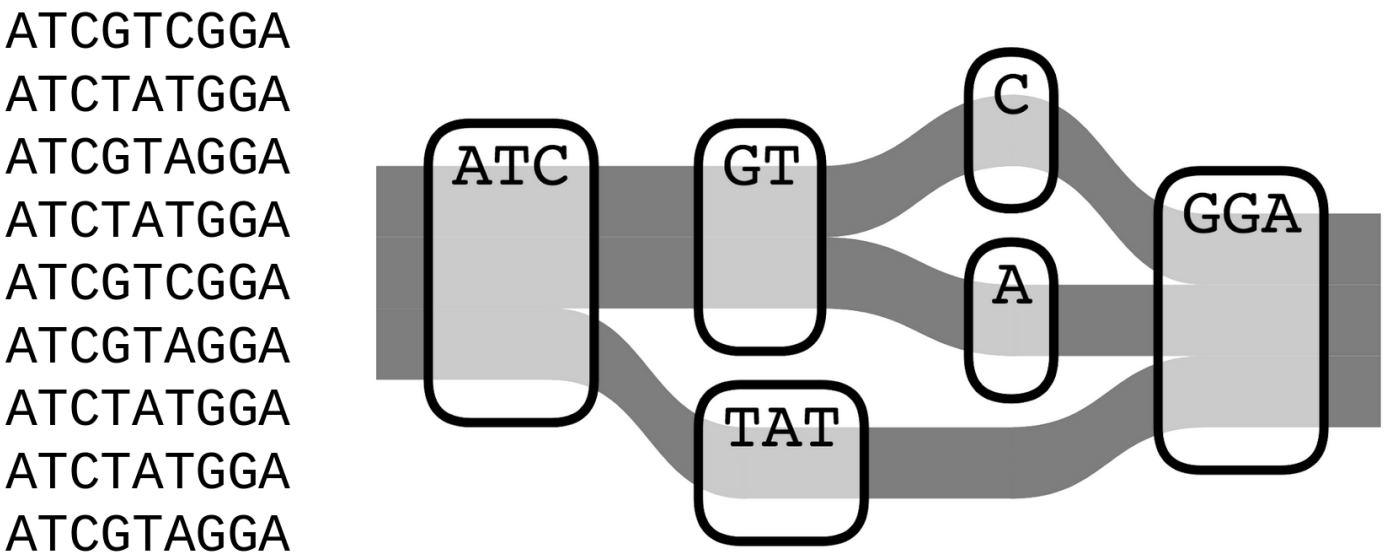

Figure 2: *An illustration of how a population reference graph (PRG) is constructed. Regions (columns) of shared sequence are collapsed into a single node. Those that differ are split into "bubbles", or branching nodes.*

Nanopore sequencing yields very long reads (up to 2.2Mbp [9]), the mean/mode read identity tends to fall in the range of 87-94% (using ONT's `guppy` basecaller), while on a consensus level, it can achieve 99.94% with assembly polishing [10].
Variant calling with Nanopore sequencing data has seen a somewhat slow development. Currently, the main tools that have had reasonable testing done and are versatile enough to detect both SNPs and (some) indels are `nanopolish` [11], Clair [12] and `medaka` [13]. A recent benchmark showed that Nanopore variant calling provides reliable diagnostic information for *Neisseria gonorrhoeae* [14]. However, to date, there has been no extensive Nanopore variant calling benchmark done for Mtb. Given the potential benefits of using genome graphs and long-read Nanopore sequencing for bacterial genomics, it makes sense to try and blend the two.

Pandora is a method being developed in the group to genotype across the *entire* pan-genome of a bacterial sample - not just the core. It does this by working with a PRG, rather than a linear reference. The method is based on the following intuition (similar to that behind the Li and Stephens model in population genetics): genomes evolve by recombination and mutation, and thus we ought to be able to approximate a $N + 1$ genome as a mosaic of the first N genomes. `pandora` maps Nanopore reads to a graph encoding of a PRG, infers a mosaic, and provides genotypes at all variants in the PanRG. Mapping is done using minimising k-mers [15] in a similar vein to that done by `minimap` [16], and is therefore fast. By using Nanopore sequencing data, it is also possible to infer gene order as, in general, a single read will contain multiple genes, as opposed to Illumina sequencing where multiple reads are required to span a single gene. Note `pandora` does not (prior to this work) include any facility for discovering novel variation.

Summary

This PhD seeks to develop new methods to enable Nanopore-based diagnostics and epidemiology for Mtb and other bacteria. By using PRGs as a strong prior [17,18], we should be able to mitigate the Nanopore error biases and indel issues. In the process, we aim to construct a high-quality reference pan-genome for Mtb that we hope will open previously inaccessible parts of its genome for investigation.

Chapter 1: Variant discovery in genome graphs

Variation in bacterial genomes can arise through a diverse range of processes. Mutations can arise during replication and are inherited vertically, genetic material can be transferred horizontally, and homologous recombination can lead to eukaryote-like gene conversion [19]. This breadth of ways in which bacteria can acquire new and varied genetic material results gives rise to the phenomenon of a pan-genome. Bacterial species with an "open" pan-genome may have individuals in their population who can share as little as 16% of their genes (*S. enterica*) [7]. Not all species' pan-genomes are this open, but it does raise the question: what do we use as a "reference" genome for such a species? One solution is to use the reference genome for the specific strain of interest. This works fine when dealing with a single sample or multiple samples of the same strain. However, when expanding to many samples from varying strains, the reference is no longer representative. An alternative solution is to focus solely on the core genome - the complement of genes found in all members of a species (see Figure 3). The issue with this approach is the loss of information about variation in all of the non-core genes, which could be a large number if the pan-genome is open. See Figure 3 for an illustration of this reference-bias problem.

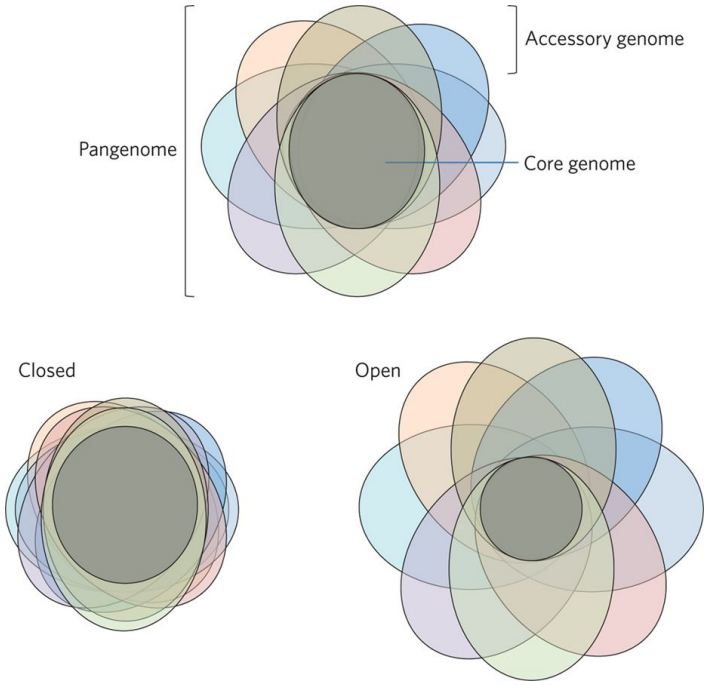

Figure 3: *Schematic representation of a pan-genome. Each oval represents the genetic content of a single individual within a species. Reference [7].*

The message we are aiming to drive home at the beginning of this chapter is that reference-based analysis fundamentally cannot access all of the variation in a bacterial species.

Prior work: Mosaic approximations and genotyping

Standard approaches to variant analysis are effectively a first-order approximation. In such an approximation, samples are considered identical to the reference, one maps reads to it, identifies clear SNPs via the read pileup, and then modifies the reference to get an estimate of the sample's genome. As mentioned in Motivation and Background, `pandora` is a method developed by a previous PhD student in the lab, Rachel Colquhoun. It works on the premise of approximating a genome as a hierarchical mosaic. At a high-level, it represents a mosaic of loci - usually genes and intergenic regions - while at the locus-level, it is a mosaic of previously-seen genomes.

`pandora` has two main workflows: `map` and `compare`. `map` aims to infer a consensus sequence from a PRG for a single sample. The consensus sequence says two things about the sample: the loci present in the sample, and the most likely path through the PRG for each present locus. Optionally, `map` can run genotyping of the sample, providing a Variant Call Format (VCF) file for the sample with respect to the consensus sequence (or a user-provided sequence). `compare` aims to infer a consensus sequence that best represents a given set of genomes and then genotype each sample with respect to this inferred "average" genome. The advantage of this approach is that the consensus best represents the specific samples provided. This allows for genotyping across the *entire* pan-genome. If a gene is present in only 2 of 50 samples then genotyping information is provided for those 2 samples and null for the other 48.

While `pandora`, before the work in this chapter, allows comparison of genomes to a level of detail provided by no other tool, there is still a significant shortcoming: it cannot discover novel variation. If a sample contains a variant not present in the PRG, the best `pandora` can do is select the path that is closest to that variant. The work in my first chapter outlines a method for removing this limitation within `pandora` and provides an analysis of the gain in recall and precision by incorporating *de novo* variant discovery into the `pandora` workflow.

Local *de novo* variant discovery in a genome graph

There are two significant difficulties in discovering *de novo* variants on a graph. The first is finding regions within the graph that look like the reads mapping to them contain variation we do not have in the PRG. Secondly, we need to generate new paths for these regions and add them back into the PRG for consideration when remapping.

