An introduction to the analysis of shotgun metagenomic data

Soyoun Park (Ph.D. candidate)
Food Microbiology

MICROBIAL GENOMICS

REVIEW

Pérez-Cobas *et al.*, *Microbial Genomics* 2020;6

DOI 10.1099/mgen.0.000409

MICROBIOLOGY SOCIETY

OPEN DATA OPEN MICROBIOLOGY

Metagenomic approaches in microbial ecology: an update on whole-genome and marker gene sequencing analyses

Ana Elena Pérez-Cobas, Laura Gomez-Valero and Carmen Buchrieser*

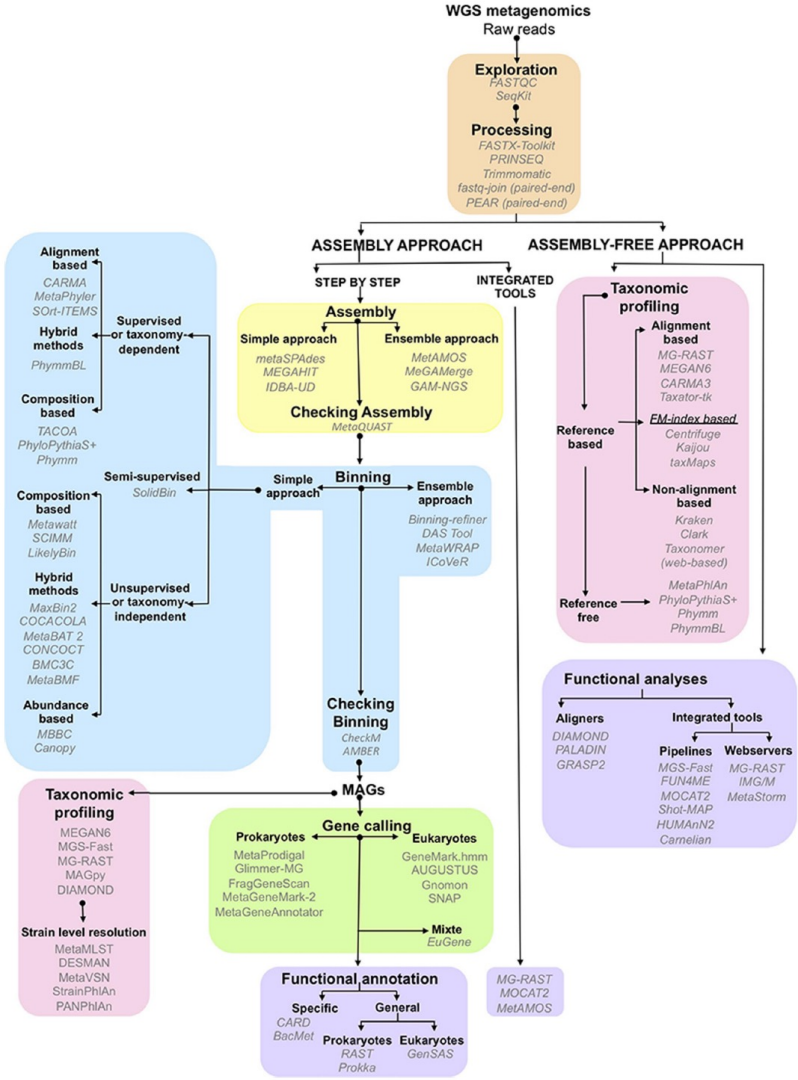


Fig. 1. Schematic representation of the main steps necessary for the analysis of WGS metagenomics derived data. The software related to each step is shown in italics.

