Bioinformatic pipeline for profiling foodborne bacterial ecology and resistome from short-read metagenomics

Arnaud Bridier, Pierre Lemée

Bioinformatic pipeline for profiling foodborne bacterial ecology and resistome from short-read metagenomics

Running title: Metagenomics pipeline for bacterial ecology and resistance

Pierre Lemée and Arnaud Bridier

Antibiotics, Biocides, Residues and Resistance Unit, Fougères Laboratory, French Agency for Food, Environmental and Occupational Health & Safety (ANSES), 35300 Fougères, France

Abstract:

Next generation sequencing revolutionized food safety management these last years providing access to a huge quantity of valuable data to identify, characterize and monitor bacterial pathogens on the food chain. Shotgun metagenomics emerged as a particularly promising approach as it enables in-depth taxonomic profiling and functional investigation of food microbial communities. In this chapter, we provide a comprehensive step-by-step bioinformatical workflow to characterize bacterial ecology and resistome composition from metagenomic short-reads obtained by shotgun sequencing.

Keywords : shotgun metagenomics, antimicrobial resistance, bioinformatical workflow, microbial ecology

1. Introduction

Despite continuous improvement of food safety practices and production techniques, the control of food contamination and subsequent diseases remains a major issue throughout the world [1]. In addition, the worrying increase in antimicrobial resistance (AMR) over the past years in various bacterial species including foodborne pathogens has become a growing public health concern and strengthen the need to monitor thoroughly resistance emergence in food microbial ecosystems [2, 3]. Understanding how bacteria colonize, develop, adapt and spread in food environments is nevertheless an ongoing challenge and required the use of dedicated approaches integrating the specificity of environmental conditions and stresses experienced in the different food sectors. In particular, the interactions of foodborne bacterial pathogens with resident microflora are known to greatly influence their survival and persistence abilities in various food-production areas [4–6]. Different studies have reported the protection by resident surface flora of pathogens including *Listeria monocytogenes*, *Pseudomonas aeruginosa*, *Staphylococcus aureus* or *Escherichia coli* for instance, when exposed to various disinfectants [7]. Besides, food environments could constitute hot spots for genetic exchange between bacteria that can lead to the dissemination of antimicrobial resistance gene [8]. This highlights the necessity to identify the nature of genetic elements involved in antimicrobial resistance and associated transmission routes by integrating data on ecological interactions and gene distribution in bacterial populations, in order to apprehend dynamics of resistance emergence and contributing factors in food environments [9].

Evolutions in sequencing technologies along with development of user-friendly bioinformatical worklflows facilitated the use of sequencing approaches and next generation sequencing (NGS) have consequently revolutionized food safety management these last years [10]. Genomics approaches (including whole genome sequencing WGS) have been successfully applied to the accurate identification of bacterial pathogens and their phylogeny, strain subtyping, the detection of outbreaks, source attribution, or the characterization of the virulence, pathogenicity and or

resistance attributes of foodborne pathogens for instance [11]. A limitation of WGS is the need to have a pure culture of the bacteria to be sequenced, while in most cases bacterial isolates are unavailable or unculturable in laboratory conditions. More recently, the dramatic reduction for sequencing cost and computing resources led to a democratization and a growing use of shotgun metagenomics, corresponding to the direct sequencing of whole DNA from samples without selective isolation of bacteria or target-specific amplification [9]. Such approach has therefore the potential to detect non-culturable bacterial pathogens and to simultaneously characterize ecological diversity and gene distribution in communities from food, industrial surfaces or environmental sample [12]. In recent years, shotgun metagenomics have thus provided valuable data on resistome dynamics and related bacterial populations in various food-associated microbial communities [13–16]. Along with the increasing use of metagenomics, the development of dedicated bioinformatical tools, processing pipelines and databases emerged as a necessity to face the huge amount of data thereby generated.

In this chapter, we describe as step-by-step processing pipeline from sequencing short-read to the characterisation of bacterial ecology and resistome.

2. Materials

1. Computal ressources: To carry out the various stages of shotgun metagenomics analysis, computational resources are essential. A minimum of 16 cores, 96GB of RAM and 500GB of storage is recommended, either from a computer or from a remote computational cluster.
2. Bioinformatical tools and databases (*see* **Note 1-2-3**): Table 1 shows the different tools and databases used for building the bioinformatic pipeline for metagenomic sequence analysis
3. Sequence files (*see* **Note 4**):
    - Paired-end raw fastq files from short reads shotgun sequencing by an Illumina instrument.
    - Animal host reference genome in fasta format.

3. Methods

Shotgun metagenomics makes it possible to analyse the bacterial population of a sample. It also provides an overview of possible functions in the population, with an analysis of antimicrobial resistance genes in particular. **Figure 1** shows an overview of the bioinformatic pipeline proposed in this chapter for analysing bacterial ecology and resistome from short-read metagenomics.

During the analysis, information about the animal host will have to be removed from the reads dataset. According to the analyses to be carried out, samples will have to be taken from places in the agri-food industry. Depending on the location and the production chain sampled, the host animal will not be the same. For example, to analyse bacterial populations in a pig slaughterhouse, surface samples will be taken. As well as bacteria, animal matter will be found in the sample (*see* **Note 5**). However, this can be problematic because more or less animal matter can be collected. The more animal matter there is, the less information we will have about the bacteria during sequencing. The DNA of the host animal can flood bacterial DNA. Nevertheless, even if the sample preparation before the sequencing is correct, the host DNA will inevitably be sequenced.