Finding candidate regions

We define a candidate region, r, as an interval within a local graph where coverage on the maximum likelihood path is less than a given threshold, c, for more than l consecutive positions. For a given read that has a mapping to r, we define s_r to be the subsequence of the read mapping to r. We define the pileup P_r as the set of all $s_r \in r$.

Enumerating paths through candidate regions

For $r \in R$, where R is the set of all candidate regions, we construct a de Bruijn graph G_r from P_r. A_L and A_R are defined as k-mers to the left and right of r in the local graph. They are anchors to allow re-insertion of new sequences found by *de novo* discovery into the local graph. If $A_L \notin G_r \vee A_R \notin G_r$ then we abandon *de novo* discovery for r.
T_r is the spanning tree obtained by performing depth-first search (DFS) on G_r from node A_L. p_r is defined as a path, from the root node A_L of T_r and ending at node A_R, which fulfills the following two conditions:

- p_r is shorter than the maximum allowed path length.

- No more than k nodes along p_r have coverage $< (n_r \times 0.1) \times e_r$, where n_r is the a counter, described below, and e_r is the expected k-mer coverage for r.

V_r is the set of all p_r. If $|V_r|$ is greater than a predefined threshold, n_r is incremented by 1 and V_r is repopulated. If $n_r \times 0.1 = 1.0$ then *de novo* discovery is abandoned for r.

Pruning the path-space in a candidate region

As `pandora` operates on both accurate and error-prone sequencing reads, the number of valid paths in G_r can be very large. In testing, this results in run-times beyond seven days (the longest any attempt was allowed to run). The increased run-time is due to cycles that can occur in G_r and exploring paths that will never reach our required end anchor A_R. In order to reduce the path-space within G_r we prune paths based on multiple criteria. Critically, this pruning happens at each step of the graph walk (path-building).
In addition to T_r, obtained by performing DFS on G_r, we produce a distance map D_r that results from running reversed breadth-first search (BFS) on G_r, beginning from node A_R. We say reversed BFS as we explore the predecessors of each node, rather than the successors. D_r is implemented as a binary search tree where each node in the tree represents a k-mer in G_r that is reachable from A_R via reversed BFS. Each node additionally has an integer attached to it that describes the shortest path from that node to A_R.
We can use D_r to prune the path-space by requiring, for each node, $n \in p_r$: is $n \in D_r$ **and** can A_R be reached from n in a minimum of i nodes, where i is defined as the maximum allowed path length minus the number of nodes walked to reach n. If one of these conditions is not met, we abandon p_r. The advantage of this pruning process is that we never explore paths that will not reach our required end point and when caught in a cycle, we will abandon the path once we have made too many iterations around the cycle.

I programmed the above methods in C++ and added them into the code base for `pandora`. They constitute 1325 lines of source code and 3486 lines of test code. I had help with the implementation of multi-threading the *de novo* component and the BFS pruning from Leandro Ishi. The overall workflow for `pandora` with *de novo* variant discovery enabled is illustrated in Figure 4.

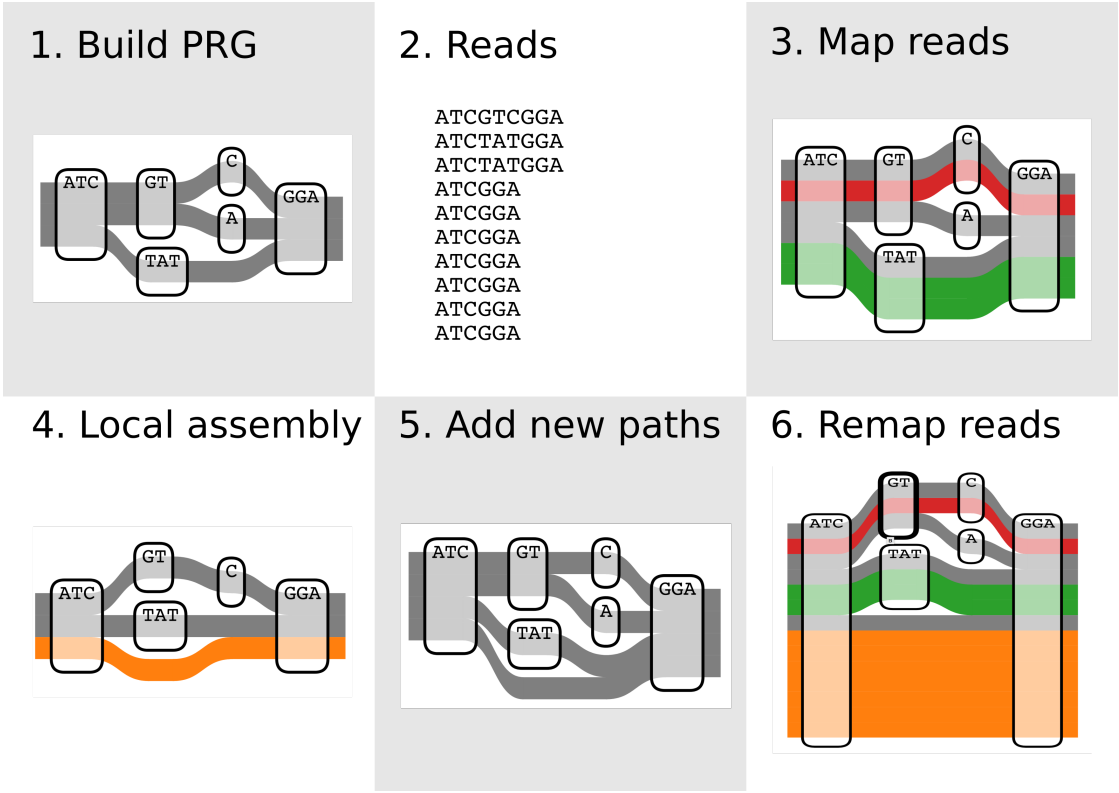

Figure 4: *Schematic of the* `pandora` *method. Note Steps 4-6 are those developed and added in the scope of chapter 1. 1) A Population Reference Graph (PRG) built from known variants or previous samples. 2) Sequencing reads to try and map on to the PRG. 3) Reads mapped on to the PRG (where possible) using minimising k-mers. Coloured segments show paths where reads map. 4) Local assembly of the reads using a de Bruijn graph due to lower than expected coverage on a section of the PRG (many unmapped reads). A new path is discovered (orange). 5) The new path is added into the original PRG. 6) Reads are remapped to the new PRG with many reads mapping to the newly included path (orange).*

Evaluation

Simulated data

The first step in evaluating the effect of adding *de novo* variant calling to `pandora` is with a simulated dataset. What we aim to show here is that the addition of *de novo* discovery allows `pandora` to improve its probability of variant detection (recall). The first step was randomly selecting 100 gene MSAs from a pool of 29702 obtained for *Escherichia coli* from the panX database [8]. Next, a PRG is constructed for each gene MSA, and a random path through each is selected using `pandora`. Each PRG's random path is then concatenated together to form a single "genome" sequence. We subsequently add SNPs to the simulated genome at a specified rate of SNPs per-gene using `snp-mutator` [20]. We then simulated Nanopore reads from this mutated genome using `nanosim-h` [21,22]. `pandora map` is then run, using the original PRG as input, along with the reads simulated from the mutated genome. With this approach, we know the exact SNPs we hope to find. After running the `map` routine, we are left with a collection of candidate paths produced by the *de novo* component. We then added these back into the PRG and iterated the `map` routine again, this time without *de novo* discovery enabled. Re-adding variants into the graph has the effect of making all of the *de novo* variants visible to genotyping. The result is a VCF file that (hopefully) contains the variants we introduced in the beginning. To compare the performance of `pandora` before and after *de novo* is added, we can compare the VCF produced from the first round of `map` with the last VCF.

To avoid error-prone conversion of linear coordinates into graph coordinates the evaluation of whether the variants called by `pandora` are correct was undertaken in a slightly more convoluted manner. We define a probe-set P as a collection of probes, p, where p represents an entry, e, in a VCF file, V. For each $e \in V$, p is constructed by the concatenation of l_w, e_c, and r_w (in that order), where

e_c is the called variant of e, and l_w and r_w are the sequences, of maximum length w, in the VCF reference to the left and right, respectively, of e_c.

A truth probe-set, P_t, was constructed from the VCF of variants added to the simulated genome and a query probe-set, P_q, from the variants called by `pandora`. We then mapped all probes from P_t to P_q using `bwa mem` [23]. We then classify each mapped probe as a false positive or true positive and calculate precision and recall for the pre-*de novo* and post-*de novo* VCF files from `pandora`. As expected, Figure 5 shows that without *de novo* variant discovery, we are unable to find almost all introduced variants. In future work, we plan to add indels to the simulations.