BACKGROUND

16S amplicon sequencing

Screening
Assembly
Filtering
Clustering
Classification
Mothur

Analysis
MicrobiomeAnalyst
R (ggplot)

```
make.file(inputdir=., type=fastq, prefix=stability)
make.contigs(file=stability.files, processors=40)
summary.seqs(fasta=stability.trim.contigs.fasta, processors=40)
# you will get an output that tells you how to set maxlength/minlength (75% and 25% percentile) in screen.seqs
screen.seqs(fasta=stability.trim.contigs.fasta, group=stability.contigs.groups, summary=stability.trim.contigs.summary, maxambig=0, maxlength=253, minlength=253, processors=40)
summary.seqs(fasta=stability.trim.contigs.good.fasta, processors=40)
unique.seqs(fasta=stability.trim.contigs.good.fasta)
count.seqs(name=stability.trim.contigs.good.names, group=stability.contigs.good.groups)
summary.seqs(fasta=stability.trim.contigs.good.unique.fasta, count=stability.trim.contigs.good.count_table, processors=40)
# align.seqs is a little slow
align.seqs(fasta=stability.trim.contigs.good.unique.fasta, reference=silva.nr_v138.pcr.unique.align, processors=40)
summary.seqs(fasta=stability.trim.contigs.good.unique.align, count=stability.trim.contigs.good.count_table, processors=40)
screen.seqs(fasta=stability.trim.contigs.good.unique.align, count=stability.trim.contigs.good.count_table, summary=stability.trim.contigs.good.unique.summary, start=1968, end=11550, maxhomop=8, processors=40)
summary.seqs(fasta=stability.trim.contigs.good.unique.good.align, count=stability.trim.contigs.good.good.count_table, processors=40)
filter.seqs(fasta=stability.trim.contigs.good.unique.good.align, vertical=T, trump=., processors=40)
unique.seqs(fasta=stability.trim.contigs.good.unique.good.filter.fasta, count=stability.trim.contigs.good.good.count_table)
pre.cluster(fasta=stability.trim.contigs.good.unique.good.filter.unique.fasta, count=stability.trim.contigs.good.unique.good.filter.count_table, diffs=2, processors=40)
split.abund(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.count_table, cutoff=5, accnos=true)
summary.seqs(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.count_table, processors=40)
chimera.uchime(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.count_table, dereplicate=t, proc
remove.seqs(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.fasta, accnos=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.denovo.uchime.accnos)
summary.seqs(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.pick.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.denovo.uchime.pick.count_tabl
# list sequences in template file
list.seqs(fasta=silva.nr_v138.pcr.unique.align)
# select those reads from the taxonomy file
get.seqs(taxonomy=silva.nr_v138.tax, accnos=silva.nr_v138.pcr.unique.accnos)
classify.seqs(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.pick.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.denovo.uchime.pick.count_tab
remove.lineage(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.pick.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.denovo.uchime.pick.count_tal
dist.seqs(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.pick.pick.fasta, cutoff=0.10, processors=40)
cluster(column=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.pick.pick.dist, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.denovo.uchime.pick.pick.count_
make.shared(list=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.pick.pick.opti_mcc.list, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.denovo.uchime.pick
classify.otu(list=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.pick.pick.opti_mcc.list, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.denovo.uchime.pic
get.relabund(shared=stability.trim.contigs.good.unique.good.filter.unique.precluster.abund.pick.pick.opti_mcc.shared, label=0.03, scale=totalotu)
```

BACKGROUND

16S amplicon sequencing

Screening
Assembly
Filtering
Clustering
Classification
Mothur

Analysis
MicrobiomeAnalyst
R (ggplot)

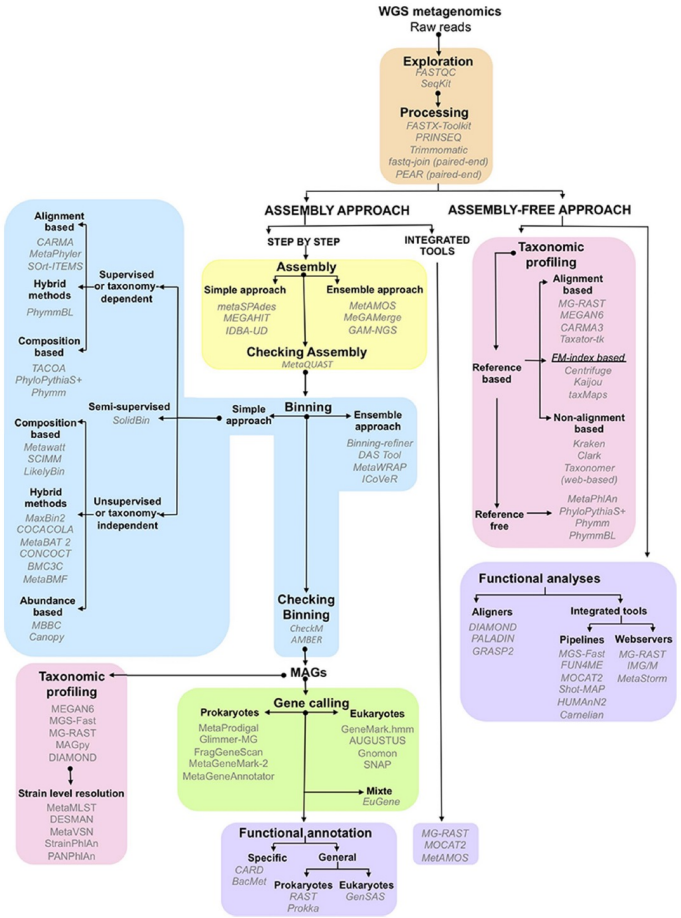
Whole-genome sequencing

Processing (trimming/quality filtering)
Error-correction
Assembly (*de novo*)
Polishing
ProkayoteAssembly

Functional annotation
Prokka

Analysis
Roary, etc

Shotgun metagenomic sequencing



BACKGROUND

16S amplicon sequencing

Screening
Assembly
Filtering
Clustering
Classification
Mothur

Analysis
MicrobiomeAnalyst
R (ggplot)

Whole-genome sequencing

Processing (trimming/quality filtering)
Error-correction
Assembly (*de novo*)
Polishing
ProkayoteAssembly

Functional annotation
Prokka

Analysis
Roary, etc

Shotgun metagenomic sequencing

Processing
BBDuk, Trimmomatic, metaWRAP
QC check
FastQC

Assembly-free approach

Taxonomic profiling
MG-RAST, KRAKAN2

Assembly approach

Assembly
metaSPAdes, MegaHIT
QC check
QUAST

Binning
MetaBAT2, MaxBIN2, CONCOCT
QC check
CheckM

Metagenome-assembled genomes
(MAGs)

