GPS PIPELINE

A Portable and Scalable Genomic Analysis Pipeline for *Streptococcus pneumoniae* Surveillance

Harry Hung, PhD

Senior Bioinformatician

Bentley Group

Global Pneumococcal Sequencing Project

wellcome sanger institute

An accessible bioinformatics pipeline!

Aims

- Easy to install and use
- Work (consistently) everywhere

Enabled by

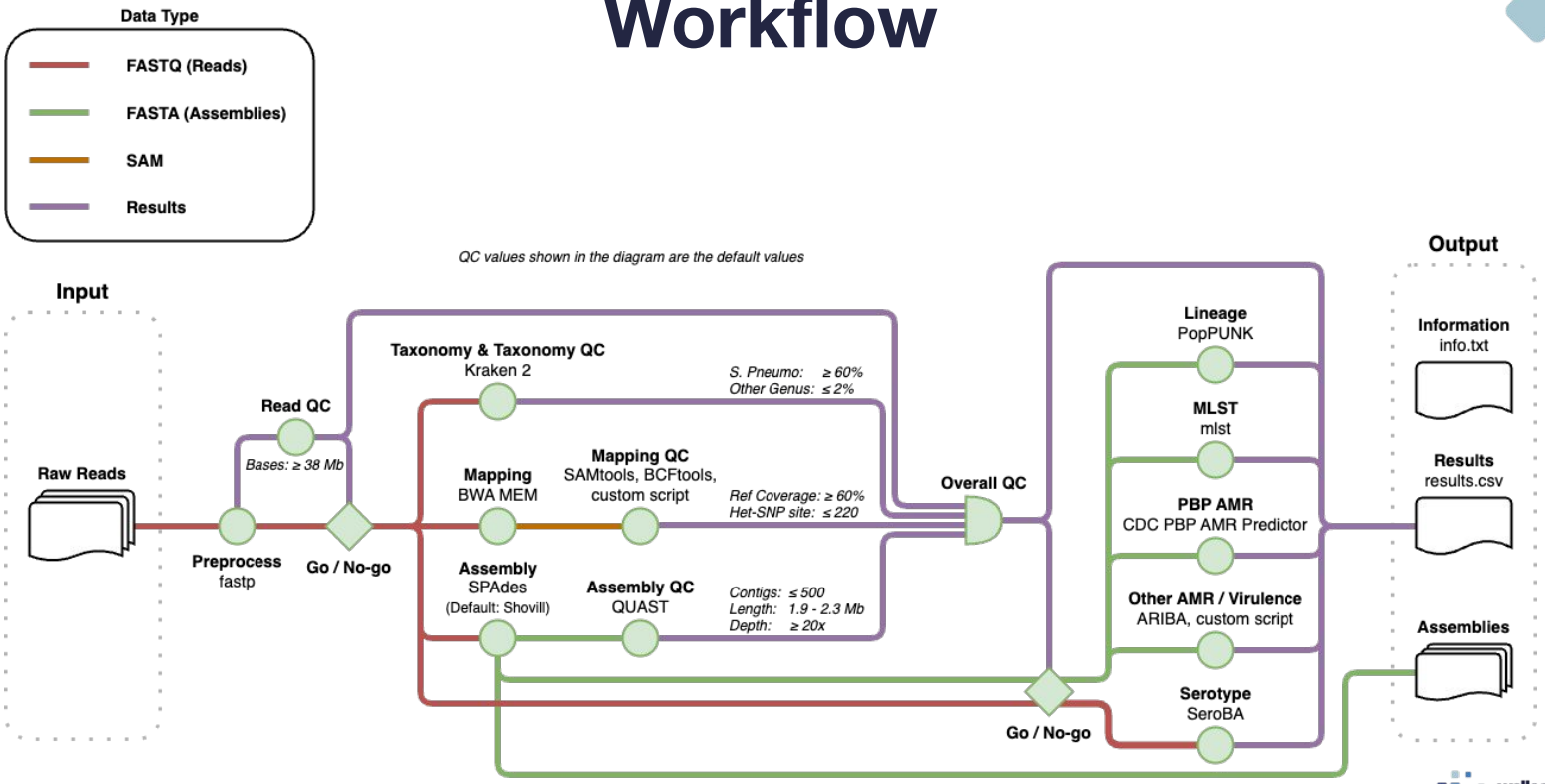
- Workflow system nextflow
- Containerisation docker

