Tonight!

- 7pm – Bay Reading Room (adjacent to Lillie 203)
- Go in the front door of Lillie (steps to left on MBL St as you come from Swope/Ebert), OR
- Go in the back door of Lillie, by the water; take the elevator up to floor 2/Library.

Metagenomics and k-mers (Day 2)

Titus Brown

MBL STAMPS 7/22/24

Library analogy!

- Suppose you stumble across a library from an ancient civilization.
- There are many small fragments of scrolls in a room! They are well mixed!
- Conveniently you also have access to a card catalog that has one card per scroll with a Dewey decimal system number on each one!
- Analogs:
  - Scroll fragment size: read size
  - Assembly: rebuilding the scrolls
  - Mapping: mapping the fragments to modern scrolls or books
  - K-mers: Analyzing all the words!

Challenges of this "library”

- Different book sizes!
- Many editions of scrolls!
- Different popularities of scrolls!
- Different languages may be present, and you may be able to read some, many, or none!

- What if: you do not know the boundaries between words/sentences??

Revisiting: What are k-mers? => WORDS

Fixed-length “words” of DNA that are extracted by sliding a window along sequence.
“k” is the window size.

```
[12]: build_kmers('ATGGACCAGATATAGGGAGAGCCAGGTAGGACA', 21)

[12]: ['ATGGACCAGATATAGGGAGAG',
       'TGGACCAGATATAGGGAGAGC',
       'GGACCAGATATAGGGAGAGCC',
       'GACCAGATATAGGGAGAGCCA',
       'ACCAGATATAGGGAGAGCCAG',
       'CCAGATATAGGGAGAGCCAGG',
       'CAGATATAGGGAGAGCCAGGT',
       'AGATATAGGGAGAGCCAGGTA',
       'GATATAGGGAGAGCCAGGTAG',
       'ATATAGGGAGAGCCAGGTAGG',
       'TATAGGGAGAGCCAGGTAGGA',
       'ATAGGGAGAGCCAGGTAGGAC',
       'TAGGGAGAGCCAGGTAGGACA']
```

You can apply this to fragments! Or books/scrolls!

Word (k-mer) based analyses:

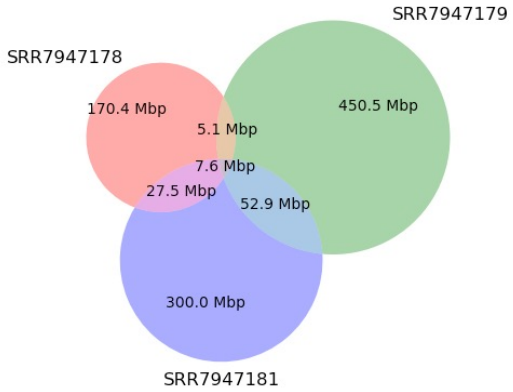
- How similar are these two books (even if I can't read them)?
- How similar are these two different piles of fragments?
- I have a book or two or three! How much matches to this pile, and how much is unknown?
  - (Both *flat/non-counting* and *abund/counting* number of fragments a word is in.)

MAG	SRA accession number and location	K-mer containment (%)	Effective coverage	Percentage of MAG detected in metagenome (%)	Number of mapped reads from MAG
Microcoleus sp. MP8IB2.171	SRR5468150 Mat lift-off from Lake Fryxell, Antarctica	99.18*	125.84	99.35	5,864,248
	SRR6266358 Polar Desert Sand Communities, Antarctica	65.02*	93.34	88.34	3,832,909
	SRR5855414 Moab Green Butte, Utah, USA	57.50*	407.19	86.11	15,915,624
	SRR2952554 Ningxia, China	41.65*	18.83	73.53	899,792
	SRR5247052 Sonoran Desert, Colorado Plateau, USA	41.10*	180.87	73.08	10,101,904
	ERR3588763 Pig Farm, UK	40.61*	9.38	76.14	329,215
	SRR5891573 Glacier Snow, China	39.54*	14.36	75.66	482,590
	ERR1333181 Mine Tailing Pool Sediment near Shaoyang, China	38.36*	28.59	73.24	1,120,980
	SRR5459769 Wastewater in Milwaukee, Wisconsin, USA	37.04*	13.67	76.29	636,988
	SRR6048908 Puca Glacier, Peru	36.30*	7.76	73.49	280,909
	SRR12473531 Negev Desert, Israel	35.71*	18.06	74.46	639,468
	ERR3192241 Southwest Germany	33.58*	8.80	69.98	288,838

Mapping validation of k-mer hits, from Lumian et al., 10.3389/fmicb.2024.1328083

Diagram 1: Metagenome comparison.

K=31, DNA. No abundance.



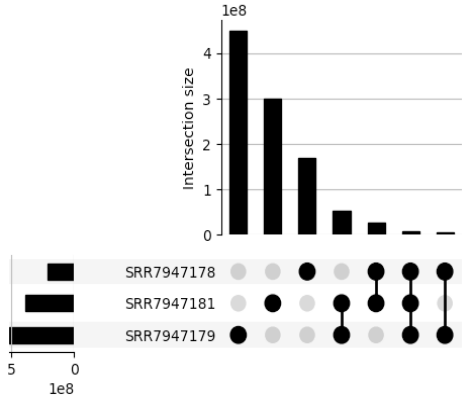
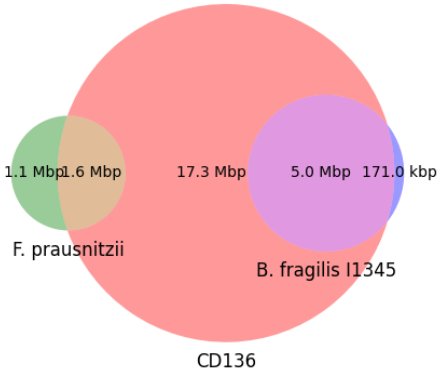


Diagram 2: Genomes and metagenomes

CD136 metagenome subset.

K=31, DNA. *No* abundance.



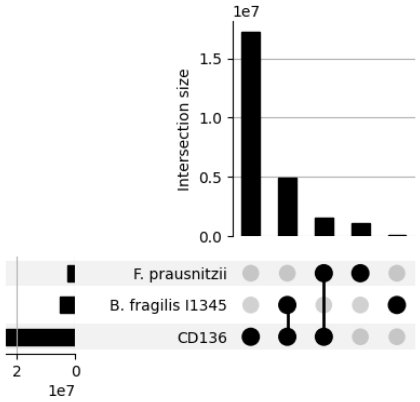
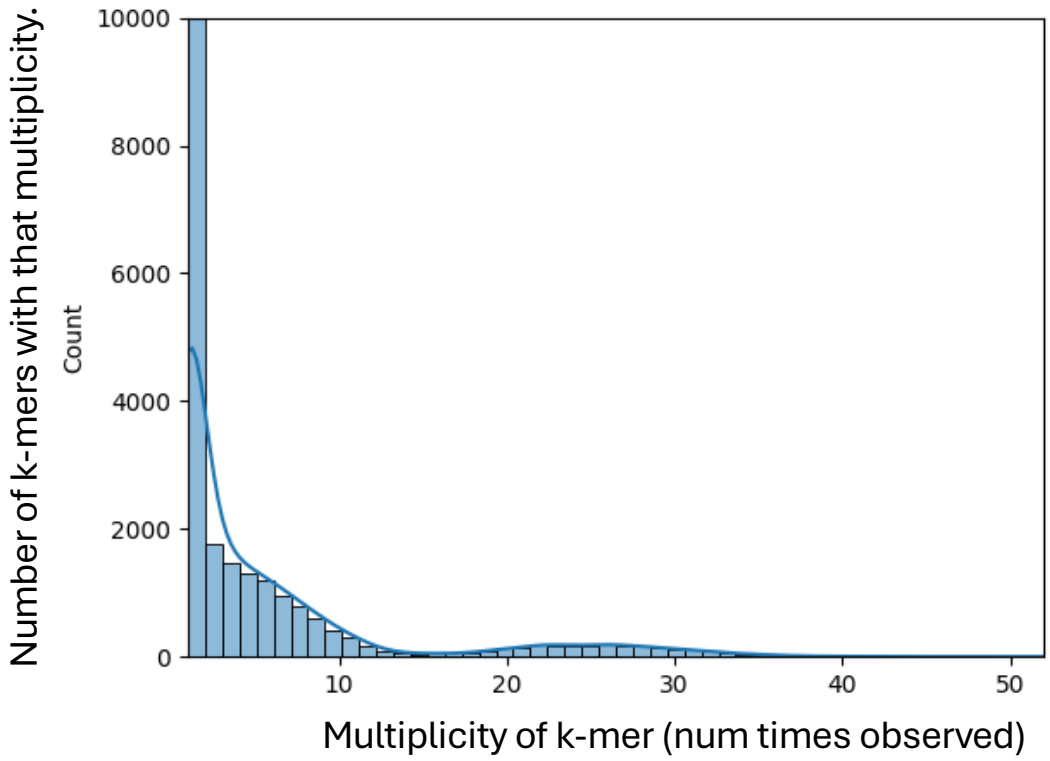


Diagram 3: Abundance histograms of k-mers in metagenomes

CD136 metagenome subset.

K=31, DNA.