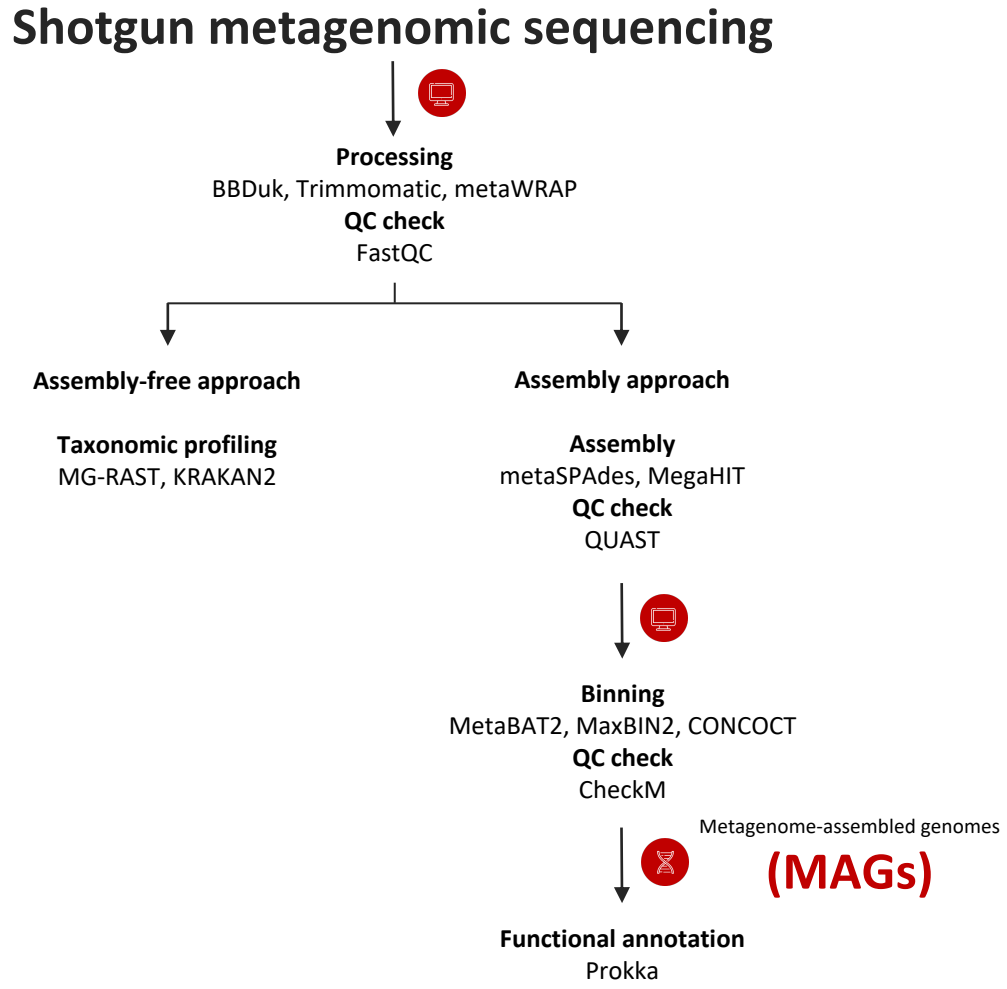
Functional annotation
Prokka

BACKGROUND

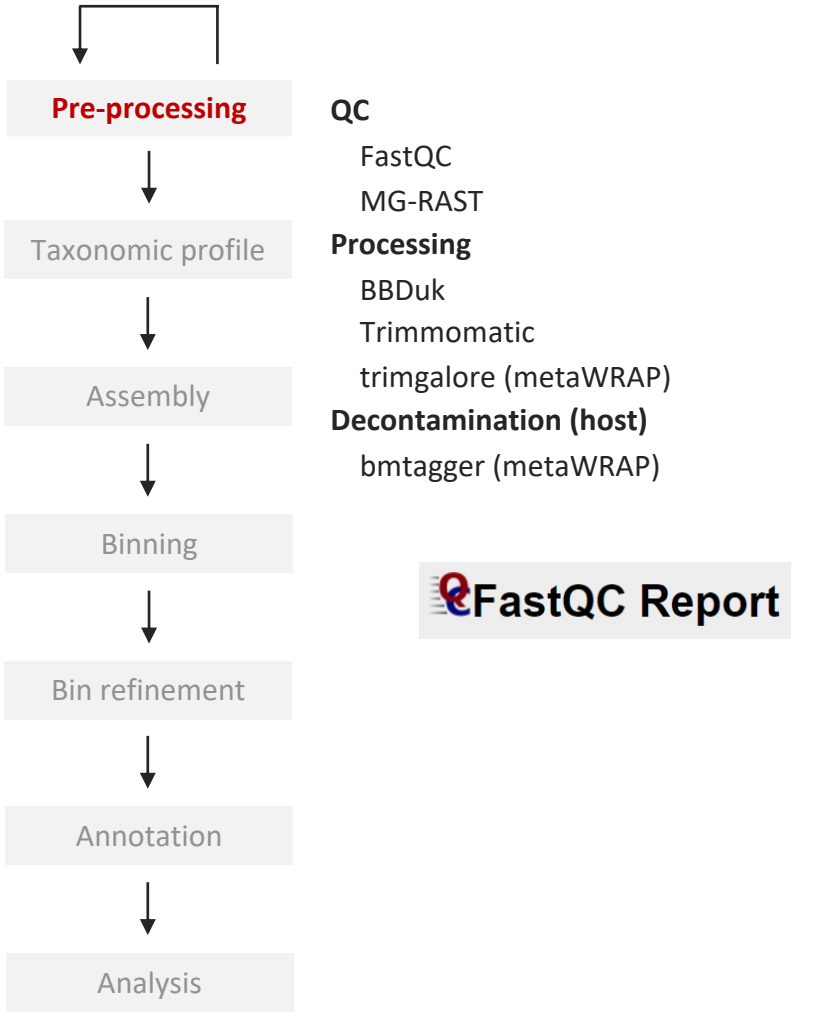
No single pipeline for all research purposes