Input

Path to Your Directory

(which containing Illumina paired-end short reads from isolates you are interested in)

Workflow


Data Type
FASTQ (Reads)
FASTA (Assemblies)
SAM
Results
QC values shown in the diagram are the default values
Input
Raw Reads
Preprocess
fastp
Read QC
Bases: ≥ 38 Mb
Go / No-go
Taxonomy & Taxonomy QC
Kraken 2
S. Pneumo: ≥ 60%
Other Genus: ≤ 2%
Mapping
BWA MEM
Mapping QC
SAMtools, BCFtools,
custom script
Ref Coverage: ≥ 60%
Het-SNP site: ≤ 220
Assembly
SPAdes
(Default: Shovill)
Assembly QC
QUAST
Contigs: ≤ 500
Length: 1.9 - 2.3 Mb
Depth: ≥ 20x
Overall QC
Go / No-go
Lineage
PopPUNK
MLST
mlst
PBP AMR
CDC PBP AMR Predictor
Other AMR / Virulence
ARIBA, custom script
Serotype
SeroBA
Output
Information
info.txt
Results
results.csv
Assemblies

Results

(in a single .csv file)

QC

- Read
- Assembly
- Mapping
- Taxonomy
- Overall

Read

- Bases

Assembly

- Contigs #
- Length
- Sequencing depth

Mapping

- Reference coverage
- Heterozygous SNP #

Taxonomy

- *S. pneumo* %
- Top non-*Strep* genus
- and its %

Lineage

- GPSC

Serotype

- Serotype

MLST

- Sequence type
- Allele ID
  - aroE
  - gki
  - spi
  - ddl
  - gdh
  - recP
  - xpt

PBP AMR

- Allele ID
  - pbp1a
  - pbp2b
  - pbp2x
- MIC & resistance
  - amoxicillin (AMO)
  - ceftriaxone (CFT)
  - cefotaxime (TAX)
  - cefuroxime (CFX)
  - meropenem (MER)
  - penicillin (PEN)

Other AMR

- MIC, Resistance & determinants
  - Chloramphenicol (CHL)
  - Clindamycin (CLI)
  - Co-Trimoxazole (COT)
  - Doxycycline (DOX)
  - Erythromycin (ERY)
  - ERY and CLI
  - Fluoroquinolones (FQ)
  - Kanamycin (KAN)
  - Levofloxacin (LFX)
  - Rifampin (RIF)
  - Sulfamethoxazole (SMX)
  - Tetracycline (TET)
  - Trimethoprim (TMP)
  - Vancomycin (VAN)

Virulence

- Expression and determinants
  - PILI-1
  - PILI-2

wellcome sanger institute

Requirements

Hardware

- 16GB RAM or above
- 50GB Free Storage (8GB for databases, 13GB for Docker images)

Operating System

- Any POSIX-compatible System
- i.e. Linux, Windows (via WSL2), macOS

Software

- Java or OpenJDK 11+
- Docker or Singularity/Apptainer
- Bash 3.2+

Validation & Benchmark

Validated on 500 random samples from the GPS Database

- 91.8% of samples have same QC assessment and *in silico* typing results
- 6% of samples have improved results
- 1.6% of samples have neutral changes
- 0.6% of samples have regressed results

Runtime

- 1 hr 4 min - 500 samples on Sanger HPC (Farm5)
- 2 hr 48 min - 100 samples on a 16-core Ubuntu-based OpenStack instance

wellcome sanger institute

Setup and Execution

1. Download or git clone it from
   https://github.com/sanger-bentley-group/gps-pipeline

2. (Optional) initialise the pipeline
   `./run_pipeline --init`
   This will download all container images and databases

3. Run the pipeline
   `./run_pipeline --reads /path/to/raw-reads-directory`
   [Only need to run this for all future runs]
   [More options available, e.g. `--output`]

wellcome sanger institute

Demo

Acknowledgement

Bentley Group @ Sanger

Dr. Stephanie W. Lo

Prof.
Stephen D. Bentley

Dr. Narender Kumar

Dr. Victoria Dyster

CGPS

- Corin Yeats

CDC

- Benjamin Metcalf
- Dr. Yuan Li
- Paulina A. Hawkins
- Dr. Lesley McGee

Global Pneumococcal Sequencing Project

wellcome

Bill & Melinda Gates foundation

CDC Centers for Disease Control and Prevention

wellcome sanger institute

https://github.com/
sanger-bentley-group/gps-pipeline

Thanks! Questions?