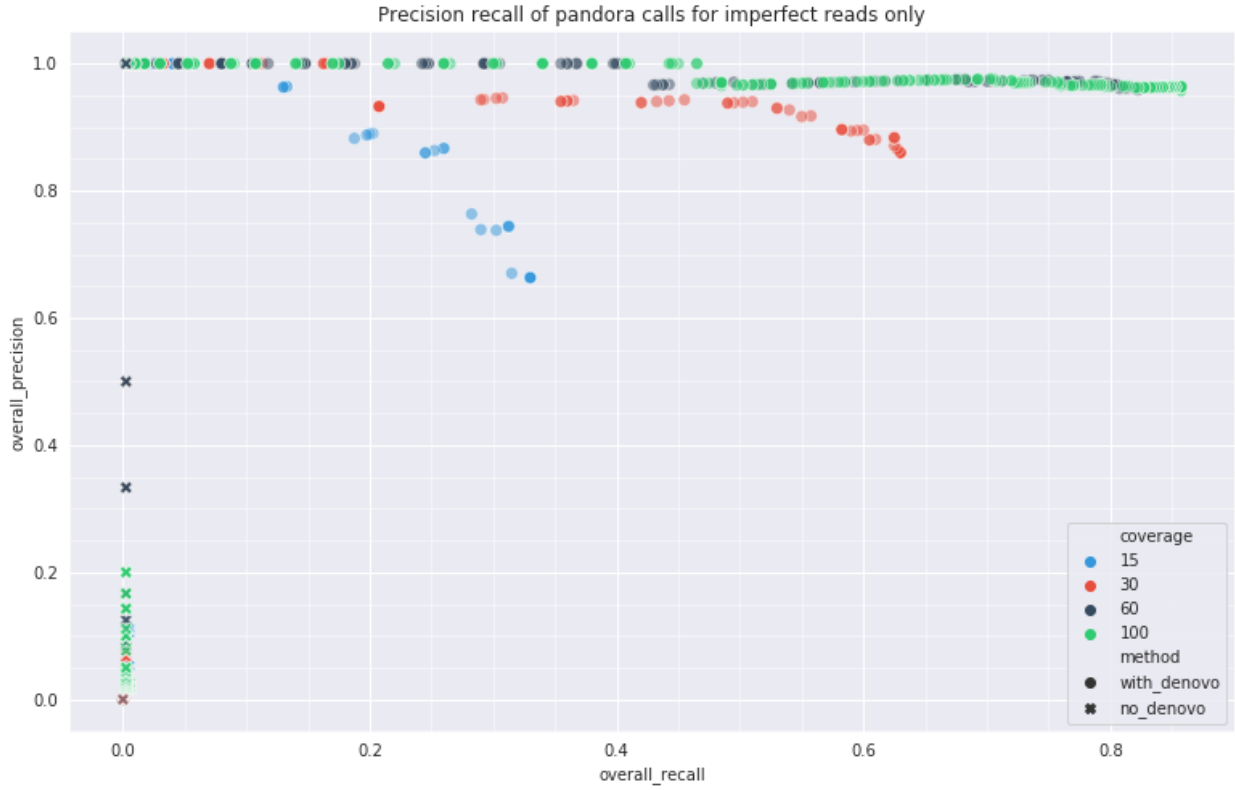

Figure 5: *Precision-recall curve for `pandora` simulations with (circles) and without (crosses) de novo variant discovery. This plot is measuring the proportion of SNPs we introduced into our simulated genome we correctly found (recall) on the X-axis. On the Y-axis is the proportion of calls we made which were correct (precision). Colours represent different coverage depth. Each point represents a different genotype confidence threshold.*

Empirical data - multi-sample comparison

This section will be the major focus for the evaluation of both *de novo and* `pandora` and will constitute the results section of the `pandora` paper (near completion). We aim to show, with the *compare* routine, the power of using a PRG instead of a single reference along with `pandora`'s performance compared to `nanopolish` and `medaka` (Nanopore) and `snippy` (Illumina) [24]. We compare 20 samples, from across the *E. coli* phylogeny. As other variant callers do not use a PRG we carefully selected 24 single-reference genomes representing the diversity of the *E. coli* phylogenetic tree as best as possible. Each variant caller was run once for each reference genome with the aim being to show that the more samples in the analysis, the lower the number of variants the caller can find.

The evaluation will again follow a similar method to the Simulated data. The truth for this analysis, however, is a bit more complicated. In the process of developing an evaluation framework, we produced a python package called `varifier` (https://github.com/iqbal-lab-org/varifier). Briefly, for precision, `varifier` creates 'probes' for each variant in the VCF, using the genome the variants we called with-respect-to. It then maps these probes to the truth genome for the sample and determines

the distance between the variant component of the probe and the part of the truth genome it maps to. For recall evaluation, `varifier` collects all differences in the pairwise alignment between the truth and VCF-reference genomes. Probes are created for these differences (based on the truth genome) and they are mapped to an augmented version of the VCF-reference genome, which has had the variants applied to it. The mappings are then evaluated in the same way as for precision.
As `pandora compare` calls variants for each sample with respect to an inferred best approximation sequence for all samples, creating the set of truth variants is slightly different. We perform a pairwise alignment for all pairs of samples and collect all the differences from this alignment. We then deduplicate this panel to ensure variants are not "double counted", meaning core genome variants would have an unbalanced effect on the overall precision and recall. We then follow the same probe-mapping approach from `varifier` with these truth variants.

We filtered `pandora` variants based on the following criteria:

- Depth less than 15x
- Less than 20% of reads are on one strand
- 50% or more of k-mers on the allele have zero coverage

After we have run this analysis, we will also most likely need to investigate applying filters to the VCF file to remove any biases relating to problems like strand bias or coverage.

Whilst I wrote nearly all of the code and associated tests for evaluating the recall for this analysis, a lot of it has since been refactored by Leandro Ishi and by Martin Hunt in the form of `varifier`. Due to the novelty of the method we are proposing, this has taken quite a lot of time and thought. I wrote 1250 lines of code to evaluate the methods, along with 3500 lines of test code to ensure there are no bugs in our evaluation. Additionally, I built the original `snakemake` [25] pipeline of approximately 3500 lines of codes to orchestrate the entire evaluation and simulations (However, much of this has been rewritten by Leandro Ishi).

Outstanding work

The major work still outstanding for this project is the direct integration of *de novo* candidates back into the PRG. The current procedure requires a fairly laborious, multi-step process for adding *de novo* candidates into the PRG, requiring the user to run a separate pipeline from `pandora`. Ultimately this will need to be handled all within the `pandora` program with no intervention from the user.
To measure whether the reference bias effect in Figure 1 is real and significant, we plan to measure it in three species: *E. coli*, *Klebsiella pneumoniae*, and *Streptococcus pneumoniae*. For each of the species, we will take a set of high-quality reference genomes, and perform the pair-wise alignment-based evaluation from the `pandora` multi-sample comparison. This work will aid the message in the introduction to my first chapter. Much of this work is being completed by Leandro Ishi, but it borrows heavily on some of the evaluation code I have written for the multi-sample comparison.
Lastly, there is an ongoing refinement of the *de novo* variant discovery process from the work in Chapters 2 and 3. The analysis in these chapters lean heavily on `pandora`, but for Mtb, which has a very different pan-genome to that of *E. coli* - which was used for most of the development of `pandora`'s methods.

Chapter 2: Applications to *M. tuberculosis* Nanopore variant calling

Public health applications for genome sequencing of Mtb generally focus on three use-cases: species identification, prediction of drug resistance, and clustering of samples for epidemiological purposes. In this chapter, we plan to focus on how the methods developed in `pandora` can be used to improve clustering of samples - generally referred to as "transmission clusters" - while, in the next chapter, we

will address the drug resistance prediction component. The intention is to be able to use Nanopore data for public health. Therefore, this chapter will focus on a head-to-head comparison of Nanopore and Illumina sequencing technologies for classifying transmission clusters Mtb. What we would like to show with the work in this chapter is that, contrary to current dogma, Nanopore sequencing technology has advanced to the point where it can be applied to this use-case to a standard acceptable by public health authorities. The work that will constitute this chapter (and the next) is a collaboration with researchers in Oxford, Birmingham, Madagascar, and South Africa, but I will be performing all of the analysis.

Data

As the work in this chapter (and the next) involves a direct comparison of two sequencing technologies - Illumina and Nanopore - the DNA sequenced by both must be identical so that we can be certain the technology is the only source of differences, if any. Each sample was sequenced on both platforms from the same isolate and DNA extraction. In total, we received 118 samples from Madagascar, 83 from South Africa, and 46 from the National Tuberculosis Reference Lab in Birmingham; giving us a total of 247 samples.
As these samples are not reference isolates, we need to be able to compare both Illumina and Nanopore to a truth. To establish how each platform compares and differs from the truth, we have additionally sequenced 35 of the Malagasy isolates with PacBio and will use the high-quality assemblies for these samples as a baseline comparison for samples without a truth.

Quality Control

The purpose of quality control (QC) is to ensure all samples used in later analysis are of the highest quality. By highest quality we mean all samples have perfectly matched Illumina and Nanopore data, sufficient coverage on both sequencing technologies, no contamination, and no evidence of a mixed Mtb population.

The first step in QC was to exclude samples where the Nanopore and Illumina data were not perfectly matched. There were some instances where isolates had to be resequenced as the Nanopore data had been accidently deleted from the sequencing laboratory's computer. Additionally, some samples did not have any matched Nanopore or Illumina sequencing. In total, we excluded 40 samples at this stage.

The remaining 207 samples were processed through a quality control pipeline (https://github.com/mbhall88/head_to_head_pipeline/tree/master/data/QC). The first step in QC is decontamination of sequencing reads. We used the decontamination database from `clockwork` (https://github.com/iqbal-lab-org/clockwork), which contains a wide range of organisms, including viral, human, Mtb, non-tuberculosis Mycobacterium (NTM), and nasopharyngeal-associated bacterial genomes. Each genome has associated metadata indicating if it is contamination or not. Reads were mapped to the database using `bwa mem` [23] (Illumina) and `minimap2` (Nanopore) [26]. The resulting alignment was used to quantify the proportion of reads considered contamination, unmapped, and wanted. A read is considered wanted if it has any mapping to a non-contamination genome in the database and is output to a final decontaminated fastq file. All other mapped reads are considered contamination.

All decontaminated fastq files were subsampled to a depth of 60x (Illumina) and 150x (Nanopore) using `rasusa` [27]. The reason for subsampling is to limit unnecessarily large read sets that can drastically slow down later steps in the analysis process without significant advantage.

