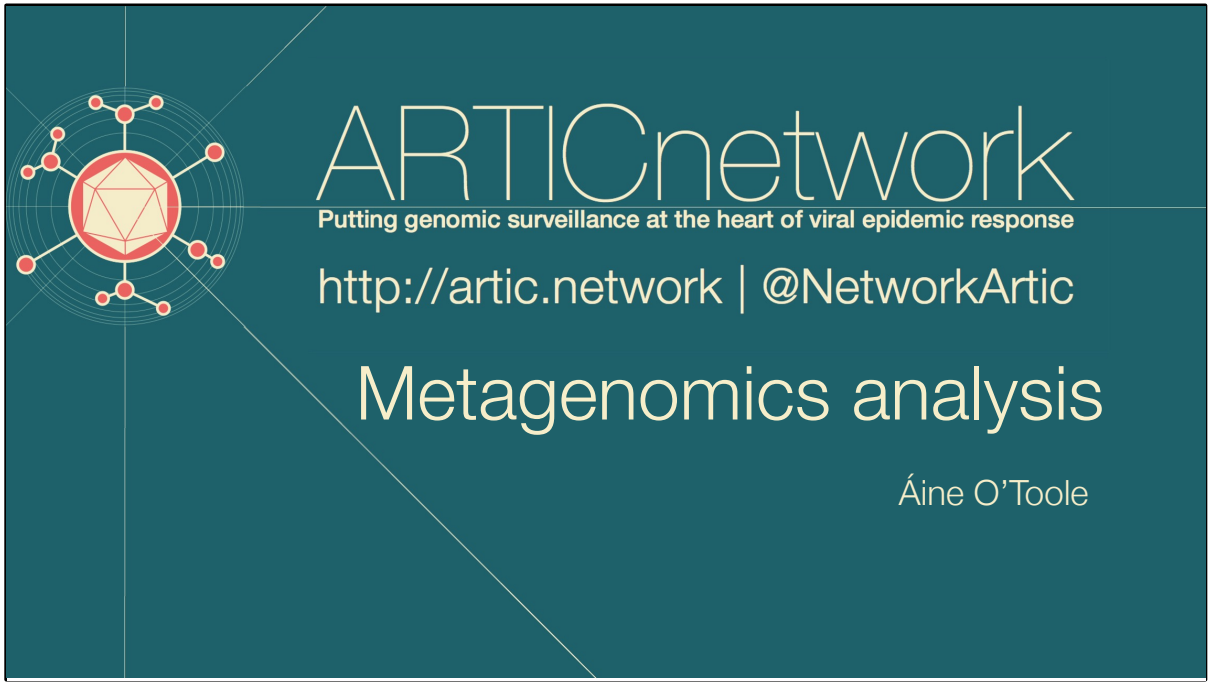
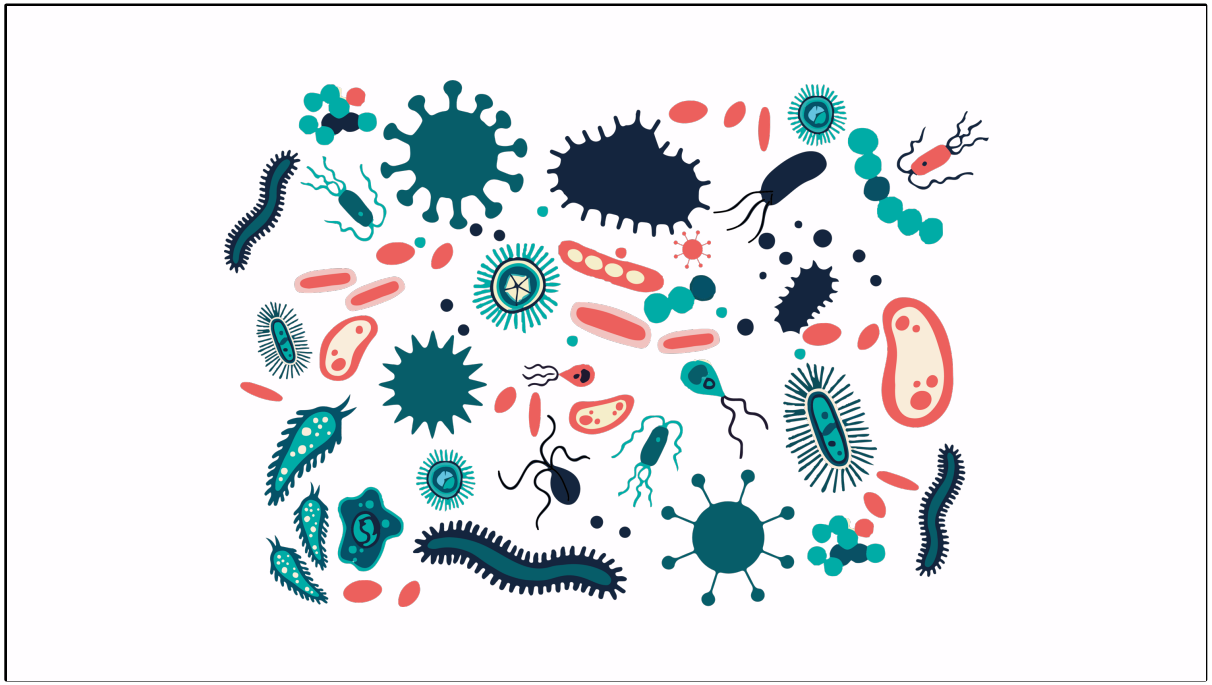
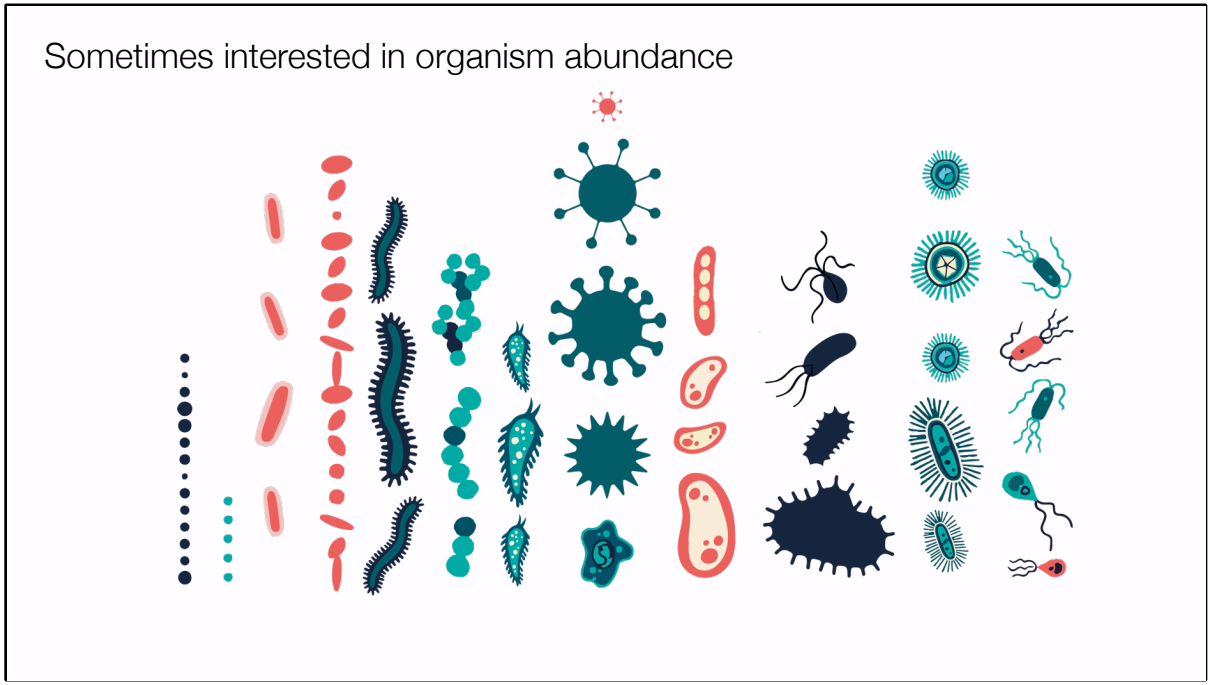


In this session we're going to cover principles of metagenomic analysis

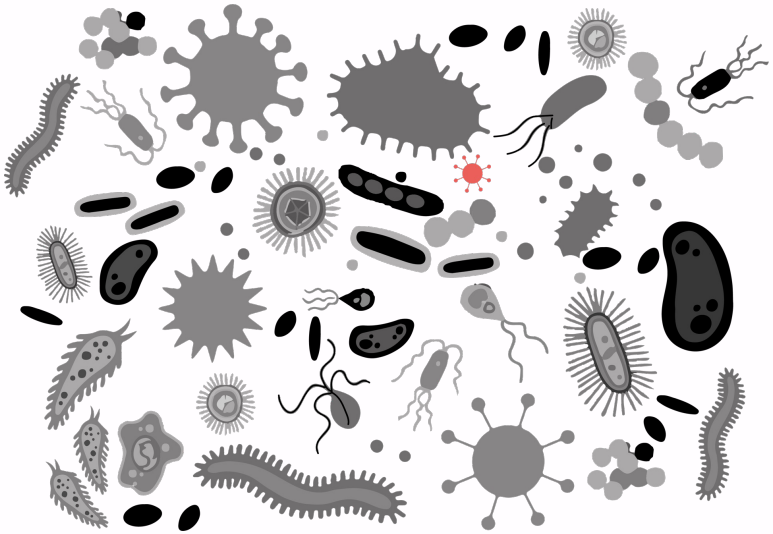


With metagenomic samples you will likely have a range of different organisms and populations within your tube

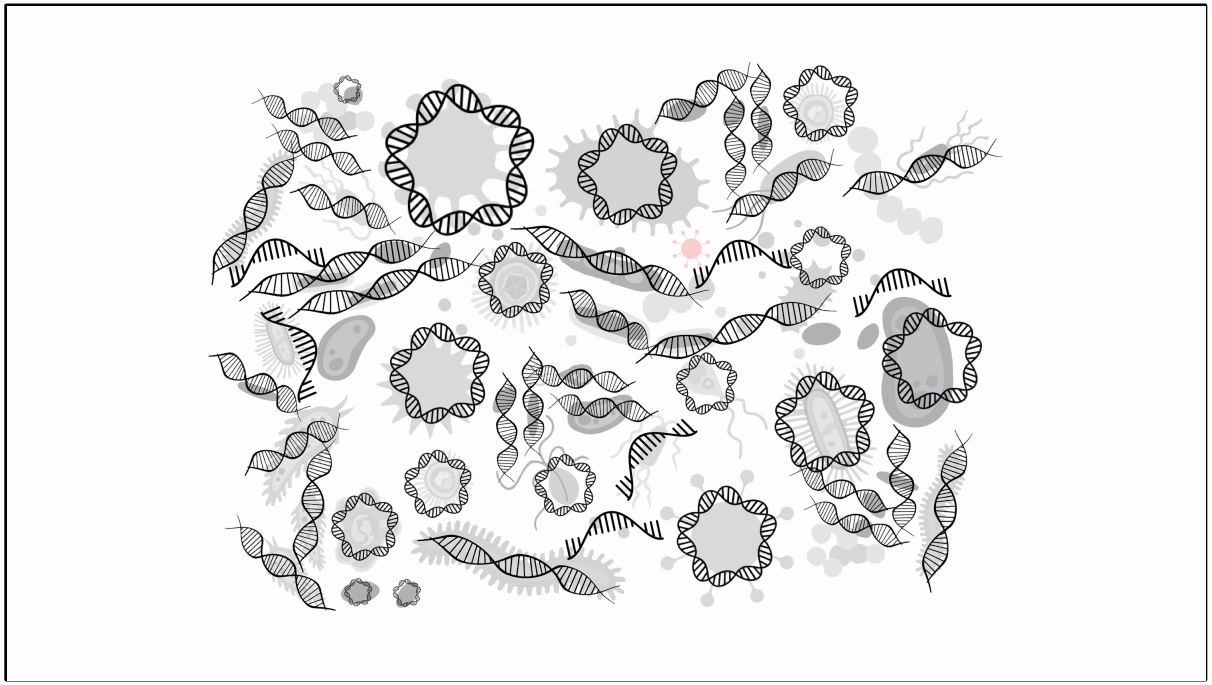


Sometimes you'll want to make sense of a metagenomic sample as a whole: for example in microbiome studies you want to get a picture of the diversity in a sample and an idea of the abundance of everything

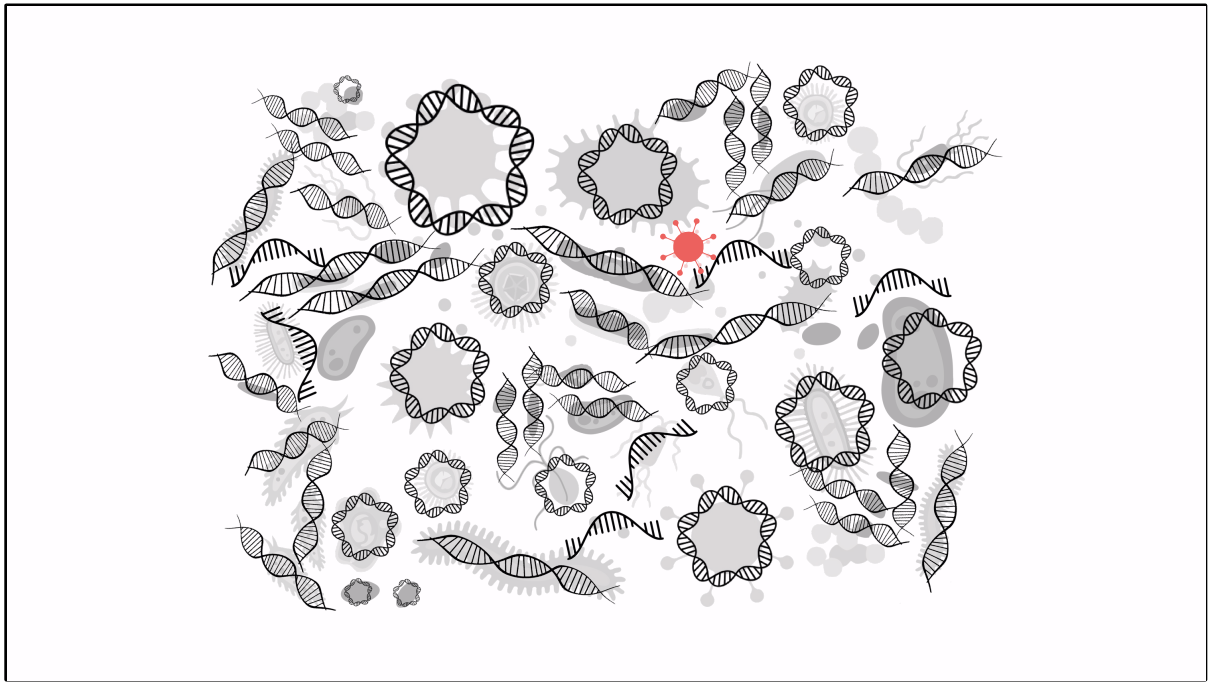
Or may want to identify a specific organism