Here Figure 1

3.1. Processing raw reads

Cleaning raw reads is an important step to avoid continuing the analysis with reads of poor quality. Reads can also be made up of adapters used for sequencing. All this information can have a negative impact on future analyses. So, we use Fastp to filter out low-quality reads and adapters. The software also provides a report on the quality of reads before and after processing.

```
fastp -i $FASTQ_R1 -I $FASTQ_R2 -o $FASTQ_CLEAN_R1 -O $FASTQ_CLEAN_R2 -h
$FASTQ_REPORT_HTML --detect_adapter_for_pe -M 25 -5 -r --correction
```

Where:

- `-i` is an option to input the first fastq reads file of a strain `$FASTQ_R1`.
- `-I` is an option to input the second fastq reads file of a strain `$FASTQ_R2`.
- `-o` is an option to output the first fastq reads file processed by Fastp `$FASTQ_CLEAN_R1`.
- `-O` is an option to output the second fastq reads file processed by Fastp `$FASTQ_CLEAN_R2`.
- `-h` is an option to output quality control report of the reads in HTML file `$FASTQ_REPORT_HTML`.
- `--detect_adapter_for_pe` is an option to detect and remove automatically sequencing adapters. It is only used for paired data.
- `-M` is an option to specify the quality threshold for others quality options. Here the threshold is setup on 25.
- `-5` is an option to drop the bases with bad quality at the start of the read. It uses a sliding window and cut if the mean quality of the bases in the window are below the thresholds specified by the option `-M`.
- `-r` is almost the same option as `-5`. The sliding window cut the right part of the read if the mean quality of the bases in the window are below the thresholds specified by the option `-M`.
- `--correction` is an option to enable base correction in overlapped regions.

The process of analysing shotgun metagenomics samples can then begin, starting with the elimination of reads from the host organism.

3.2. Recovery of bacterial reads.

To recover reads from bacteria only, it is essential to know the host organism in which the sample was taken. The reference genome of the species is used in the next step. It can be retrieved from the NCBI database. The reference assembly method will be employed, using the host genome as a reference. All reads that map to the genome will be removed from the dataset. Bowtie2 software will be used to carry out the assembly.

3.2.1. Reference alignment

To begin with, an index of the reference genome must be built using bowtie2-build.

```
bowtie2-build $GENOME_FASTA SPECIES
```

Where:

- $GENOME_FASTA is the genome fasta file.
- SPECIES is an index name used to call the genome index.

After that, the alignment step can be carried out with Bowtie2.

```
bowtie2 -x SPECIES -1 $FASTQ_CLEAN_R1 -2 $FASTQ_CLEAN_R2 -S $ASSEMBLY_SAM
```

Where:

- -x is an option to specify the genome index SPECIES build by bowtie2.
- -1 is an option for calling the first fastq file of reads processed by fastp $FASTQ_CLEAN_R1.
- -2 is an option for calling the second fastq file of reads processed by fastp $FASTQ_CLEAN_R2.
- -S is an option to specify the name of the sam $ASSEMBLY_SAM file that will be output by the process.

All mapped and unmapped reads are now present in the SAM file. The Samtools toolbox will be used to extract the unmapped reads.

3.2.2. Remove host reads

The SAM file needs to be converted into a BAM file to be processed (*see* **Note 6**).

```
samtools view -b $ASSEMBLY_SAM > $ASSEMBLY_BAM
```

Where:

- `view` is the tool to convert and process SAM/BAM file.
- `-b` is an option used to specify that the output will be in BAM format.
- `$ASSEMBLY_SAM` is the file in SAM format.
- `$ASSEMBLY_BAM` is the same file but in BAM format.

Unmapped reads can now be extracted from the BAM file.

```
samtools view -f 12 -F 256 $ASSEMBLY_BAM > $READS_UNMAPPED_BAM
```

Where:

- `-f` is an option to specify reads that can only be extracted from the file, followed by a flag. Here, flag 12 specifies read unmapped and paired read unmapped.
- `-F` if an option to specify reads that cannot be extracted from the file, followed by a flag. Here, flag 256 specifies reads that are not primary alignments.
- `$ASSEMBLY_BAM` is the BAM file with all the reads.
- `$READS_UNMAPPED_BAM` is the file that contain only the reads not mapped to the reference genome.

The final step is to separate this file of unmapped reads. Some software will require paired files to work.

The BAM file is sorted to have paired reads next to each other.

```
samtools sort -n $READS_UNMAPPED_BAM -o $READS_UNMAPPED_SORTED_BAM
```

Where:

- `sort` is the tool to sort BAM file.

- `-n` is an option for sorting by name.
- `$READS_UNMAPPED_BAM` is the file containing unsorted reads.
- `-o` is an option to specify the output name of the file `$READS_UNMAPPED_SORTED_BAM` containing sorted reads.

Now the reads file can easily be split into two fastq files for the two paired reads files.

```
samtools fastq $READS_UNMAPPED_SORTED_BAM -1 $HOST_READS_REMOVED_FASTQ_R1 -2 $HOST_READS_REMOVED_FASTQ_R2
```

Where:

- `fastq` is the tool to convert a BAM file to FASTQ. It will automatically compress the file if the file name has a .gz extension (*see* **Note 7**).
- `$READS_UNMAPPED_SORTED_BAM` is the file that contain sorted reads.
- `-1` is an option to specify the name of the first paired reads file `$HOST_READS_REMOVED_FASTQ_R1`.
- `-2` is an option to specify the name of the second paired reads file `$HOST_READS_REMOVED_FASTQ_R2`.

Reads from the host are now eliminated from paired fastq files. Several analysis steps can now be performed using these processed files, such as meta-assembly.

3.3. Analysis of bacterial communities

An important step in the analysis of a shotgun metagenomics sample is to know its bacterial diversity. The analysis is performed on the read with the host remove from the previous part. The MetaPhlAn software will be used to profile the bacterial community in the sample. It can detect Bacteria but also Archaea and Eukaryotes, which will be filtered out of the result because the analysis is carried out on bacteria only.

3.3.1.Build MetaPhlAn database

To search for the bacterial population present in the metagenome, the database used by MetaPhlAn must be installed.

```
metaphlan --install
```

Where:

- `--install` is an option for building the latest MetaPhlAn database. If the database is already installed, the build will not be started.

The database is installed in the Conda environment of the MetaPhlAn software.

3.3.2. Speciation

MetaPhlAn can now be launched, the software can handle paired-end metagenomes but it will not use the paired-end information. So here, the two fastq files can be used separately but also merged.

```
metaphlan  $HOST_READS_REMOVED_FASTQ_R1,$HOST_READS_REMOVED_FASTQ_R2  --bowtie2out
$INTERMEDIATE_METAPHLAN --input fastq > $METAPHLAN_RESULT
```

Where:

- `$HOST_READS_REMOVED_FASTQ_R1,R2` is the two fatsq files.
- `--bowtie2out` is a recommended for saving the intermediate Bowtie2 output `$INTERMEDIATE_METAPHLAN` to quickly rerun MetaPhlAn if needed.
- `--input_type` is an option to specify the input type. Here, it is in fastq format.
- `$METAPHLAN_RESULT` is the output file containing all the MetaPhlAn results.