The last step in the QC pipeline is to assign lineages for each sample. A panel of lineage-defining SNPs [28,29,30] was used in conjunction with a sample's Illumina VCF from the Baseline variant analysis for the lineage assignment. At each lineage-defining position in the sample's VCF we determine if the called allele is the same as the panel allele. If it is, we add the full lineage that allele defines (e.g. 4.1.1) to a list of called lineages. For this analysis, if more than one heterozygous call was made at lineage-defining positions, we abandon lineage assignment for that sample. After classifying all of a sample's lineage-defining positions we then produce a lineage assignment based on the list of called lineages. The most recent common ancestor of all the called lineages is used as the lineage assignment. For example, if the called lineages were `[4, 4.2.3, 4.2.5]` the lineage assignment would be 4.2. If there is more than one called lineage from a different major lineage group, a mixed lineage assignment is given. For example `[4, 4.2.3, 4.2.5, 3.2]` would still be called lineage 4.2, however, `[4, 4.2.3, 4.2.5, 3.2, 3.1]` would be called mixed.

In the end, we chose to exclude samples from further analysis if they met any of the following criteria:

- Illumina coverage below 20x
- Nanopore coverage below 30x
- Evidence of mixed infection - i.e. mixed lineage classification
- Unknown lineage assignment - no valid SNPs at lineage-defining sites

This filtering criteria led to a further 57 samples being excluded; leaving us with a total of 150 samples to use for the remainder of this work.

In addition to the QC of the Illumina/Nanopore data, we sadly had to exclude 26/35 PacBio sequencing datasets due to mismatched Illumina/Nanopore data or PacBio coverage lower than 20x.

Genetic clustering of samples

Although there is scientific interest in the question of identifying transmission chains from genetic data, all the actionable public health information exists in the identification of transmission clusters [31,32].

The first step towards clustering a set of genomes is determining a distance matrix. Typically, this is done either by feeding aligned genomes into a phylogenetic tree-building tool, or more coarsely, by merely counting SNP differences and clustering based on these [31,32,33]. For the majority of bacteria, there is a necessary step of identifying recombination tracts - which will contain a high density of SNPs - and removing them. Removal of these SNPs is necessary as they will have arrived at a different rate to the putative molecular clock and will artefactually extend branch lengths on the phylogenetic tree [31,32]. In the case of Mtb, however, there is virtually no recombination [34,35,36], so this step in not required.

There are three main issues we need to address or keep in mind for the clustering component of this chapter:

- When running a variant caller on a single sample, typically the tool only makes a non-reference call when it finds a definite difference from the reference. Therefore, it is impossible to tell the difference between a reference-matching site and one in which there is genotyping uncertainty. One solution is to make a de-duplicated list of variants found in all samples and genotype the samples at those positions. Solving this reference-bias issue is the strength of the `compare` routine within `pandora`. It tackles the problem in this manner while also handling issues of overlapping variants in the list of all-sample-variants.
- What distance measure do we use? Do we exclude positions where any sample has missing data? Missing data of this type could be due to low coverage, but it could also be because a section of the

genome is not present. We will extensively cover this issue for *E. coli* in Chapter 1. The pan-genome of Mtb is small, but it does exist nonetheless [36,37]. There is also a secondary issue that if certain regions have clustered genetic variation in some lineages of Mtb then reads will not map well to the reference. Both of the above issues are handled by `pandora` due to its use of a PRG, but we still need to make a choice about missing data.
- How do we handle sites where there is evidence of heterozygosity - i.e. a mixed sample?

For this chapter, we define genetic distance to be the sum of genetic discordances, where missing data and heterozygosity do not cause discordance and study the clustering this definition generates.

Baseline variant analysis

The truth set of variants for the Illumina data in this chapter come from running the Public Health England pipeline, COMPASS [38]. This pipeline will act as a guide to inform us about whether the results from the Nanopore data are comparable with those being used in real public health settings. COMPASS effectively uses `samtools` [39] to call variants and then applies a series of complex variant filtering. As a baseline for the Nanopore data, we use `bcftools` [40], with some filtering of variants to remove low-quality calls and with a mask to avoid repetitive and structurally variable regions of the Mtb genome.

Nanopore SNP concordance with Illumina

Whilst the Illumina SNP calls from COMPASS are filtered as part of the pipeline, we had to settle on filters for the Nanopore SNP calls from `bcftools`. We used the methodology from the section Comparing Illumina and Nanopore SNPs to truth assemblies to refine the filters in an iterative process. In the end we filter all SNPs with quality (QUAL column in VCF) below 60, a read position bias less than 0.05, a segregation-based metric above -0.5, or a variant distance bias below 0.002.

To assess how well the Nanopore SNPs agree with Illumina we first look at SNP concordance. Two metrics of interest here are the call rate - what proportion of COMPASS alternate alleles does `bcftools` make a reference/alternate call - and the concordance - what proportion of COMPASS alternate alleles does `bcftools` genotype agree with. Figure 6 shows that concordance is very high between the two technologies, with nearly all samples having a concordance greater than 99.5%. Call rate is a little lower than this, with the majority of samples being above 97%.

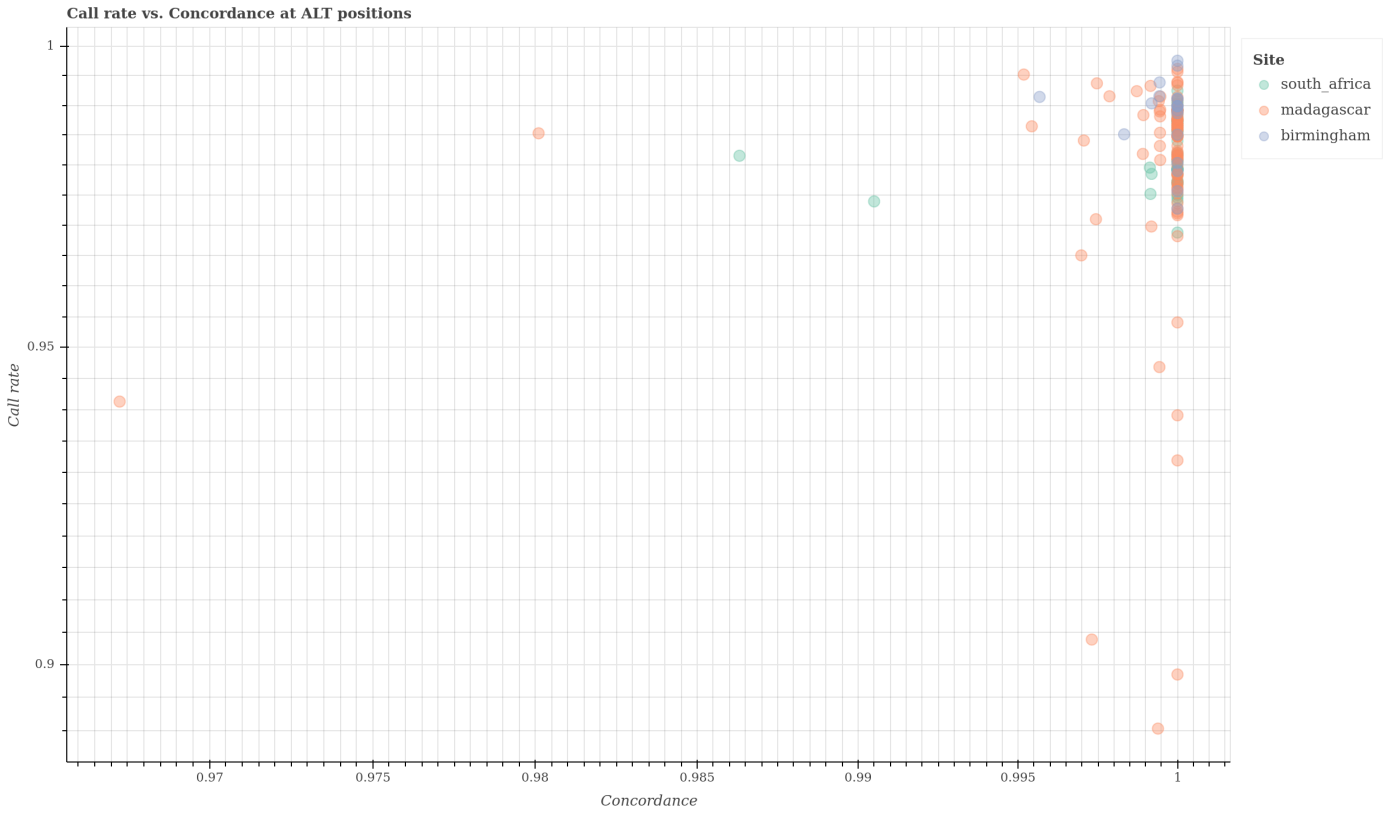

Figure 6: *Call rate (Y-axis) and concordance (X-axis) of* `bcftools` *SNP calls to Illumina COMPASS calls. Call rate is what proportion of COMPASS alternate alleles does* `bcftools` *make a reference/alternate call. Concordance is what proportion of COMPASS alternate alleles does* `bcftools` *genotype agree with. Each point represents a sample, with samples coloured by the site the data came from.*

As transmission clusters are ultimately defined based on a SNP distance matrix, it is important to understand how such matrices differ between Illumina and Nanopore variant calls. As we saw in Figure 6 that the Nanopore SNPs are reasonably concordant with Illumina, but how does this relate to distances between samples? To investigate this a consensus sequence was generated from the filtered VCFs by replacing reference positions with called alternate bases. Any positions with a null genotype or that failed the filtering we masked by replacing the reference positions with an N. Positions which do not appear in the VCF (i.e. no reads mapped to this region) were also masked, as were positions in a previously-defined genome mask of repetitve regions [41]. We then generate a pairwise SNP distance matrix for each sequencing technology from their respective consensus genomes using `snp-dists` [42].
Figure 7 shows the relationship of these distances between pairs of samples based on the sequencing technology used. While the relationship across all samples of all distances is interesting, in the context of defining transmission clusters, it is slightly misleading. Transmission clusters by grouping together samples that are within a certain number of SNPs. The threshold used for this grouping is generally in the order of tens-of-SNPs [31] so it makes more sense to look at the distance relationship for samples that are closer to each other. In Figure 8, we zoom in on the bottom left of Figure 7, to samples within an Illumina SNP distance of 100. It shows that, at this scale, the relationship between Illumina- and Nanopore-defined SNP distance is much closer. The correlation between the two can be quantified by the linear equation $y = 0.93x + 0.84$, where y is the predicted Nanopore distance between two samples, given the Illumina distance x. We can use this equation as a way of translating transmission cluster SNP thresholds for Illumina data to Nanopore. For instance, if clusters are defined as samples within 12 SNPs of each other, we can use this as x and define our Nanopore transmission clusters as $y = 0.93 \times 12\backslash + 0.84 = 12.0$. So at a threshold of 12 SNPs, the Nanopore threshold would be the same as Illumina.