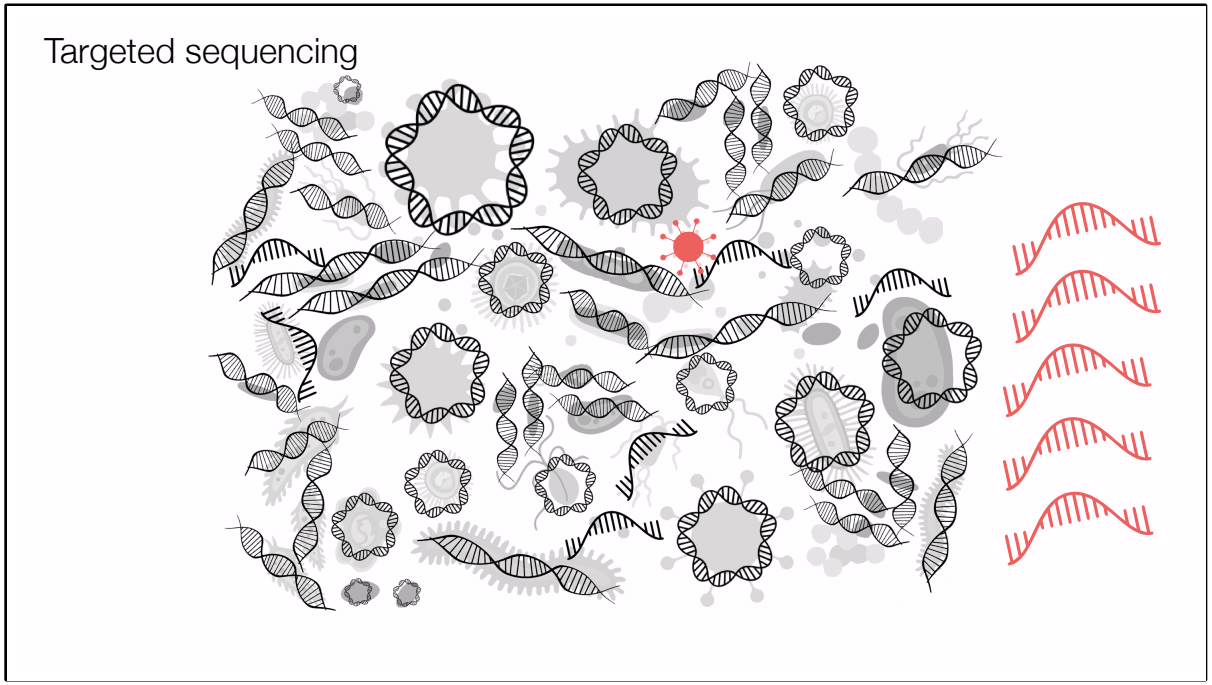
But in other cases you will want to identify the relevant organism, perhaps one responsible for a disease, in your sample



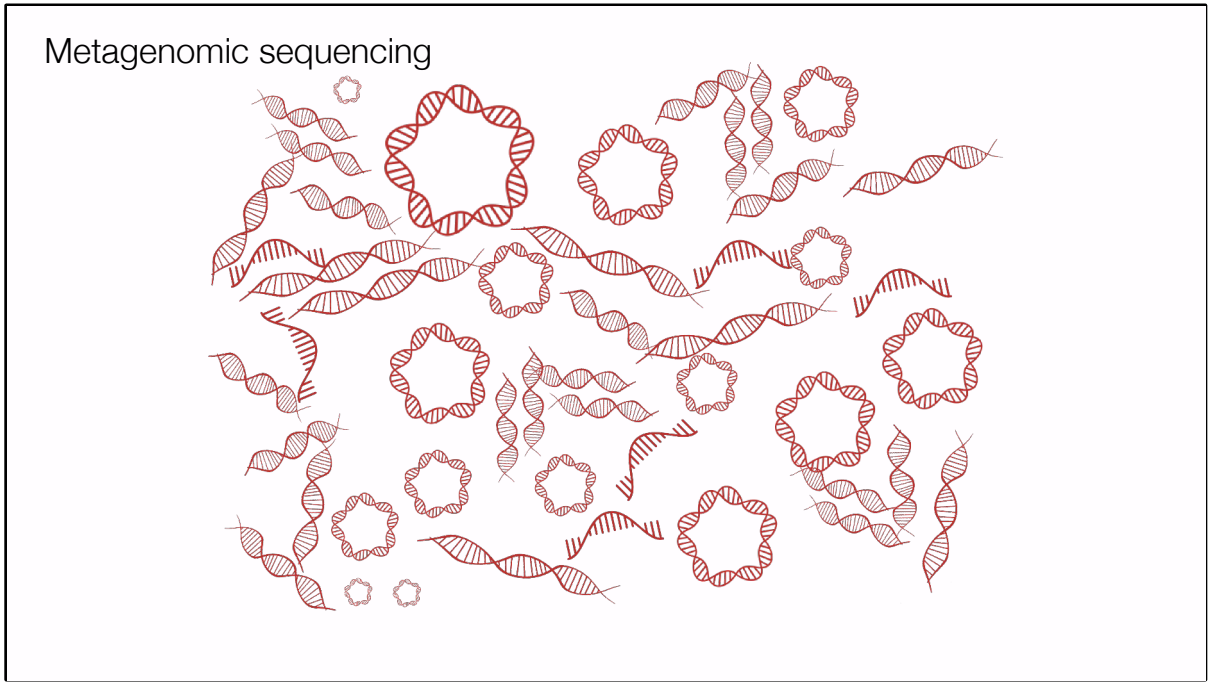
And when you're doing metagenomics, what this means is you'll need to somehow sort through all the nucleotide sequence generated from the diverse population in your tube and identify the relevant sequencing reads



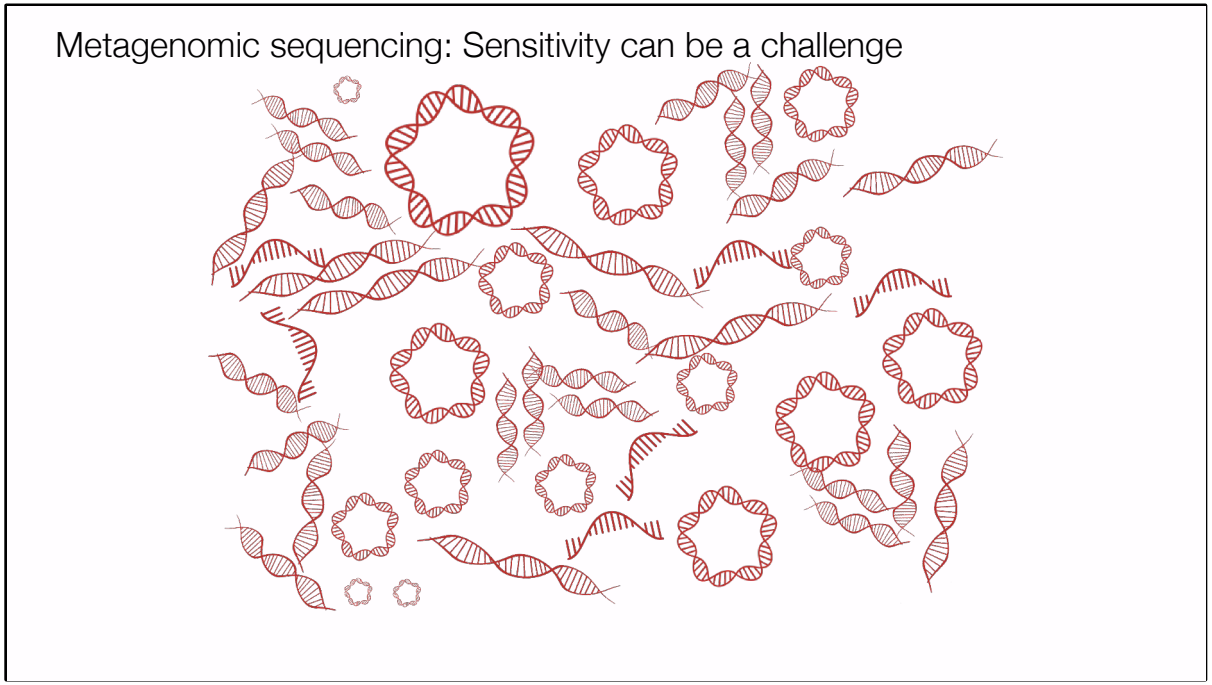
If you know what you're looking for (or have a fairly good idea) you can do targeted sequencing



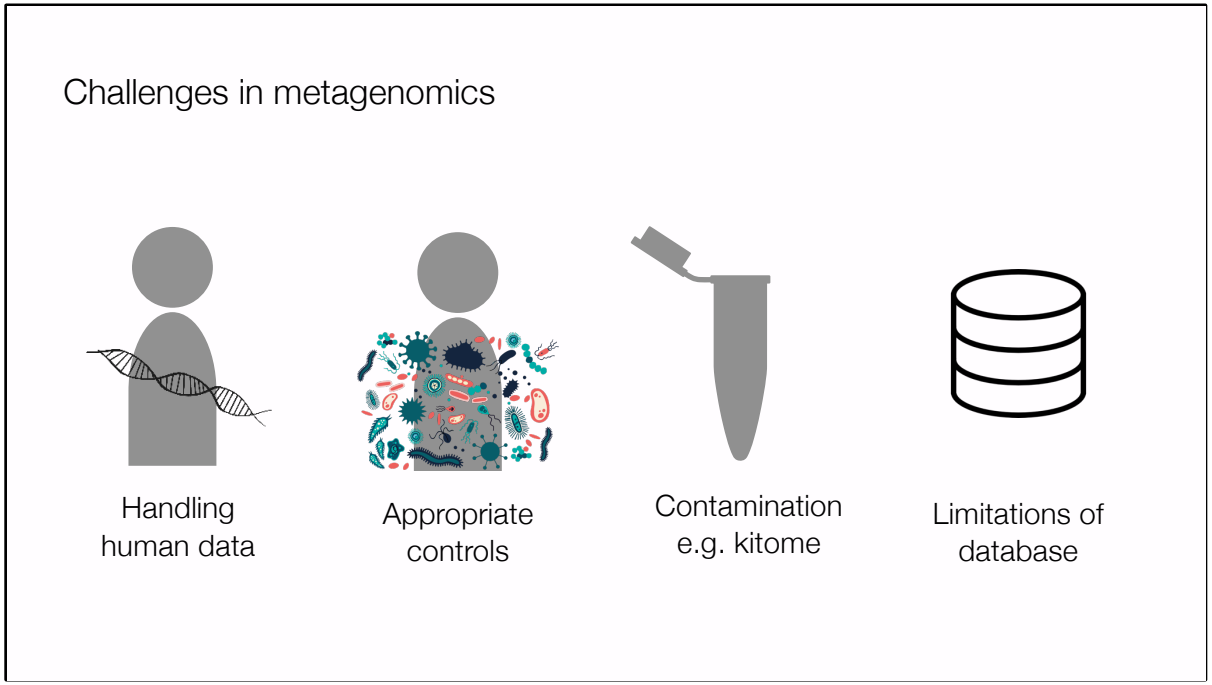
And hopefully just amplify and sequence the relevant genetic material, such as with the artic tiled amplicon schemes. Using that approach, we can specifically target only a portion of the diversity in a sample



But with metagenomic sequencing we're sequencing everything and then somehow need to filter out the noise to get to the actual signal.



This means that if what you're looking for is in low abundance- and has a small genome relative to the rest of the things comprising your sample, this approach can have low sensitivity.



Some other challenges include planning and acquiring appropriate controls for your samples, how you might need to handle sensitive human data, potential contaminants and limitations of the database you're comparing against

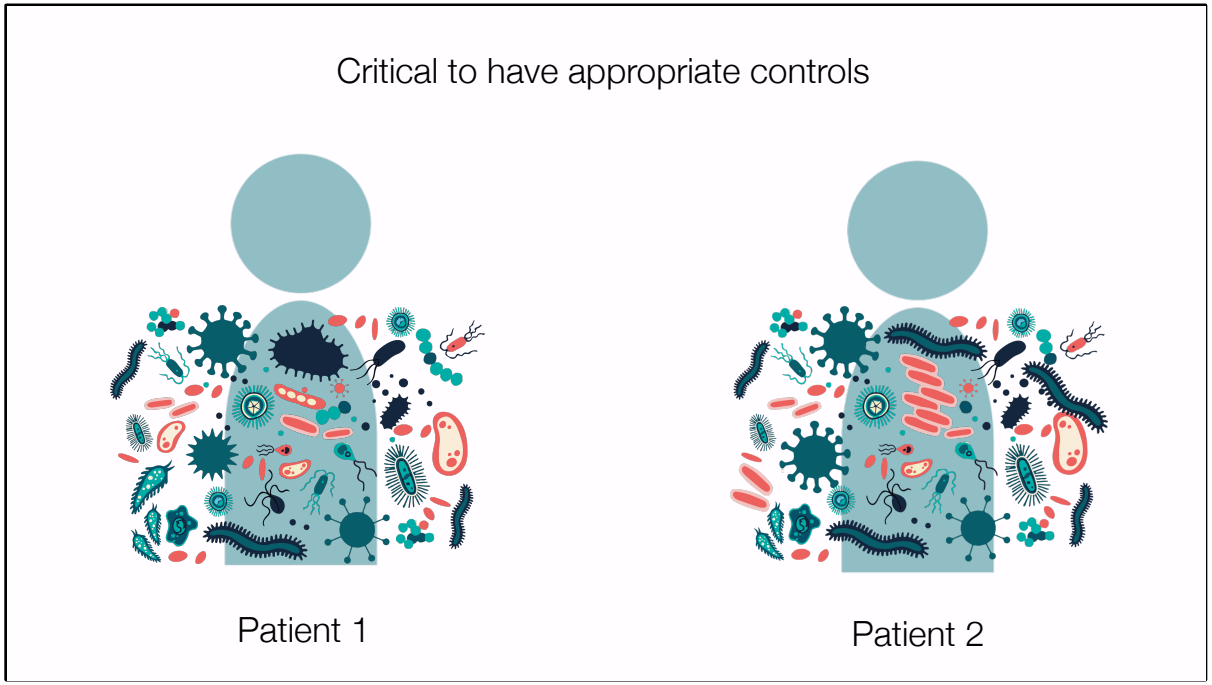
Need to handle human data

Ethics considerations for sharing data

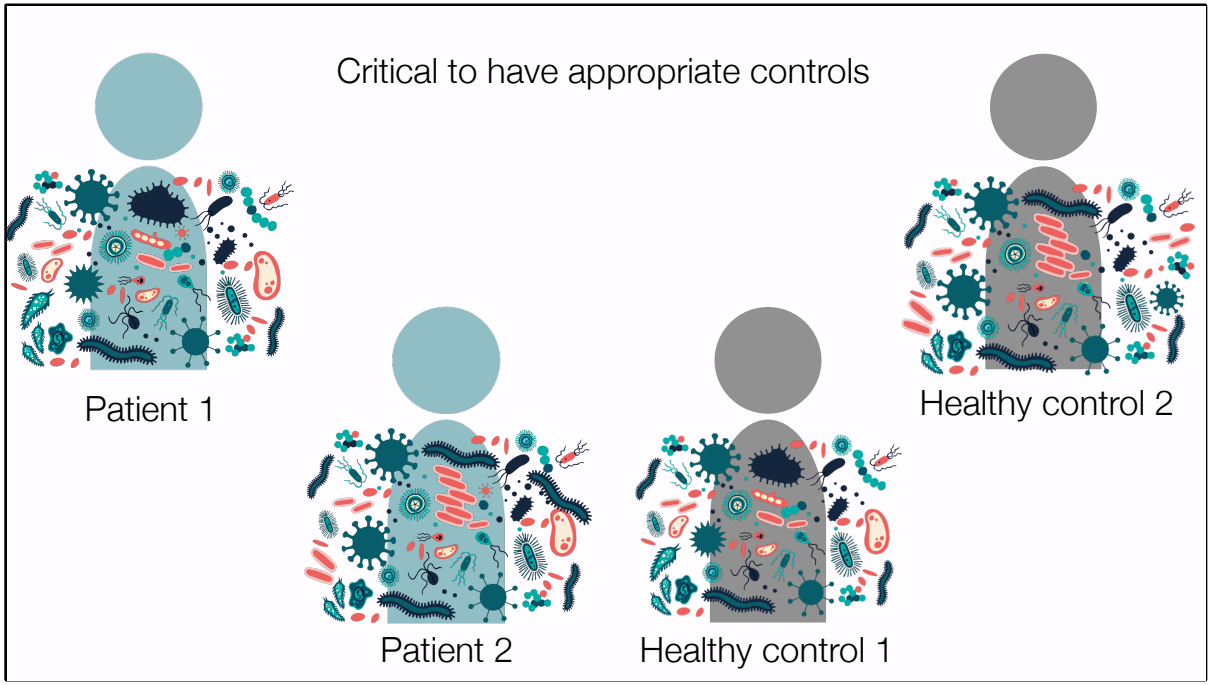
Identifiability

De-humanisation

Another difference with metagenomic sequencing is you need to have considerations for dehumanising the data. With targeted sequencing you can be fairly confident that you are going to amplify up your target, and hopefully off-targets are minimal, but with metagenomics you can be inadvertently sequencing a lot of human or rna. That human data could be of the patient that was sampled, or it could be of yourself too! Either way, this will need to be handled for ethical reasons before uploading or sharing the raw metagenomic data.