The resulting file contains relative abundances of each bacterium at species level. This data may contain Archaea that need to be filtered to avoid analysis.

To go further and analyse the bacterial diversity of our sample using this file, two parameters can be calculated with the R package vegan and RStudio: alpha and beta diversity. Alpha diversity is a

measure of diversity in a single sample. It is, for example, the number of species present. But there are many indices that can be used to calculate this diversity in the vegan package. Beta diversity measures the diversity between different samples. A multivariate analysis of relative abundances using ordinal methods such as principal component analysis (PCA) is often carried out to represent this beta diversity.

Example of possible analysis and representation after processing the results with R. We want to know the bacterial populations in a pig slaughterhouse between 2017 and 2019. Two samples are taken, one at the beginning (in) and one et the end (out) of the slaughter line.

Here Figure 2 and Figure 3

Here is an example of how to represent the data obtained with metaphlan using 4 samples. The most common genus found are shown as pie plot (figure 2). The diversity of species present in samples from the beginning of the slaughter chain is lower than in samples from the end of the chain between 2017 and 2019. The diversity for each sample is also represented with the alpha diversity in the form of the Shannon index and the species richness (figure 3A). We can see the same conclusions as with the pie plots in the beta diversity graph. The two samples at the start of the chain are close together in the PCA, whereas the samples at the end of the chain are far apart.

3.4. Metagenome assembly

Assembling the metagenome will provide, after annotation, an overview of the biological functions presents in this metagenome. To carry out this assembly, the host's clean reads will be used.

3.4.1. Assembly

MEGAHIT software will be used to carry out the assembly. This stage of the analysis requires significant computational resources to operate.

```
megahit -1 $HOST_READS_REMOVED_FASTQ_R1 -2 $HOST_READS_REMOVED_FASTQ_R2 -o $MEGAHIT_RESULT_FOLDER
```

Where:

- -1 is an option to specify the name of the first paired reads file $HOST_READS_REMOVED_FASTQ_R1.
- -2 is an option to specify the name of the second paired reads file $HOST_READS_REMOVED_FASTQ_R2.
- -o is an option used to specify the output folder of megahit.

The newly assembled metagenome is found in the output folder. It is made up of several contigs placed one after the other. Newly formed contigs must be filtered to ensure that the metagenome is as clean as possible.

3.4.2. Contigs filtering

To filter out potentially poor-quality contigs in the assembly, two parameters are considered: contig size and depth. In terms of depth, the reads used to assemble the metagenome will be used to carry out an assembly using the newly constructed metagenome as a reference. Thus, contigs with little or no depth will be considered of poor quality.

So, as in the host read elimination step (3.2.1), bowtie2 will be used to perform reference assembly.

```
bowtie2-build $METAGENOME_ASSEMBLY METAGENOME
```

Where:

- $METAGENOME_ASSEMBLY is the metagenome assembly file in fasta format.
- METAGENOME is the basename chosen to call the index.

```
bowtie2 -x METAGENOME -1 $HOST_READS_REMOVED_FASTQ_R1 -2 $HOST_READS_REMOVED_FASTQ_R2 -S $METAGENOME_ASSEMBLY_SAM
```

Where:

- `-x` is the basename of the index `METAGENOME` for the reference genome.
- `-1` is an option for calling the first fastq file of host's removed reads `$HOST_READS_REMOVED_FASTQ_R1`.
- `-2` is an option for calling the second fastq file of host's removed reads `$HOST_READS_REMOVED_FASTQ_R2`.
- `-S` is an option to specify the name of the sam `$METAGENOME_ASSEMBLY_SAM` file that will be output by the process.

Now that the reference assembly is complete, it remains to estimate the depth for each contig using the Samtools toolbox. As in step 3.2.2, the SAM file from the previous step will be converted into a BAM file (*see* **Note 6**), which will be sorted as follows.

```
samtools view -b $METAGENOME_ASSEMBLY_SAM > $METAGENOME_ASSEMBLY_BAM
```

Where:

- `-b` is an option used to specify that the output will be in BAM format.
- `$METAGENOME_ASSEMBLY_SAM` is the file in SAM format.
- `$METAGENOME_ASSEMBLY_BAM` is the same file but in BAM format.

```
samtools sort $METAGENOME_ASSEMBLY_BAM -o $METAGENOME_ASSEMBLY_SORT_BAM
```

Where:

- `$METAGENOME_ASSEMBLY_BAM` is the file containing unsorted reads in the BAM assembly.
- `-o` is an option to specify the output name of the file `$METAGENOME_ASSEMBLY_SORT_BAM` containing sorted BAM reads.

The BAM file needed to be sorted to calculate the number of reads mapped to each contig in the metagenome.

```
samtools coverage --reference $METAGENOME_ASSEMBLY -o $METAGENOME_ASSEMBLY_COVERAGE $METAGENOME_ASSEMBLY_SORT_BAM
```

Where:

- `coverage` is the tool to produce an output of reads coverage by reference sequence.
- `--reference` is a generic option for setting the reference fasta file. This is mainly used to match the IDs of the contigs in the metagenome with the IDs of the contigs that will be obtained in the output.
- `-o` is an option for defining the name of the output `$METAGENOME_ASSEMBLY_COVERAGE` in .cov format.
- `$METAGENOME_ASSEMBLY_SORT_BAM` is the input sort BAM file.

Now that the coverage for each contig is known in the `$METAGENOME_ASSEMBLY_COVERAGE` file, the metagenome can be filtered according to two parameters: the size and depth of the contig (*see* **Note 8**).

```
awk '$7>=1 && $3>=500' $METAGENOME_ASSEMBLY_COVERAGE | awk '{print $1}' > $CONTIGS_TO_KEEP
```

Where:

- `awk` is a language for processing files line by line like spreadsheets.
- `$METAGENOME_ASSEMBLY_COVERAGE` is the file analysed by awk. It contains several columns with parameters on the size and coverage of the contigs. The filters are applied to column 7, which represents the average depth of the contig, and column 3, which indicates the size of the contig. Here, contigs with an average depth greater than or equal to 1 and a length greater than or equal to 500nt are kept.