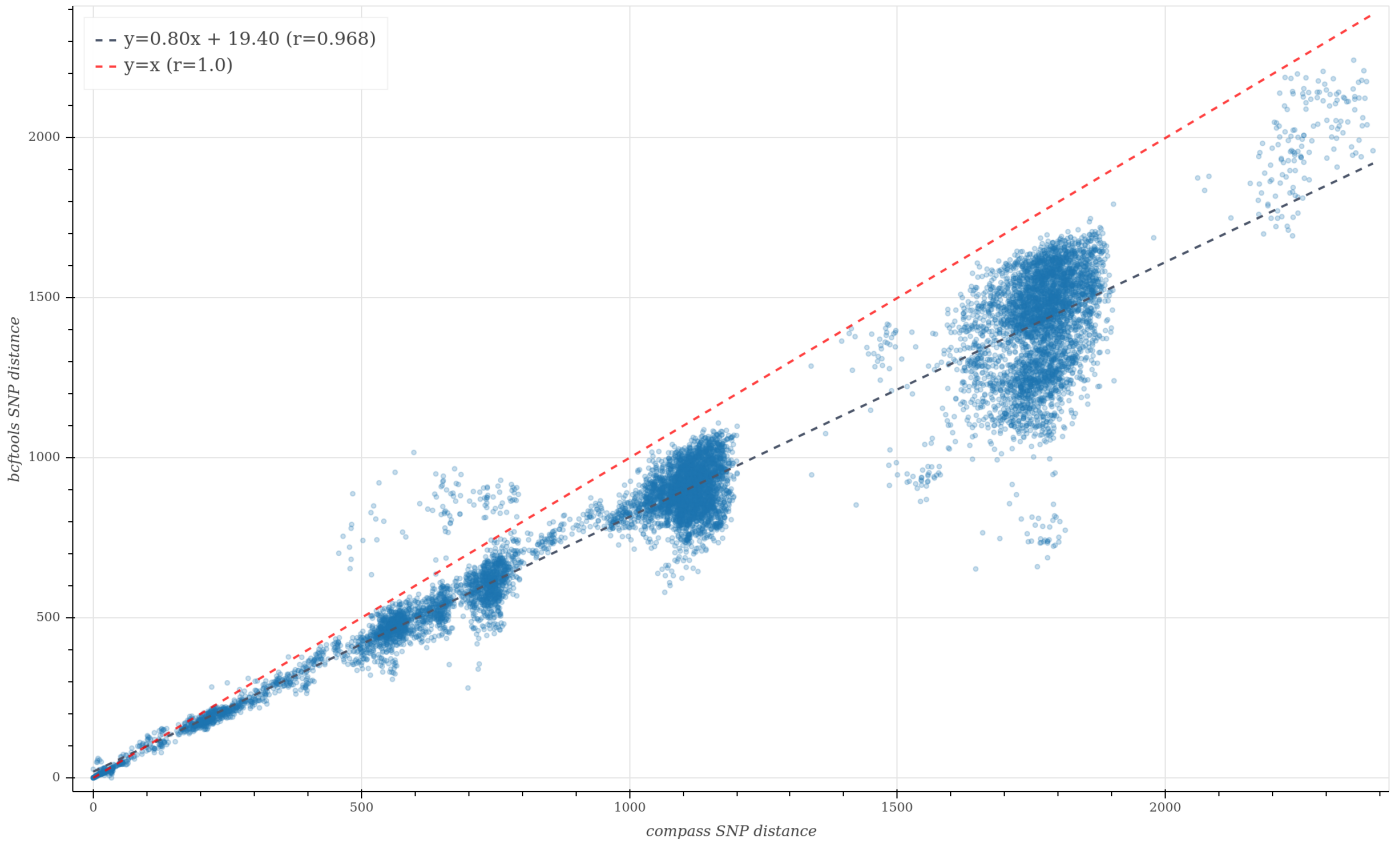

Figure 7: *Relationship between pairwise SNP distance for Illumina (COMPASS; X-axis) and Nanopore (`bcftools`; Y-axis). Each point represents a pair of samples. The red diagonal line is the identity line, which is where the points should lie if the distance between samples is the same for each technology. The black line shows the line of best fit for the data. The legend also shows the equations for these lines, along with their correlation coefficient (r).*

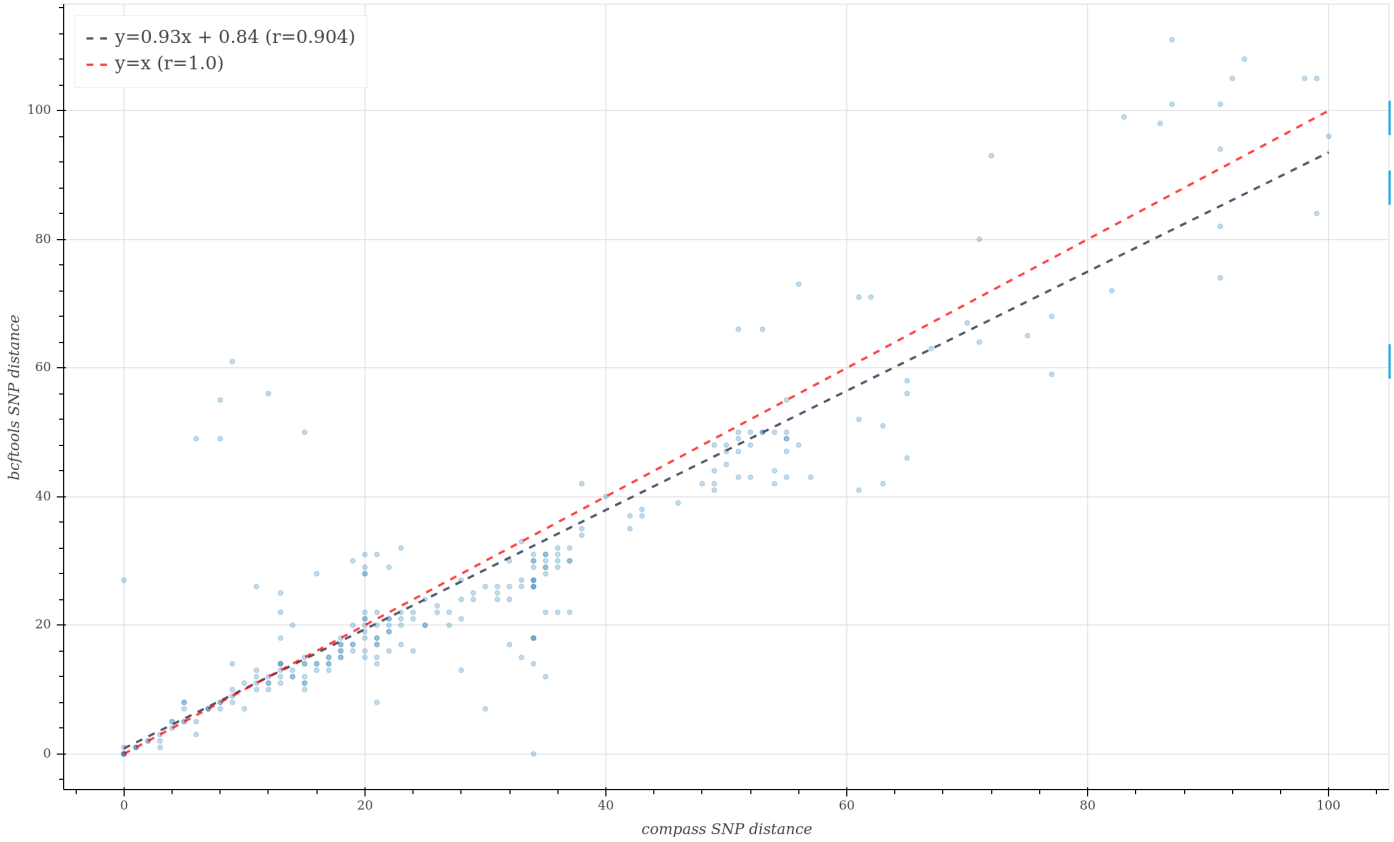

Figure 8: *Relationship between pairwise SNP distance for Illumina (COMPASS; X-axis) and Nanopore (`bcftools`; Y-axis) for samples within 100 SNPs of each other (based on Illumina distance). Each point represents a pair of samples. The red diagonal line is the identity line, which is where the points should lie if the distance between samples is the same for each technology. The black line shows the line of best fit for the data. The legend also shows the equations for these lines, along with their correlation coefficient (r).*

Comparing Illumina and Nanopore SNPs to truth assemblies

The analyses so far have treated the Illumina SNPs as a kind of "truth". In order to get a sense of how "correct" the SNP calls are for each technology we need to compare then to a "truth". For the nine samples with PacBio CCS data that passed QC, we generated assemblies (using only the CCS reads) with `flye` [43]. We masked any positions in the assembly where mapped Illumina reads did not have more than 90% agreement with the assembly, or had less than 10 reads. One sample was excluded due to the detection of other species contigs within the assembly. We then used `varifier` to assess the precision and recall of the SNP calls for the eight samples with high quality assemblies. Figure 9 shows that ...