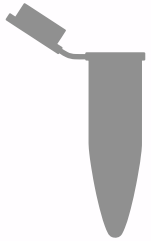


It’s very difficult to attribute disease to a particular organism, as the causative agent may be challenging to distinguish between various viruses and co-circulating bacterial pathogens, and some organisms can be opportunistic pathogens! There may be so much diversity within a given sample that It’s only with appropriate controls can we attribute disease to a specific organism. This adds to the expense of metagenomics, because you may be sequencing twice as many samples or more



We suggest having matched healthy controls from the same location (e.g. within household or within a hospital ward)

Controlling for contamination



Skin of investigators

Air contaminants

Equipment in lab

Kitome

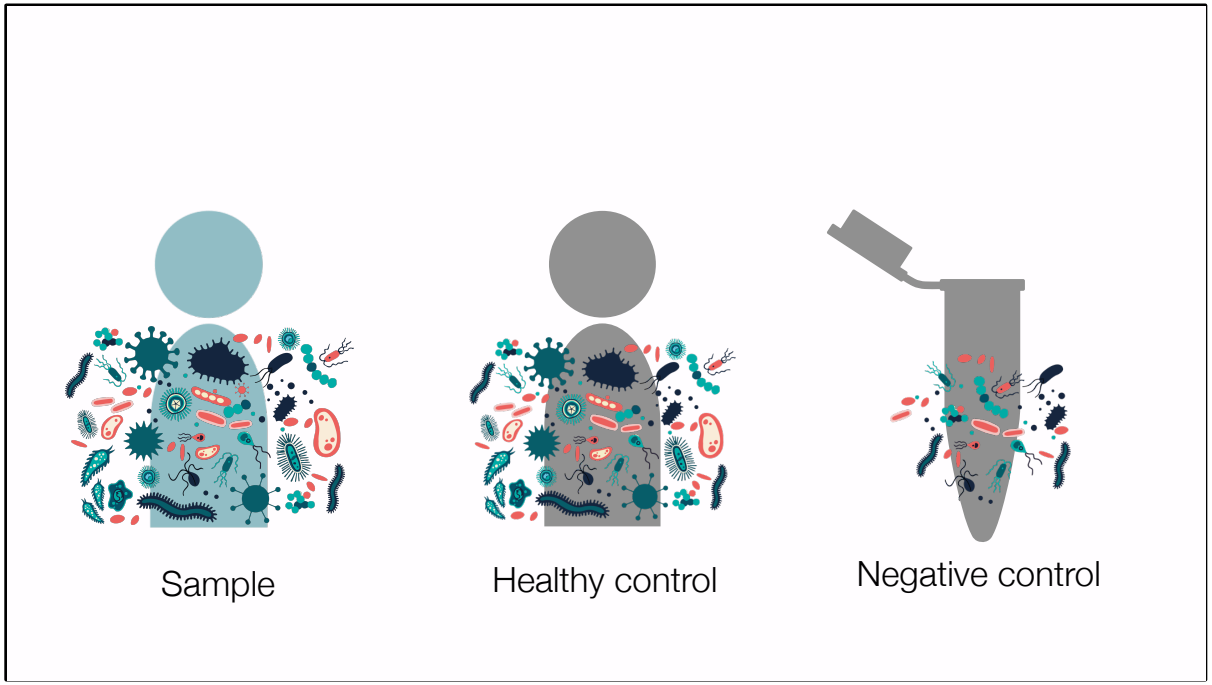
The kitome can include contaminants in extraction kits, pcr enzymes, library kits, even molecular water used can contain contaminants that will show up in sequencing

Controlling for contamination

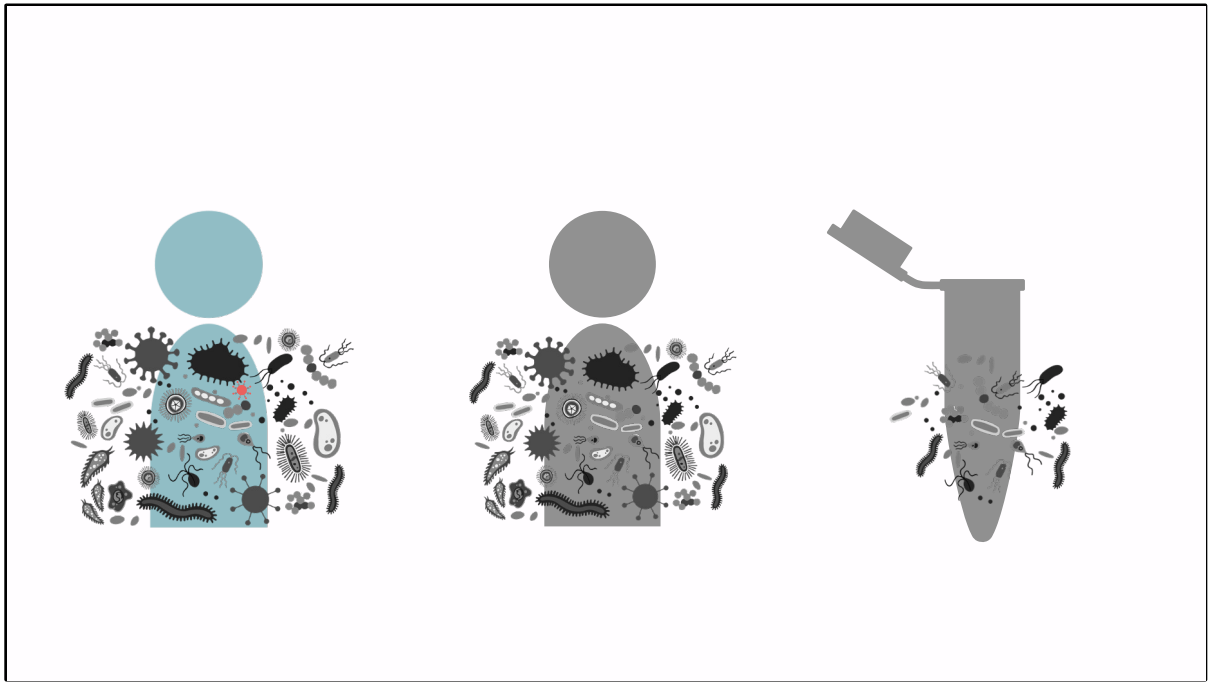
What to do?

- Sequence a negative control (not just a gel)
- Be aware of plate geography
- Consider batch effects

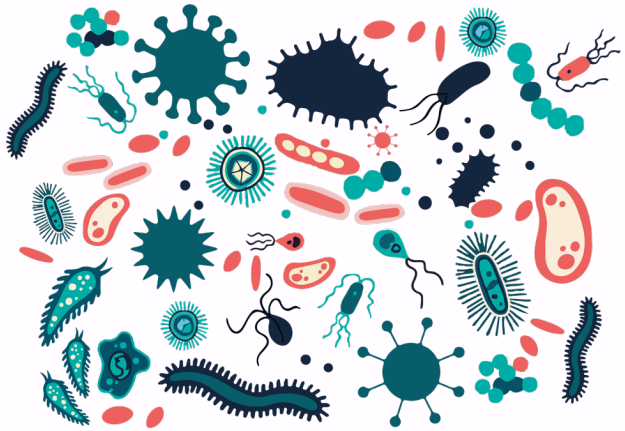
Plate geography includes being aware of which wells are adjacent to one another and which order samples were put into the wells. Could there be cross-contamination between them?



Then hopefully when you've designed the appropriate controls,

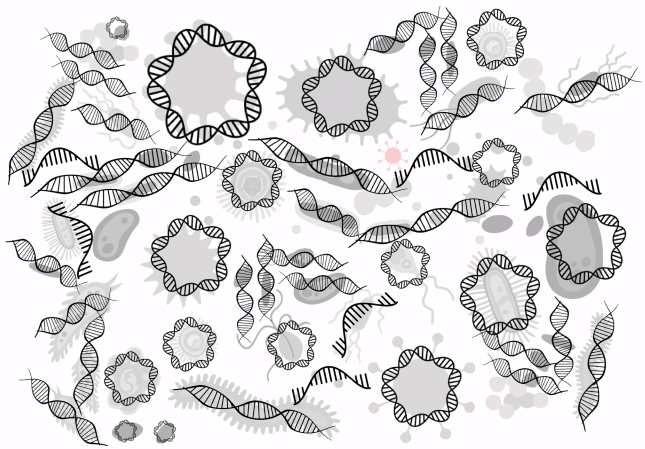


we can bioinformatically pull out the significant signals and rule out things common to the kitome and the control volunteers