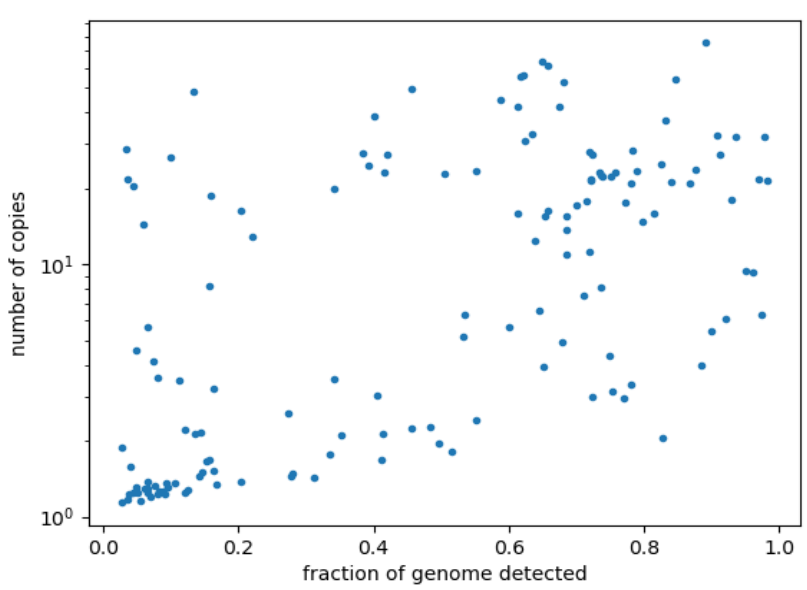
With abundance.



query	p_genome	avg_abund	p_metag	metagenome name
B. fragilis I1345	96.7%	7.3	27.5%	CD136
F. prausnitzii	58.4%	25.3	30.7%	CD136

Diagram 4: Which genomes are present in this metagenome?

SRR7947178
K=31, DNA. Abund.
All points are robustly observed under a naïve null model.



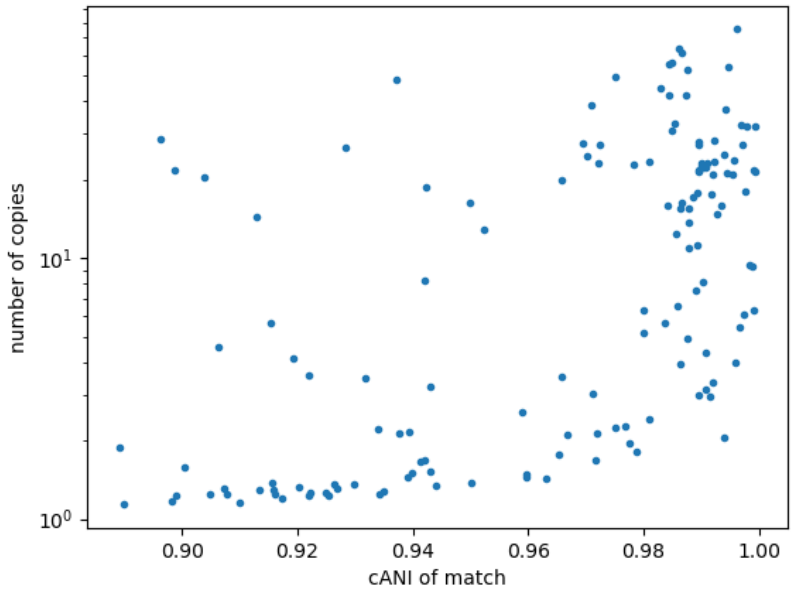
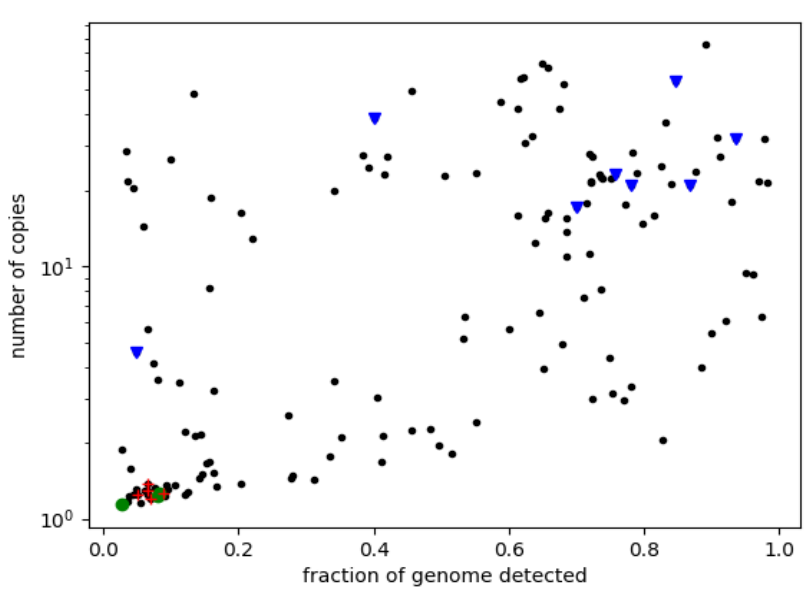
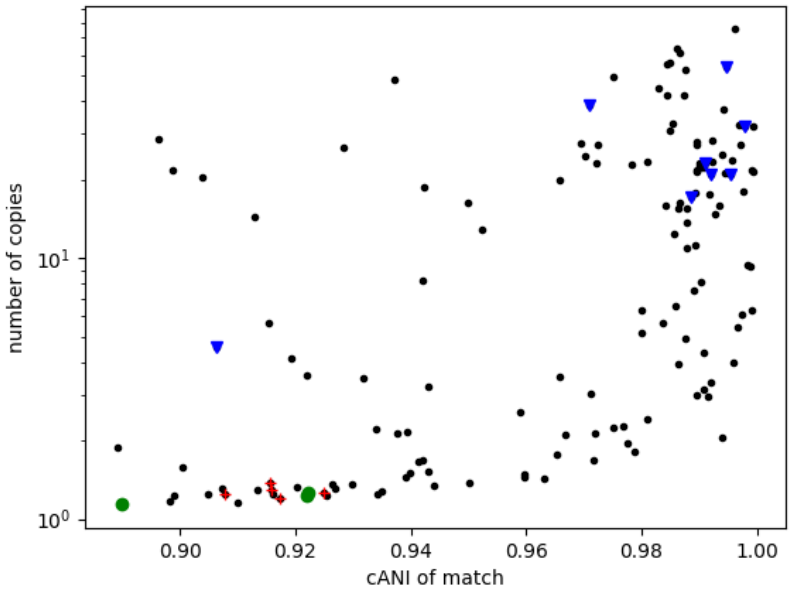


Diagram 4: Which genomes are present in this metagenome?

SRR7947178
K=31, DNA. Abund.
All points are robustly observed under a naïve null model.





Blue triangles: *Ruminococcus* Red crosses: *Bacteroides* Green circles: *Alistipes*

MAG	SRA accession number and location	K-mer containment (%)	Effective coverage	Percentage of MAG detected in metagenome (%)	Number of mapped reads from MAG
Microcoleus sp. MP8IB2.171	SRR5468150 Mat lift-off from Lake Fryxell, Antarctica	99.18*	125.84	99.35	5,864,248
	SRR6266358 Polar Desert Sand Communities, Antarctica	65.02*	93.34	88.34	3,832,909
	SRR5855414 Moab Green Butte, Utah, USA	57.50*	407.19	86.11	15,915,624
	SRR2952554 Ningxia, China	41.65*	18.83	73.53	899,792
	SRR5247052 Sonoran Desert, Colorado Plateau, USA	41.10*	180.87	73.08	10,101,904
	ERR3588763 Pig Farm, UK	40.61*	9.38	76.14	329,215
	SRR5891573 Glacier Snow, China	39.54*	14.36	75.66	482,590
	ERR1333181 Mine Tailing Pool Sediment near Shaoyang, China	38.36*	28.59	73.24	1,120,980
	SRR5459769 Wastewater in Milwaukee, Wisconsin, USA	37.04*	13.67	76.29	636,988
	SRR6048908 Puca Glacier, Peru	36.30*	7.76	73.49	280,909
	SRR12473531 Negev Desert, Israel	35.71*	18.06	74.46	639,468
	ERR3192241 Southwest Germany	33.58*	8.80	69.98	288,838

Mapping validation of k-mer hits, from Lumian et al., 10.3389/fmicb.2024.1328083

We calculated the presence/absence of MAGs across sites by utilising MinHash sketching techniques implemented with sourmash (102). Specifically, we created sourmash signatures of all MAGs and quality filtered metagenomic read sets from all four sites using a k-mer size of 21 and scaled to 1000. These settings allowed us to query for the presence of a MAG within a site with greater sensitivity than by using a higher k-mer or scale value. We selected these metrics to minimize the false negative of not finding a MAG that is present, though we acknowledge this possibly increases the false positive, something recently found by other groups (103). Finally, we included abundance weighting in the signature creation step of the MAG and metagenome signatures so we could measure relative abundances of MAGs present. To query presence/absence, we used sourmash *gather* on each MAG against each metagenome to calculate containment scores. These scores are a proxy for Average Nucleotide Identity where a containment score of 0.2 is equivalent to an ANI of 0.95 for a signature created with a k-mer of 21. We therefore concluded that a MAG was present within a metagenome if its score was greater than 0.2, to account for the varying completeness levels of the MAGs and to again minimize the false negative rate. We measured MAG abundances within each site using the abundance weighted percent match between the MAG and the metagenome. This correlates to the proportion of metagenomic reads that would map to the MAG, a proxy for abundance. The final set of MAGs was deduplicated for all analyses using a 95% ANI cutoff. Where duplicate MAGs were found, the MAG from the site with its highest abundance was retained.

From: doi:10.21203/rs.3.rs-4445835/v1

A paper to read...