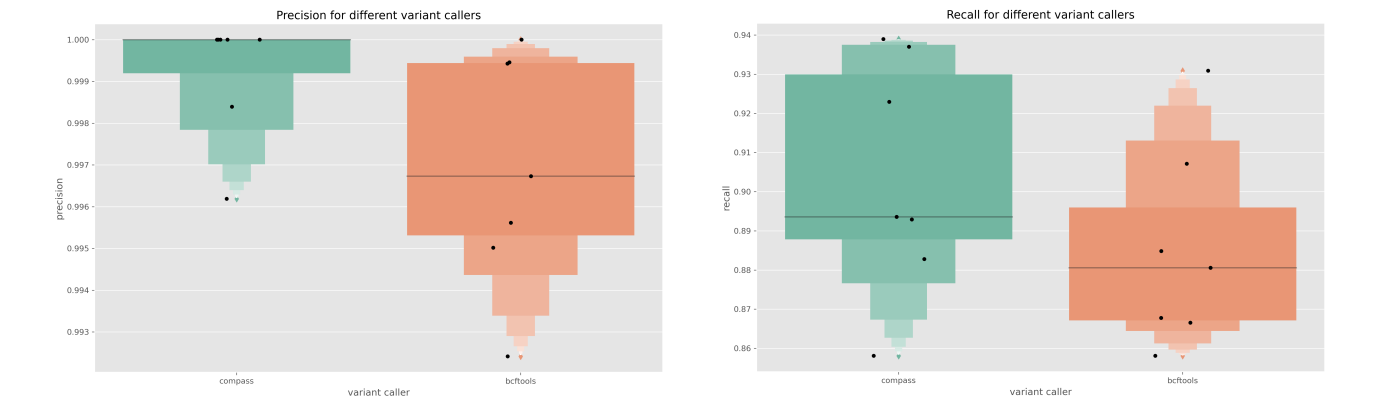

Figure 9: *Evaluation of the Illumina (COMPASS; green) and Nanopore (`bcftools`; orange) SNP calls to the PacBio CCS assemblies for eight samples using `varifier`. The left plot shows precision and the right is recall. Each point is one of the eight samples.*

Per-sample variant calls with `pandora`

The aim of this section will be to try varying degrees of PRG complexity for Mtb sample analysis. At this stage, we have two varieties in mind:

- Sparse PRG - H37Rv and all variants from a random selection of 100 samples from each lineage in the CRyPTIC (Comprehensive Resistance Prediction for Tuberculosis: an International Consortium) dataset [5].
- Dense PRG - The same as Sparse PRG, but 2,000 samples from each lineage.

In all of the above PRGs, we will apply the same mask from the baseline analysis and divide the genome into genes and intergenic regions, with a local PRG for each.

For each of the PRGs, we plan to perform the following analysis. Quantify the number of SNPs and indels `pandora` calls per-sample and see how this compares to the baseline and truth. We will report on the concordance rate, which is the proportion of shared sites between `pandora` and the truth that agree. Additionally, we will investigate how the complexity of the PRG effects the call rates and what the cost in computational performance is.

Multi-sample comparison

Multi-sample comparison suffers from two main challenges. First, large chunks of DNA may be present or absent across samples - this is the pan-genome effect. This effect causes significant issues with single-reference approaches, as outlined previously, but `pandora` was developed to address this. Second, when comparing a set of samples, the choice of reference affects how one describes variants. `pandora`, by design, chooses a reference for each loci PRG based on the current dataset,

intending to maximise the succinctness of variant descriptions (see Figure 1). For example, we want to see SNPs as what they are - single-base variants - not as a nested region (as in Figure 1).

As mentioned, we handle these cases implicitly in `pandora`, however, the aim of this chapter and the next, in addition to comparing across technologies, will also be to compare methods for gaining drug resistance and epidemiological clustering information. To be able to compare with other methods though, we will need to be able to compare variants called with respect to a single-reference by other tools, with those from `pandora`.

With that in mind, this section will focus on producing a distance matrix for all samples using the result of the `pandora compare` routine and contrast this to those obtained from the single-reference methods.

Reproducing "truth" Illumina transmission clusters

In this section we will examine how well we can recreate the transmission clusters produced from COMPASS SNP calls with Nanopore data. Ultimately we will conclude with a recommendation of which method to call variants with: `bcftools`, `pandora map` (single-sample), or `pandora compare` (multi-sample). And what SNP threshold is required to ensure clusters are as similar as possible - if it *is* possible to get comparable clusters.

Outstanding work

I am currently producing the `pandora` variant calls. This has required an iterative refinement of the *de novo* variant discovery process - thus improving it's ability to detect clustered variants. Additionally, there is likely to be some software development time required to allow the `compare` routine to deal with samples of varying coverage. This will entail refining some prototyping that has been done already on genotype confidence percentile, which allows for normalising over coverage.
Once the `pandora` variants are in-hand, there just remains the clustering of samples. We don't anticipate any impediments to generating clusters and it *should* be a fairly quick process.

Chapter 3: Applications to improving *M. tuberculosis* drug resistance prediction

The genetic basis for drug resistance in Mtb is only partially understood. For the four first-line drugs, it is possible to detect the majority of resistant strains with high confidence [5], but for second-line, novel, and repurposed drugs, this is much harder.

Previous work from our group has shown that using whole-genome sequencing (WGS) it is possible to create a panel of resistance markers and then successfully use this panel to predict drug resistance from Mtb sequence data [3,4]. This work involved the development of a software program called `mykrobe` to automate this inference and can use either Illumina or Nanopore data [6]. During the previous year, the predictive power of `mykrobe` has expanded and become even more accurate [44] (I played a small role in improving the likelihood calculations). This update of the `mykrobe` panel was mostly due to the recently published work from the CRyPTIC project [5]. CRyPTIC aims to perform drug susceptibility testing and WGS on 40,000 Mtb samples (many MDR) from all over the globe, and combine this with WGS data from another 60,000 samples. The goal of the project is to improve genotypic resistance prediction by expanding our catalogue of resistance mutations.

The proposed work for this chapter is based on the assumption that a large part of the work by this consortium (which our group is a critical part of) will be available. While there has already been a

significant amount of data produced from CRyPTIC, there will be more coming during the remainder of my PhD. Given the collection of SNPs and indels identified as being necessary for resistance to the 14 major drugs tested, we want to show that we can detect them as well with Nanopore data as we can with Illumina.

Limitations of existing methods

Although there are many tools available for predicting drug resistance in Mtb from Illumina WGS, only two support the use of Nanopore data - `mykrobe` and `tb-profiler` [45]. In the recent update to `mykrobe` from our group [44], we have shown that the primary factor determining how well a tool performed was the catalogue of resistance mutations used. However, given the same panel, different tools do not perform identically, and therefore methodology is still important. While some studies have focused specifically on Nanopore data, all used a small sample size (Votintseva *et al.* 2017 n=5, Hunt *et al.* 2019 n=5, `tb-profiler` 2019 n=3). Additionally, `tb-profiler` and `mykrobe` both have limitations with their methods.

`tb-profiler` uses an approach similar to a read pileup and then taking consensus at each site. This means they will never perform particularly well with indel calls as pileups do not work well for Illumina [46], let alone Nanopore. Given that indel calls are critical for some drugs in Mtb such as pyrazinamide [5], this becomes a ceiling for the `tb-profiler` method. On the other hand, `mykrobe` uses a conservative, precise approach of mapping 21-mers to a de Bruijn graph. With an approximate error rate of 10% in Nanopore data, the probability of a correct 21-mer is 11%. Meaning that `mykrobe` requires high depth-of-coverage to get sufficient, correct 21-mers. By comparison, `pandora` maps entire reads to genes - rather than k-mers in isolation as with `mykrobe` - confirms that the read really does overlap the gene, and only then includes the k-mers in the calculation. Theoretically, this allows for the use of a smaller k-mer size and therefore reduces coverage requirements.
A second limitation of both `tb-profiler` and `mykrobe` is that they only genotype with respect to know variants - i.e. they cannot detect novel variants. The CRyPTIC consortium recently introduced a new approach whereby if an unknown mutation is identified in a gene known to be involved in resistance, they refuse to make a call and instead send the sample for phenotyping [5]. On their 10,000 samples, this achieved a specificity and sensitivity for first-line drugs that was acceptable for clinical usage. This method is now in use at Public Health England for all Mtb samples in England. Hunt *et al.* 2019 [44] quantified the cost of the pure-genotyping approach of `mykrobe`, showing that 2.4-4.6% of resistant samples were missed. By introducing *de novo* discovery into `pandora`, I enable us to address this issue, and that is the focus for this chapter.

Drug susceptibility prediction for *M. tuberculosis* using `pandora`

The work in this chapter will aim to predict drug-resistance for Mtb using `pandora` and its new *de novo* component introduced in Chapter 1. The first step in this will be producing a gene-succinct PRG that includes variants from the above-mentioned CRyPTIC work that are known to cause resistance or susceptibility. This PRG will be easy to build as the alleles and probes for these variants-of-interest are already defined for `mykrobe`. I will write a software program (either an extension of `pandora` or a standalone tool) that takes the output of `pandora` used with this PRG and makes predictions about resistance, susceptibility, and whether phenotyping should be performed. As we know whether an allele causes resistance or susceptibility, this prediction will be straightforward to implement.