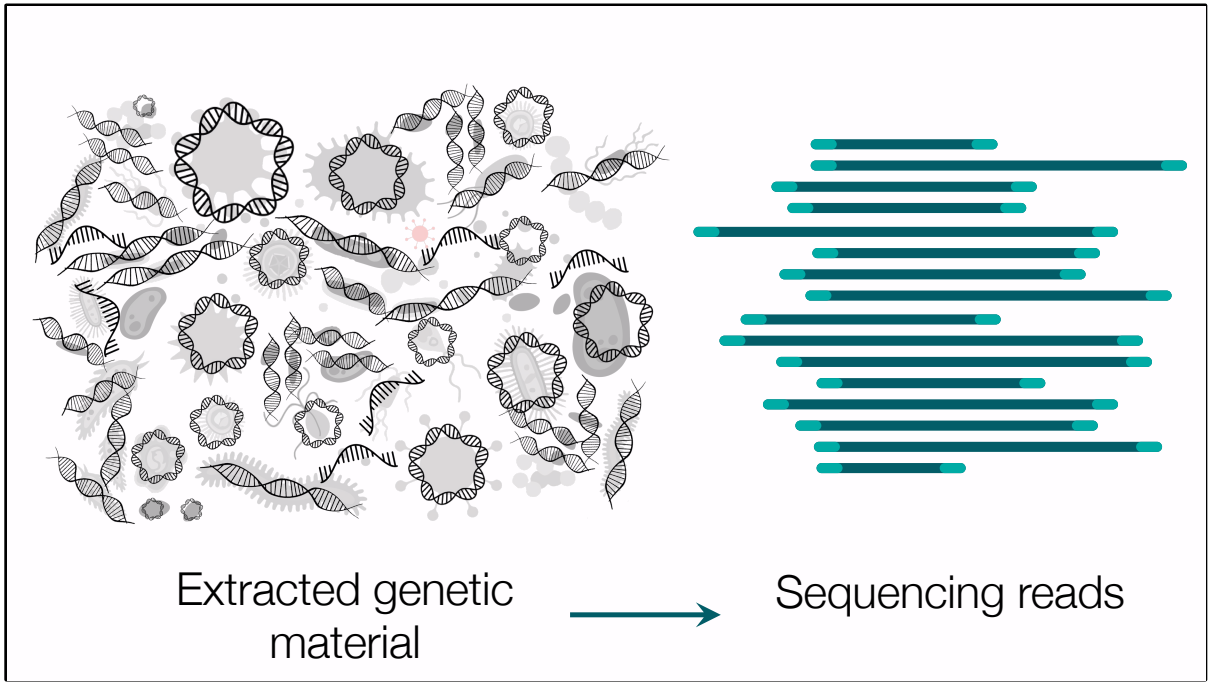
Diverse sample

So with all these things in mind, we have taken our diverse sample



Extracted genetic material

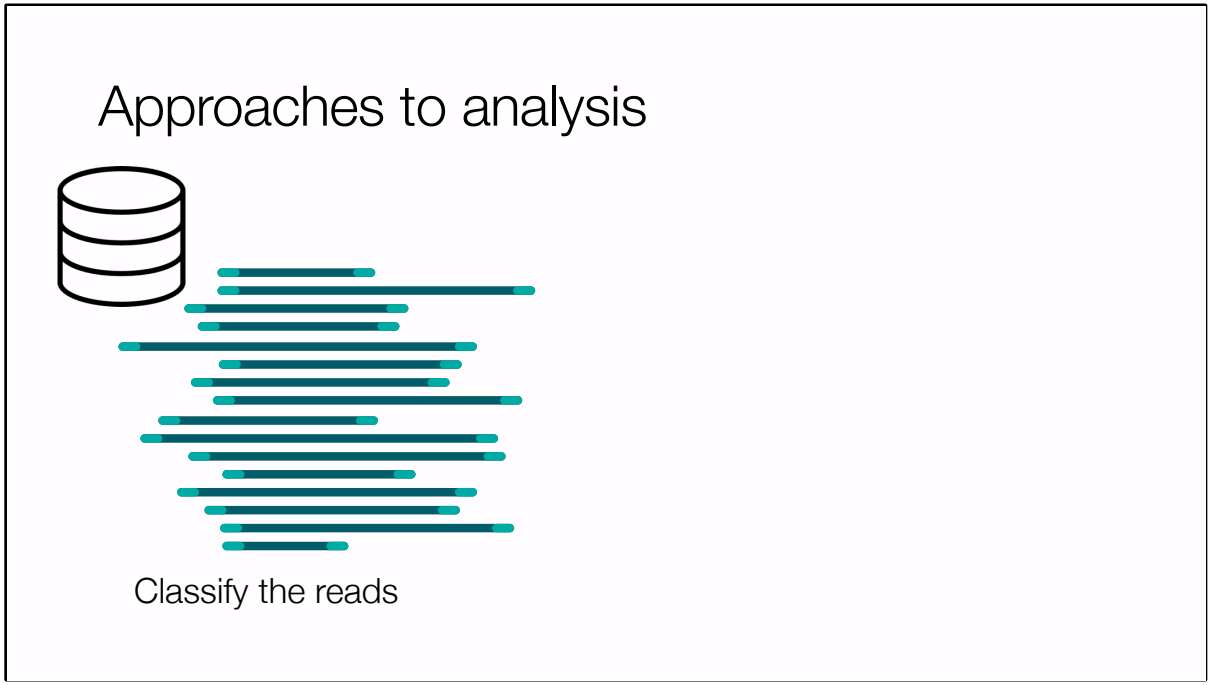
Extracted the nucleic acid material



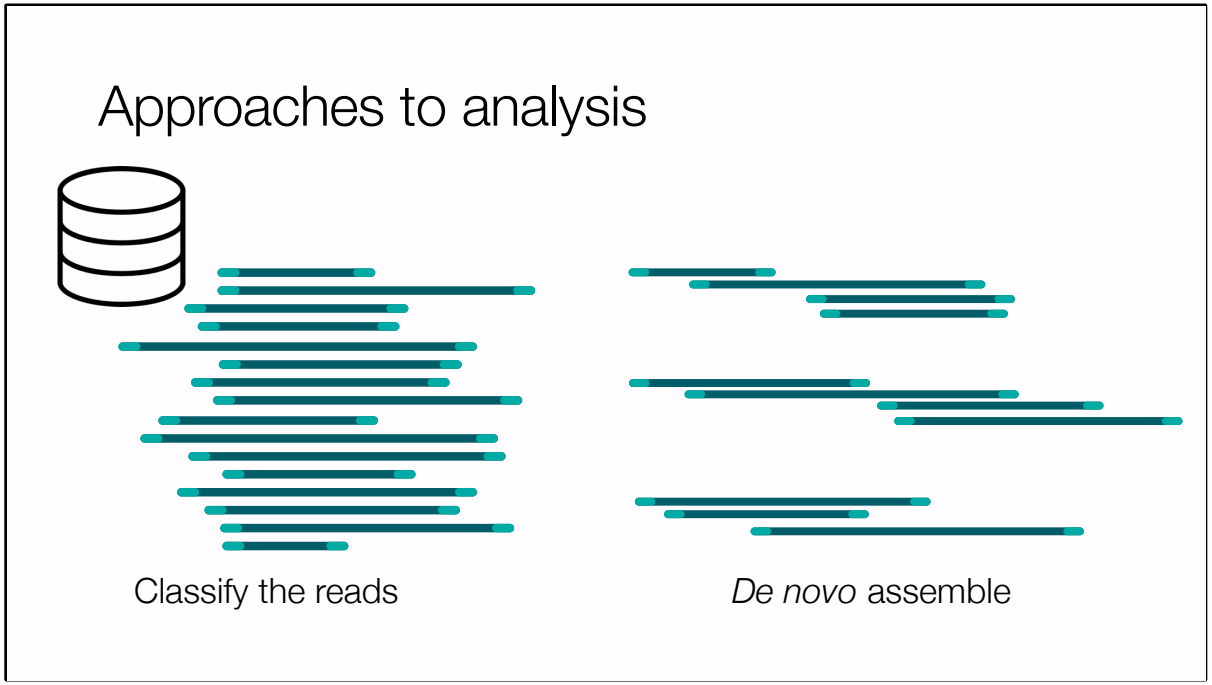
Done some unbiased metagenomic sequencing, and generated some sequencing reads

Approaches to analysis

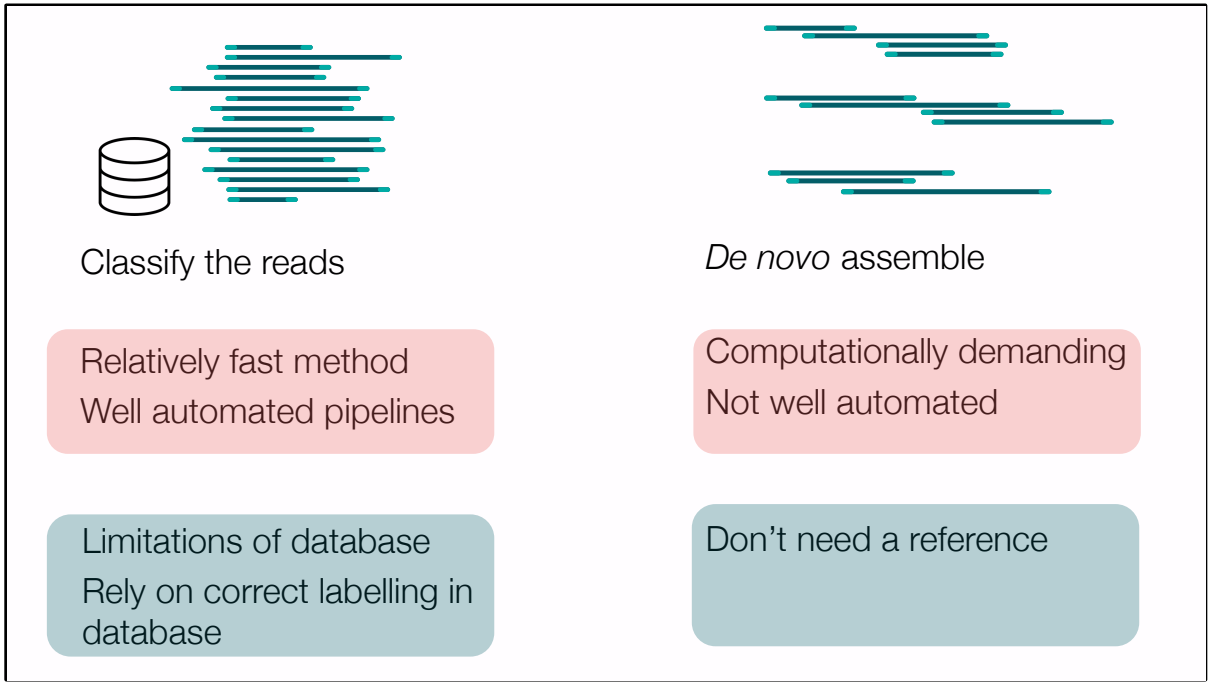
Where do we go from here? How exactly do we get useful information out of our data?



One approach is classification of reads- match reads against a database of known things



An alternative approach is to assemble the data- take the reads and try join up all the different bits that look like they match together (into what we call contigs), and then figure out what you've got in your sample after that



There are pros and cons to both approaches, and in reality you'll probably want to give both methods a try when working with metagenomic data

Classify the reads

Relatively fast method
Well automated pipelines

Limitations of database
Rely on correct labelling in database

ARTIC: EPI2ME Labs

wf-metagenomics

Kraken2 for read classification

The artic wf-metagenomics uses the classification approach

Genome Biology

Short Report | Open Access | Published: 28 November 2019

Improved metagenomic analysis with Kraken 2

Derrick E. Wood, Jennifer Lu & Ben Langmead

Genome Biology **20**, Article number: 257 (2019) | Cite this article

76k Accesses | **1675** Citations | **117** Altmetric | Metrics

John Hopkins University, Baltimore

Sequence classification tool

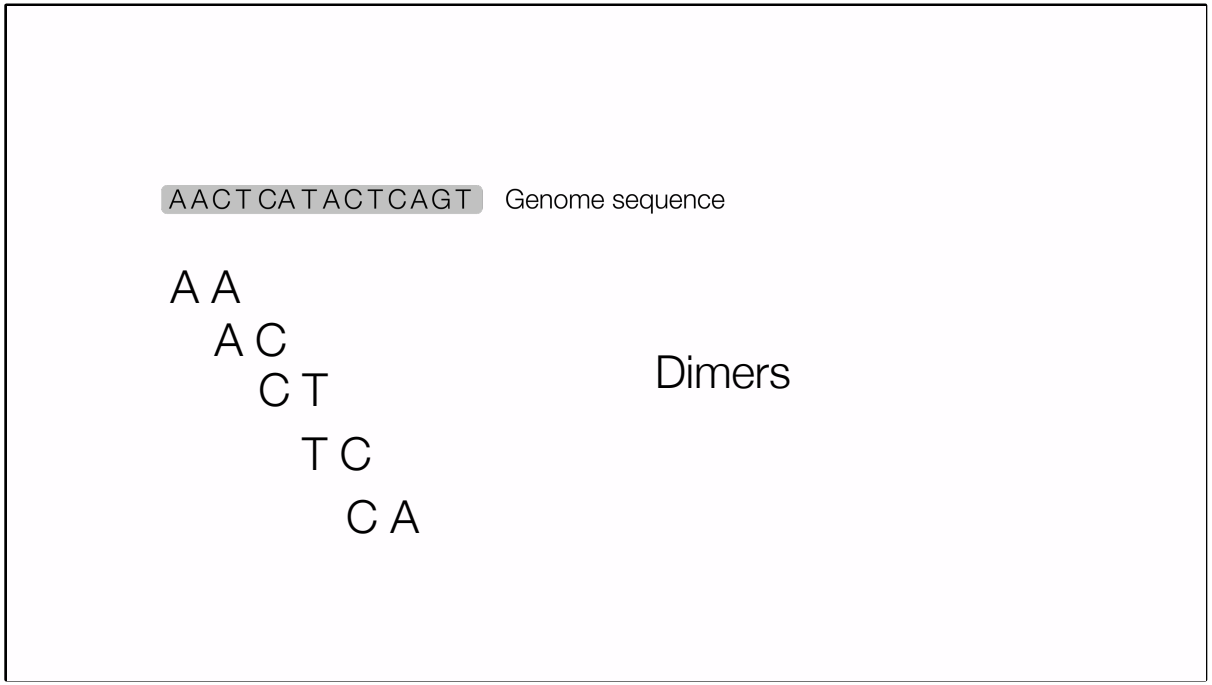
Uses exact-match database queries of k-mers

Kraken 2 is a popular classification tool that uses k-mer matching to taxonomically classify the metagenomic reads

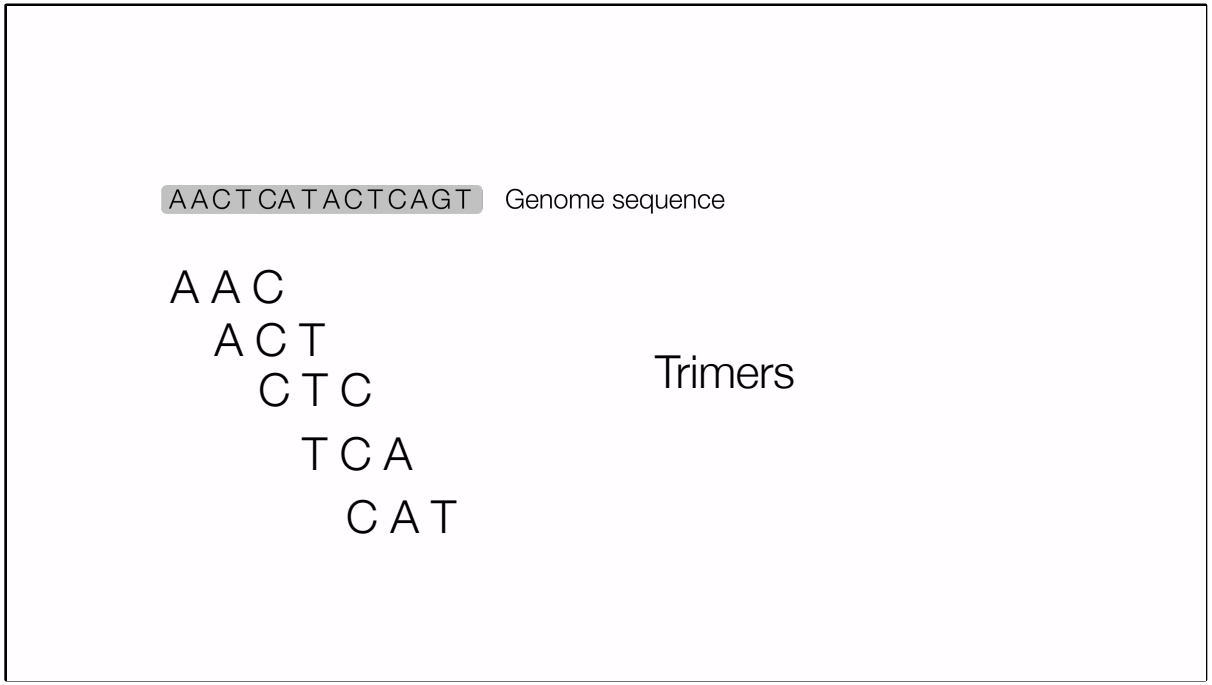
AACTCATACTCAGT Genome sequence

What's a k-mer?

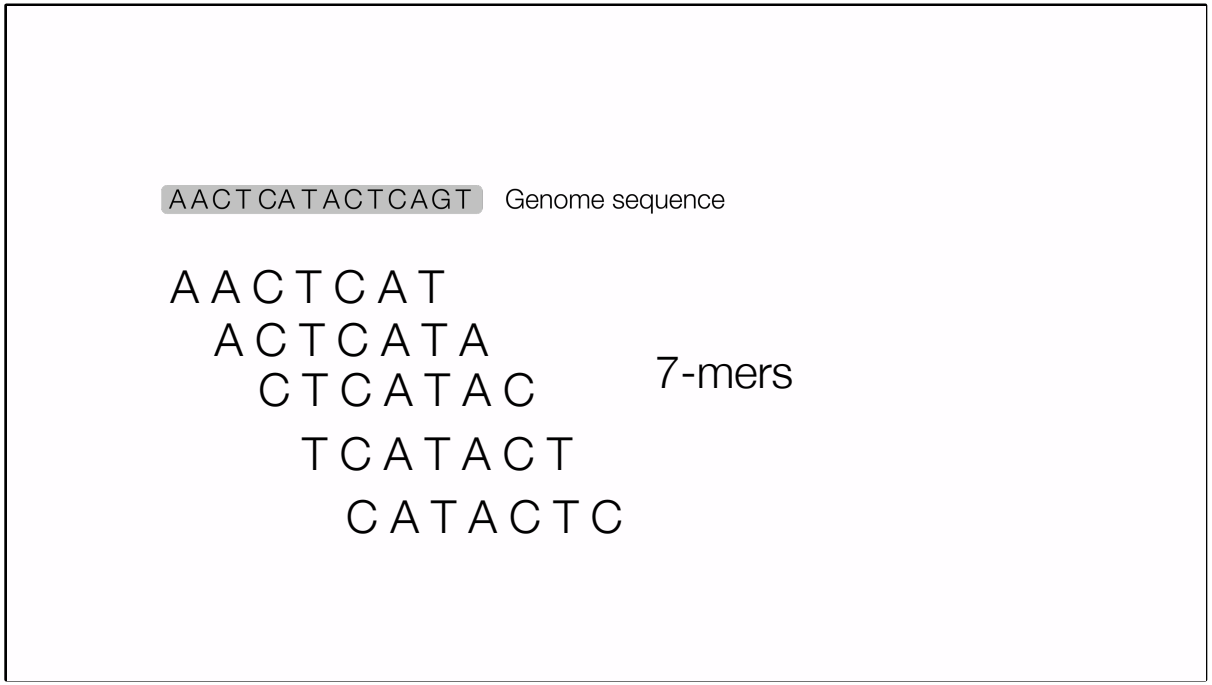
A k-mer is a dna or rna sequence of a certain length, where k represents the length. Often these refer to subsets of longer sequences



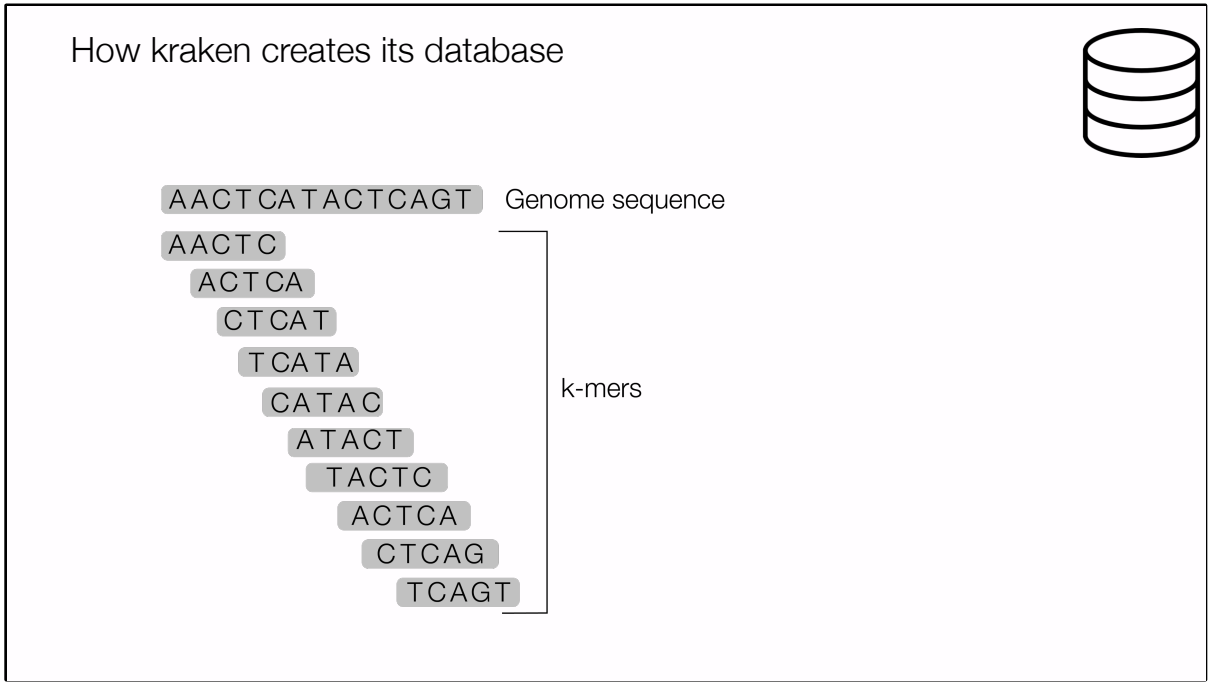
For example a dimer (or a 2-mer), is just a sequence of length 2. Here we can see dimers along a “sliding window” of the original genome sequence.