wellcome sanger institute

wellcome
sanger
institute

Extra Information

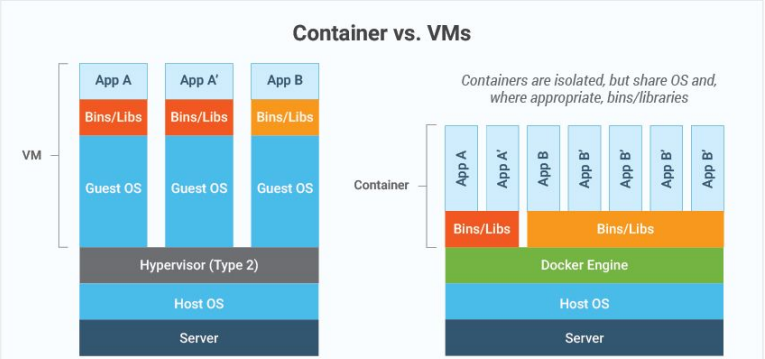
nextflow

Designed for making scalable and reproducible scientific pipelines

● **Simple config**	Installation-free; POSIX-compatible system with Java 11+
● **Scalable**	Run from laptop to server farm with only config changes
● **Parallelism**	All processes are inherently parallel
● **Scripting languages**	Support Bash/Python/Perl scripts directly
● **Reproducibility**	Support Docker and Singularity containers and more
● **Checkpoints**	Resume from last successful executed step

IMHO: Easy to understand syntax, good documentation, good community

- Containerisation platforms
- Packages the program, its dependencies and environment together
- Containerised program should work virtually the same everywhere
- Much more lightweight than VM



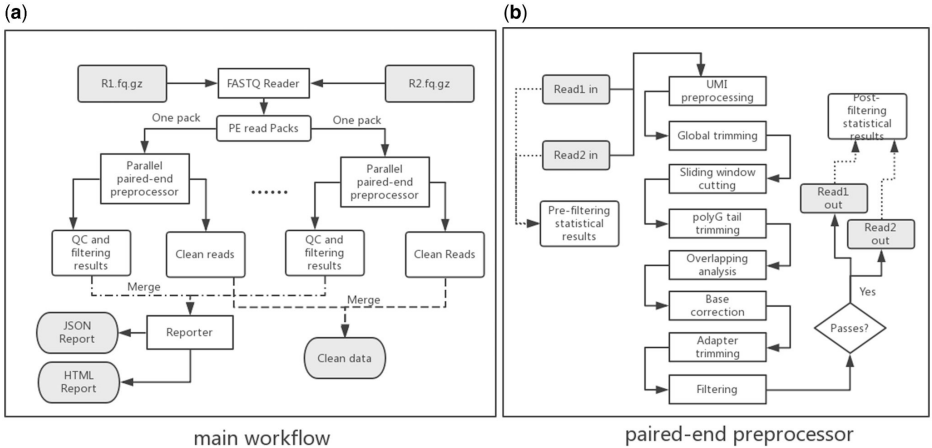
Adapted from: https://www.eginnovations.com/blog/containers-vs-vms/

fastp [preprocessing + reads QC]

Preprocess FASTQ files and check total base count

- High performance due to written in C++ with multithreading
- Automatic adapter trimming (no adapter sequences required)
- Filtering low quality reads
- Tail trimming
- Base correction

Chen, Shifu, et al. "fastp: an ultra-fast all-in-one FASTQ preprocessor." *Bioinformatics* 34.17 (2018): i884-i890.

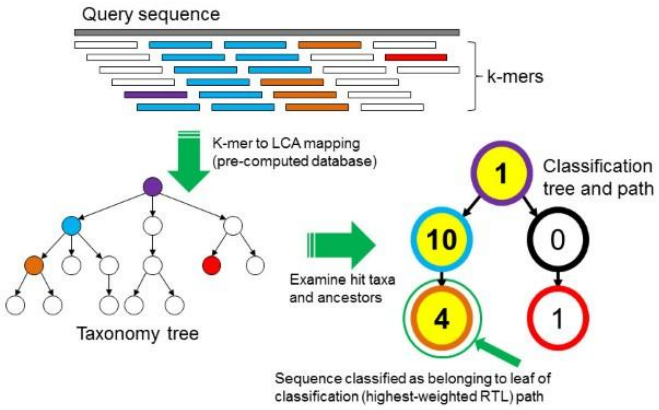


Kraken2 [taxonomy classification + QC]