I plan to validate this approach with the data from Chapter 2, comparing concordance with `mykrobe` for Illumina and Nanopore. In particular, the analysis will focus on the (hopefully) lower coverage required by `pandora` to achieve the same, or better, results as `mykrobe`, and the increased detection power provided by *de novo* variant discovery.

Part B: Training and career development

Publication Strategy

> Briefly outline the publication strategy; set priorities if needed.

List of publications, papers in press, preprints, manuscripts submitted/in preparation to date

Work plan and timeline for thesis submission

> Provide a work plan to be completed prior to thesis writing and a timeline for submission.

List of scientific courses and conferences attended to date and planned for next year

> Please list all events in reverse chronological order.

List of additional training, teaching and other relevant activities to date

> Please list all training and other activities in reverse chronological order.

Career development plan

Please describe your current long-term career aims (i.e. 3-5 years after PhD).

Please comment on the types of position you would like to apply to for after the PhD and your expected application timeline?

> If applying for postdoc positions, which fields/fellowships are you considering? If you are applying for non-academic careers, what type of role? Do you have target companies or organisations in mind?

What do you see as your strengths (2-3 skills)?

What do you see as your areas for improvement1 (2-3 areas)?

What are your career development priorities until the end of your contract?

> Please take into account both the scientific and non-scientific skills you will need to work on a successful and timely completion of the PhD. List the skills you still need to acquire to achieve your longer-term career aims.

What actions will you take to develop these skills?

> This may include training courses, opportunities to practice and develop these skills, or seeking feedback/guidance. N.B. fellows should take 1-2 non-scientific trainings or career development workshops per year, and at least one international conference during their PhD.

Part C: Impact of COVID-19 pandemic

> As outlined in the Procedures for COVID-19 related fellows' contract extensions, students and their GTL should document the impact of the corona crisis on the PhD project. The TAC should assess the approximate project delay and discuss how this could be mitigated over the remaining PhD period. Please use the space below to outline the impact of corona crisis on your project (e.g. lost productivity / project delays).

References

1. **Global tuberculosis report 2019**
   Organisation mondiale de la santé
   (2019)
   ISBN: 9789241565714

2. **Mycobacterial DNA Extraction for Whole-Genome Sequencing from Early Positive Liquid (MGIT) Cultures**
   Antonina A. Votintseva, Louise J. Pankhurst, Luke W. Anson, Marcus R. Morgan, Deborah Gascoyne-Binzi, Timothy M. Walker, T. Phuong Quan, David H. Wyllie, Carlos Del Ojo Elias, Mark Wilcox, … Derrick W. Crook
   Journal of Clinical Microbiology (2015-04) https://doi.org/f65dtt
   DOI: 10.1128/jcm.03073-14 · PMID: 25631807 · PMCID: PMC4365189

3. **Rapid antibiotic-resistance predictions from genome sequence data for Staphylococcus aureus and Mycobacterium tuberculosis**
   Phelim Bradley, N. Claire Gordon, Timothy M. Walker, Laura Dunn, Simon Heys, Bill Huang, Sarah Earle, Louise J. Pankhurst, Luke Anson, Mariateresa de Cesare, … Zamin Iqbal
   Nature Communications (2015-12-21) https://doi.org/f755tg
   DOI: 10.1038/ncomms10063 · PMID: 26686880 · PMCID: PMC4703848

4. **Whole-genome sequencing for prediction of Mycobacterium tuberculosis drug susceptibility and resistance: a retrospective cohort study**
   Timothy M Walker, Thomas A Kohl, Shaheed V Omar, Jessica Hedge, Carlos Del Ojo Elias, Phelim Bradley, Zamin Iqbal, Silke Feuerriegel, Katherine E Niehaus, Daniel J Wilson, … Tim EA Peto
   The Lancet Infectious Diseases (2015-10) https://doi.org/f3jjtq
   DOI: 10.1016/s1473-3099(15)00062-6 · PMID: 26116186 · PMCID: PMC4579482

5. **Prediction of Susceptibility to First-Line Tuberculosis Drugs by DNA Sequencing**
   The CRyPTIC Consortium and the 100,000 Genomes Project
   New England Journal of Medicine (2018-10-11) https://doi.org/d9kj
   DOI: 10.1056/nejmoa1800474 · PMID: 30280646 · PMCID: PMC6121966

6. **Same-Day Diagnostic and Surveillance Data for Tuberculosis via Whole-Genome Sequencing of Direct Respiratory Samples**
   Antonina A. Votintseva, Phelim Bradley, Louise Pankhurst, Carlos del Ojo Elias, Matthew Loose, Kayzad Nilgiriwala, Anirvan Chatterjee, E. Grace Smith, Nicolas Sanderson, Timothy M. Walker, … Zamin Iqbal
   Journal of Clinical Microbiology (2017-05) https://doi.org/f94vt4
   DOI: 10.1128/jcm.02483-16 · PMID: 28275074 · PMCID: PMC5405248

7. **Why prokaryotes have pangenomes**
   James O. McInerney, Alan McNally, Mary J. O'Connell
   Nature Microbiology (2017-03-28) https://doi.org/gfw8gq
   DOI: 10.1038/nmicrobiol.2017.40 · PMID: 28350002

8. **panX: pan-genome analysis and exploration**
   Wei Ding, Franz Baumdicker, Richard A Neher
   Nucleic Acids Research (2018-01-09) https://doi.org/gczkbr
   DOI: 10.1093/nar/gkx977 · PMID: 29077859 · PMCID: PMC5758898

9. **BulkVis: a graphical viewer for Oxford nanopore bulk FAST5 files**
   Alexander Payne, Nadine Holmes, Vardhman Rakyan, Matthew Loose
   Bioinformatics (2019-07-01) https://doi.org/gfnkxj
   DOI: 10.1093/bioinformatics/bty841 · PMID: 30462145 · PMCID: PMC6596899

10. **Performance of neural network basecalling tools for Oxford Nanopore sequencing**
    Ryan R. Wick, Louise M. Judd, Kathryn E. Holt

Genome Biology (2019-06-24) https://doi.org/gf4jwm
DOI: 10.1186/s13059-019-1727-y · PMID: 31234903 · PMCID: PMC6591954

11. **Real-time, portable genome sequencing for Ebola surveillance**
Joshua Quick, Nicholas J. Loman, Sophie Duraffour, Jared T. Simpson, Ettore Severi, Lauren Cowley, Joseph Akoi Bore, Raymond Koundouno, Gytis Dudas, Amy Mikhail, … Miles W. Carroll
Nature (2016-02-03) https://doi.org/f88652
DOI: 10.1038/nature16996 · PMID: 26840485 · PMCID: PMC4817224

12. **Exploring the limit of using a deep neural network on pileup data for germline variant calling**
Ruibang Luo, Chak-Lim Wong, Yat-Sing Wong, Chi-Ian Tang, Chi-Man Liu, Chi-Ming Leung, Tak-Wah Lam
Nature Machine Intelligence (2020-04-06) https://doi.org/d9kq
DOI: 10.1038/s42256-020-0167-4

13. **nanoporetech/medaka**
GitHub
https://github.com/nanoporetech/medaka

14. **High precision *Neisseria gonorrhoeae* variant and antimicrobial resistance calling from metagenomic Nanopore sequencing**
Nicholas D. Sanderson, Jeremy Swann, Leanne Barker, James Kavanagh, Sarah Hoosdally, Derrick Crook, Teresa L. Street, David W. Eyre, The GonFast Investigators Group
Genome Research (2020-09) https://doi.org/ghcbjr
DOI: 10.1101/gr.262865.120 · PMID: 32873606

15. **Reducing storage requirements for biological sequence comparison**
M. Roberts, W. Hayes, B. R. Hunt, S. M. Mount, J. A. Yorke
Bioinformatics (2004-07-15) https://doi.org/dkhs8w
DOI: 10.1093/bioinformatics/bth408 · PMID: 15256412

16. **Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences**
Heng Li
Bioinformatics (2016-07-15) https://doi.org/f8zxc3
DOI: 10.1093/bioinformatics/btw152 · PMID: 27153593 · PMCID: PMC4937194

17. **A Natural Encoding of Genetic Variation in a Burrows-Wheeler Transform to Enable Mapping and Genome Inference**
Sorina Maciuca, Carlos del Ojo Elias, Gil McVean, Zamin Iqbal
Lecture Notes in Computer Science (2016) https://doi.org/d9ks
DOI: 10.1007/978-3-319-43681-4_18

18. **Improved genome inference in the MHC using a population reference graph**
Alexander Dilthey, Charles Cox, Zamin Iqbal, Matthew R Nelson, Gil McVean
Nature Genetics (2015-04-27) https://doi.org/f7dt83
DOI: 10.1038/ng.3257 · PMID: 25915597 · PMCID: PMC4449272

19. **Genetic variation: molecular mechanisms and impact on microbial evolution**
Werner Arber
FEMS Microbiology Reviews (2000-01) https://doi.org/b4x7zf
DOI: 10.1111/j.1574-6976.2000.tb00529.x · PMID: 10640595

20. **CFSAN SNP Pipeline: an automated method for constructing SNP matrices from next-generation sequence data**
Steve Davis, James B. Pettengill, Yan Luo, Justin Payne, Al Shpuntoff, Hugh Rand, Errol Strain
PeerJ Computer Science (2015-08-26) https://doi.org/d9np
DOI: 10.7717/peerj-cs.20