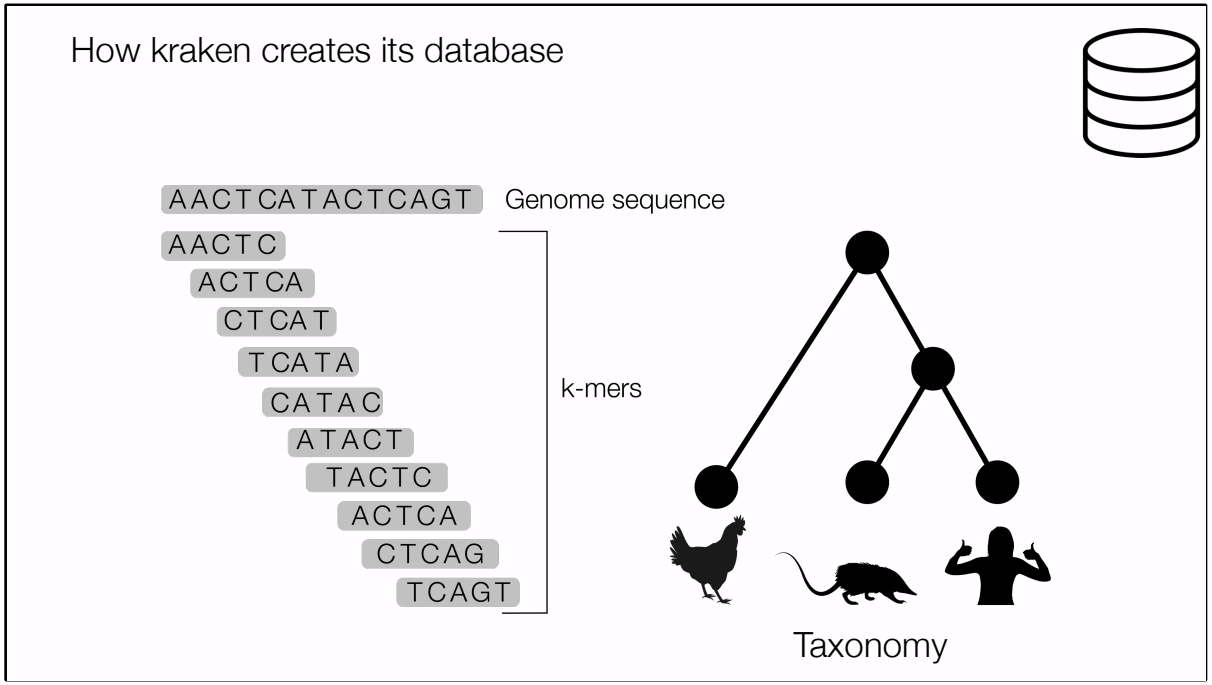
Here we can see a trimer (or 3-mer) sliding window of the same sequence



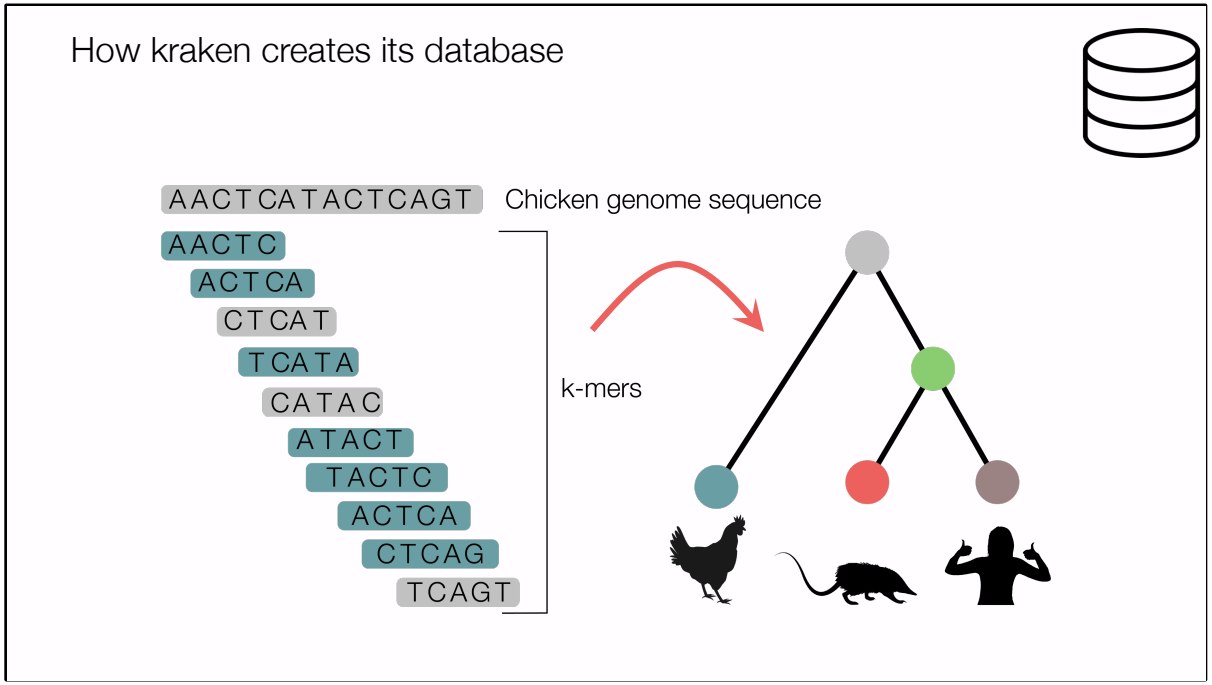
And a 7-mer sliding window



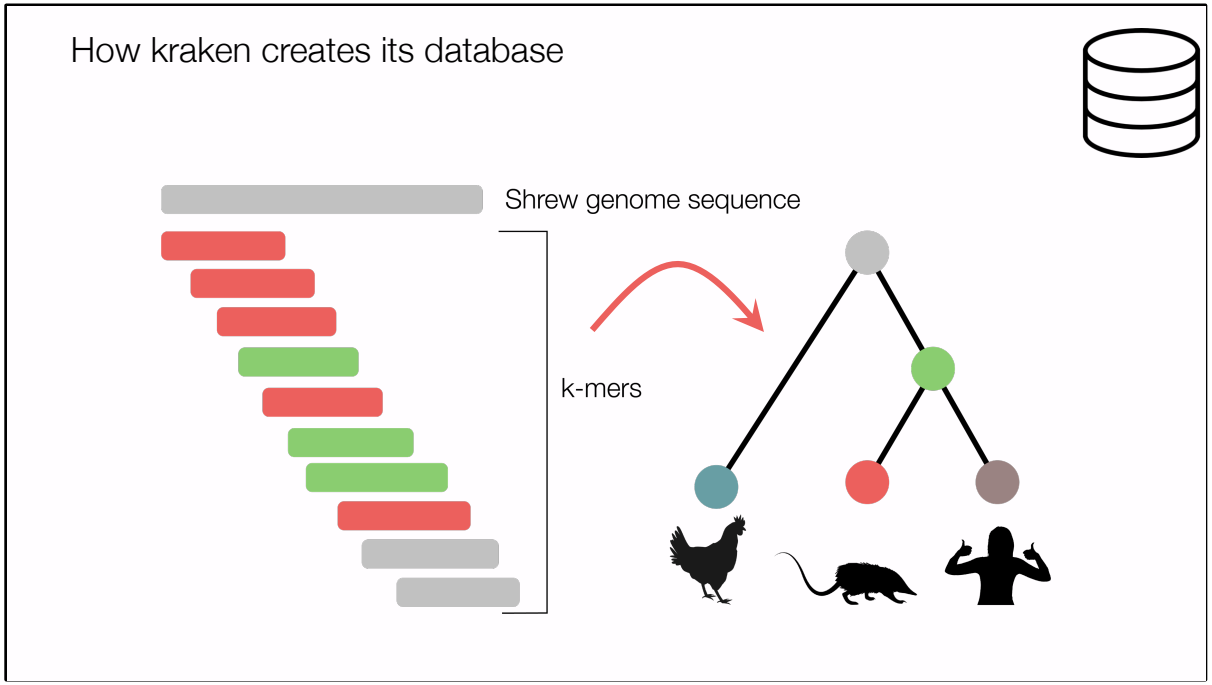
How kraken works is it takes all the kmers of a given sequence (of a certain size, here we're using 5-mers as an example), and it catalogues them



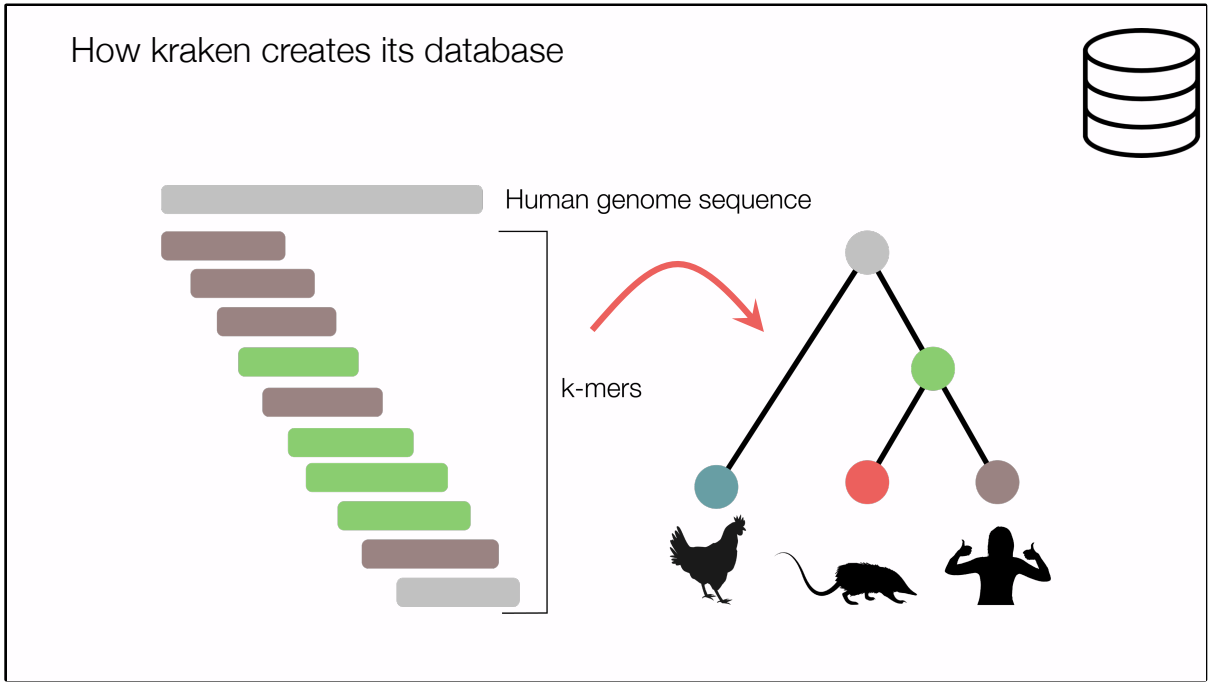
Kraken also accesses taxonomy information



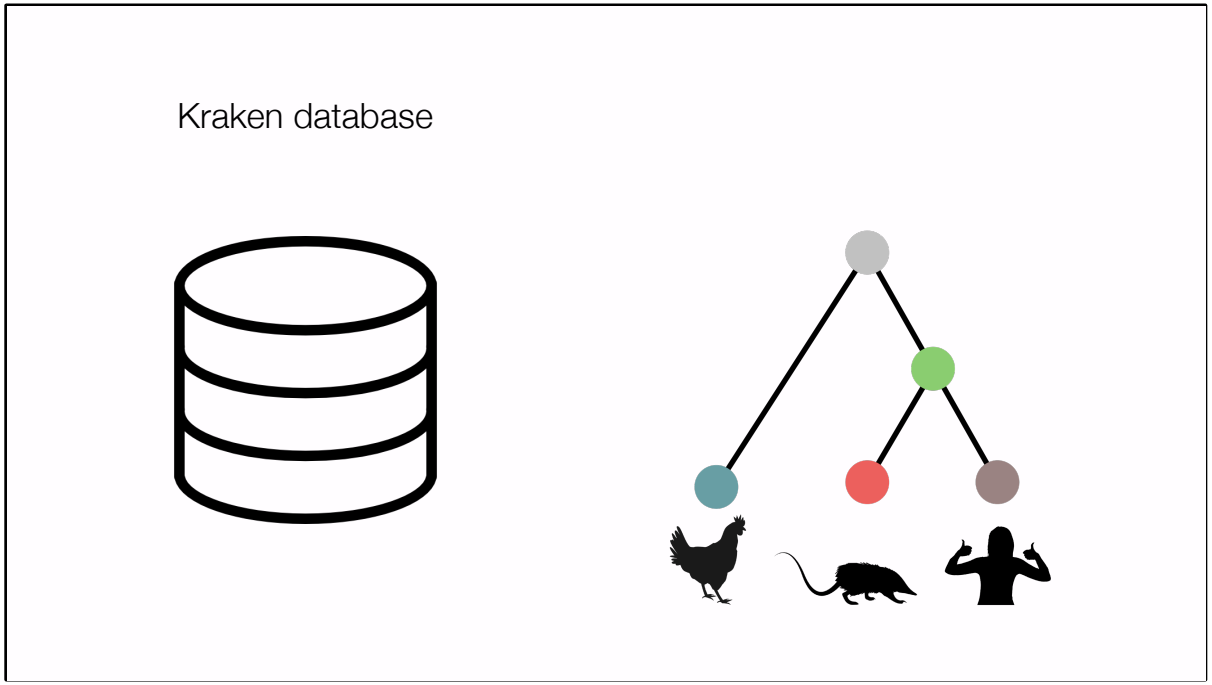
For an example, here we see a chicken genome sequence. Kraken breaks it up into kmers and then identifies within the whole database, which kmers are unique to chicken only (blue) and keeps a record of that. Some kmers unique to chicken here, and so are classed as uninformative- or here assigned to the root of the taxonomy (grey)



It does the same for the shrew genome here. Now we see that some of the kmers are specific to shrew (red), but actually some of them (in green) are common to both shrew and human genomes (possibly to all mammals?). In this case, kraken will record those kmers at the most specific taxonomic level possible (here, let's say Mammalia), but those kmers cannot tell you which mammal exactly that sequence is derived from. We can see there are also a couple of kmers that are not specific to any in the database.