YACHT: an ANI-based statistical test to detect microbial presence/absence in a metagenomic sample

David Koslicki [1,2,3,4,†,*], Stephen White[5,†], Chunyu Ma [3,†], Alexei Novikov[5]

[1]Department of Computer Science and Engineering, Pennsylvania State University, State College, PA 16802, United States
[2]Department of Biology, Pennsylvania State University, State College, PA 16802, United States
[3]Huck Institutes of the Life Sciences, Pennsylvania State University, State College, PA 16802, USA
[4]One Health Microbiome Center, Pennsylvania State University, State College, PA 16802, United States
[5]Department of Mathematics, Pennsylvania State University, State College, PA 16802, United States

*Corresponding author. Department of Computer Science and Engineering, Pennsylvania State University, Westgate Building W205C, State College, PA 16802, USA. E-mail: dmk333@psu.edu (D.K.)

[†]These authors contributed equally to the work.

Associate Editor: Alfonso Valencia

Diagram 5: Presence/absence content plots

K=31, DNA.
10kb regions from known *B. fragilis* genomes.
No abund.

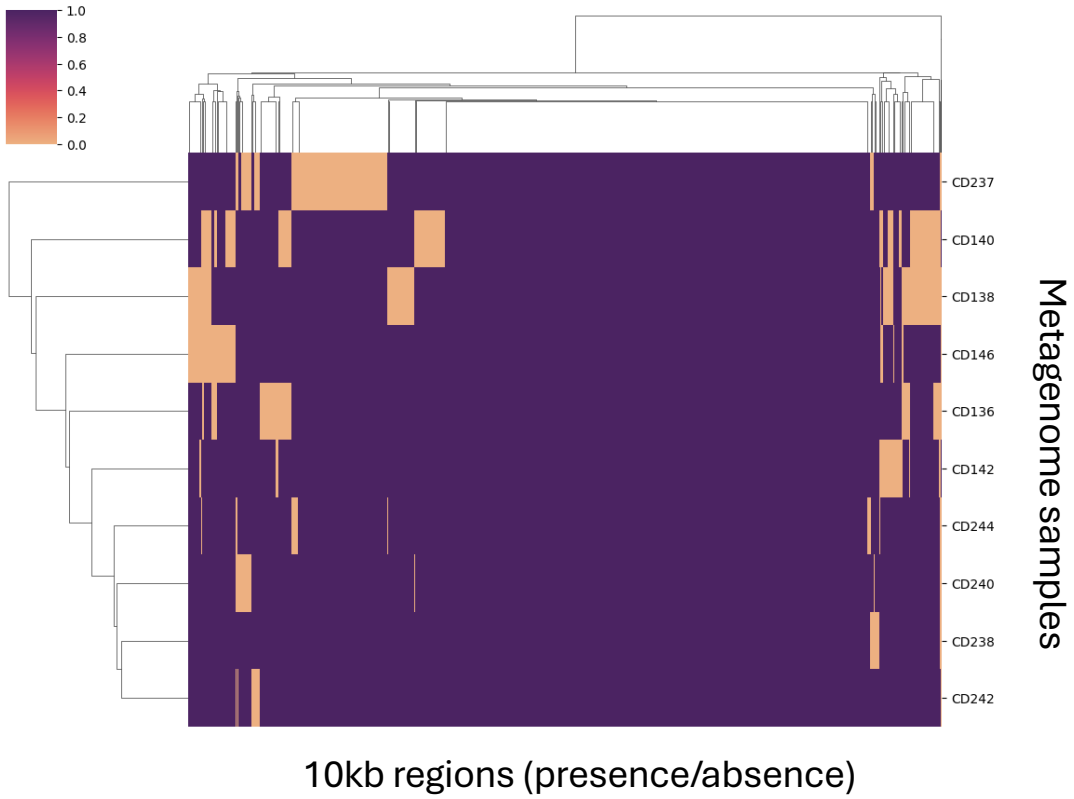
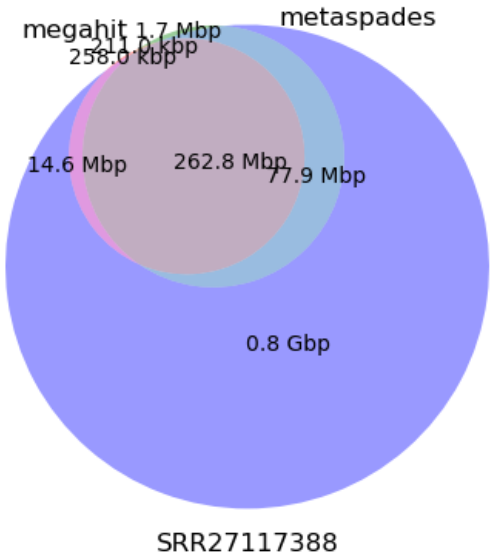
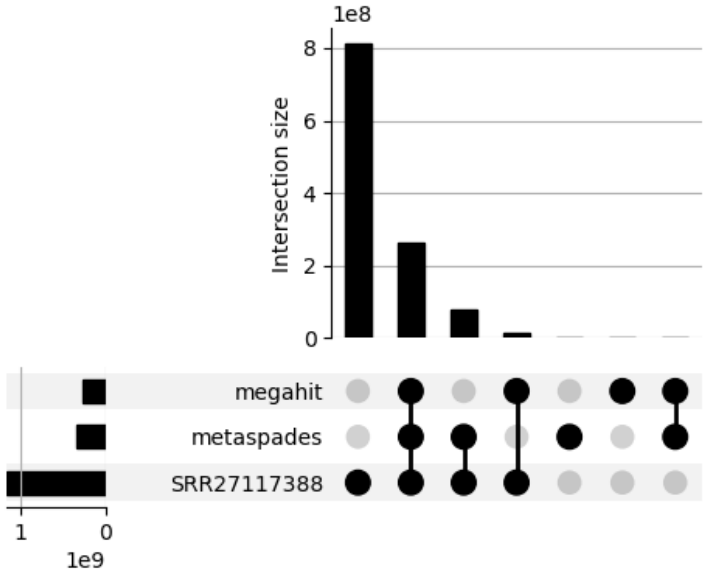


Diagram 6: comparing assembly content



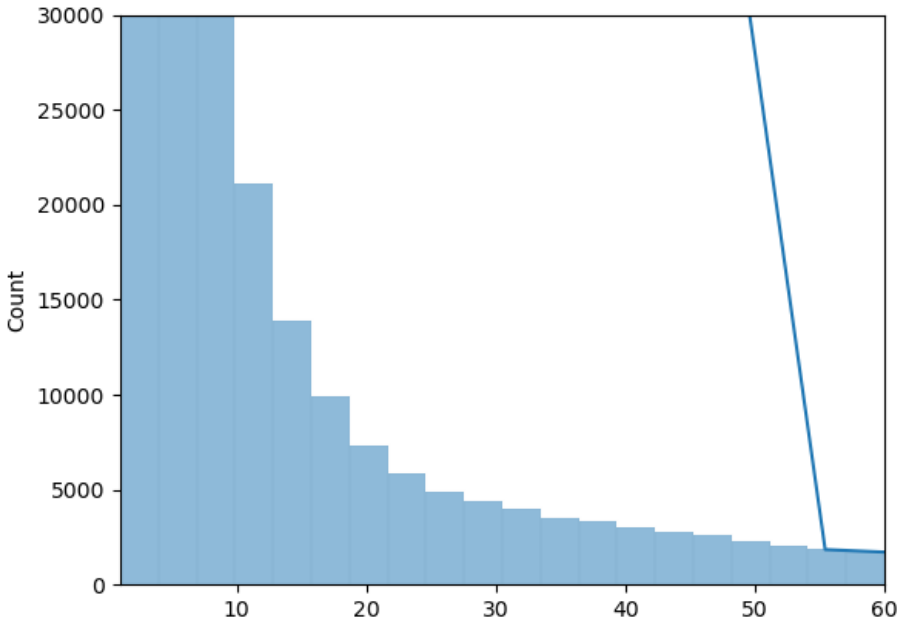


```
overlap     p_query p_match avg_abund
---------   ------- ------- ---------
340.6 Mbp      89.8%   99.5%      31.6    metaspades
277.3 Mbp       1.6%    5.3%      12.8    megahit
found less than 50.0 kbp in common. => exiting

found 2 matches total;
the recovered matches hit 91.4% of the abundance-weighted query.
the recovered matches hit 30.4% of the query k-mers (unweighted).
```