Q. Who is the major bacterial species?

Q. What can the major bacterial species do?



1. Pre-processing – Trimming and filtering



QC
- FastQC
- MG-RAST

Processing
- BBDuk
- Trimmomatic
- trimgalore (metaWRAP)

Decontamination (host)
- bmtagger (metaWRAP)

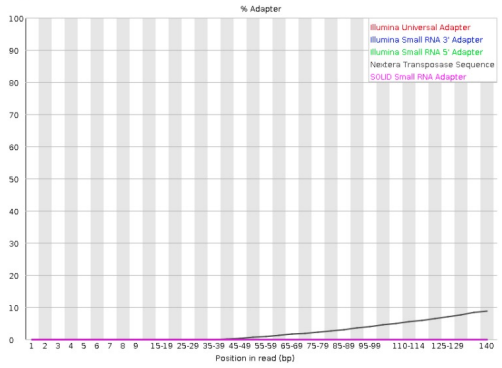
FastQC Report

Raw sequences
(151 bp)
Low quality sequences
Mixed sequences (bacteria, fungi, host, etc)
Adaptor sequences

Basic Statistics

Measure	Value
Filename	190204C88Q3_1.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	48885515
Sequences flagged as poor quality	0

Adapter Content

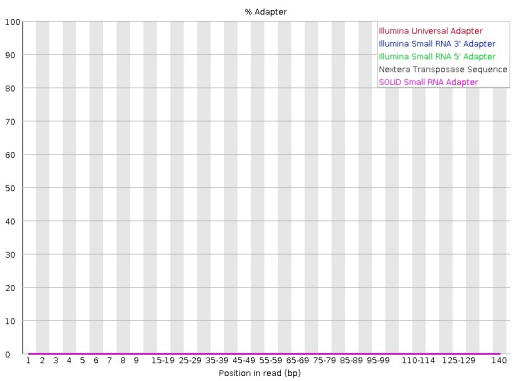


Cleaned sequences
(20 – 151 bp)
Good quality sequences
No host DNA, no adaptor sequences

Basic Statistics

Measure	Value
Filename	final_pure_reads_1.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	649838
Sequences flagged as poor quality	0

Adapter Content



2. Taxonomy Profiles – MG-RAST and Kraken2

Pre-processing

↓

Taxonomic profile

↓

Assembly

↓

Binning

↓

Bin refinement

↓

Annotation

↓

Analysis

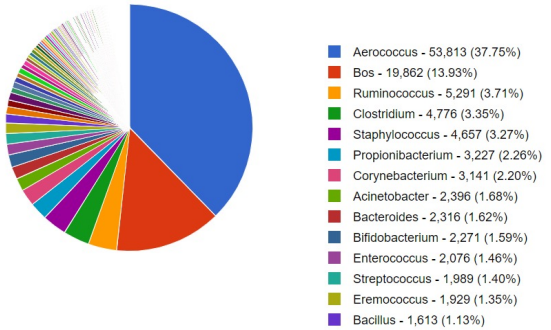
Assembly-free approach

Taxonomic profiling

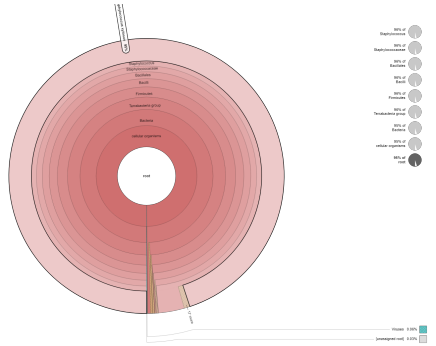
- MG-RAST
- Kraken2 (metaWRAP)

Cleaned sequences
(20 – 151 bp)
Good quality sequences
No host DNA, no adaptor sequences

