Guided discussion using interactive polls

GenEpi-BioTrain Virtual training 23

Virulence Profiling of *E. coli* Pathotypes

João Cardoso (joacar@dtu.dk)
Technical University of Denmark (DTU), National Food Institute

10/03/2026

Impact of Low QC Scores

Increased base-calling errors lead to false SNPs and indels, biasing variant calling and downstream association or outbreak analyses

Poor-quality tails reduce mapping quality and can cause reads to map to incorrect regions or fail to map at all, decreasing effective coverage

Low-quality and adapter-contaminated reads fragment assemblies, leading to shorter contigs and mis-assemblies

Filtered low-quality reads increase runtime and resource usage for mapping, assembly, and downstream tools without adding reliable information

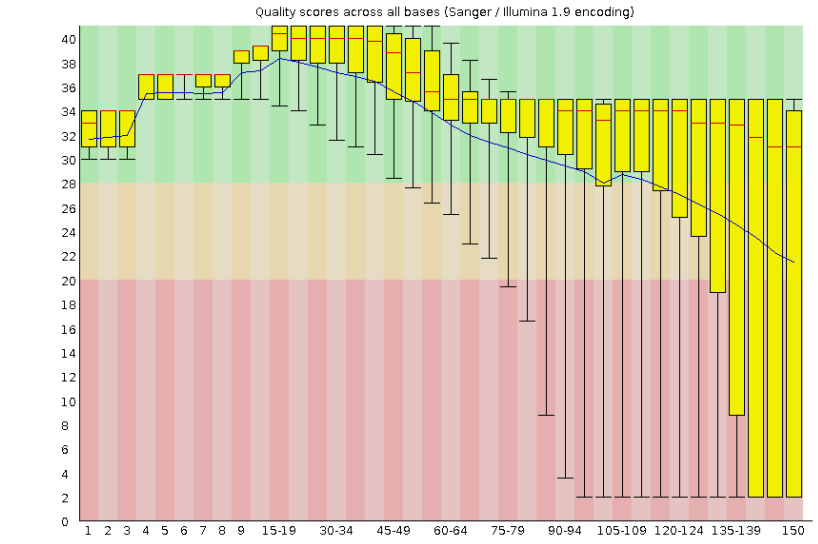
Example QC Report

Sample VFS17

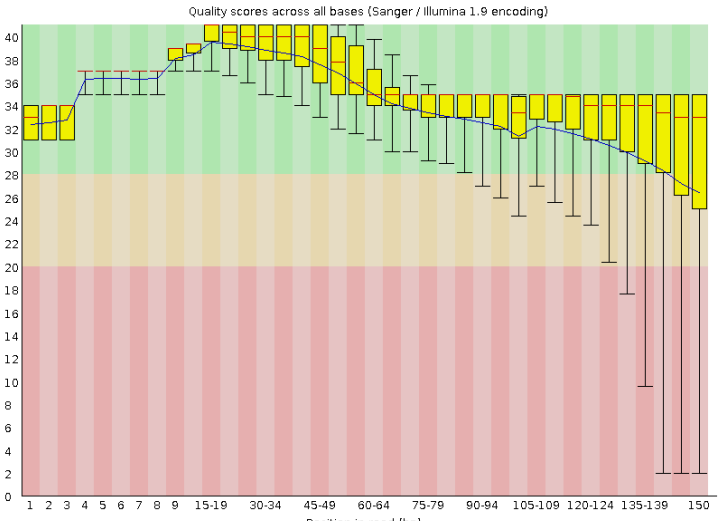