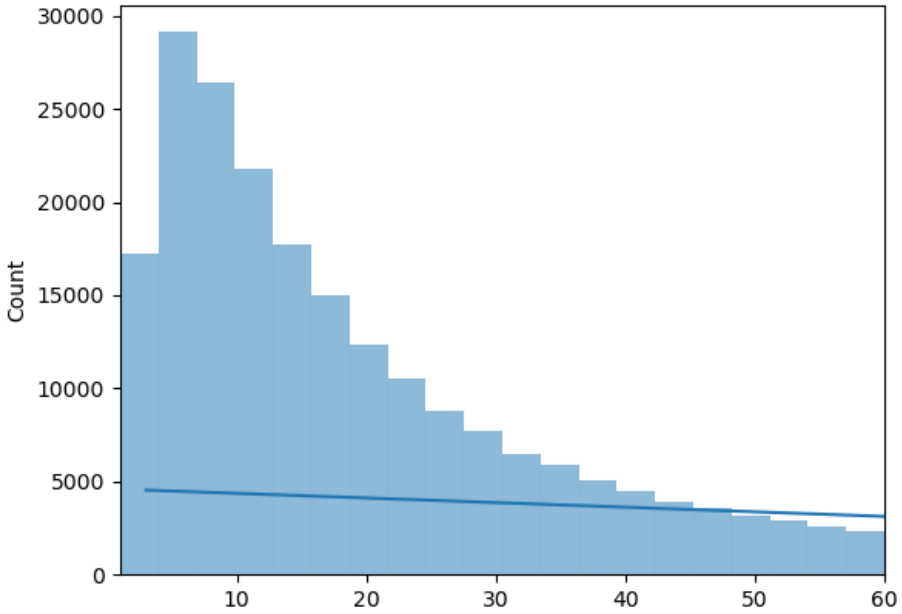
Abundance histogram of SRR27117388

K=31, DNA



Abundance histogram of k-mers assembled by megahit from SRR27117388

K=31, DNA



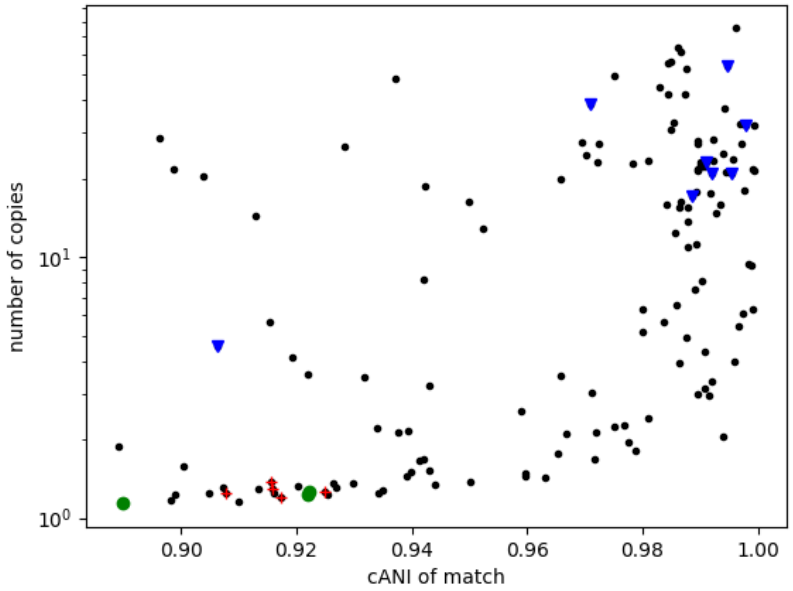
Summary of k-mer diagrams and perspective

- K-mers can be used to investigate genomes and metagenomes – overlap, containment, similarity
- Their measures are different from, but related to, mapping.
- *However*, mapping **requires** a reference, k-mers do not.
- Lots and lots of tools do k-mer analyses; sourmash is just one multitool that Titus likes.

Part (ii): Dude, what's in my metagenome??

- There are 143 genomes from GTDB rs214 detected (via sourmash gather) in SRR7947178.

K=31, DNA. Abund.



Blue triangles: *Ruminococcus*

Red crosses: *Bacteroides*

Green circles: *Alistipes*

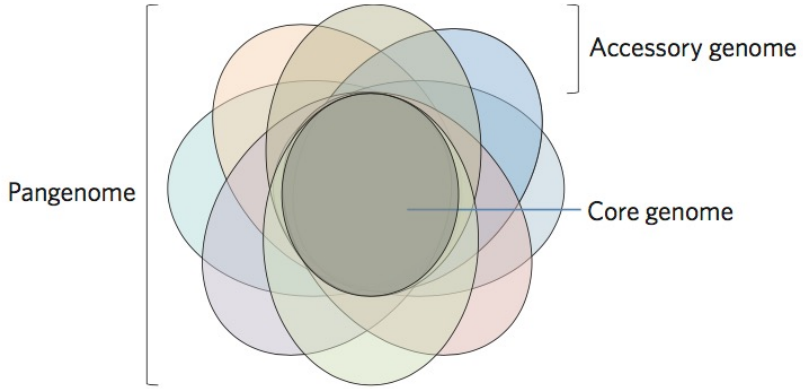
My (our!) goal is strain resolved metagenomics.

- We want to know ***which genomes are in my metagenome?***
- Our solution should be capable of working at large scale - ~million genomes.
- Our solution should *also* deal with the problem of having highly overlapping genomes in the databases.

Conceptual challenge: genomes with shared content.

Many genomes share substantial content with other genomes – most especially, because of ***species pangenomes.***

How do you assign metagenome content in this case??



McInerny et al., DOI: 10.1038/nmicrobiol.2017.40

Strain resolved metagenomics is our goal, but also very challenging!

- Different strains of the same microbial species may have very different function – e.g. *E. coli* can be harmless, or can be pathogenic, due to differences in strain content.
- Metagenomes always contain *mixtures* of strains, in practice.
- There will inevitably be very large databases of strain information, and there is no good way to condense them!
- Strains always have significant overlap with other strains of the same species!

Our genome databases are *massively* redundant!

e.g. for zymo mock community, SRR12324253, there are many overlapping genomes – but most are redundant!

Overlap with SRR12324253

Number of genomes >= 100kb overlap	species name
15964	Staphylococcus aureus
30158	Listeria monocytogenes
75036	Escherichia coli
258360	Salmonella enterica

(using genbank - 700,000 genomes)

I’m going to *slightly* change the question…

From:

What *genomes* are in my *metagenome?*

To:

What is the *shortest list* of genomes that covers *all* of the *known* parts of my metagenome?

And *this* turns out to be a well known CS problem - the “min set cover” problem.

By analogy – what’s the smallest set of additional circles that you need to cover the blue dots (== “known” k-mers)?

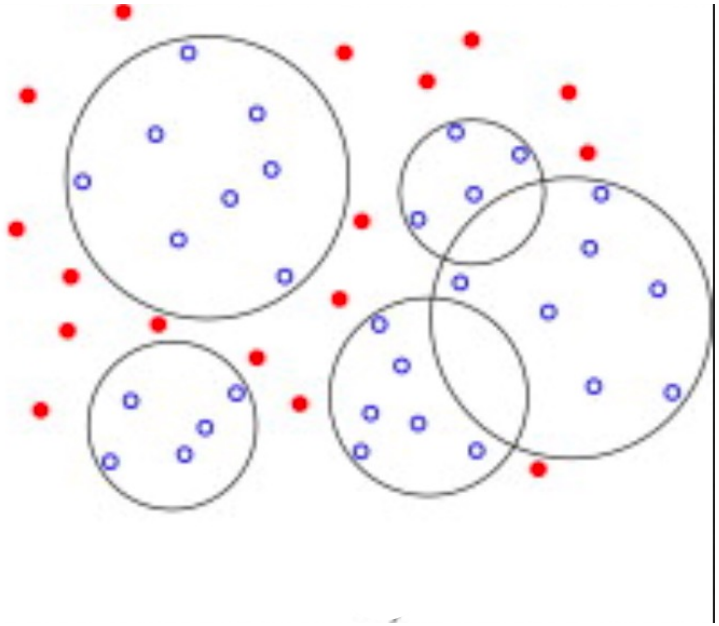


Figure from Bereg et al., 2012 doi.org/10.1016/j.comgeo.2012.01.014

min-set-cov has a straightforward “best approximate” solution!

Find circle that contains the most points; remove those points; repeat.

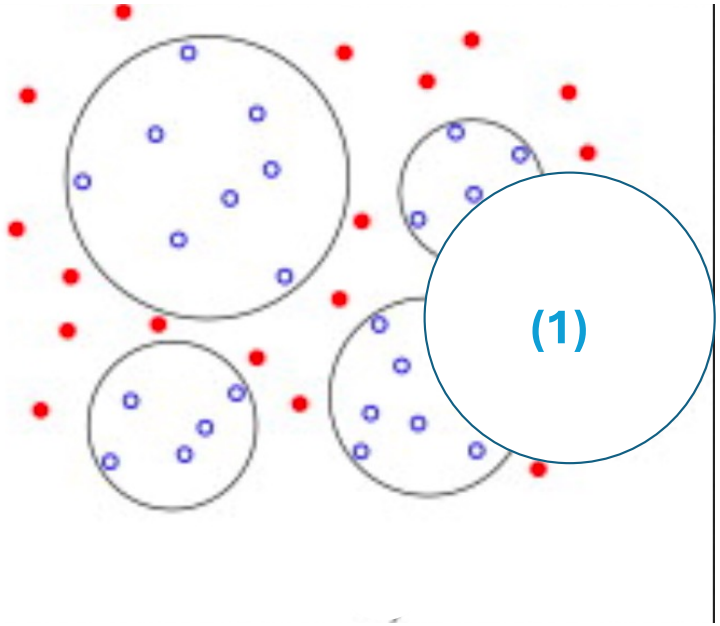


Figure from Bereg et al., 2012 doi.org/10.1016/j.comgeo.2012.01.014

min-set-cov has a straightforward “best approximate” solution!

Find circle that contains the most points; remove those points; repeat.

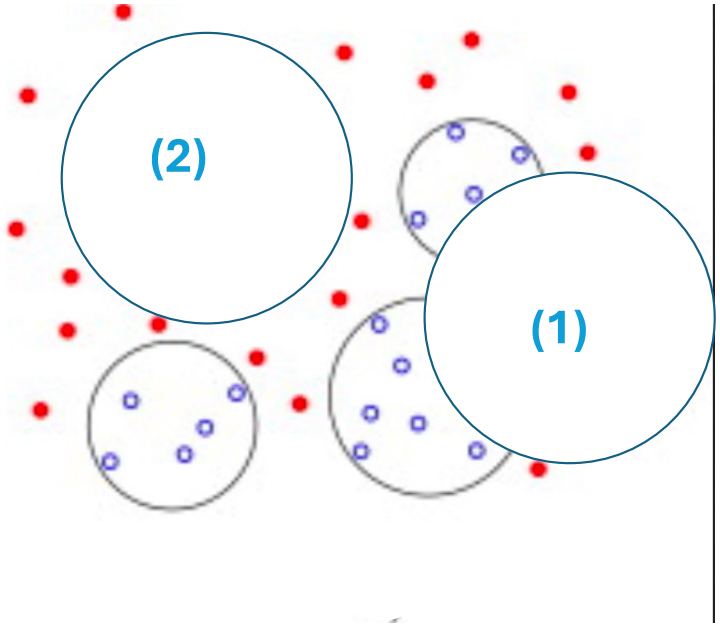


Figure from Bereg et al., 2012 doi.org/10.1016/j.comgeo.2012.01.014

-- and min-set-cov has a straightforward "best approximate" solution!