MG-RAST
metagenomics analysis server



Kraken 2



3. Assembly – megaHIT vs metaSPAdes

Pre-processing

↓

Taxonomic profile

↓

Assembly

↓

Binning

↓

Bin refinement

↓

Annotation

↓

Analysis

Assembler

megaHIT

metaSPAdes

Report

QUAST

Cleaned sequences
(20 – 151 bp)
Short reads

Contigs
(various)
contiguous DNA

Quast

Worst Median Best Show heatmap

Statistics without reference	megaHIT	metaSPAdes
# contigs	1844	2242
# contigs (>= 0 bp)	4144	14 239
# contigs (>= 1000 bp)	429	494
# contigs (>= 5000 bp)	127	116
# contigs (>= 10000 bp)	81	77
# contigs (>= 25000 bp)	33	38
# contigs (>= 50000 bp)	9	10
Largest contig	108 429	126 462
Total length	4 026 619	4 396 583
Total length (>= 0 bp)	4 904 219	7 539 866
Total length (>= 1000 bp)	3 085 127	3 230 976
Total length (>= 5000 bp)	2 604 293	2 677 769
Total length (>= 10000 bp)	2 274 370	2 392 407
Total length (>= 25000 bp)	1 416 654	1 735 135
Total length (>= 50000 bp)	636 939	774 309
N50	16 609	14 956
N75	1079	932
L50	62	62
L75	366	564
GC (%)	36.87	37.52
Mismatches		
# N's	0	4940
# N's per 100 kbp	0	112.36

4. Binning – CONCOCT vs MaxBIN2 vs metaBAT2

Pre-processing

↓

Taxonomic profile

↓

Assembly

↓

Binning

↓

Bin refinement

↓

Annotation

↓

Analysis

Algorithm

CONCOCT

MaxBIN2

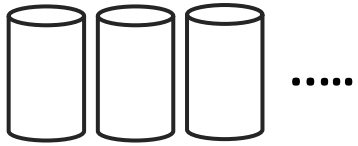
mataBAT2

Quality

CheckM

Contigs
(various in size)
contiguous DNA

Contigs from mixed and different origins



Bins
(various in size)
Contigs belonging to the same biological taxon (species, subspecies, or genus)

What is a good quality Bin?
> 70% complete and < 10% contaminated bin

CheckM

Bin Id	Marker lineage	# genomes	# markers	# marker sets	0	1	2	3	4	5+	Completeness	Contamination	Strain heterogeneity
bin.1	c__Bacilli (UID285)	586	324	180	1	317	6	0	0	0	99.44	2.69	16.67
bin.0	c__Bacilli (UID285)	586	324	180	1	317	6	0	0	0	99.44	2.69	16.67
bin.2	c__Bacilli (UID285)	586	324	180	38	281	5	0	0	0	90.56	2.13	20.00