Summary

- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content

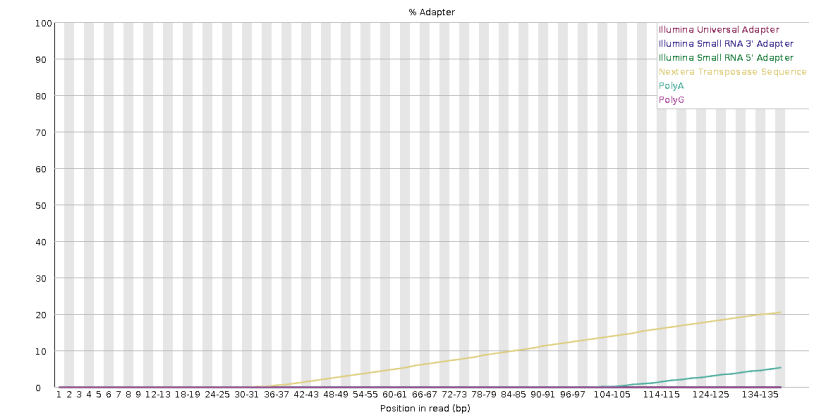
Per base sequence quality



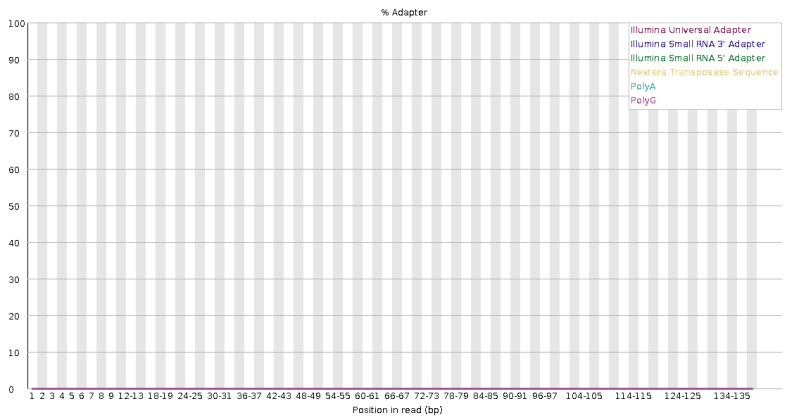
Per base sequence quality



Adapter Content



Adapter Content



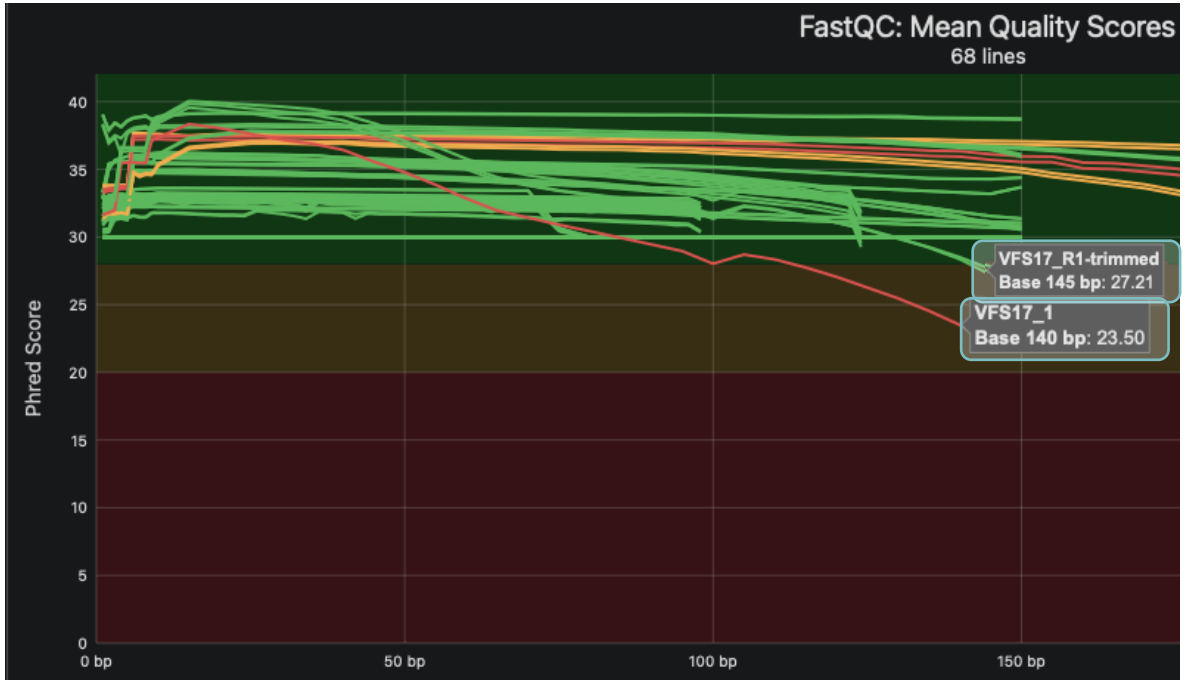
Summary

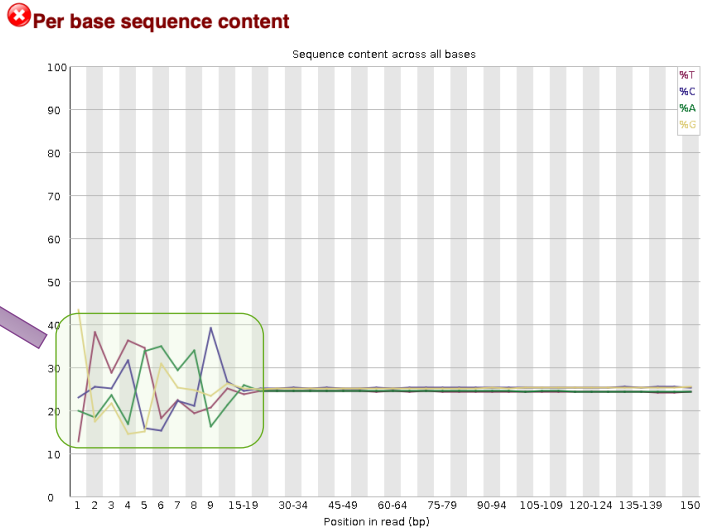
- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content

How Can We Solve Low QC Score Problems?

EXAMPLE

Using FastP flags:
-trim_front2 15 -> Trim first 15 bases
-trim_tail2 15 -> Trim last 15 bases





After Filter

Before Filter

EXAMPLE (FastP)

-g -> polyG tail trimming
-x -> enable polyX trimming
-n 0 -> Maximum N bases allowed per read
-2 -> enable adapter detection
-q 20 -> Minimum Quality Score
-l 50 -> Minimum length required

Impact of Contamination on WGS Analysis