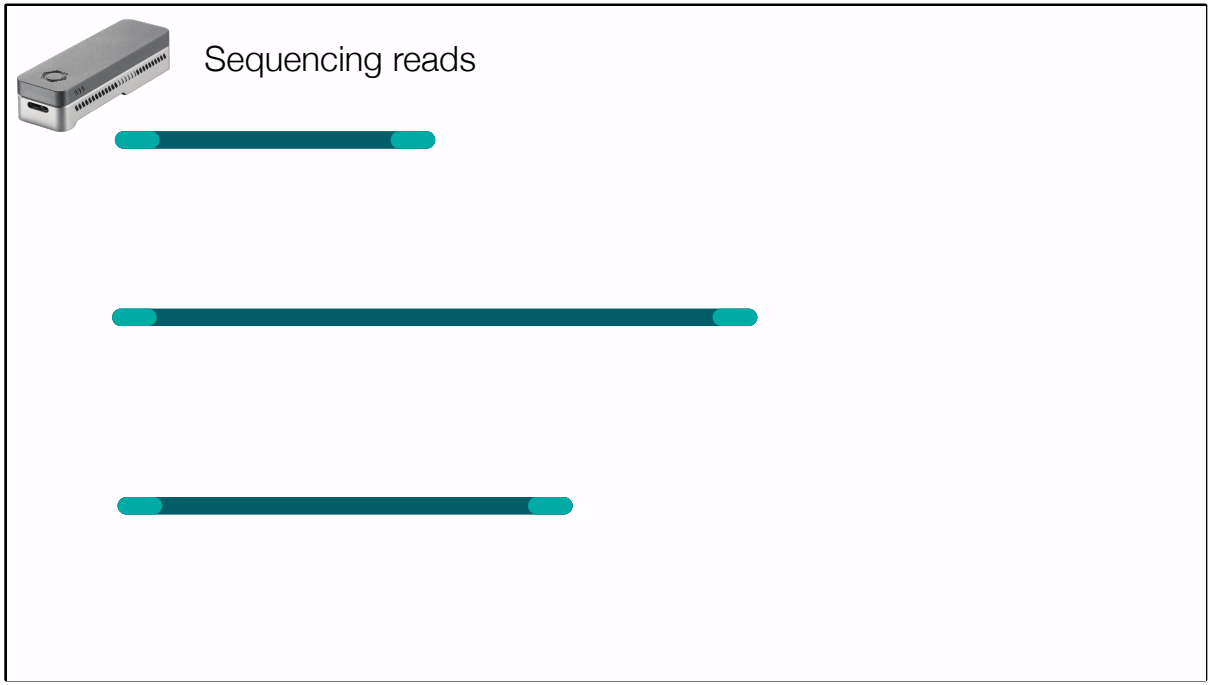


Kraken does the same for everything in the database, here we see the human genome kmers have some human specific (brown), some mammal-specific (green), and some uninformative (grey) kmers

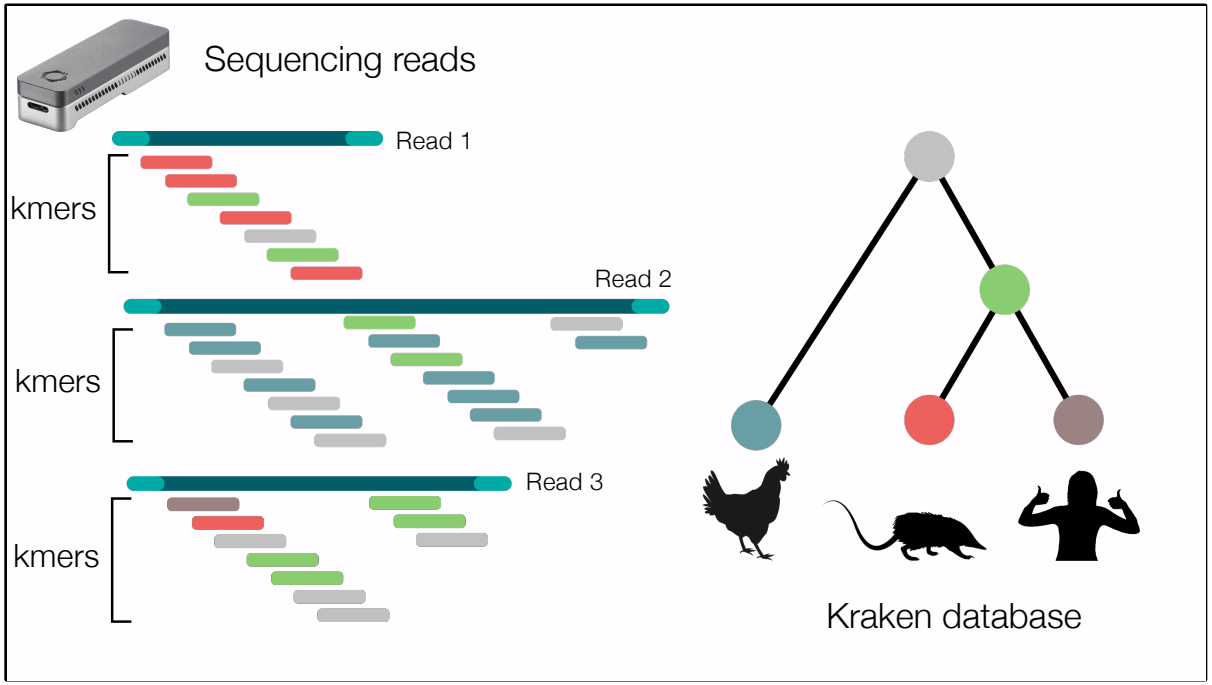


So this information is what kraken stores as its database. In reality, there is a much larger set of taxa, and much greater numbers of genomes represented.

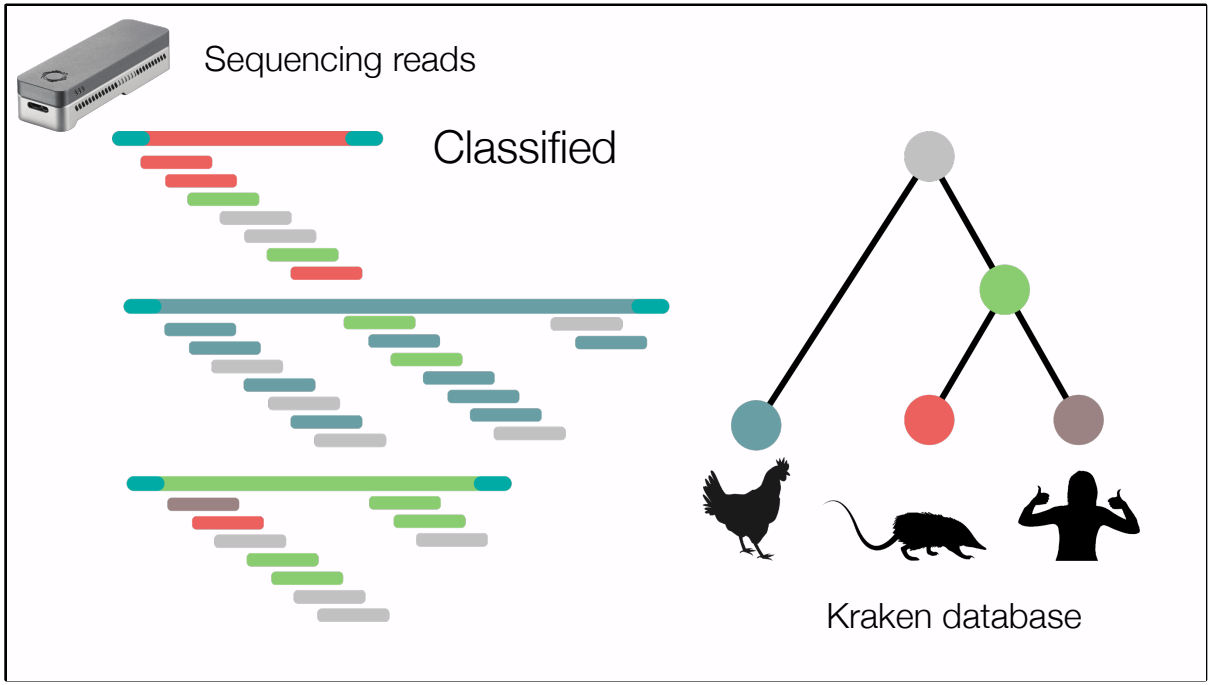
Kraken accepts sequencing reads as the query against the kraken database



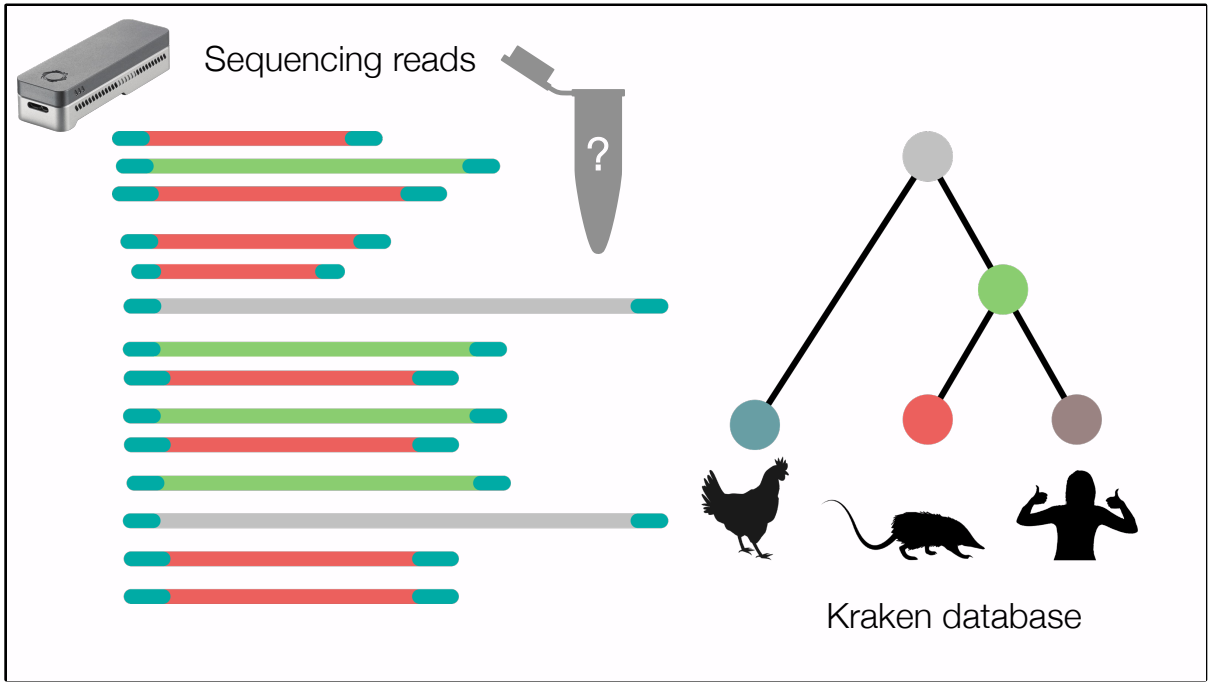
Kraken applies the same treatment to the reads as it did to the genomes in the database



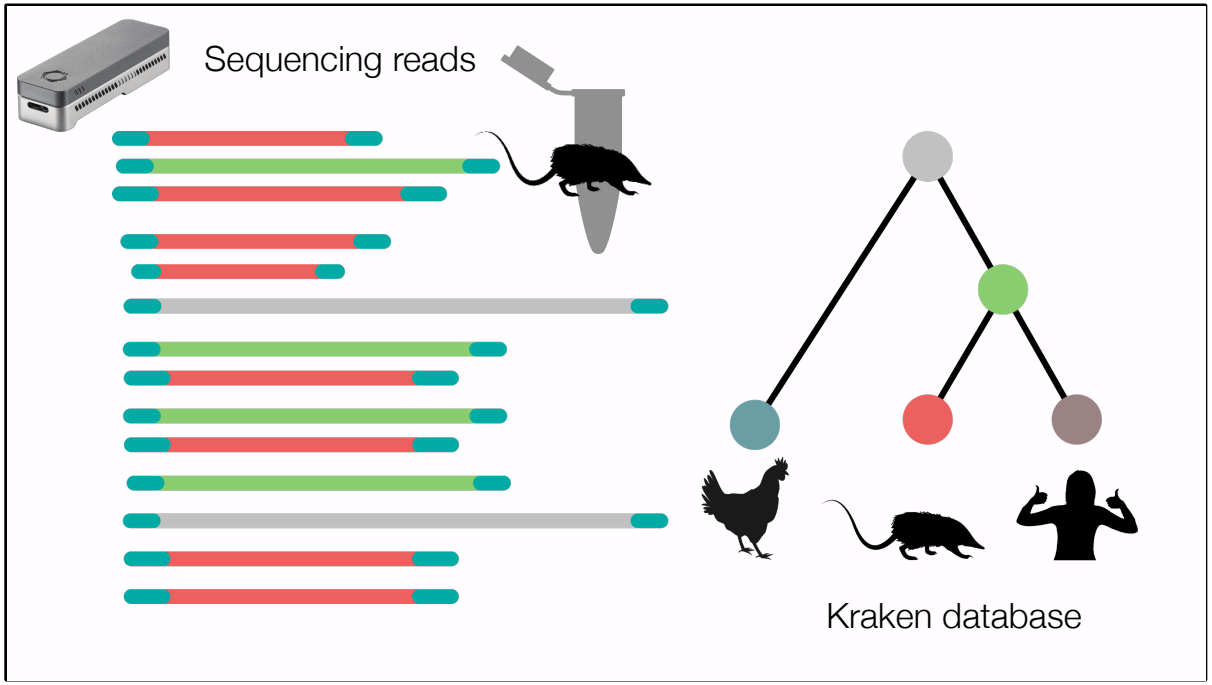
It splits each read into a series of kmers, and uses the database to look up which taxonomic assignment can be given to each kmer. Here we see three example reads split into kmers.
Read 1 has some shrew-specific kmers, some mammal-specific kmers, and one uninformative one. The majority of kmers are shrew though, if you count them all up. And mammal kmers don't necessarily conflight with the shrew signal either. For read 2, most of the kmers are chicken, with a few uninformative or mammalian kmers thrown in. The mammalian kmers do conflict with the taxonomy, but kmer-classification is noisy, so this can happen. The final read has a majority of kmers assigned to mammals, with some human and shrew and uninformative.