- Then, if the conditions are met, another awk command is run after the pipe '|' symbol to store the name of contig contained in the first column of the $METAGENOME_ASSEMBLY_COVERAGE file in another file $CONTIGS_TO_KEEP used as a list.

Now that a list of contigs to keep has been compiled. All that remains is to filter the metagenome. The seqtk toolbox will be used for filtration.

```
seqtk subseq $METAGENOME_ASSEMBLY $CONTIGS_TO_KEEP > $METAGENOME_FASTA
```

Where:

- subseq is a tool for extracting sequences from a file using a metadata file.
- $METAGENOME_ASSEMBLY is the starting metagenome file.
- $CONTIGS_TO_KEEP is a list of contigs ID that will be kept in the metagenome.
- $METAGENOME_FASTA is the final metagenome file.

The metagenome is now filtered and can be used for further analysis.

3.4.3.Assembly quality control

Before moving on to the next stage, it may be useful to obtain statistics on the assembly of the metagenome using the MetaQUAST software.

```
metaquast -o $METAQUAST_RESULT -1 $HOST_READS_REMOVED_FASTQ_R1 -2 $HOST_READS_REMOVED_FASTQ_R2 $METAGENOME_FASTA
```

Where:

- -o is an option used to define the name of the MetaQUAST result output file $METAQUAST_RESULT.
- -1 is an option to define the first fastq file of reads $HOST_READS_REMOVED_FASTQ_R1.
- -2 is an option to define the second fastq file of reads $HOST_READS_REMOVED_FASTQ_R2.

- $METAGENOME_FASTA is the file of the genome in fasta format.

MetaQUAST will output a report that contains statistics like the number of reads used, the number of contigs, the mean length of the contigs, etc.

3.5. Annotation

Using the metagenome obtained in the previous step, a search for biological function can be carried out. This analysis will begin by annotating the genes present in the metagenome using the Prodigal tool.

```
prodigal -i $METAGENOME_FASTA -o $METAGENOME_GBK -a $PROTEINS_METAGENOME -d $NUCLEOTIDES_METAGENOME -p meta
```

Where:

- -i is an option to define the input file of the metagenome $METAGENOME_FASTA in fasta format.
- -o is an option to specify output file. Here a file in genbank format is requested.
- -a is an option to define the proteins translations file $PROTEINS_METAGENOME. It contains the genes predicted by Prodigal in fasta proteomic format with amino acids sequences.
- -d is an option to define the nucleotide sequences file $NUCLEOTIDES_METAGENOME. It contains the genes predicted by Prodigal in fasta nucleotide format.
- -p is an option for defining the analysis mode. Here, it's in meta format for metagenomics.

The metagenome is now partially annotated, meaning that Prodigal has only predicted the possible genes between each start and stop codon. Two files are therefore obtained as output, containing the predicted genes in fasta nucleotide and protein format. Subsequently, when searching for genes in databases or software, one or other of the files will be used as input.

3.6. Resistance gene research

Based on the partial annotation carried out in the previous step, it is now possible to search for resistance genes in the metagenome. To do this, various online databases (*see* **Note 9**) and software will be used for this purpose. There are several different tools for searching for antibiotic, biocide and metal resistance genes, but we will look at an example for each type of resistance here.

3.6.1. Antibiotic resistance genes

ABRicate software will be used to search the metagenome for antibiotic resistance genes. The tool is directly associated with different gene databases that can be changed as the files are processed. Here, we will use the NCBI AMRFinderPlus database.

```
abricate --db ncbi $NUCLEOTIDES_METAGENOME > $ANTIBIOTIC_RESISTANCE_GENES
```

Where:

- `--db` is an option to specify the database that will be used for the process. Here, we use the NCBI AMRFinderPlus database with the variable “ncbi”.
- `$NUCLEOTIDES_METAGENOME` is the file containing all the predicted gene of the metagenome in fasta nucleotide format.
- `$ANTIBIOTIC_RESISTANCE_GENES` is the output that will contains all the result from abricate.

The output file contains all the antibiotic resistance genes identified by abricate. Along with the gene name and the target antibiotic, two parameters are important to consider the percentage of coverage and identity. We can filter out genes with less than 90% identity and coverage to interpret only correct results.

3.6.2.Biocide and metal resistance genes

The BacMet database will be used to search for resistance genes to biocides and metals. The Blast tool and the Perl language must be installed to run the tool implemented with the database. The tool and database can then be downloaded from the BacMet website

(http://bacmet.biomedicine.gu.se/index.html). For research purposes, we use the experimental database that contains experimentally verified data. The other BacMet database has much more data, but all of them are predicted. The last step before launching the search is to bring the tool and the database together in the same directory.

```
./BacMet-Scan.pl -i $PROTEINS_METAGENOME -protein -blast -table -counts -o $BIOCIDE_AND_METAL_RESISTANCE_GENES -d $BACMET_DATABASE_FOLDER
```

Where:

- `BacMet-Scan.pl` is a Perl script used to search the database. It is launched with "./" before, since it is not installed but just present in the directory.
- `-i` is an option used to specify the input file. The proteins `$PROTEINS_METAGENOME` file of the metagenome will be input.
- `-protein` is an option indicating that the input contains proteins.
- `-blast` is an option calling the tool Blast for searching BacMet.
- `-table` is an option to specify that the BacMet-Scan report will be in table format.
- `-counts` is an option to output a list of counts for each gene in the database.
- `-o` is an option to specify the output file of the process `$BIOCIDE_AND_METAL_RESISTANCE_GENES`.
- `-d` is an option to indicate the database `$BACMET_DATABASE_FOLDER` to be used.

The BacMet results file shows the IDs of the genes in the database that are found in the proteins of the metagenome. It is therefore necessary to link the information from these gene IDs to the resistances specific to them to continue the analysis. To do this, we need to link the IDs contained in the results file with the IDs in the database metadata file (*see* **Note 10**). On we have succeeded in linking the genes contained in the results and the associated resistances, it is possible to interpret the results. All types of resistances to biocides and metals present in the metagenome are highlighted.

Genes for resistance to antibiotics, biocides and metals have been identified in the metagenome. To go further than simply concluding that these genes are present, it may be interesting to obtain their abundance to quantify them in the metagenome. We will see how to obtain the abundance of all the genes predicted by Prodigal in the next step.