False positives in virulence and resistance gene detection: contaminant genomes can carry virulence or AMR genes that are wrongly attributed to the target isolate

Distorted phylogenies and clustering: contaminant reads/contigs introduce spurious SNPs and mixed signals, leading to incorrect tree topologies and outbreak inferences

Contaminant sequences can assemble together with target reads, creating chimeric contigs and incorrect gene predictions or genome structure

Misleading conclusions on gene presence/absence and functional profiles (e.g. overestimating virulence potential or resistance repertoire)

How Can We Detect the Contamination?

Intra-Species

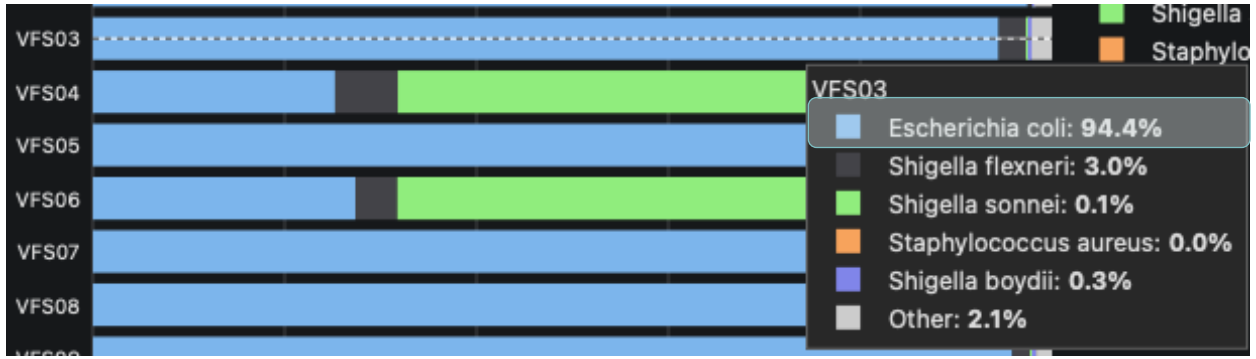
Sample Name	Dups	GC	Avg len	Median len	Failed	Seqs
VFS03_1	37.6 %	50.0 %	144 bp	151 bp	10 %	4.7 M
VFS03_2	37.4 %	50.0 %	144 bp	151 bp	10 %	4.7 M
VFS03_R1-trimmed	37.8 %	50.0 %	143 bp	151 bp	10 %	4.6 M
VFS03_R2-trimmed	37.5 %	50.0 %	143 bp	151 bp	10 %	4.6 M