Find circle that contains the most points; remove those points; repeat.

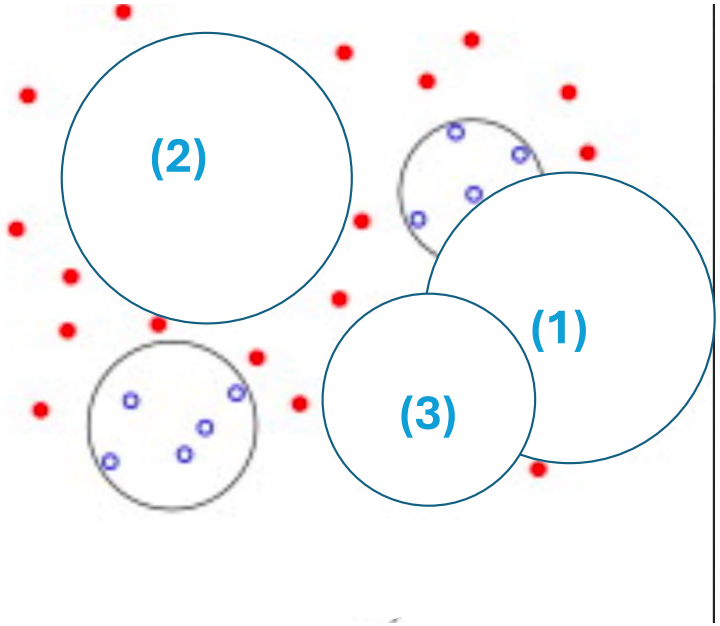


Figure from Bereg et al., 2012 doi.org/10.1016/j.comgeo.2012.01.014

-- and min-set-cov has a straightforward "best approximate" solution!

Find circle that contains the most points; remove those points; repeat.

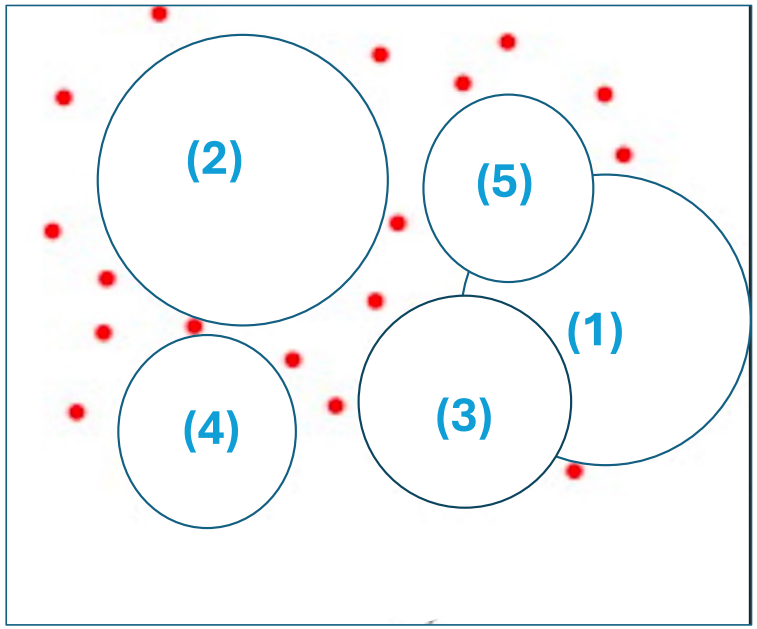
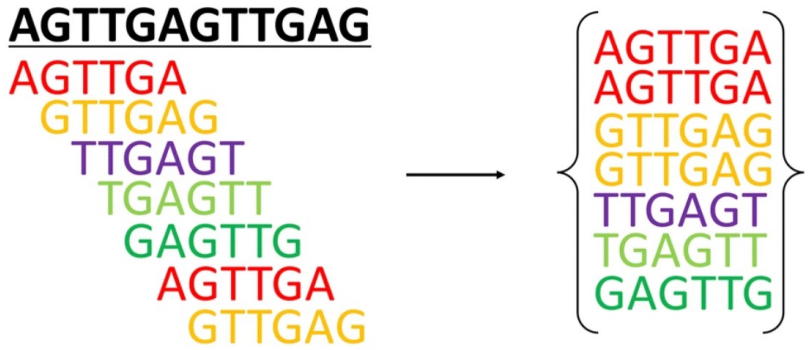


Figure from Bereg et al., 2012 doi.org/10.1016/j.comgeo.2012.01.014

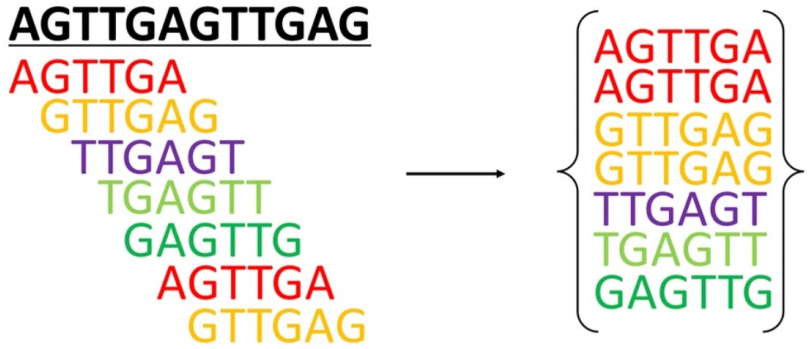
We can implement the min-set-cov algorithm for genomes using k-mers.



(Typically we choose k to be unique at the strain or species level, so k=31 or k=51.)

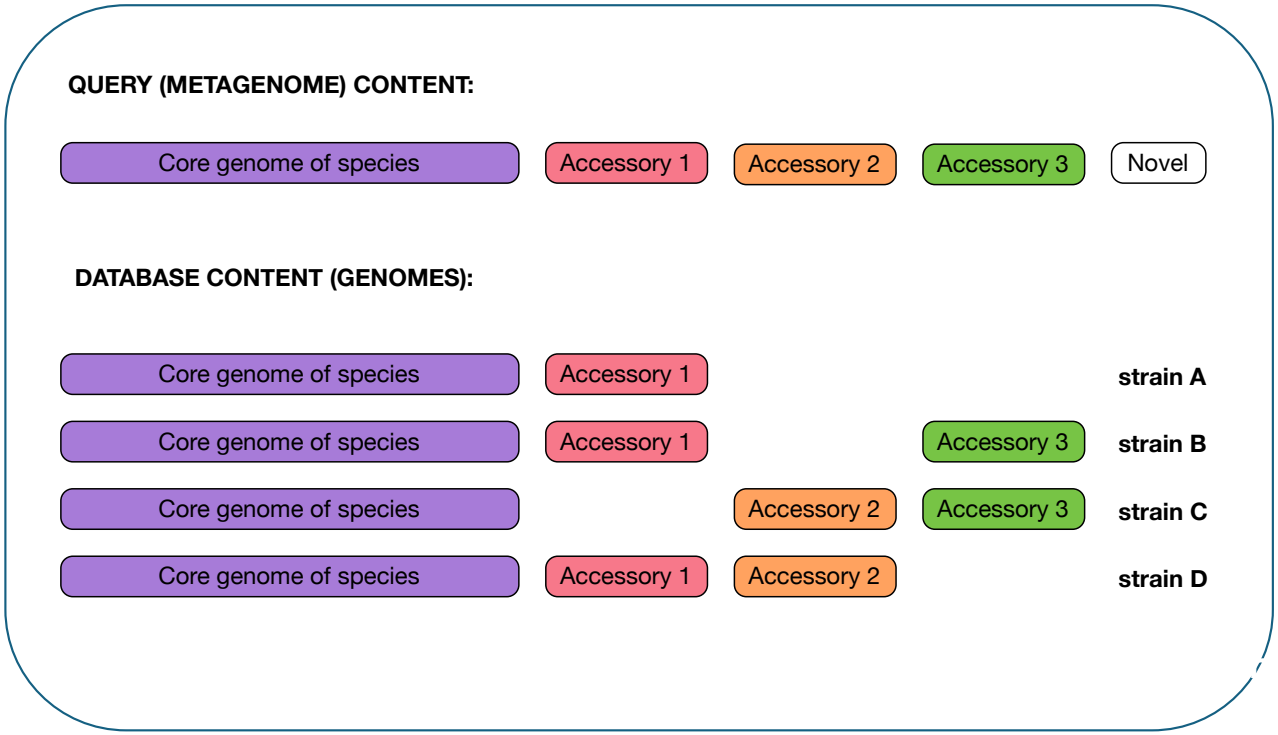
Figure from Hua and Zhang, *BMC Genomics* **volume 20.**

We can implement the min-set-cov algorithm for genomes using k-mers.