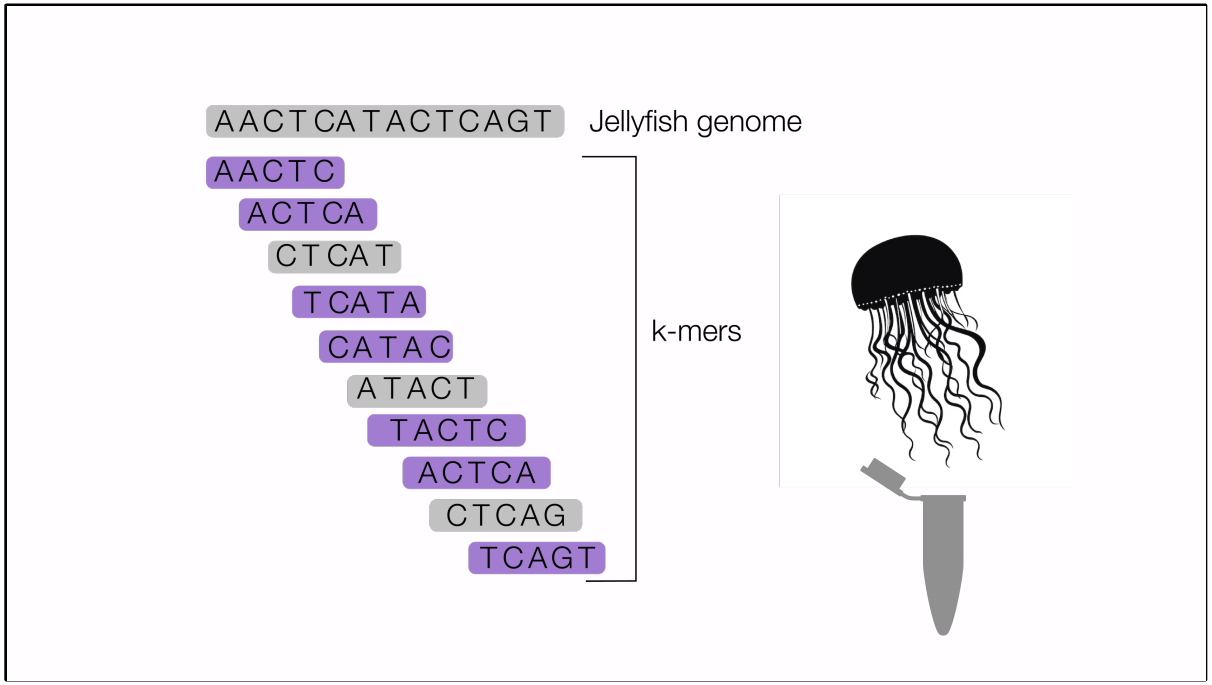
The classification of the reads goes with the taxonomic label with the most kmer support, so read 1 gets classified as shrew, read 2 as chicken and read 3 as mammal.



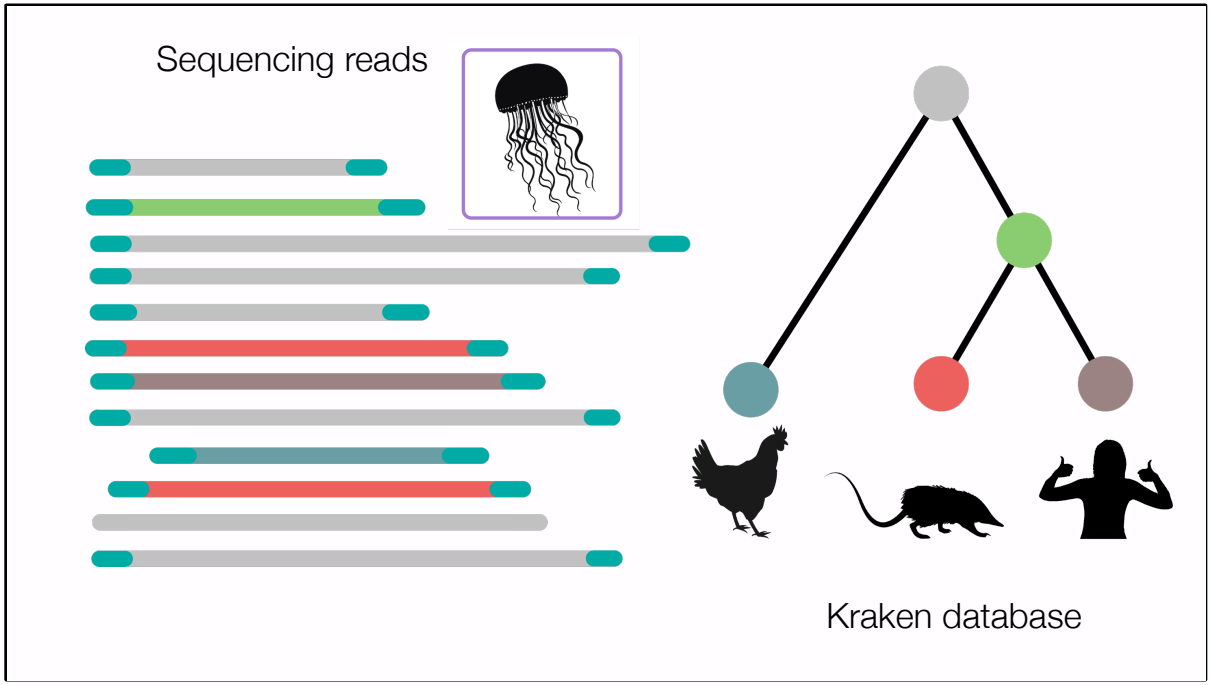
For all the reads from a given sample, each read will get the most specific taxonomic classification possible. However bear in mind that reads are intrinsically noisy, and can have high error rates. The classification itself is noisy, and just an estimate of what sample you've got.
This mystery sample has some shrew reads, some mammalian and some uninformative reads.



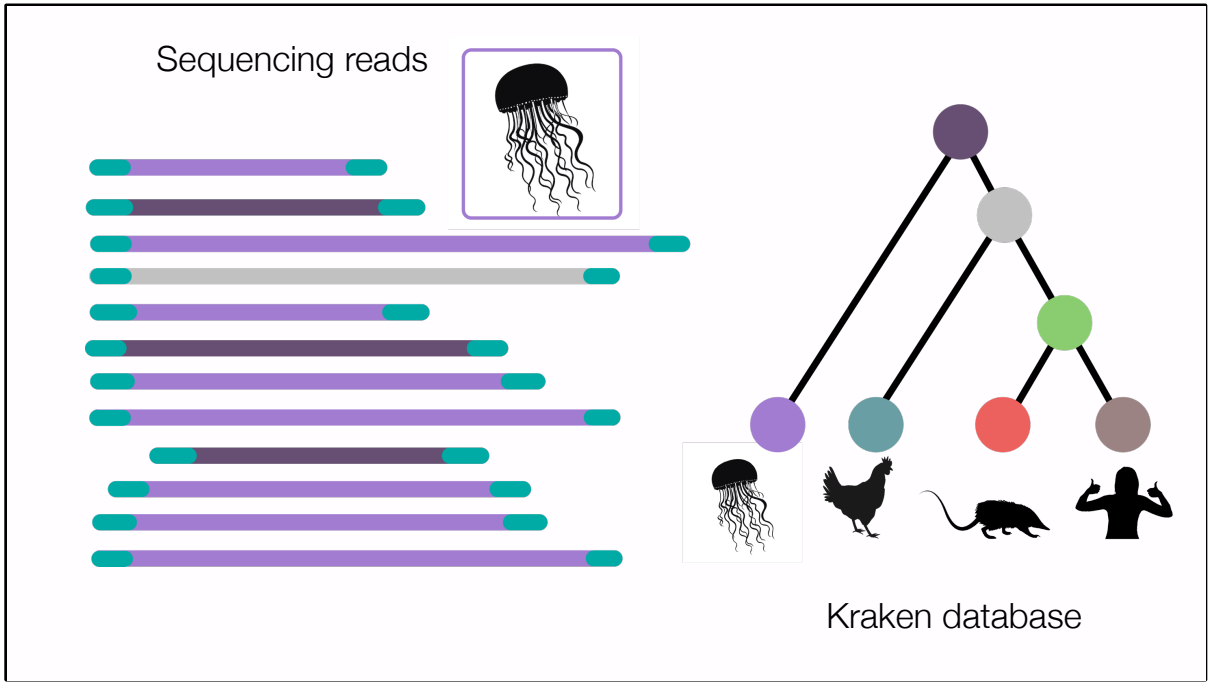
Because the majority of reads point to shrew, we could say this sample likely is a shrew sample, or some related mammal that is closer to shrews than humans (bear in mind our database only has 3 species in it!)



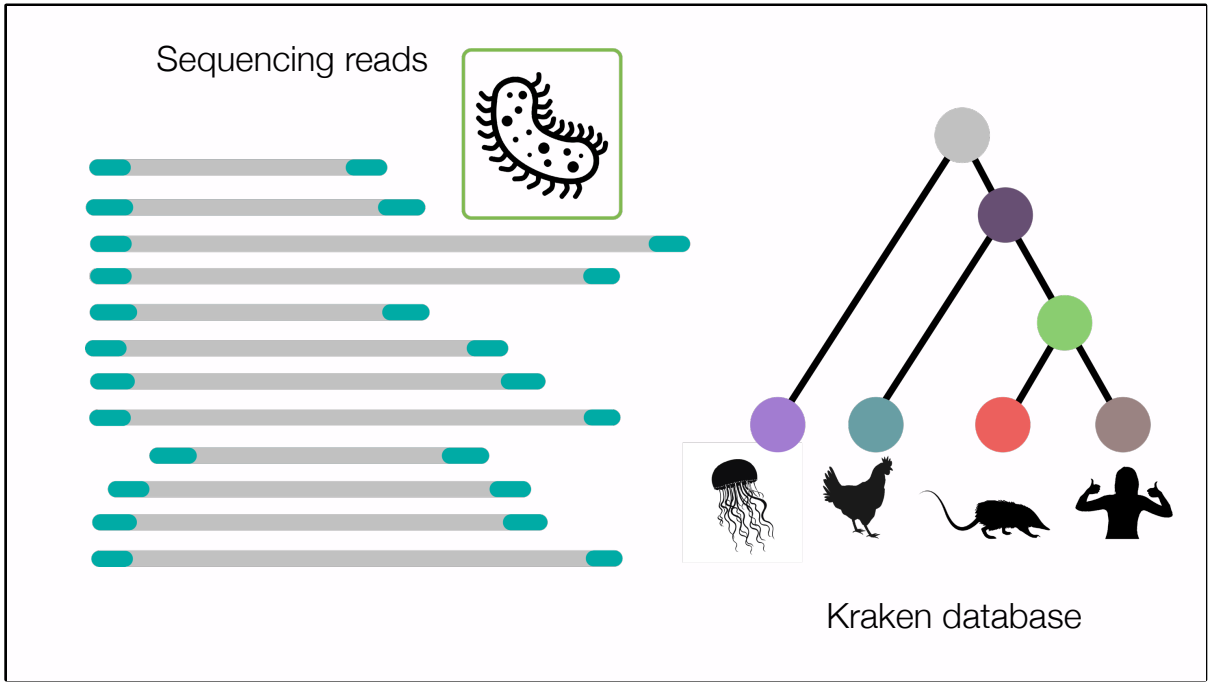
To give you an example of how limited our database is- what happens if our sample is something not represented in the database? For example here a jellyfish



In this case, we might by chance have some related kmers that match to the species in our database. Here we see some sequencing reads assigned mammal, shrew, chicken and human

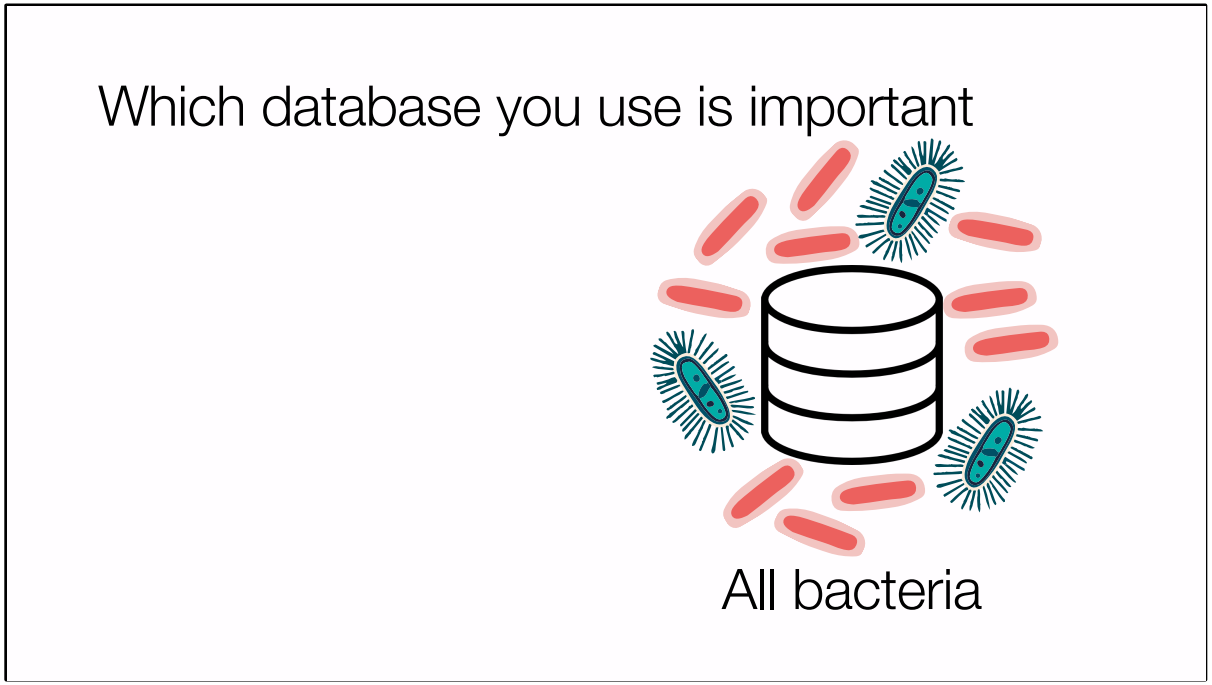


However if we matched against a different database that had a closely-related jellyfish reference, we might see a very different result! We need to be very aware of how limitations or biases in our database can heavily impact the end result

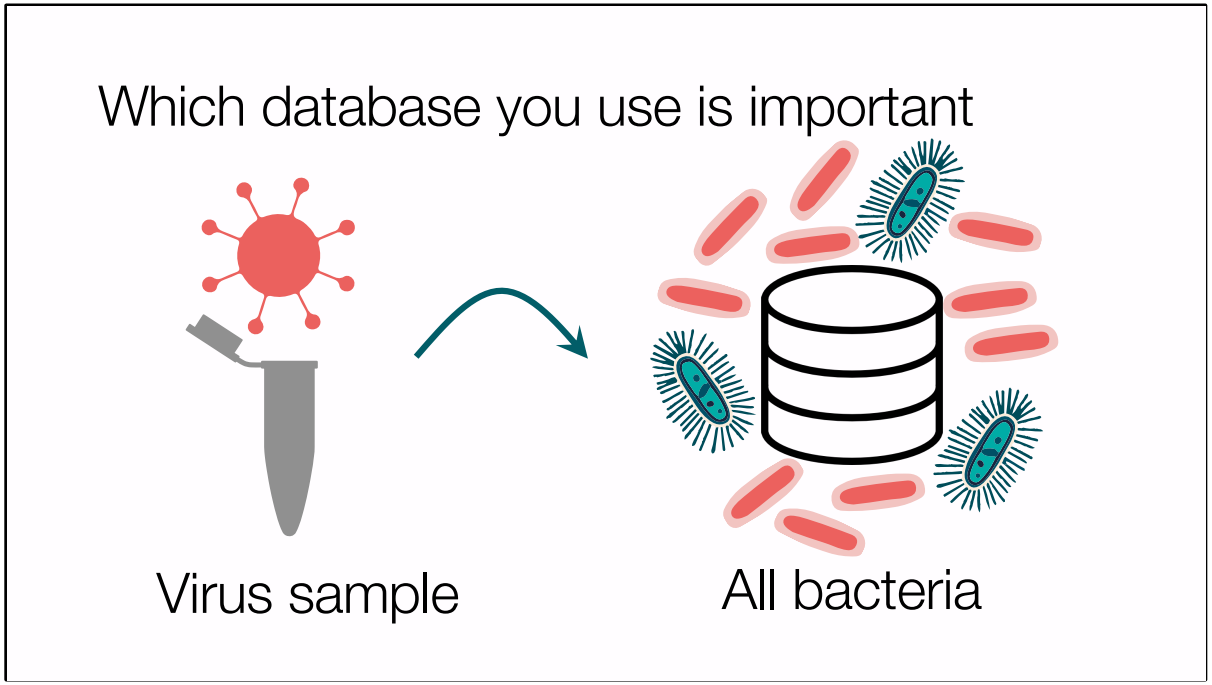


Imagine if we had a bacterial sample for our animal database? What would you expect to see?