3.7. Gene abundance

To obtain the abundance of genes in the metagenome, we will use the genes predicted by Prodigal in fasta nucleotide format. We will work on the principle that the gene abundance will depend on the number of reads that align with the gene. So, the more reads that align with a gene, the more abundant it will be in the metagenome.

3.7.1. Gene abundance

To align the reads with the genes, we will once again use reference genome assembly using the Prodigal predicted gene file in fasta nucleotide format. Bowtie2 will be used for the assembly and Samtools toolbox for the process of the BAM file.

```
bowtie2-build $NUCLEOTIDES_METAGENOME NUCLEOTIDES_GENES
```

Where:

- $NUCLEOTIDES_METAGENOME is the list file of predicted gene of the metagenome in fasta format.
- NUCLEOTIDES_GENES is the basename chosen to call the index.

```
bowtie2 -x NUCLEOTIDES_GENES -1 $HOST_READS_REMOVED_FASTQ_R1 -2 $HOST_READS_REMOVED_FASTQ_R2 -S $GENES_ASSEMBLY_SAM
```

Where:

- -x is the basename of the index NUCLEOTIDES_GENES for the reference genome.

- `-1` is an option for calling the first fastq file of host's removed reads `$HOST_READS_REMOVED_FASTQ_R1`.
- `-2` is an option for calling the second fastq file of host's removed reads `$HOST_READS_REMOVED_FASTQ_R2`.
- `-S` is an option to specify the name of the sam `$GENES_ASSEMBLY_SAM` file that will be output by the process.

```
samtools view -b $GENES_ASSEMBLY_SAM > $GENES_ASSEMBLY_BAM
```

Where:

- `-b` is an option used to specify that the output will be in BAM format.
- `$GENES_ASSEMBLY_SAM` is the file in SAM format.
- `$GENES_ASSEMBLY_BAM` is the same file but in BAM format.

```
samtools sort $GENES_ASSEMBLY_BAM -o $GENES_ASSEMBLY_SORT_BAM
```

Where:

- `$METAGENOME_ASSEMBLY_BAM` is the file containing unsorted reads in the BAM assembly.
- `-o` is an option to specify the output name of the file `$METAGENOME_ASSEMBLY_SORT_BAM` containing sorted BAM reads.

Now that the BAM file is well sorted, we can extract the number of reads that have successfully mapped onto the genes. First, we need to create an index for the BAM file.

```
samtools index $GENES_ASSEMBLY_SORT_BAM
```

Where:

- `index` is a tool to create BAI index file of BAM file.
- `$GENES_ASSEMBLY_SORT_BAM` is the bam sorted file of the assembly on metagenome genes

The index file is essential for extracting assembly statistics on metagenomes genes. Even if it is not indicated in the output, the file is still created in the directory.

```
samtools idxstats $GENES_ASSEMBLY_SORT_BAM > $GENES_STATS_ABUNDANCE
```

Where:

- idxstats is a tool used to output alignment summary statistics from a BAM file. It requires a BAM index for processing.
- $GENES_ASSEMBLY_SORT_BAM is the bam sorted file of the assembly on metagenome genes
- $GENES_STATS_ABUNDANCE is the output file that contains all the statistics about the assembly.

Now that we have statistics on the number of reads that have mapped onto each gene, it is important to normalize all this data to interpret the results.

3.7.2. Normalization

Data normalization avoids any bias linked to gene-mapped reads. It also makes it possible to compare different metagenomics samples. These two samples have different numbers of total reads. So, normalizing on the number of mapped reads provides a way to compare two sets of data.

Therefore, we will apply the gene coverage per million formula for each gene:

$$\text{GCMP (t)} = \frac{\left(\frac{counts\,(t)}{gene\ length\,(t)}\right)\times 10^6}{\sum_{i=1}^{n}\frac{counts\,(i)}{gene\ length\,(i)}}$$

Where:

- counts (t) = number of mapped reads to the gene (t).
- gene length (t) = the length of gene (t)
- n = number of all predicted genes.

Now that we have a quantification for each predicted genes, we can link the ID of predicted genes with the ID of genes identified in a resistance. This makes it possible to quantify resistance to antibiotics, biocides and metals in our metagenomic sample.

For example, here is a representation of the results obtained for the metal resistance genes for the 4 samples also used as examples in the MetaPhlAn step (2.3.2) (figure 4).

Here Figure 4

4. Notes

1. It is mandatory to use a UNIX environment to perform shotgun metagenomics analysis. Some software is only available on UNIX. In addition, the permeability of the environment will facilitate software installation and execution. So, a basic knowledge of the Bash language is essential.
2. To install each software, it is recommended to use a single Conda environment for each of them. This will facilitate use and avoid version conflicts between each software. So, before each use of a tool, the Conda environment where it will be installed should be activated.
3. All software has an option to indicate the number of CPUs that will be used to run it. This option often has a base value that way be insufficient, making execution take longer. It is therefore important to specify this option in all commands.
4. All the files described in the method section are called up in the form of a global variable. It takes the following the form: $NAME_OF_THE_FILE. In fact, each use of this variable with the same file name refers to the same file throughout the analysis.
5. Sample preparation is a critical step in the entire analysis, well before the bioinformatics analyses begin. As explained in the methods section, it is essential to avoid having animal

DNA in the sequenced sample as much as possible, as this will result in DNA being sequenced unnecessarily.

6. The SAM file is a very large file, so it is important to delete it to save space in the workspace, as it is not important to keep it for later.
7. It can be important to always compress fastq files to conserve storage space, as these are generally large files. Most software can use .fastq.gz files as input.
8. It can be interesting to play on these parameters to influence the quality of the metagenome and obtain stringent results, depending on the objectives of the study.
9. Pay attention to the versions of the databases that will be used. If it is already several years old, it is better to use a more recent database. While paying attention to the data contained in this database, it is best to use manually or experimentally processed data.
10. It is possible to perform this join using the Excel spreadsheet or by using Python or R scripts.

5. Data availability

All the data used as examples in this work are available on NCBI in the BioProject PRJNA1018717.