5. Bin Refinement – Quality of the refined bins

Pre-processing

↓

Taxonomic profile

↓

Assembly

↓

Binning

↓

Bin refinement

RefineM

Quality

CheckM

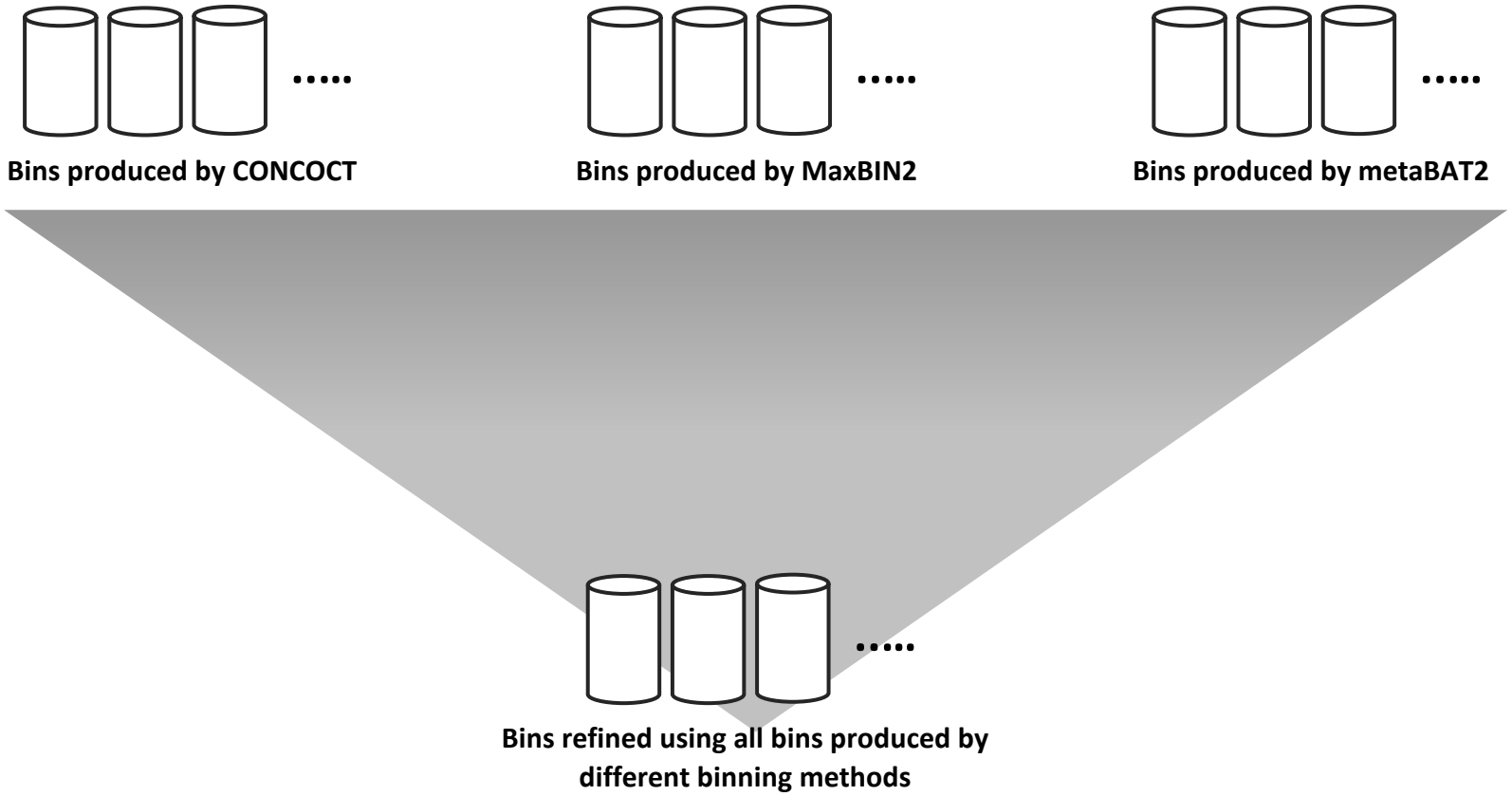
PATRIC

↓

Annotation

↓

Analysis



5. Bin Refinement – Quality of the refined bins

Pre-processing

↓

Taxonomic profile

↓

Assembly

↓

Binning

↓

Bin refinement

RefineM

Quality

CheckM

PATRIC

↓

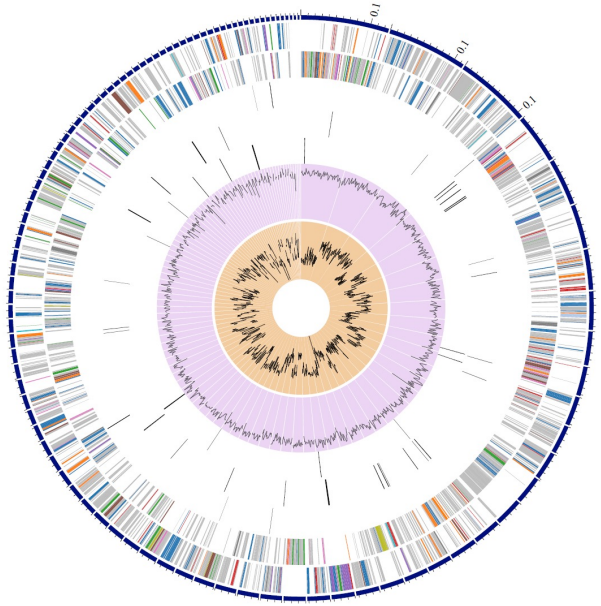
Annotation

↓

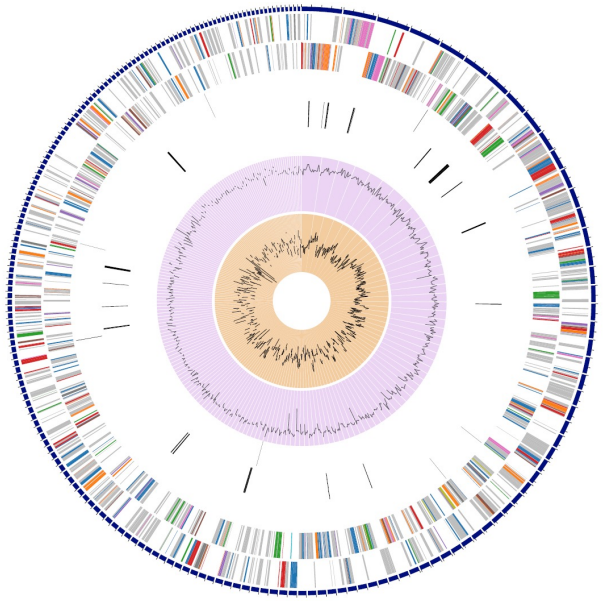
Analysis

PATRIC 3.6.12

Sample: 190507C7Q2
MAG: Bin.2 produced by metaBAT2
Species: ***Staphylococcus xylosus***
Genome quality: poor (too short contigs)



Sample: 190204C88Q3
MAG: Bin.2 produced by metaBAT2
Species: ***Aerococcus urinaeequi***
Genome quality: poor (too short contigs)



6. Annotation

Pre-processing
↓
Taxonomic profile
↓
Assembly
↓
Binning
↓
Bin refinement
↓
Annotation Prokka
↓
Analysis

Sample: 190507C7Q2
MAG: Bin.2 produced by metaBAT2
Species: ***Staphylococcus xylosus***
Genome quality: poor (too short contigs)