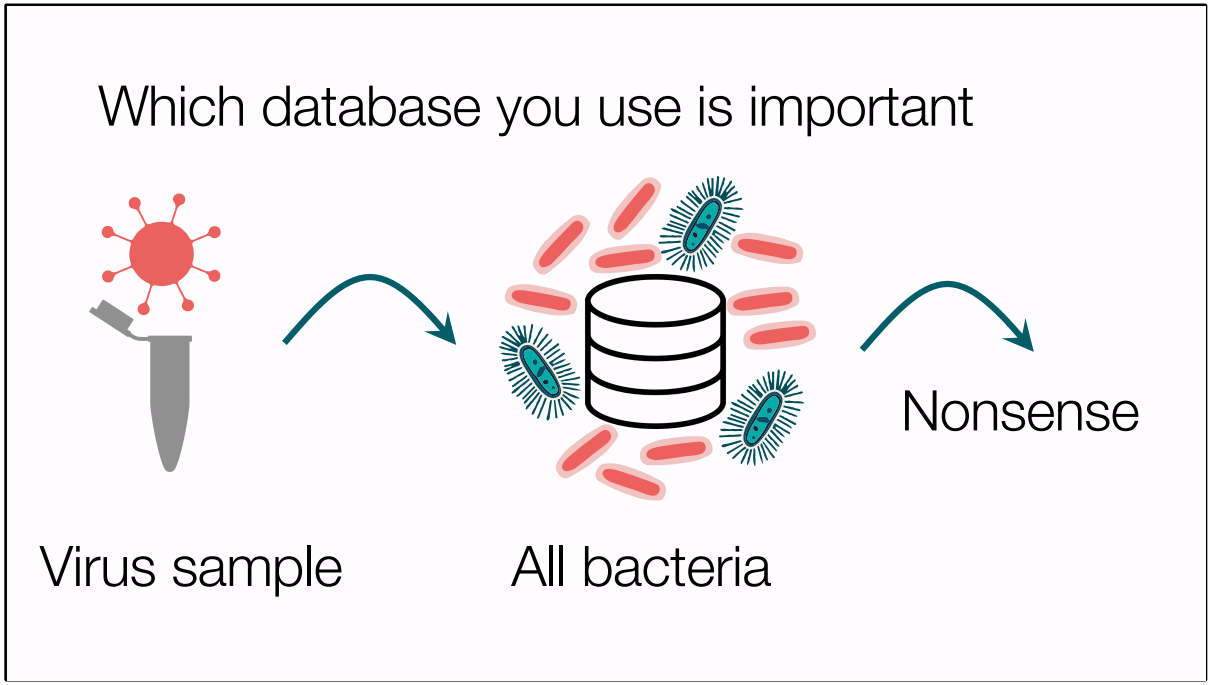
Which database you use is important



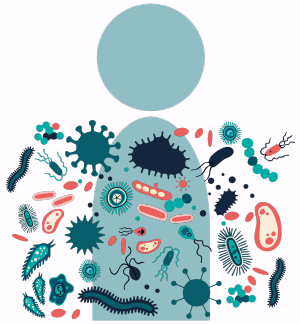
If you have a kraken database containing all known bacteria



But the sample that you're sequencing is a virus sample

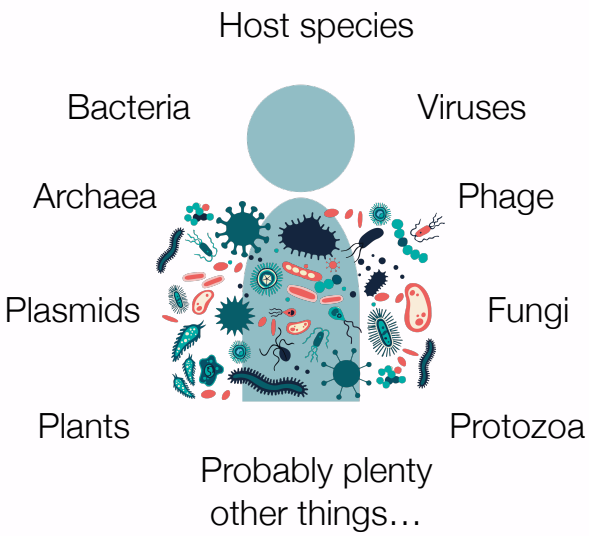


You won’t get any useful assignments from the data, except perhaps about the contaminants/ background diversity in the tube

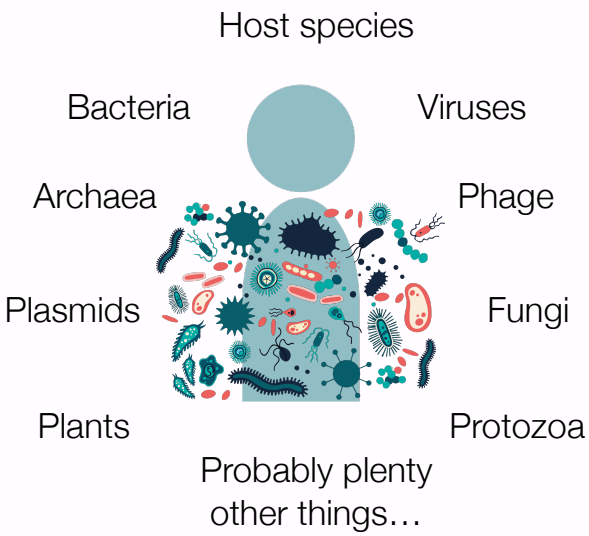


What's in my sample?

And the sample you sequence can have a bit impact on the diversity of organisms present in the metagenomic data



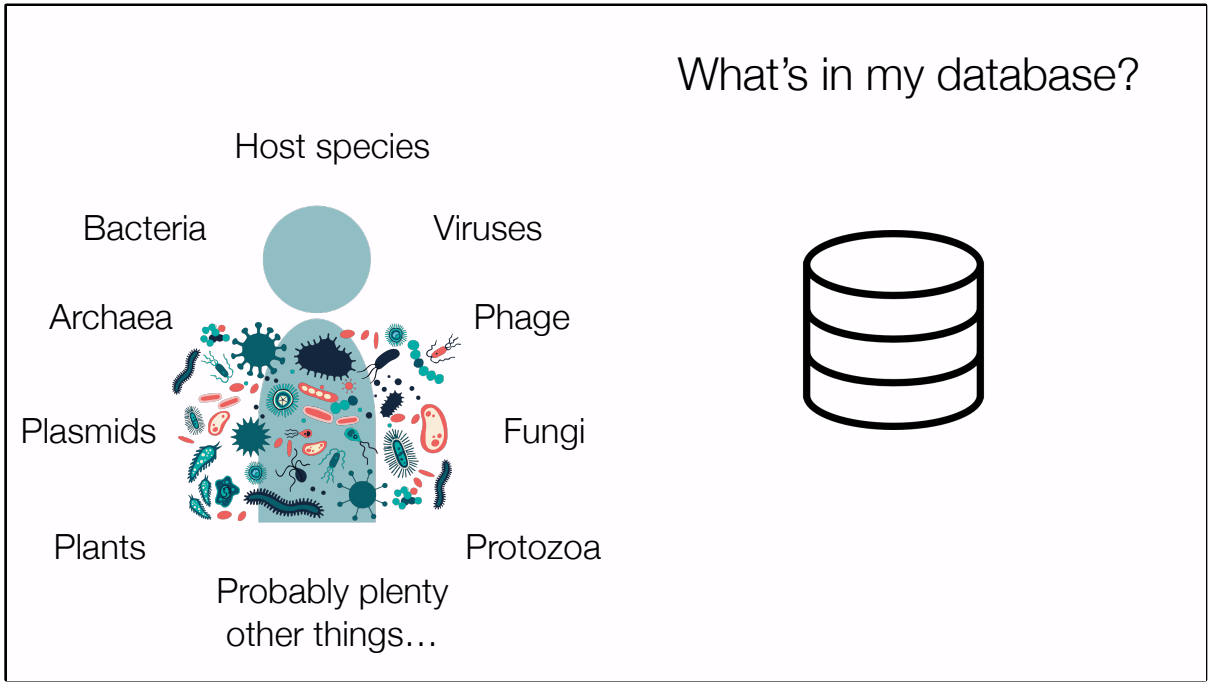
What's in my sample?



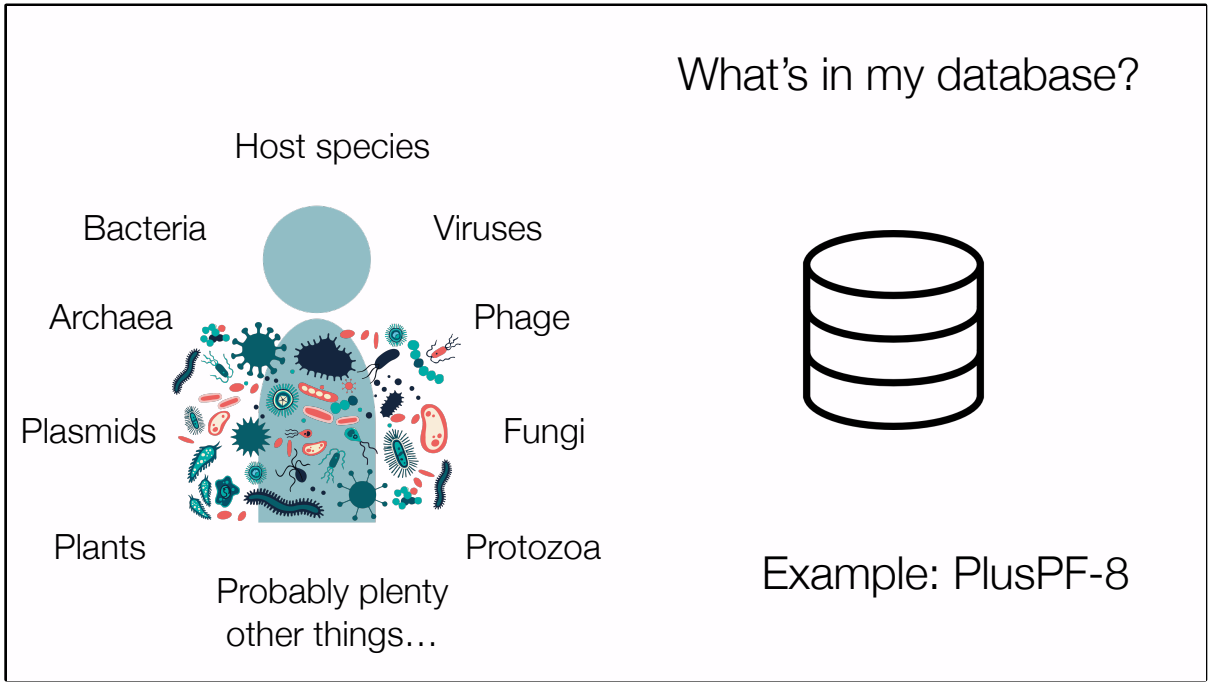
What's in my sample?

What am I interested in
in my sample?

What you're interested in in the sample might be the viral pathogens?



So when you go to identify a given set of sequencing reads, you're very much dependent on what's known and what is included in the database.



So when you go to identify a given set of sequencing reads, you're very much dependent on what's known and what is included in the database.

What's in my database?

Archaea
Bacteria
Viral
Plasmid
Human
UniVec_Core
(vectors, adapters, common primers)
Protozoa
Fungi

No plants!!

Example: PlusPF-8

This example database contains…

But it doesn't contain plant genomes- so plant matter won't get assigned. This is a time space-saving shortcut, as plant genomes can be enormous and complex.

What's in my database?

Archaea
Bacteria
Viral
Plasmid
Human
UniVec_Core
(vectors, adapters, common primers)
Protozoa
Fungi

No plants!!
Genomic dark matter

Example: PlusPF-8

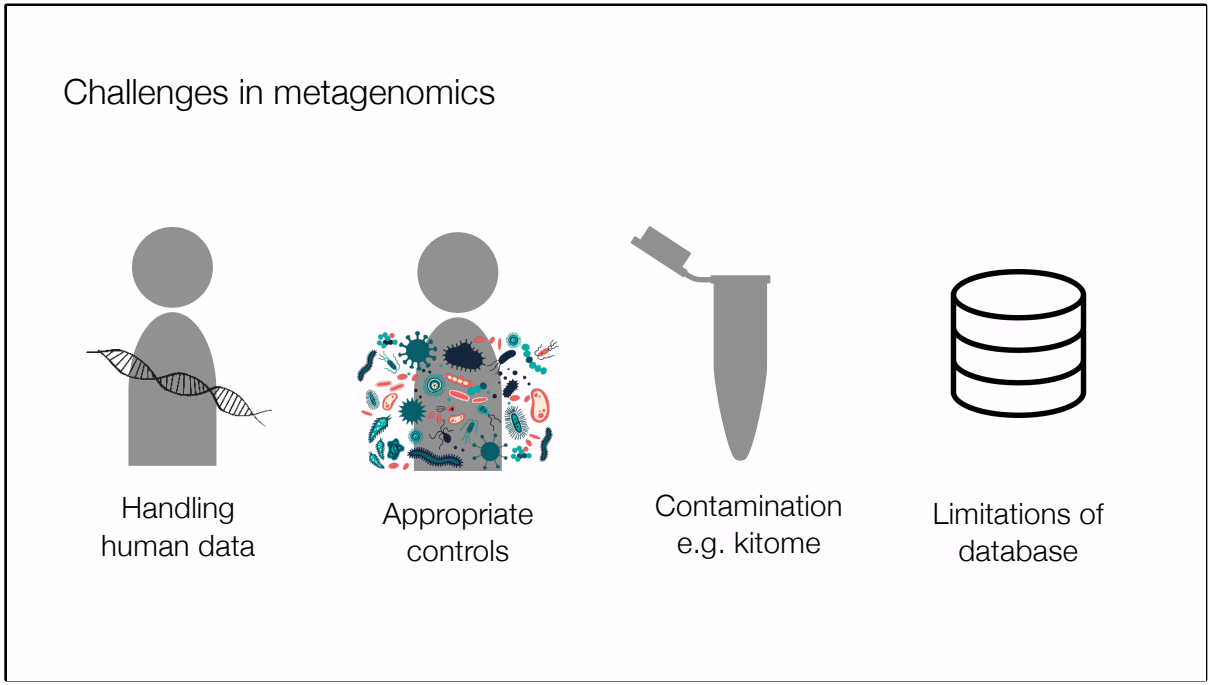
There's also genetic material that's too different from anything known to match to anything, so that won't get classified

Archaea
Bacteria
Viral ← Not every virus
Plasmid
Human
UniVec_Core
(vectors, adapters, common primers)
Protozoa
Fungi

No plants!!
Genomic dark matter

Example: PlusPF-8

And there's going to be limitations to this database- it may contain viruses, but it won't contain every virus you're might come across. For example, this PlusPF-8 database doesn't contain poliovirus!



So when you go to interpret your results, bear in mind these challenges and limitations associated with metagenomics.

ARTICnetwork

Putting genomic surveillance at the heart of viral epidemic response

http://artic.network | @NetworkArtic

Metagenomics analysis

Áine O'Toole