References

1. European Food Safety Authority, European Centre for Disease Prevention and Control (2022) The European Union One Health 2021 Zoonoses Report. EFS2 20:. https://doi.org/10.2903/j.efsa.2022.7666

2. Mancuso G, Midiri A, Gerace E, Biondo C (2021) Bacterial Antibiotic Resistance: The Most Critical Pathogens. Pathogens 10:1310. https://doi.org/10.3390/pathogens10101310

3. Sagar P, Aseem A, Banjara SK, Veleri S (2023) The role of food chain in antimicrobial resistance spread and One Health approach to reduce risks. International Journal of Food Microbiology 391–393:110148. https://doi.org/10.1016/j.ijfoodmicro.2023.110148

4. Yang X, Wang H, Hrycauk S, Holman DB, Ells TC (2023) Microbial Dynamics in Mixed-Culture Biofilms of Salmonella Typhimurium and Escherichia coli O157:H7 and Bacteria Surviving Sanitation of Conveyor Belts of Meat Processing Plants. Microorganisms 11:421. https://doi.org/10.3390/microorganisms11020421

5. Fagerlund A, Langsrud S, Møretrø T (2021) Microbial diversity and ecology of biofilms in food industry environments associated with Listeria monocytogenes persistence. Current Opinion in Food Science 37:171–178. https://doi.org/10.1016/j.cofs.2020.10.015

6. Visvalingam J, Zhang P, Ells TC, Yang X (2019) Dynamics of Biofilm Formation by Salmonella Typhimurium and Beef Processing Plant Bacteria in Mono- and Dual-Species Cultures. Microb Ecol 78:375–387. https://doi.org/10.1007/s00248-018-1304-z

7. Sanchez-Vizuete P, Orgaz B, Aymerich S, Le Coq D, Briandet R (2015) Pathogens protection against the action of disinfectants in multispecies biofilms. Front Microbiol 6:705. https://doi.org/10.3389/fmicb.2015.00705

8. Bergšpica I, Kaprou G, Alexa EA, Prieto-Maradona M, Alvarez-Ordóñez A (2020) Identification of risk factors and hotspots of antibiotic resistance along the food chain using next-generation sequencing. EFSA J 18:e181107. https://doi.org/10.2903/j.efsa.2020.e181107

9. Bridier A (2019) Exploring Foodborne Pathogen Ecology and Antimicrobial Resistance in the Light of Shotgun Metagenomics. Methods Mol Biol 1918:229–245. https://doi.org/10.1007/978-1-4939-9000-9_19

10. Jagadeesan B, Gerner-Smidt P, Allard MW, Leuillet S, Winkler A, Xiao Y, Chaffron S, Van Der Vossen J, Tang S, Katase M, McClure P, Kimura B, Ching Chai L, Chapman J, Grant K (2019) The use of

next generation sequencing for improving food safety: Translation into practice. Food Microbiology 79:96–115. https://doi.org/10.1016/j.fm.2018.11.005

11. Allard MW, Bell R, Ferreira CM, Gonzalez-Escalona N, Hoffmann M, Muruvanda T, Ottesen A, Ramachandran P, Reed E, Sharma S, Stevens E, Timme R, Zheng J, Brown EW (2018) Genomics of foodborne pathogens for microbial food safety. Curr Opin Biotechnol 49:224–229. https://doi.org/10.1016/j.copbio.2017.11.002

12. Billington C, Kingsbury JM, Rivas L (2022) Metagenomics Approaches for Improving Food Safety: A Review. J Food Prot 85:448–464. https://doi.org/10.4315/JFP-21-301

13. Doster E, Thomas KM, Weinroth MD, Parker JK, Crone KK, Arthur TM, Schmidt JW, Wheeler TL, Belk KE, Morley PS (2020) Metagenomic Characterization of the Microbiome and Resistome of Retail Ground Beef Products. Front Microbiol 11:541972. https://doi.org/10.3389/fmicb.2020.541972

14. Moon SH, Udaondo Z, Abram KZ, Li X, Yang X, DiCaprio EL, Jun S-R, Huang E (2022) Isolation of AmpC- and extended spectrum β-lactamase-producing Enterobacterales from fresh vegetables in the United States. Food Control 132:108559. https://doi.org/10.1016/j.foodcont.2021.108559

15. Rubiola S, Macori G, Chiesa F, Panebianco F, Moretti R, Fanning S, Civera T (2022) Shotgun metagenomic sequencing of bulk tank milk filters reveals the role of Moraxellaceae and Enterobacteriaceae as carriers of antimicrobial resistance genes. Food Res Int 158:111579. https://doi.org/10.1016/j.foodres.2022.111579

16. Bloomfield SJ, Zomer AL, O'Grady J, Kay GL, Wain J, Janecko N, Palau R, Mather AE (2023) Determination and quantification of microbial communities and antimicrobial resistance on food through host DNA-depleted metagenomics. Food Microbiol 110:104162. https://doi.org/10.1016/j.fm.2022.104162

17. Feldgarden M, Brover V, Gonzalez-Escalona N, Frye JG, Haendiges J, Haft DH, Hoffmann M, Pettengill JB, Prasad AB, Tillman GE, Tyson GH, Klimke W (2021) AMRFinderPlus and the Reference Gene Catalog facilitate examination of the genomic links among antimicrobial resistance, stress response, and virulence. Sci Rep 11:12728. https://doi.org/10.1038/s41598-021-91456-0

18. Pal C, Bengtsson-Palme J, Rensing C, Kristiansson E, Larsson DGJ (2014) BacMet: antibacterial biocide and metal resistance genes database. Nucleic Acids Res 42:D737-743. https://doi.org/10.1093/nar/gkt1252

19. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990) Basic local alignment search tool. J Mol Biol 215:403–410. https://doi.org/10.1016/S0022-2836(05)80360-2

20. Langmead B, Salzberg SL (2012) Fast gapped-read alignment with Bowtie 2. Nat Methods 9:357–359. https://doi.org/10.1038/nmeth.1923

21. Chen S, Zhou Y, Chen Y, Gu J (2018) fastp: an ultra-fast all-in-one FASTQ preprocessor. Bioinformatics 34:i884–i890. https://doi.org/10.1093/bioinformatics/bty560

22. Li D, Liu C-M, Luo R, Sadakane K, Lam T-W (2015) MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. Bioinformatics 31:1674–1676. https://doi.org/10.1093/bioinformatics/btv033