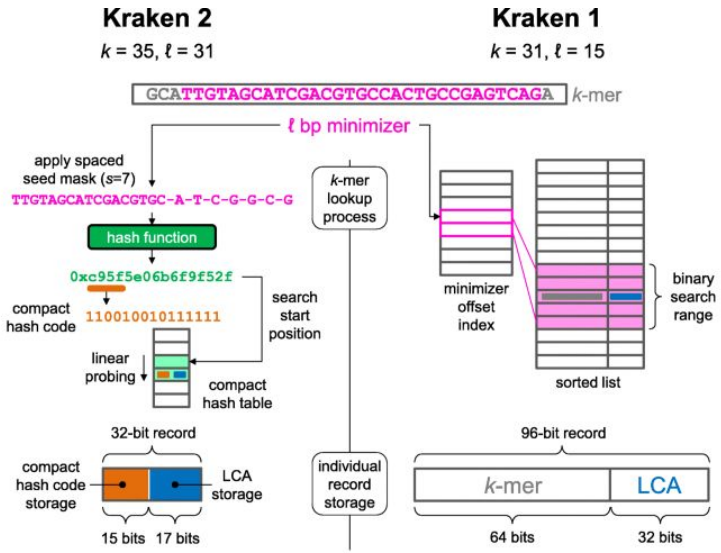
Check the species which the reads belong to

- k-mer (fixed-length DNA subsequence)-based search on database
- matches each k-mer within a query sequence to the lowest common ancestor (LCA) of all genomes containing the given k-mer

Wood, Derrick E., and Steven L. Salzberg. "Kraken: ultrafast metagenomic sequence classification using exact alignments." Genome biology 15.3 (2014): 1-12.



Wood, Derrick E., Jennifer Lu, and Ben Langmead. "Improved metagenomic analysis with Kraken 2." Genome biology 20 (2019): 1-13.



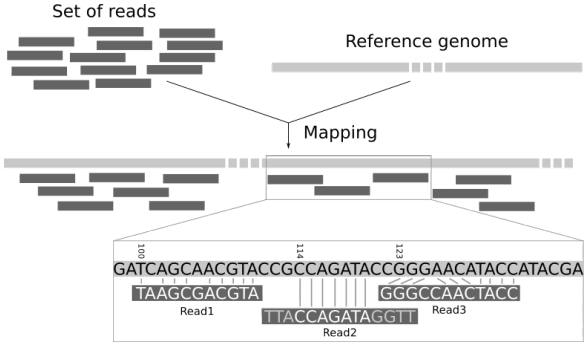
BWA-MEM + SAMtools + BCFtools + custom script [mapping + QC]

Check coverage and purity of isolates

1. BWA-MEM maps reads to reference genome
2. SAMtools calculates coverage and generates mapped+sorted BAM
3. BCFtools calls SNP
4. Python script calculates non-cluster* heterozygous SNP count

*Het-SNP within 50bp (i.e. cluster) might just be side effect of recombination or mapping inaccuracies in the repeated regions

Adapted from: https://gxy.io/GTN:T00237



Adapted from: https://www.htslib.org/doc/samtools-coverage.html

```
samtools coverage -A -w 32 -r chr1:1M-12M input.bam

chr1 (249.25Mbp)
>  24.19% | .                                | Number of reads: 528695
>  21.50% |::                                |     (132000 filtered)
>  18.81% |::                                | Covered bases:   1.07Mbp
>  16.12% |::                              :  | Percent covered: 9.727%
>  13.44% |::  :  .       ::             : :| Mean coverage:   3.5x
>  10.75% |:: ::  :       ::           : : :| Mean baseQ:      34.4
>   8.06% |:::::  :       ::         : : : :| Mean mapQ:       55.8
>   5.37% |::::: ::      :::       : ::::: :|
>   2.69% |::::: :::     :::  ::: :::::::::| Histo bin width: 343.8Kbp
>   0.00% |:::::::::::. :::::::::::::::::::| Histo max bin:   26.873%
        1.00M     4.44M     7.87M       12.00M
```

Shovill + QUAST [assembly / QC]

Assembly the reads and check the assemblies quality

- Shovill uses SPAdes at its core, with modified pre- and post-assembly step to speed up the process with similar results
- QUAST evaluates genome assemblies quality