Inter-Species

Sample Name	Dups	GC	Avg len	Median len	Failed	Seqs
VFS10_1	54.7 %	48.0 %	123 bp	150 bp	20 %	6.8 M
VFS10_2	50.3 %	48.0 %	123 bp	150 bp	30 %	6.8 M
VFS10_R1-trimmed	54.8 %	48.0 %	124 bp	151 bp	20 %	6.7 M
VFS10_R2-trimmed	50.6 %	48.0 %	124 bp	151 bp	30 %	6.7 M

How Can We Detect the Contamination?

Intra-Species



Inter-Species

Kraken+Bracken

Reads

KmerFinder 3.2 results:

Template	Num	Score	Expected	Template_length	Query_Coverage
NZ_CP069882.1 Escherichia coli strain FDAARGOS_1289 chromosome, complete genome	13684	24770700	4729	167386	63.43
NZ_CP007390.1 Escherichia coli strain ST540, complete genome	9370	5538013	11387	152044	14.18
NZ_CP065615.1 Escherichia coli strain FDAARGOS_943 chromosome, complete genome	23345	1448175	13569	159988	3.71

KmerFinder 3.2 results:

Template	Num	Score	Expected	Template_length	Query_Coverage
NZ_CP050862.1 Escherichia coli strain 8-3-DC15 chromosome, complete genome	23752	31762984	5405	148404	64.42
NZ_CP015817.1 Staphylococcus aureus strain FORC_039, complete genome	17478	10615624	10470	119390	21.53
NZ_CP065624.1 Escherichia coli strain FDAARGOS_941 chromosome, complete genome	20536	1264494	16526	149714	2.56

How Can We Detect the Contamination?

Assembly

Rapid assessment of genome bin quality using machine learning.

Copy table | Configure columns | Scatter plot | Violin plot | Export as CSV... | Showing 16/16 rows and 7/12 columns. | + Summarize ta

Sample Name	Predicted Completeness	Predicted Contamination	Coding Density	Average Gene Length	Genome Size	GC Content	Total Coding Sequences
VFS01	100.0 %	0.20 %	0.863	280 a.a.	4 877 513	0.51	5 022
VFS02	100.0 %	0.15 %	0.878	304 a.a.	5 367 782	0.50	5 178
VFS03	100.0 %	51.52 %	0.873	201 a.a.	9 768 362	0.51	14 201
VFS04	100.0 %	0.00 %	0.852	277 a.a.	4 767 551	0.51	4 896
VFS05	100.0 %	0.03 %	0.874	279 a.a.	5 378 078	0.50	5 626
VFS10	100.0 %	0.20 %	0.864	278 a.a.	4 872 545	0.51	5 058