23. Manghi P, Blanco-Míguez A, Manara S, NabiNejad A, Cumbo F, Beghini F, Armanini F, Golzato D, Huang KD, Thomas AM, Piccinno G, Punčochář M, Zolfo M, Lesker TR, Bredon M, Planchais J, Glodt J, Valles-Colomer M, Koren O, Pasolli E, Asnicar F, Strowig T, Sokol H, Segata N (2023) MetaPhlAn 4 profiling of unknown species-level genome bins improves the characterization of diet-associated microbiome changes in mice. Cell Rep 42:112464. https://doi.org/10.1016/j.celrep.2023.112464

24. Mikheenko A, Saveliev V, Gurevich A (2016) MetaQUAST: evaluation of metagenome assemblies. Bioinformatics 32:1088–1090. https://doi.org/10.1093/bioinformatics/btv697

25. Hyatt D, Chen G-L, Locascio PF, Land ML, Larimer FW, Hauser LJ (2010) Prodigal: prokaryotic gene recognition and translation initiation site identification. BMC Bioinformatics 11:119. https://doi.org/10.1186/1471-2105-11-119

26. Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, Whitwham A, Keane T, McCarthy SA, Davies RM, Li H (2021) Twelve years of SAMtools and BCFtools. Gigascience 10:giab008. https://doi.org/10.1093/gigascience/giab008

Tables

Table 1. Bioinformatic tools and databses uses for metagenomics sequences analysis

Name	Description	Source
ABRicate	Contig analysis tool for the antimicrobial resistance gene. It is associated with other databases.	https://github.com/tseemann/abricate
AMRFinderPlus	NCBI database of antimicrobial resistance genes.	[17]
Bacmet	Antibacterial biocide and metal resistance genes database. Genes are added either manually, with experimental verification of resistance, or predictively by sequence similarity with other known genes.	[18]
BLAST	Tool to compare two or more sequences to find similar regions.	[19]
Bowtie2	Tool to align sequencing reads to reference sequences.	[20]
fastp	Tool to control the quality of data from high-throughput sequencing.	[21]
Megahit	Designated software to assemble metagenomes from NGS data.	[22]
Metaphlan	Tool to profile microbial communities from metagenomic shotgun sequencing data. It relies on 5.1M unique clade-specific marker genes.	[23]
MetaQuast	Tool to evaluate and to compare	[24]

	metagenome assemblies.	
Prodigal	Predict protein-gene coding sequences for prokaryotic genomes.	[25]
Samtools	Toolbox for processing sequence alignment file in BAM, SAM or CRAM format.	[26]
Seqtk	Toolbox for processing sequence files in FASTA or FASTQ format.	https://github.com/lh3/seqtk

Figure 1. Metagenomic analysis pipeline (available at https://github.com/Arnaud-Bridier/METARes).

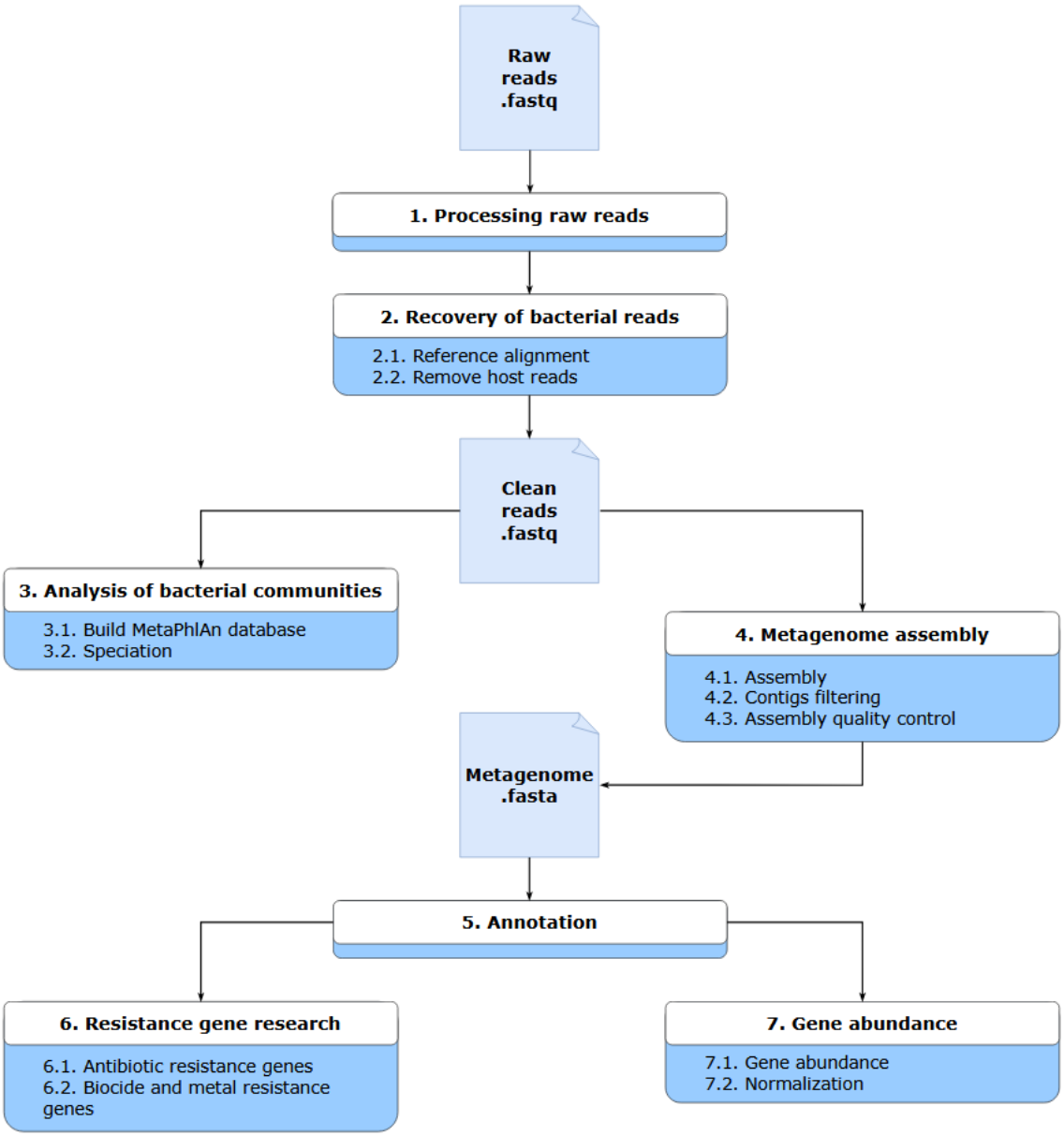

Figure 2. Proportional abundance of bacterial genus between 2017 and 2019 at the beginning (in) and end (out) of the pig slaughter chain.