	S. xylosus
Contigs	102
Bases	2,360,379 bp
CDS	2,234
tRNA	20

Sample: 190204C88Q3
MAG: Bin.2 produced by metaBAT2
Species: ***Aerococcus urinaeequi***
Genome quality: poor (too short contigs)

	A. urinaeequi
Contigs	199
Bases	1,542,727 bp
CDS	1,373
tRNA	9

7. Analysis - Potential antagonistic factors

Pre-processing

↓

Taxonomic profile

↓

Assembly

↓

Binning

↓

Bin refinement

↓

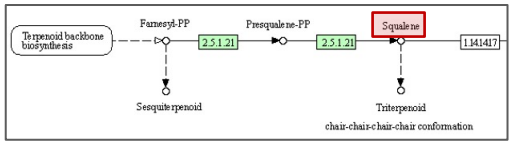
Annotation

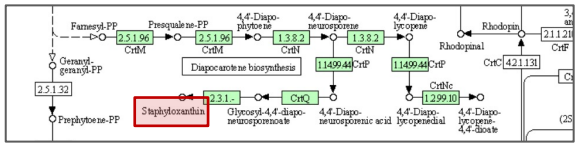
↓

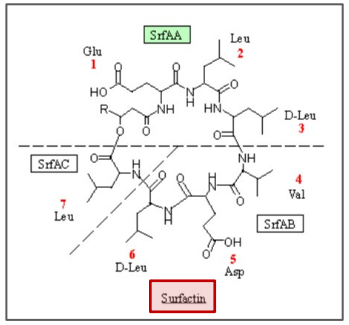
Analysis

BAGEL4
antiSMASH
KEGG
Roary

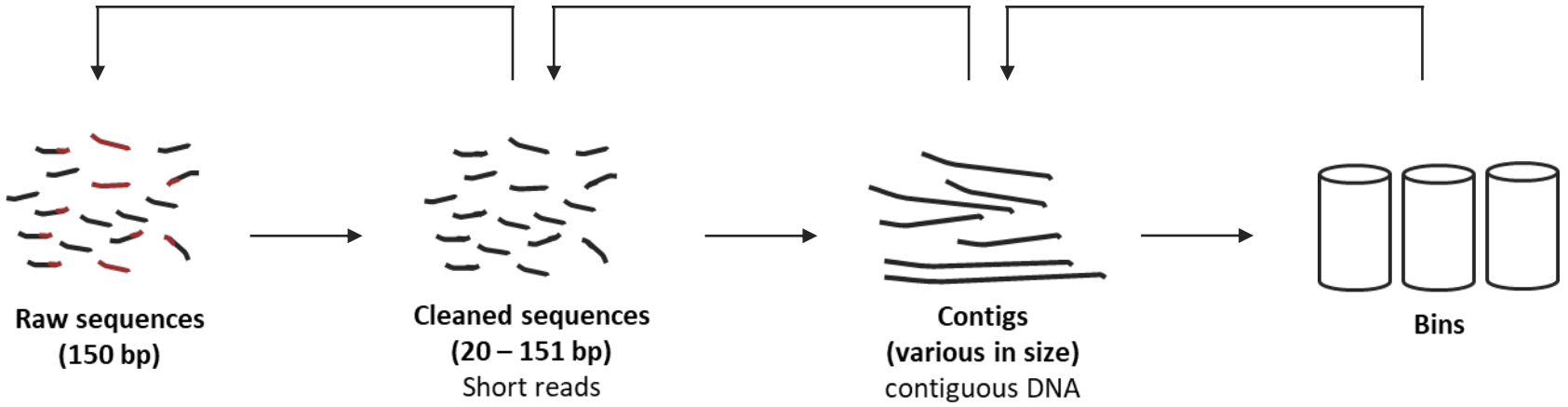
Region	Type	From	To	Most similar known cluster		Similarity
Region 3.1	NRPS	1	42,035			
Region 6.1	siderophore	8,317	23,316	staphyloferrin A	Other:Non-NRP siderophore	100%
Region 9.1	T3PKS	8,207	48,395	capsular polysaccharide	Saccharide:Exopolysaccharide	3%
Region 21.1	terpene	4,837	25,712			
Region 83.1	terpene	1	8,072			