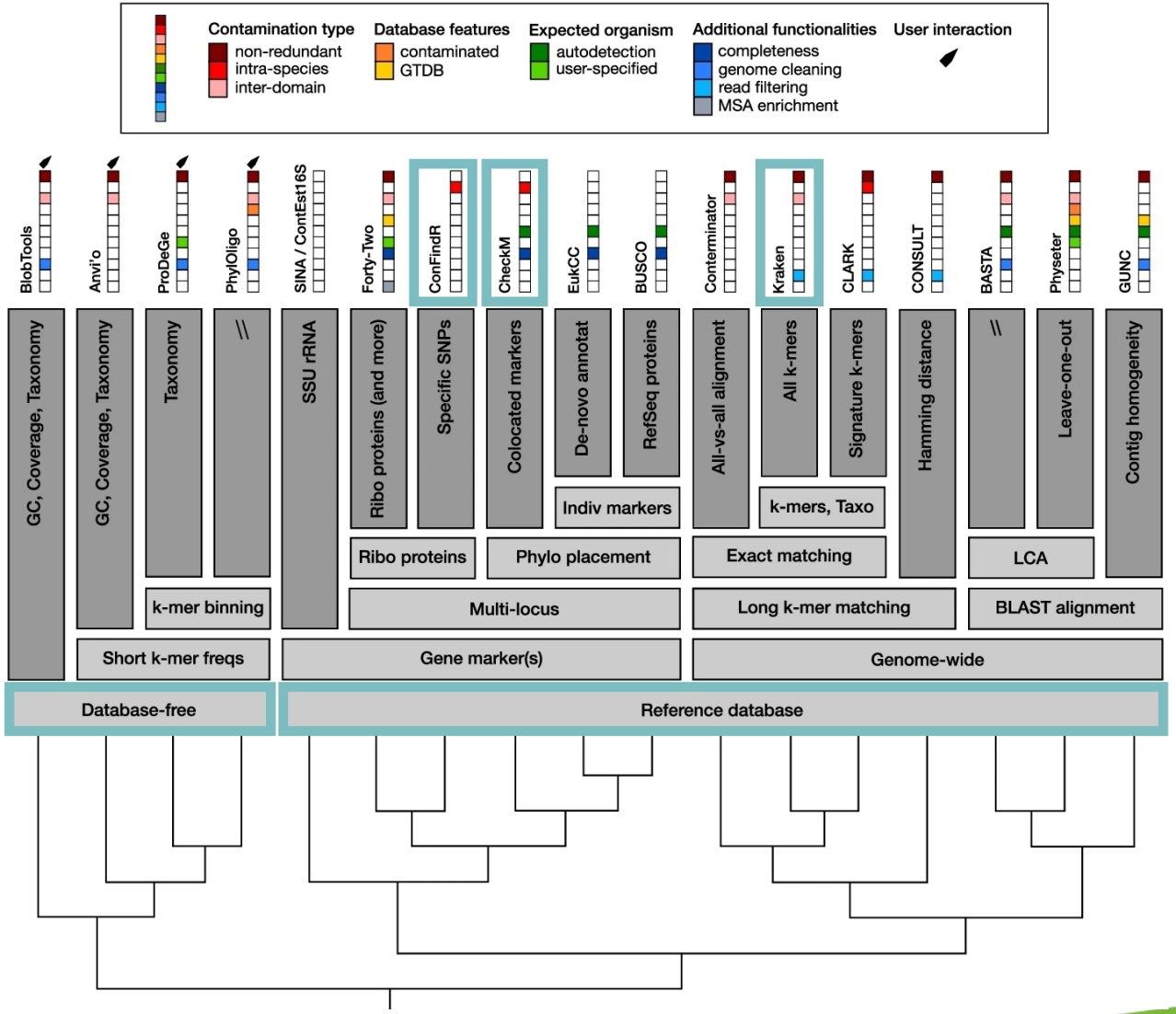
Intra-Species

Inter-Species

Read-based Contamination Assessment

3. Read-based approaches

- Contaminants may be present at **low abundance** and may not assemble well into contigs.
- **Read-based** classification uses **all reads**, including those that never assemble.
- **Different tools** focus on **different questions** (intra versus intra-species)
- No single tool detects all contamination types → combining tools is best



Cornet L, et al. (2022). Genome Biol. 23(1):60; LCA, Lowest Common Ancestor

How Can We Solve Contamination?

Classified reads with Kraken2 (and Bracken if used) against a comprehensive database to identify reads belonging to non-target taxa

Removed reads assigned to non-target species or higher-level taxa above a chosen threshold (e.g. discard reads assigned outside the target genus or family).

Re-run QC (FastQC/fastp) and mapping/assembly on the cleaned read sets to confirm improved GC profiles and species purity

For samples with high-level contamination even after filtering, we considered them unreliable and excluded them from further analyses.

How Can We Solve Contamination?

Taxonomic Read Profiling (Kraken2)

We map the raw paired-end sequence reads against a reference database. Using exact k-mer matches, every single read is assigned a taxonomy ID (TaxID) so we know exactly what organisms are present in the sample.

```
kraken2 --db $DB_PATH --paired --threads $THREADS --report report.txt --output sample.kraken R1-trimmed.fastq.gz R2-trimmed.fastq.gz
```