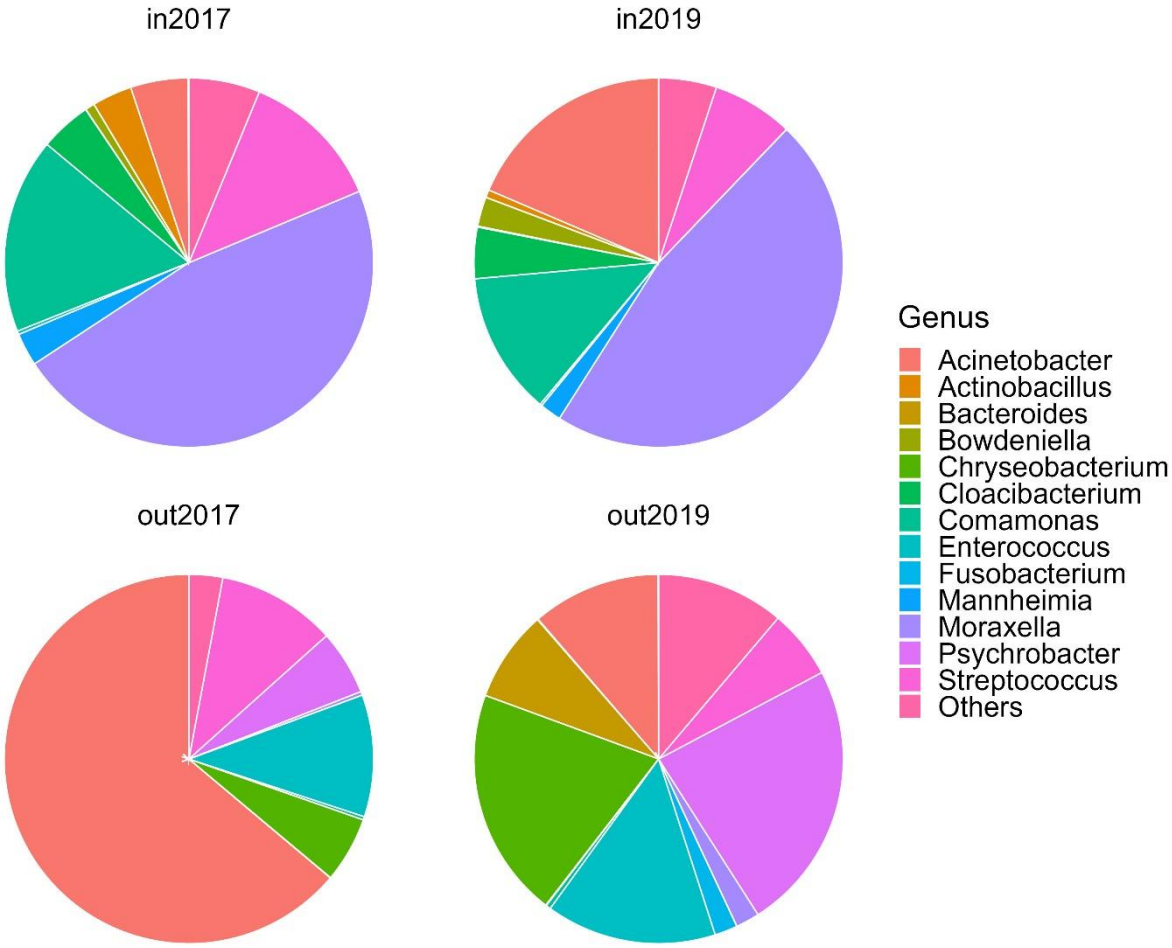

Figure 3. (A) Intra-sample alpha diversity of each bacterial populations using Shannon index and species richness. (B) Inter-sample beta diversity with Bray-Curtis index.

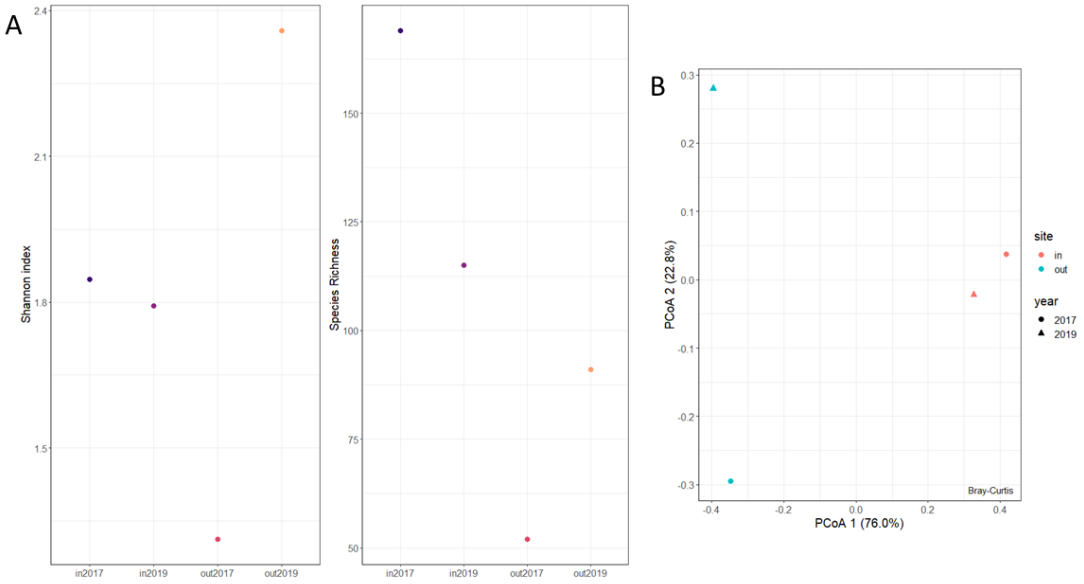

Figure 4. Mean abundance of metal resistance gene between 2017 and 2019 at the beginning (in) and end (out) of the pig slaughter chain.