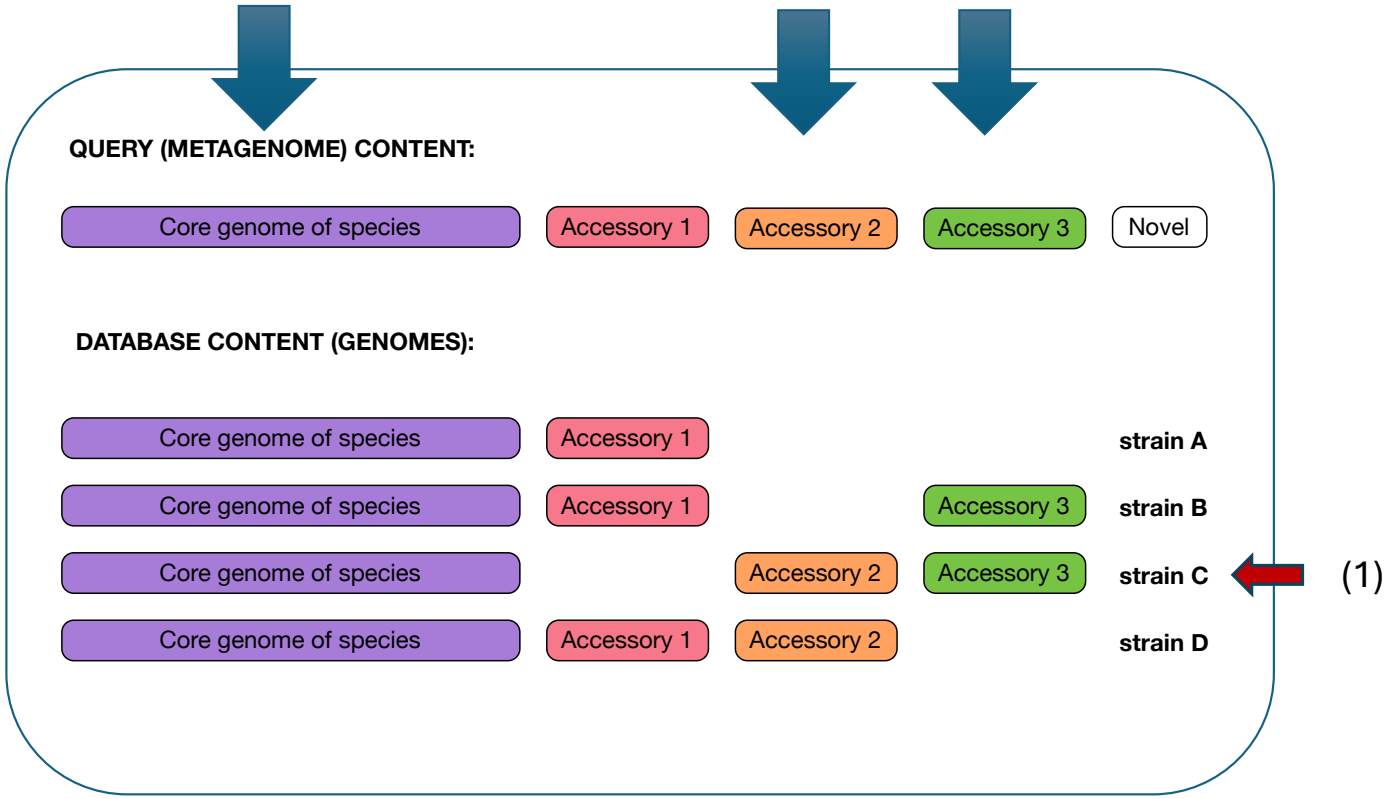


Reframed: what is the smallest collection of genomes { G } in database D such that all of the k-mers shared between G and D are in the collection { G }?

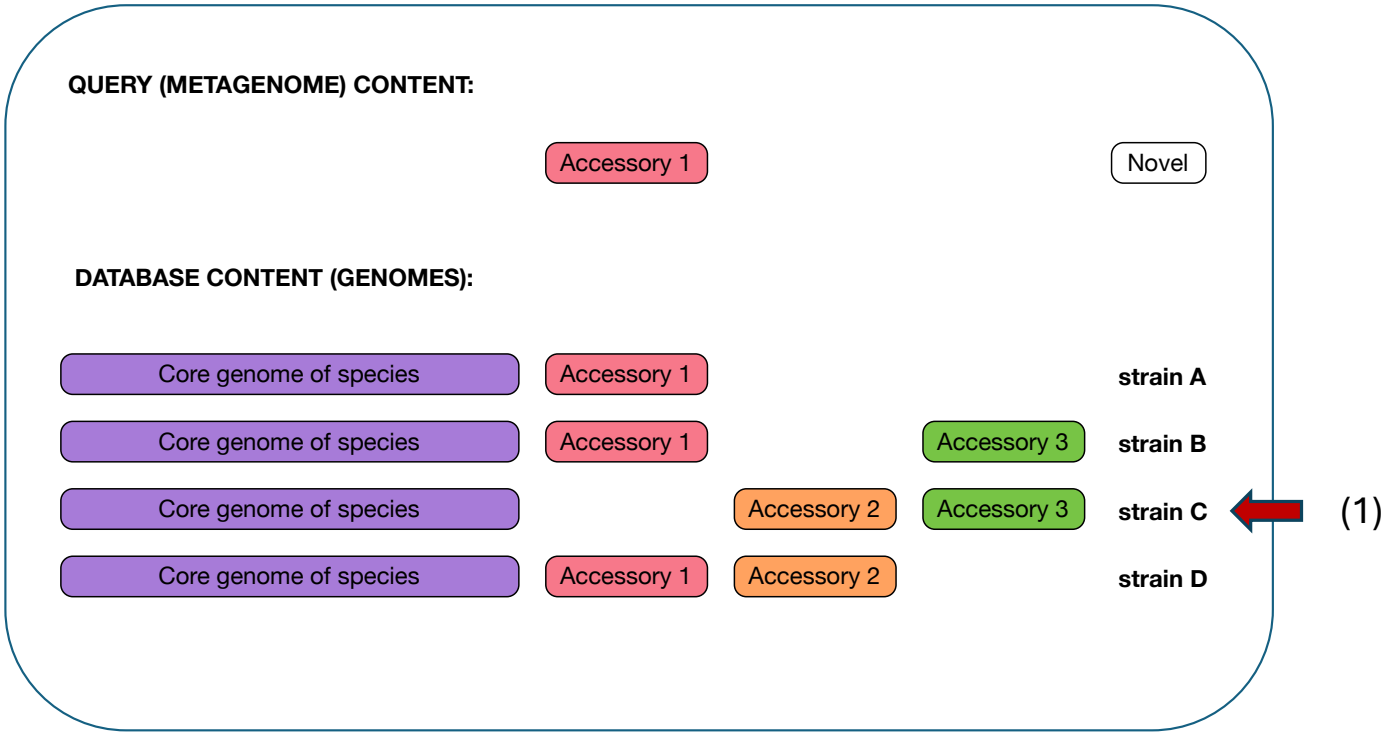
Strain-level resolution for queries with k-mers – what's the smallest set of matches?