Adapted from: https://www.x.com/pathogenomenick/status/781147338798669824/

tseemann / shovill

Shovill pipeline

1. Estimate genome size - *KmerStream*
2. Correct errors in reads - *Lighter*
3. Overlap PE reads - *FLASH*
4. Choose k-mer values from reads - *SeqTK*
5. Assemble reads into contigs - *SPAdes*
   a. Multiple k-mer values - *my logic*
6. ~~Scaffold contigs~~
7. Correct mistakes in contigs - *PILON*

Adapted from: https://hoytpr.github.io/bioinformatics-semester/

QUAST

Quality Assessment Tool for Genome Assemblies by CAB

17 October 2019, Thursday, 09:46:13

View in Icarus contig browser

All statistics are based on contigs of size >= 500 bp, unless otherwise noted (e.g., "# contigs (>= 0 bp)" and "Total length (>= 0 bp)" include all contigs).

Suggestion: assemblies abyss21, abyss25, abyss31, soap21, soap31, soap41, velvet21, velvet31, velvet41 contain continuous fragments of N's longer than or equal to 10 bp. You may consider rerunning QUAST using --scaffolds (-s) option!

Aligned to "ref" | 562 643 bp | 1 fragment | 38.49 % G+C

Worst Median Best ☑ Show heatmap

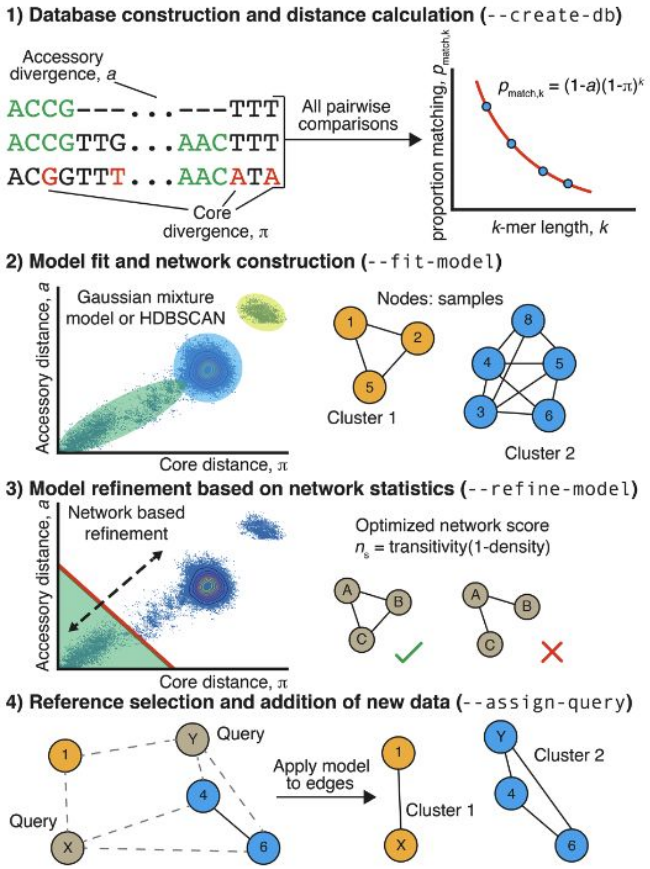
Genome statistics	abyss21	abyss25	abyss31	soap21	soap31	soap41	velvet21	velvet31	velvet41
Genome fraction (%)	98.123	98.059	99.043	97.113	97.769	98.131	97.979	98.22	98.22
Duplication ratio	1.007	1.006	1.011	1.003	1.002	1.002	1.001	1.003	1.003
Largest alignment	136 924	136 998	136 952	172 445	172 681	310 008	183 007	288 933	288 933
Total aligned length	554 260	554 235	563 136	545 725	549 558	551 942	551 377	553 466	553 466
NGA50	80 749	80 495	84 032	136 334	136 423	310 008	100 712	288 933	288 933
LGA50	3	3	3	2	2	1	2	1	1
Misassemblies									
# misassemblies	1	1	1	0	0	0	1	3	3
Misassembled contigs length	91 465	92 472	92 506	0	0	0	249 502	391 243	391 243
Mismatches									
# mismatches per 100 kbp	16.85	13.78	16.15	10.98	18.36	14.49	10.34	12.3	12.3
# indels per 100 kbp	11.41	9.61	8.97	98.95	25.63	21.92	11.97	11.58	11.58
# N's per 100 kbp	365.62	204.43	79.16	752.54	352.23	246.93	179.88	365.91	365.91
Statistics without reference									
# contigs	15	14	15	15	14	12	11	7	7
Largest contig	137 043	136 998	136 952	172 601	172 681	310 008	249 502	391 243	391 243
Total length	556 865	556 674	563 416	549 605	551 623	553 598	552 031	554 231	554 231
Total length (>= 1000 bp)	556 865	556 674	563 416	546 465	549 884	552 772	551 176	554 231	554 231
Total length (>= 10000 bp)	520 714	521 422	536 060	524 752	525 712	527 498	539 282	542 558	542 558