21. **NanoSim: nanopore sequence read simulator based on statistical characterization**
Chen Yang, Justin Chu, René L Warren, Inanç Birol
GigaScience (2017-04) https://doi.org/gbj69k
DOI: 10.1093/gigascience/gix010 · PMID: 28327957 · PMCID: PMC5530317

22. **Karel-Brinda/Nanosim-H: Nanosim-H 1.1.0.4**
Karel Břinda, Chen Yang, Justin Chu, Jasper Linthorst, Wiktor Franus
Zenodo (2018-08-07) https://doi.org/d9nq
DOI: 10.5281/zenodo.1341250

23. **Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM**
Heng Li
arXiv (2013-05-28) https://arxiv.org/abs/1303.3997

24. **tseemann/snippy**
GitHub
https://github.com/tseemann/snippy

25. **Snakemake–a scalable bioinformatics workflow engine**
J. Koster, S. Rahmann
Bioinformatics (2012-08-20) https://doi.org/gd2xzq
DOI: 10.1093/bioinformatics/bts480 · PMID: 22908215

26. **Minimap2: pairwise alignment for nucleotide sequences**
Heng Li
Bioinformatics (2018-09-15) https://doi.org/gdhbqt
DOI: 10.1093/bioinformatics/bty191 · PMID: 29750242 · PMCID: PMC6137996

27. **Rasusa: Randomly subsample sequencing reads to a specified coverage**
Michael Hall
Zenodo (2020-03-27) https://doi.org/d9rz
DOI: 10.5281/zenodo.3731394

28. **Evolutionary pathway analysis and unified classification of East Asian lineage of Mycobacterium tuberculosis**
Egor Shitikov, Sergey Kolchenko, Igor Mokrousov, Julia Bespyatykh, Dmitry Ischenko, Elena Ilina, Vadim Govorun
Scientific Reports (2017-08-23) https://doi.org/gbvbxh
DOI: 10.1038/s41598-017-10018-5 · PMID: 28835627 · PMCID: PMC5569047

29. **Multiple Introductions of Mycobacterium tuberculosis Lineage 2–Beijing Into Africa Over Centuries**
Liliana K. Rutaihwa, Fabrizio Menardo, David Stucki, Sebastian M. Gygli, Serej D. Ley, Bijaya Malla, Julia Feldmann, Sonia Borrell, Christian Beisel, Kerren Middelkoop, … Sebastien Gagneux
Frontiers in Ecology and Evolution (2019-04-16) https://doi.org/d9r2
DOI: 10.3389/fevo.2019.00112

30. **Mycobacterium tuberculosis lineage 4 comprises globally distributed and geographically restricted sublineages**
David Stucki, Daniela Brites, Leïla Jeljeli, Mireia Coscolla, Qingyun Liu, Andrej Trauner, Lukas Fenner, Liliana Rutaihwa, Sonia Borrell, Tao Luo, … Sebastien Gagneux
Nature Genetics (2016-10-31) https://doi.org/f9dg9j
DOI: 10.1038/ng.3704 · PMID: 27798628 · PMCID: PMC5238942

31. **Beyond the SNP Threshold: Identifying Outbreak Clusters Using Inferred Transmissions**
James Stimson, Jennifer Gardy, Barun Mathema, Valeriu Crudu, Ted Cohen, Caroline Colijn
Molecular Biology and Evolution (2019-03) https://doi.org/d9r7
DOI: 10.1093/molbev/msy242 · PMID: 30690464 · PMCID: PMC6389316

32. **Interpreting whole genome sequencing for investigating tuberculosis transmission: a systematic review**
Hollie-Ann Hatherell, Caroline Colijn, Helen R. Stagg, Charlotte Jackson, Joanne R. Winter, Ibrahim Abubakar
BMC Medicine (2016-03-23) https://doi.org/f8gsk2
DOI: 10.1186/s12916-016-0566-x · PMID: 27005433 · PMCID: PMC4804562

33. **Whole-genome sequencing to delineate Mycobacterium tuberculosis outbreaks: a retrospective observational study**
Timothy M Walker, Camilla LC Ip, Ruth H Harrell, Jason T Evans, Georgia Kapatai, Martin J Dedicoat, David W Eyre, Daniel J Wilson, Peter M Hawkey, Derrick W Crook, … Tim EA Peto
The Lancet Infectious Diseases (2013-02) https://doi.org/f2g6zn
DOI: 10.1016/s1473-3099(12)70277-3 · PMID: 23158499 · PMCID: PMC3556524

34. **Bottlenecks and broomsticks: the molecular evolution of Mycobacterium bovis**
Noel H. Smith, Stephen V. Gordon, Ricardo de la Rua-Domenech, Richard S. Clifton-Hadley, R. Glyn Hewinson
Nature Reviews Microbiology (2006-09) https://doi.org/fhqqkv
DOI: 10.1038/nrmicro1472 · PMID: 16912712

35. **Evidence for Recombination in Mycobacterium tuberculosis**
Xiaoming Liu, Michaela M. Gutacker, James M. Musser, Yun-Xin Fu
Journal of Bacteriology (2006-12-01) https://doi.org/ftp6r2
DOI: 10.1128/jb.01062-06 · PMID: 16997954 · PMCID: PMC1698211

36. **Genomic analysis of smooth tubercle bacilli provides insights into ancestry and pathoadaptation of Mycobacterium tuberculosis**
Philip Supply, Michael Marceau, Sophie Mangenot, David Roche, Carine Rouanet, Varun Khanna, Laleh Majlessi, Alexis Criscuolo, Julien Tap, Alexandre Pawlik, … Roland Brosch
Nature Genetics (2013-01-06) https://doi.org/f4mrqv
DOI: 10.1038/ng.2517 · PMID: 23291586 · PMCID: PMC3856870

37. **Computational genomics-proteomics and Phylogeny analysis of twenty one mycobacterial genomes (Tuberculosis & non Tuberculosis strains)**
Fathiah Zakham, Othmane Aouane, David Ussery, Abdelaziz Benjouad, Moulay Ennaji
Microbial Informatics and Experimentation (2012) https://doi.org/d9r8
DOI: 10.1186/2042-5783-2-7 · PMID: 22929624 · PMCID: PMC3504576

38. **Towards standardisation: comparison of five whole genome sequencing (WGS) analysis pipelines for detection of epidemiologically linked tuberculosis cases**
Rana Jajou, Thomas A Kohl, Timothy Walker, Anders Norman, Daniela Maria Cirillo, Elisa Tagliani, Stefan Niemann, Albert de Neeling, Troels Lillebaek, Richard M Anthony, Dick van Soolingen
Eurosurveillance (2019-12-12) https://doi.org/d9r9
DOI: 10.2807/1560-7917.es.2019.24.50.1900130 · PMID: 31847944 · PMCID: PMC6918587

39. **The Sequence Alignment/Map format and SAMtools**
H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin, 1000 Genome Project Data Processing Subgroup
Bioinformatics (2009-06-08) https://doi.org/ff6426
DOI: 10.1093/bioinformatics/btp352 · PMID: 19505943 · PMCID: PMC2723002

40. **Improving SNP discovery by base alignment quality**
H. Li
Bioinformatics (2011-02-13) https://doi.org/fw7k5k
DOI: 10.1093/bioinformatics/btr076 · PMID: 21320865 · PMCID: PMC3072548

41. **Assessment of Mycobacterium tuberculosis transmission in Oxfordshire, UK, 2007–12, with whole pathogen genome sequences: an observational study**
Timothy M Walker, Maeve K Lalor, Agnieszka Broda, Luisa Saldana Ortega, Marcus Morgan, Lynne Parker, Sheila Churchill, Karen Bennett, Tanya Golubchik, Adam P Giess, … Christopher P Conlon

The Lancet Respiratory Medicine (2014-04) https://doi.org/f3hxn7
DOI: 10.1016/s2213-2600(14)70027-x · PMID: 24717625 · PMCID: PMC4571080

42. **Source code for snp-dists software**
Torsten Seemann
Zenodo (2018-09-09) https://doi.org/d9zj
DOI: 10.5281/zenodo.1411986

43. **Assembly of long, error-prone reads using repeat graphs**
Mikhail Kolmogorov, Jeffrey Yuan, Yu Lin, Pavel A. Pevzner
Nature Biotechnology (2019-04-01) https://doi.org/gfzbrd
DOI: 10.1038/s41587-019-0072-8 · PMID: 30936562

44. **Antibiotic resistance prediction for Mycobacterium tuberculosis from genome sequence data with Mykrobe**
Martin Hunt, Phelim Bradley, Simon Grandjean Lapierre, Simon Heys, Mark Thomsit, Michael B. Hall, Kerri M. Malone, Penelope Wintringer, Timothy M. Walker, Daniela M. Cirillo, … Zamin Iqbal
Wellcome Open Research (2019-12-02) https://doi.org/ggd835
DOI: 10.12688/wellcomeopenres.15603.1 · PMID: 32055708 · PMCID: PMC7004237

45. **Integrating informatics tools and portable sequencing technology for rapid detection of resistance to anti-tuberculous drugs**
Jody E. Phelan, Denise M. O'Sullivan, Diana Machado, Jorge Ramos, Yaa E. A. Oppong, Susana Campino, Justin O'Grady, Ruth McNerney, Martin L. Hibberd, Miguel Viveiros, … Taane G. Clark
Genome Medicine (2019-06-24) https://doi.org/d949
DOI: 10.1186/s13073-019-0650-x · PMID: 31234910 · PMCID: PMC6591855

46. **High-throughput microbial population genomics using the Cortex variation assembler**
Zamin Iqbal, Isaac Turner, Gil McVean
Bioinformatics (2013-01-15) https://doi.org/d95c
DOI: 10.1093/bioinformatics/bts673 · PMID: 23172865 · PMCID: PMC3546798