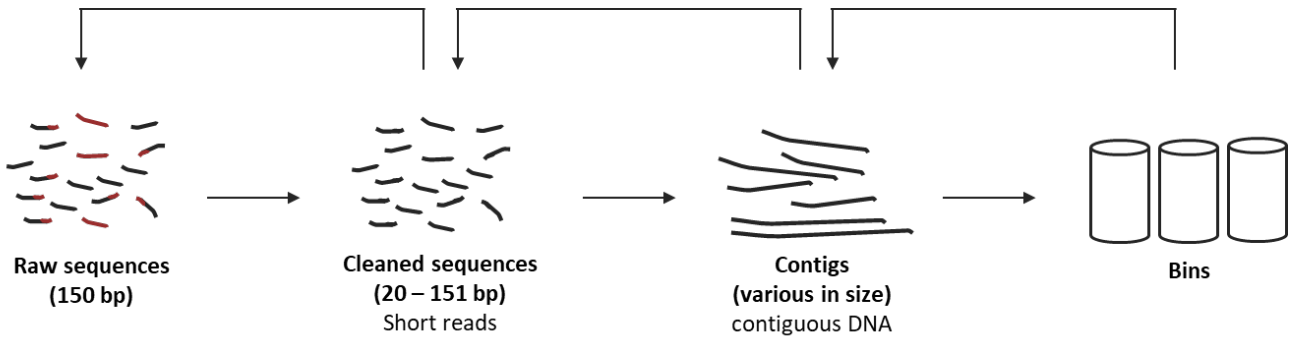




SUMMARY


Raw sequences
(150 bp)
Cleaned sequences
(20 – 151 bp)
Short reads
Contigs
(various in size)
contiguous DNA
Bins

SUMMARY



Shotgun metagenomic sequencing

Processing
BBDuk, Trimmomatic, metaWRAP
QC check
FastQC

Assembly-free approach

Taxonomic profiling
MG-RAST, KRAKAN2

Assembly approach

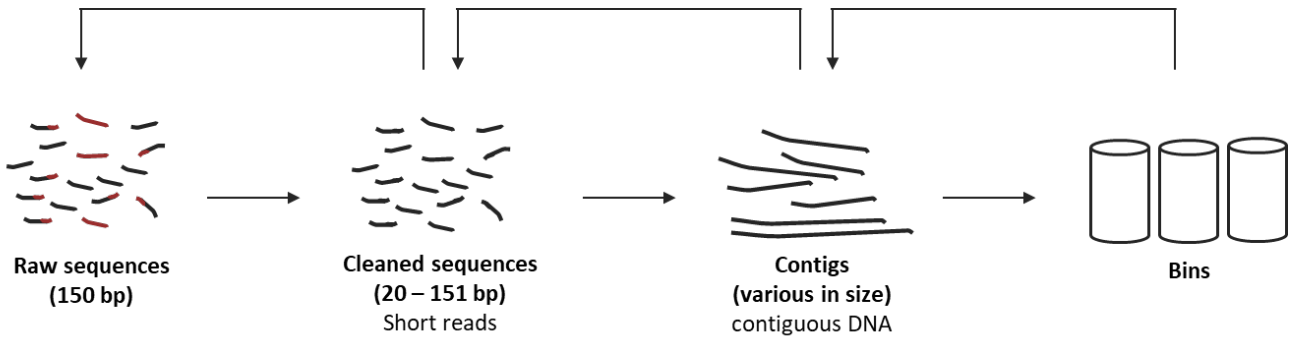
Assembly
metaSPAdes, MegaHIT
QC check
QUAST

Binning
MetaBAT2, MaxBIN2, CONCOCT
QC check
CheckM

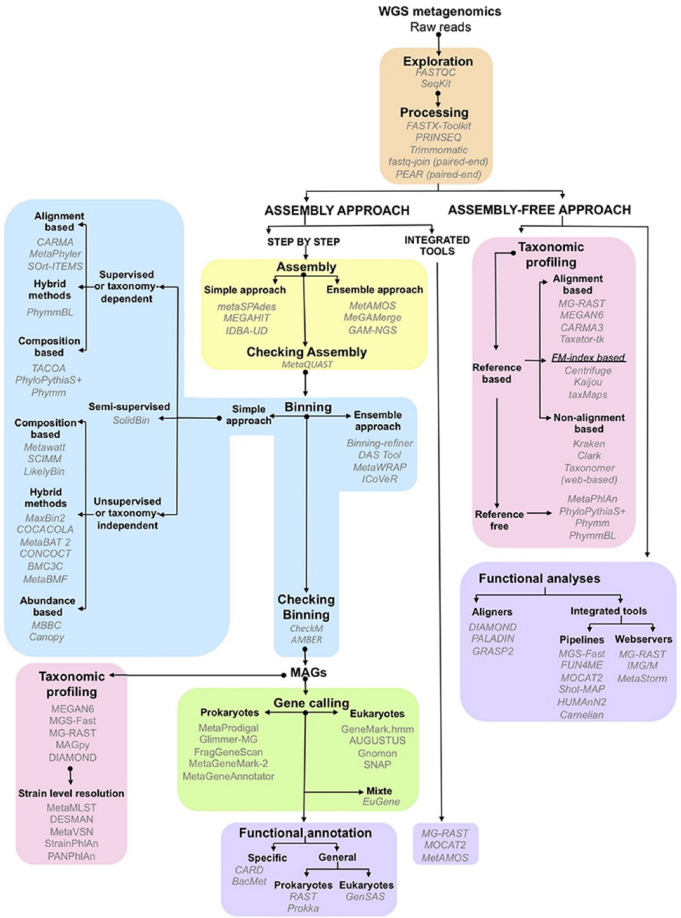
Metagenome-assembled genomes
(MAGs)

Functional annotation
Prokka

SUMMARY



Shotgun metagenomic sequencing



Thank you