0.12	7945	7945	U	0	unclassified
99.88	6732609	98818	R	1	root
98.4	6633037	153	R1	131567	cellular organisms
98.39	6632316	36550	D	2	Bacteria
63.43	4275695	1356	K	3379134	Pseudomonadati
63.41	4274236	4426	P	1224	Pseudomonadota
63.34	4269455	25059	C	1236	Gammaproteobacteria
62.96	4243873	46358	O	91347	Enterobacterales
62.26	4196814	3172850	F	543	Enterobacteriaceae
12.08	814550	65553	G	561	Escherichia
10.59	713637	695232	S	562	Escherichia coli
34.21	2305634	7908	F	90964	Staphylococcaceae
34.09	2297637	1976263	G	1279	Staphylococcus
4.47	301102	300098	S	1280	Staphylococcus aureus
0.01	499	499	S1	869816	Staphylococcus aureus subsp. aureus JKD6159
0	111	111	S1	548473	Staphylococcus aureus subsp. aureus TCH60
0	87	87	S1	548470	Staphylococcus aureus subsp. aureus MN8
0	52	52	S1	273036	Staphylococcus aureus RF122

Abundance Re-estimation (Bracken)

Kraken2 tells us what is there, but Bracken tells us how much. It mathematically re-distributes reads to estimate the precise relative abundance (%) of the top species and families in the sample.

```
bracken -d $DB_PATH -i report.txt -o sample.bracken -r $READ_LENGTH -l S -t $THREADS
```

name	taxonomy_id	taxonomy_lvl	kraken_assigned_reads	added_reads	new_est_reads	fraction_total_reads
Escherichia coli	562	S	713637	2689240	3402877	0.50561
Escherichia albertii	208962	S	13948	978	14926	0.00222
Escherichia fergusonii	564	S	7576	1506	9082	0.00135
Escherichia marmotae	1499973	S	7421	451	7872	0.00117
Escherichia sp. E4742	2044467	S	4713	399	5112	0.00076
Staphylococcus aureus	1280	S	301102	1987640	2288742	0.34007
Staphylococcus argenteus	985002	S	3941	521	4462	0.00066
Staphylococcus schweitzer	1654388	S	2280	400	2680	0.0004
Staphylococcus sp. IVB6181	2929481	S	533	24	557	0.00008
Staphylococcus sp. 17KM08	2583989	S	115	1	116	0.00002

How Can We Solve Contamination?

Identify The "Good" Taxonomy ID

Using the Kraken2 reports, we need to identify the specific Taxonomy ID (TaxID) of the primary, intended species/family that we want to keep (e.g., *Escherichia coli*, (Species level, taxID: 562) or *Enterobacteriaceae* (family level, taxID: 543).

0.12	7945	7945	U	0	unclassified
99.88	6732609	98818	R	1	root
98.4	6633037	153	R1	131567	cellular organisms
98.39	6632316	36550	D	2	Bacteria
63.43	4275695	1356	K	3379134	Pseudomonadati
63.41	4274236	4426	P	1224	Pseudomonadota
63.34	4269455	25059	C	1236	Gammaproteobacteria
62.96	4243873	46358	O	91347	Enterobacterales
62.26	4196814	3172850	F	543	Enterobacteriaceae
12.08	814550	65553	G	561	Escherichia
10.59	713637	695232	S	562	Escherichia coli
34.21	2305634	7908	F	90964	Staphylococcaceae
34.09	2297637	1976263	G	1279	Staphylococcus
4.47	301102	300098	S	1280	Staphylococcus aureus
0.01	499	499	S1	869816	Staphylococcus aureus subsp. aureus JKD6159
0	111	111	S1	548473	Staphylococcus aureus subsp. aureus TCH60
0	87	87	S1	548470	Staphylococcus aureus subsp. aureus MN8
0	52	52	S1	273036	Staphylococcus aureus RF122

Active Contamination Removal (KrakenTools Rescue)

When a read matches our "Good" dictionary, its sequence and structural data are completely extracted, while the contaminated garbage reads are permanently ignored

Kraken Tools

```
python3 extract_kraken_reads.py -k sample.kraken -s R1.fastq.gz -s2 R2.fastq.gz -t
<Target_TaxID> --include-children --fastq-output -o Clean_R1.fastq -o2 Clean_R2.fastq
```

Acknowledgements

The creation of this training material was commissioned by ECDC to Technical University of Denmark (DTU) with the direct involvement of João Cardoso (Research Assistant at DTU, Msc – joacar@food.dtu.dk)