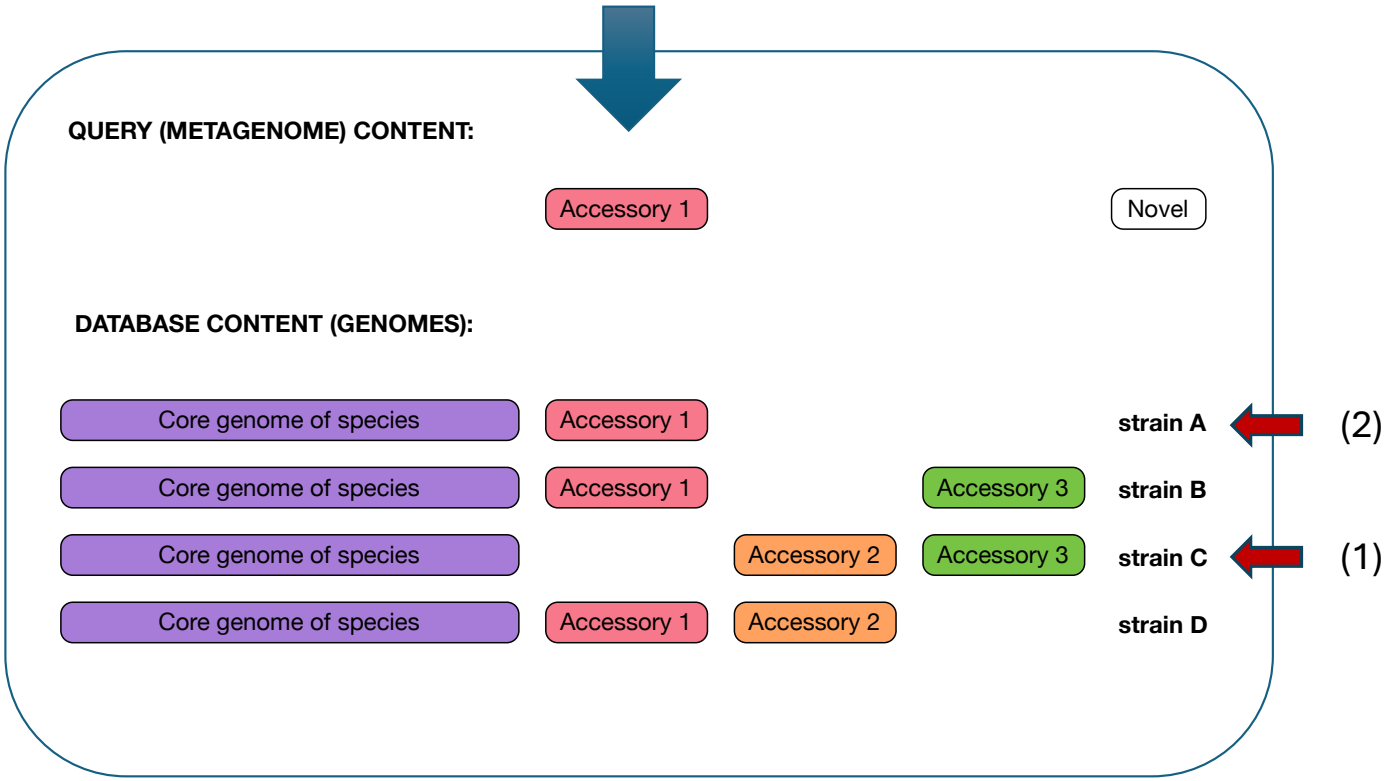
Step 1 – find a “best” match…



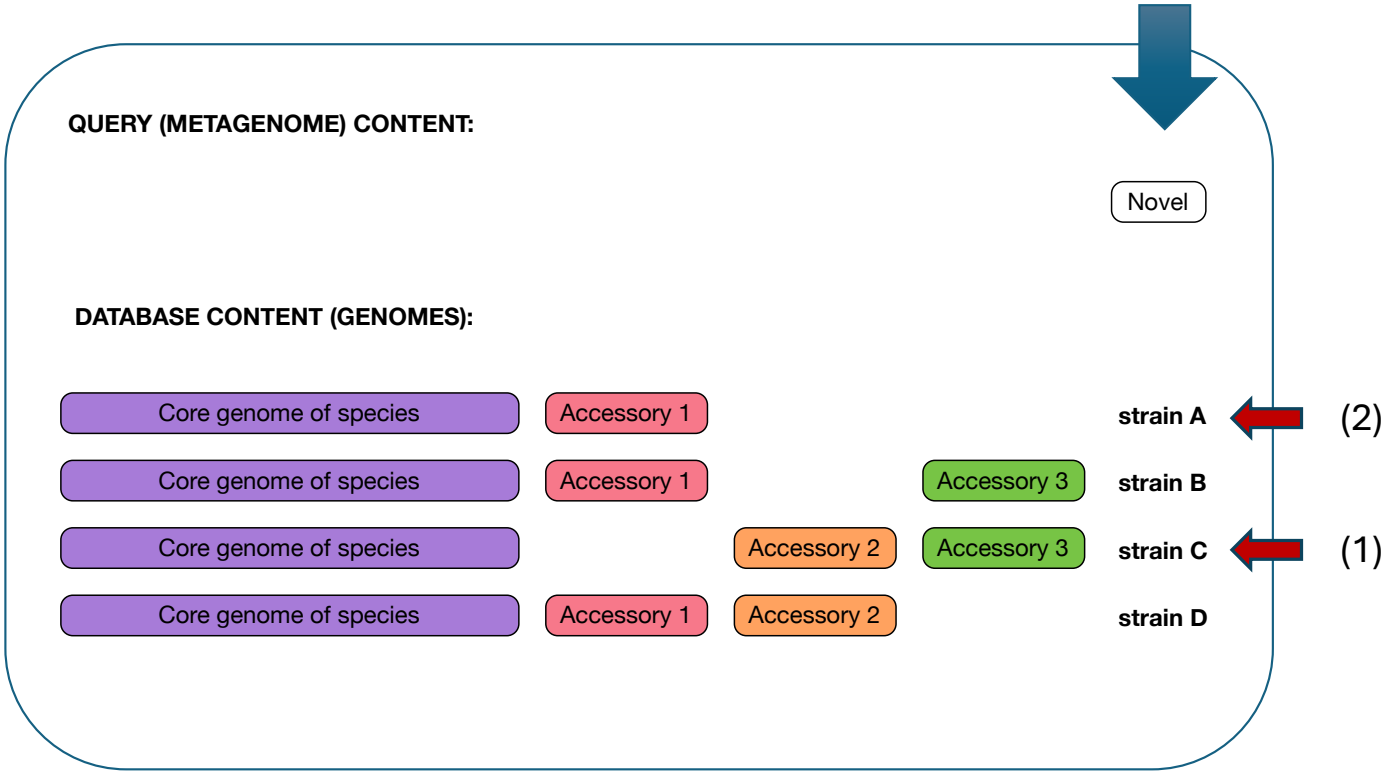
Step 2 – remove best match from query!



Step 3 – find next best match, remove.



...& continue until you run out of matches.



Using k-mers and min-set cov to make a list of genomes in a metagenome.

We frame the challenge of finding the relevant set of genomes as a min-set-cover problem:

What is the ***smallest*** collection of genomes { G } in database D such that all of the k-mers shared between G and D are in the collection { G }?

There is a simple & efficient polynomial-time solution!

Our genome databases are *massively* redundant!

e.g. for zymo mock community, SRR12324253, there are many overlapping genomes – but most are redundant!

Overlap with SRR12324253

Number of genomes >= 100kb overlap	species name
15964	Staphylococcus aureus
30158	Listeria monocytogenes
75036	Escherichia coli
258360	Salmonella enterica

(using genbank - 700,000 genomes)

min-set-cov dramatically reduces the list of relevant genomes

Which genomes have significant overlap, vs which genomes are comprise the *minimal* list of genomes that cover everything?

data set	genomes >= 100kb overlap	min-set-cov
zymo mock (SRR12324253)	405,839	19
podar mock (SRR606249)	5800	74
p8808mo11 (iHMP)	96,423	99
hu-s1 oil well (SRR1976948)	1235	135

(using genbank - 700,000 genomes)

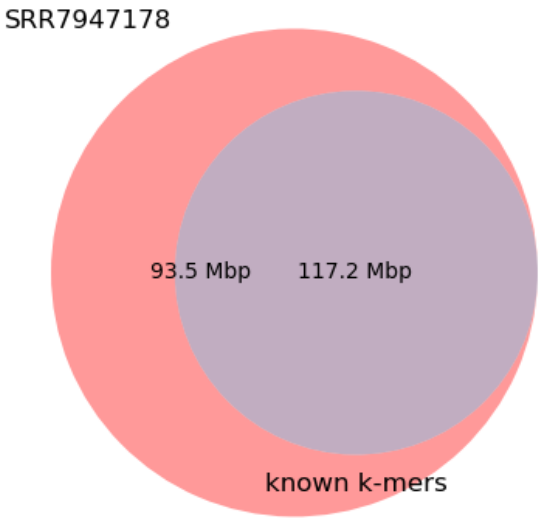
What does this look like in practice??

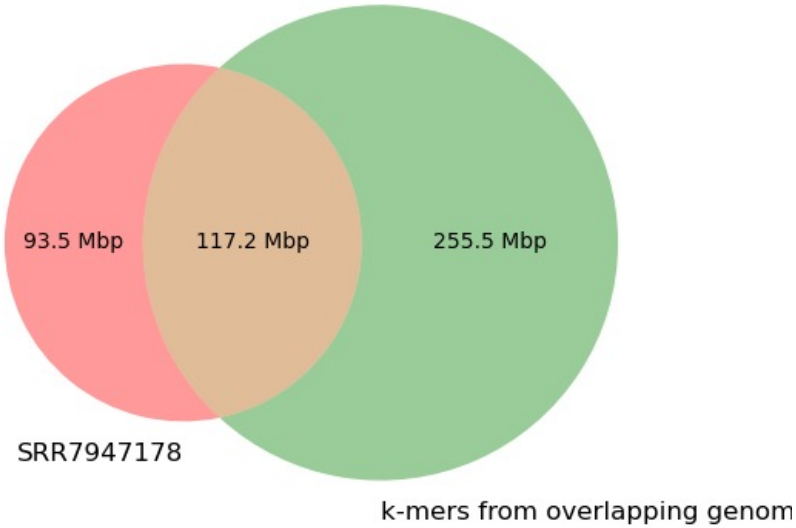
overlap	p_query	p_match	avg_abund	
6.1 Mbp	5.6%	97.0%	21.7	GCF_020540665.1 Blautia producta str...
5.6 Mbp	4.1%	89.0%	17.9	GCF_000765245.1 Blautia producta str...
5.3 Mbp	2.1%	96.2%	9.3	GCF_903935665.1 Klebsiella pneumonia...
4.9 Mbp	1.3%	96.5%	6.3	GCF_020732745.1 Escherichia coli str...
5.0 Mbp	1.9%	89.8%	9.4	GCF_018420935.1 Klebsiella aerogenes...
4.6 Mbp	2.2%	71.7%	11.3	GCF_020559295.1 Enterocloster boltea...
4.6 Mbp	6.3%	90.2%	32.3	GCF_024463855.1 Clostridium sp. DFI....
4.4 Mbp	11.7%	64.5%	63.7	GCA_000466465.2 Clostridium sp. KLE ...
3.7 Mbp	0.6%	87.7%	4.0	GCF_002301735.1 Clostridioides diffi...
3.1 Mbp	0.2%	51.3%	1.8	GCF_902385905.1 Enterocloster aspara...

```
found 143 matches total;
the recovered matches hit 88.4% of the abundance-weighted query.
the recovered matches hit 55.6% of the query k-mers (unweighted).
```

SRR7947178, k=31, against GTDB rs214

Overlaps – known and unknown!