PopPUNK [lineage / GPSC]

Assign GPSC using assemblies

- Clustering genomes based on estimation of core and accessory genome distances between samples
- Clusters are variable-length-k-mer cluster (VLKCs), typically representing distinct strains
- Use sketching (approx. compact summary of genome) to increase efficiency

Lees, John A., et al. "Fast and flexible bacterial genomic epidemiology with PopPUNK." Genome research 29.2 (2019): 304-316.



SeroBA [serotype]

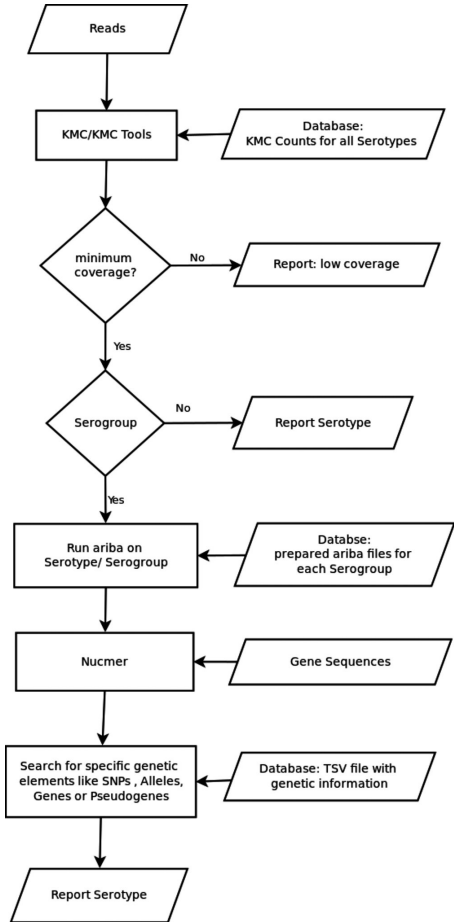
Predicts serotypes using reads

- Identifying the *cps* locus directly from raw whole genome sequencing read data using a k-mer based method

  1. Intersect k-mers of input and k-mers of serotype database to select serotype/group

  2. If not yet determined, check local assembly and alignment of *cps* sequence

Epping, Lennard, et al. "SeroBA: rapid high-throughput serotyping of Streptococcus pneumoniae from whole genome sequence data." Microbial genomics 4.7 (2018): e000186.



mlst [MLST]

Predict MLST using assemblies

- BLASTn contigs against pubMLST typing schemes
- Prediction gets a score based on its scoring system

Adapted from: https://github.com/tseemann/mlst

```
% mlst --legacy --scheme neisseria *.fa
FILE      SCHEME     ST    abcZ  adk  aroE  fumC  gdh  pdhC  pgm
NM003.fa  neisseria  11    2     3    4     3     8    4     6
NM009.fa  neisseria  11149 672   3    4     3     8    4     6
MN043.fa  neisseria  11    2     3    4     3     8    4     6
NM051.fa  neisseria  11    2     3    4     3     8    4     6
NM099.fa  neisseria  1287  2     3    4    17     8    4     6
NM110.fa  neisseria  11    2     3    4     3     8    4     6
```

CDC PBP AMR Predictor [PBP/β-lactam AMR]

Predict resistance for β-lactam antibiotics using a Machine Learning algorithm and Penicillin Binding Protein (PBP) types

1. Assigning an allele code to *pbp1A*, *pbp2B* and *pbp2X* genes
2. Using a machine learning approach to estimate the MICs (minimum inhibitory concentration) for a set of antimicrobials
3. Resistance phenotype is then interpreted using CLSI guidelines

ARIBA + custom script [other AMR & virulence]

Capture known gene presence/absence or mutations that lead to various antimicrobial resistance (AMR) and virulence

1. ARIBA detects known gene presence/absence or mutation that will lead to AMR
2. Python script uses the ARIBA results and custom logic based on published research articles to deduce resistance phenotypes