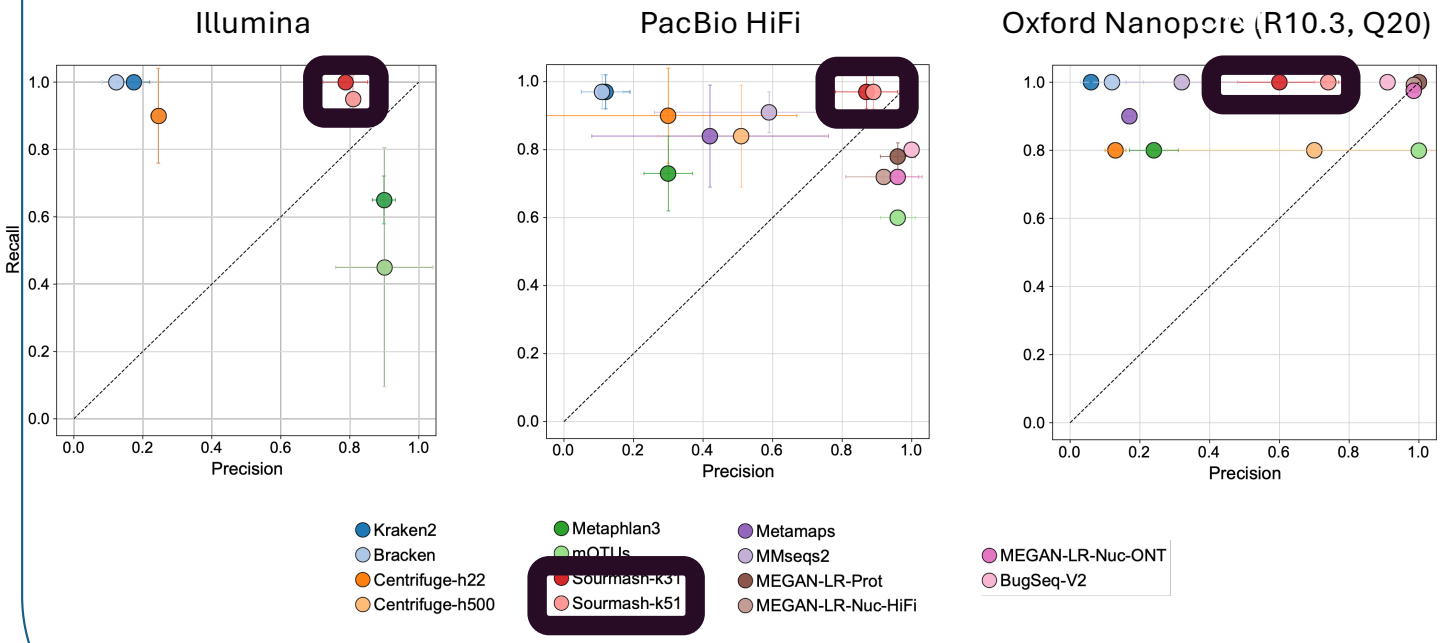
What does this look like in practice??

overlap	p_query	p_match	avg_abund	
6.1 Mbp	5.6%	97.0%	21.7	GCF_020540665.1 Blautia producta str...
5.6 Mbp	4.1%	89.0%	17.9	GCF_000765245.1 Blautia producta str...
5.3 Mbp	2.1%	96.2%	9.3	GCF_903935665.1 Klebsiella pneumonia...
4.9 Mbp	1.3%	96.5%	6.3	GCF_020732745.1 Escherichia coli str...
5.0 Mbp	1.9%	89.8%	9.4	GCF_018420935.1 Klebsiella aerogenes...
4.6 Mbp	2.2%	71.7%	11.3	GCF_020559295.1 Enterocloster boltea...
4.6 Mbp	6.3%	90.2%	32.3	GCF_024463855.1 Clostridium sp. DFI....
4.4 Mbp	11.7%	64.5%	63.7	GCA_000466465.2 Clostridium sp. KLE ...
3.7 Mbp	0.6%	87.7%	4.0	GCF_002301735.1 Clostridioides diffi...
3.1 Mbp	0.2%	51.3%	1.8	GCF_902385905.1 Enterocloster aspara...

```
found 143 matches total;
the recovered matches hit 88.4% of the abundance-weighted query.
the recovered matches hit 55.6% of the query k-mers (unweighted).
```

SRR7947178, k=31, against GTDB rs214

sourmash performs well for species-level taxonomic assignment.

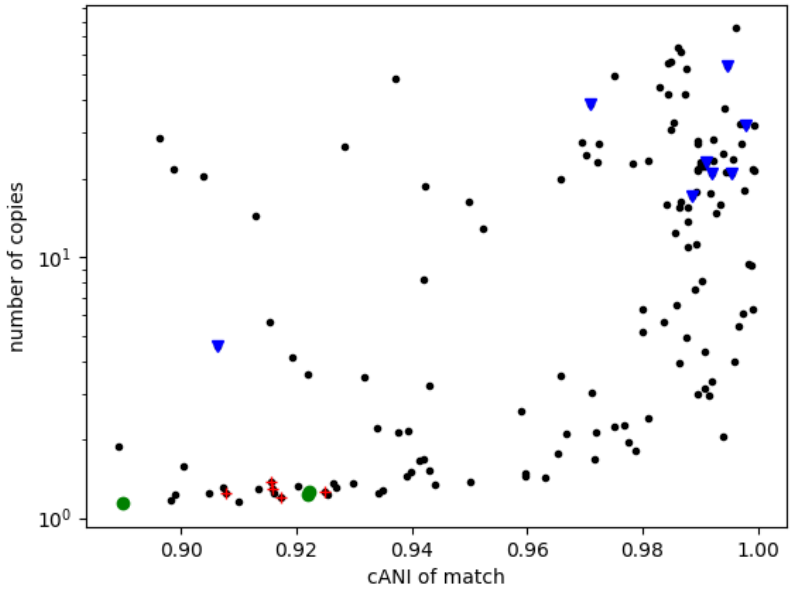


Portik et al., 2022

How do we get to taxonomy??

- There are 143 genomes from GTDB rs214 detected (via sourmash gather) in SRR7947178.

K=31, DNA. Abund.



Blue triangles: *Ruminococcus*

Red crosses: *Bacteroides*

Green circles: *Alistipes*

Sourmash taxonomy results.

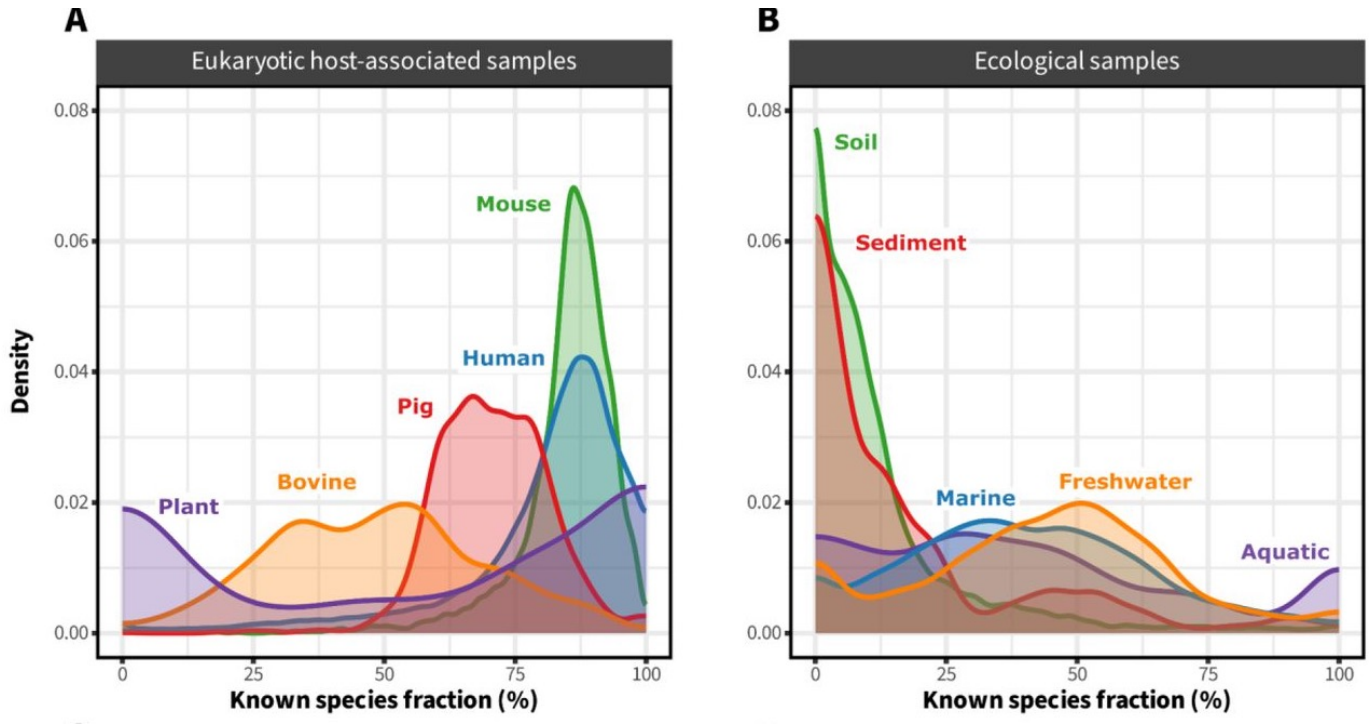
sample name	proportion	cANI	lineage
SRR7947178	71.8%	96.8%	d__Bacteria;p__Bacillota_A;c__Clostridia
SRR7947178	11.6%	-	unclassified
SRR7947178	7.0%	90.3%	d__Bacteria;p__Bacillota;c__Bacilli
SRR7947178	6.1%	93.2%	d__Bacteria;p__Pseudomonadota;c__Gammaproteobacteria
SRR7947178	3.2%	87.5%	d__Bacteria;p__Actinomycetota;c__Coriobacteriia
SRR7947178	0.2%	88.0%	d__Bacteria;p__Bacteroidota;c__Bacteroidia
SRR7947178	0.1%	83.7%	d__Bacteria;p__Bacillota_C;c__Negativicutes

Can use either NCBI or GTDB taxonomy; custom references; etc.

Some thoughts and concluding points

- Some parts of metagenomes remain refractory to analysis against reference genomes and are simply “unknown”.
  - Error? Garbage? Unknown genomes, esp viruses & euks?
- This includes accounting for assembly-based and MAG-based analysis!
- If doing a reference-based analysis, always measure how much of your metagenome is unknown
  - Sourmash does this for you with ’sourmash gather’!
  - It’s a lower bound (i.e. the true number is always greater than what is reported by sourmash)
  - (Use the weighted number.)

The landscape of metagenome classification



(Sandpiper & singleM are fantastic!!)

doi: 10.1101/2024.01.30.578060

Open Lab

- You can run these programs and commands yourself, and be able to generate all the plots!
- It is straightforward to run these on your own metagenomes of interest, as well as with adding your own collections of genomes.