Introducing Microbiome Bioinformatics

Part 6.

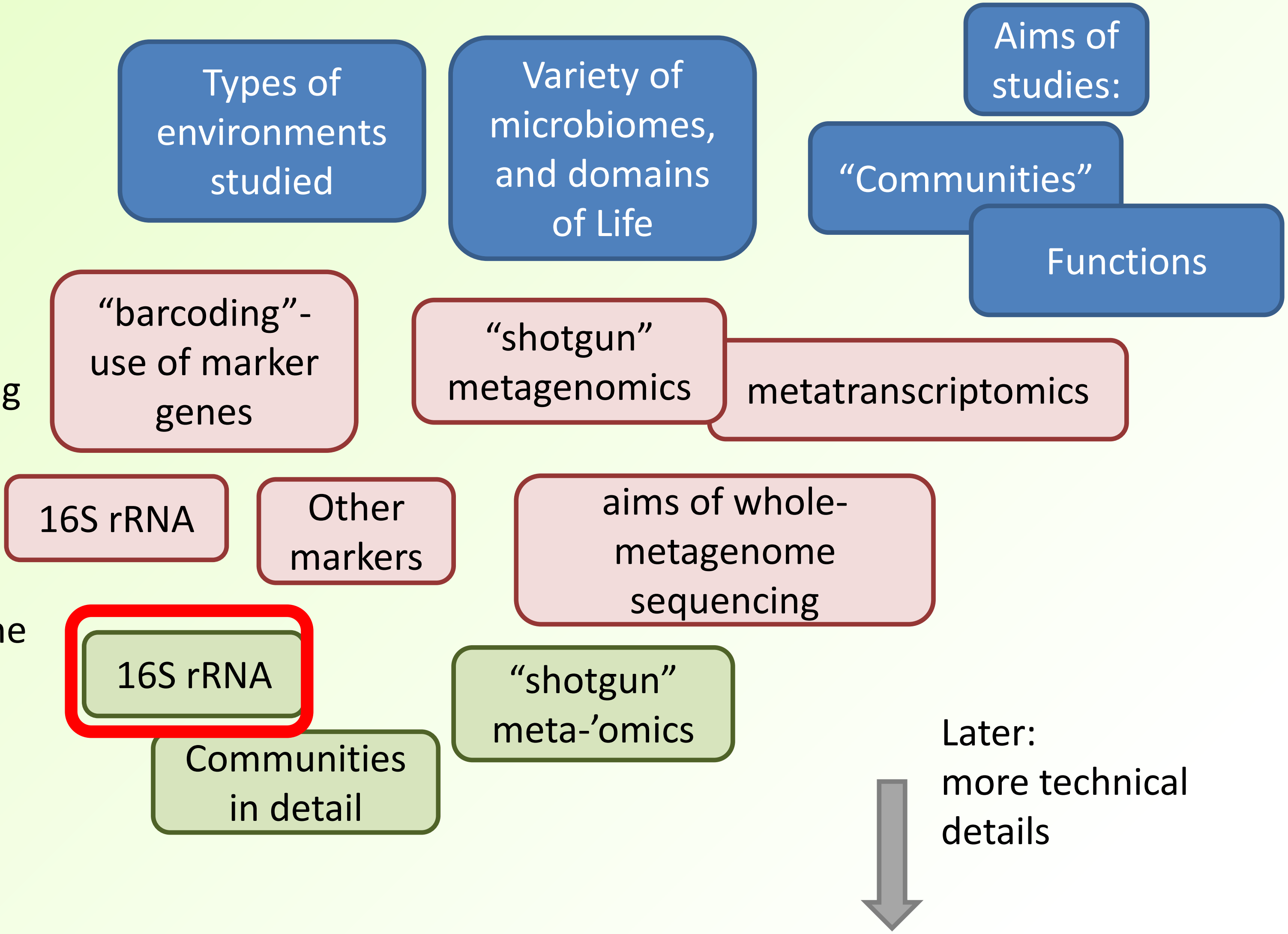
07/04/2017

John Walshaw, GHFS, IFR

Recap: Aims

- **Microbiome analysis**
  - with particular regard to **sequence informatics concepts**
- "Top down" – putting analysis tools and resources in context
- No highly detailed technicalities (yet)
  - No instructions on how to run particular programs
- Why you are using the bioinformatics approaches you use; pros, cons; alternatives

Topics, top-down



Introduction

Types of environments studied

Variety of microbiomes, and domains of Life

Aims of studies:

“Communities”

Functions

Aims of DNA/ RNA sequencing approaches

“barcoding”- use of marker genes

“shotgun” metagenomics

metatranscriptomics

16S rRNA

Other markers

aims of whole-metagenome sequencing

Dealing with the data: Informatics concepts

16S rRNA

Communities in detail

“shotgun” meta-’omics

Later: more technical details

Series of talks

- 5 so far
- Open ended… as long there is demand
- Expected to be every 2 weeks, but all dates will be confirmed in advance
  - *Bite-size bioinformatics mailing list*
- The next few will cover: (*not necessarily in this order…*)
  - 16S analysis for community profiling
  - Clustering and classification issues (taxonomies etc)
  - Analysing richness and diversity of those communities
  - Dealing with sequencing and other errors
- Informal and flexible
  - Please interrupt and ask questions
  - Suggestions for topics for further focus

Series of talks

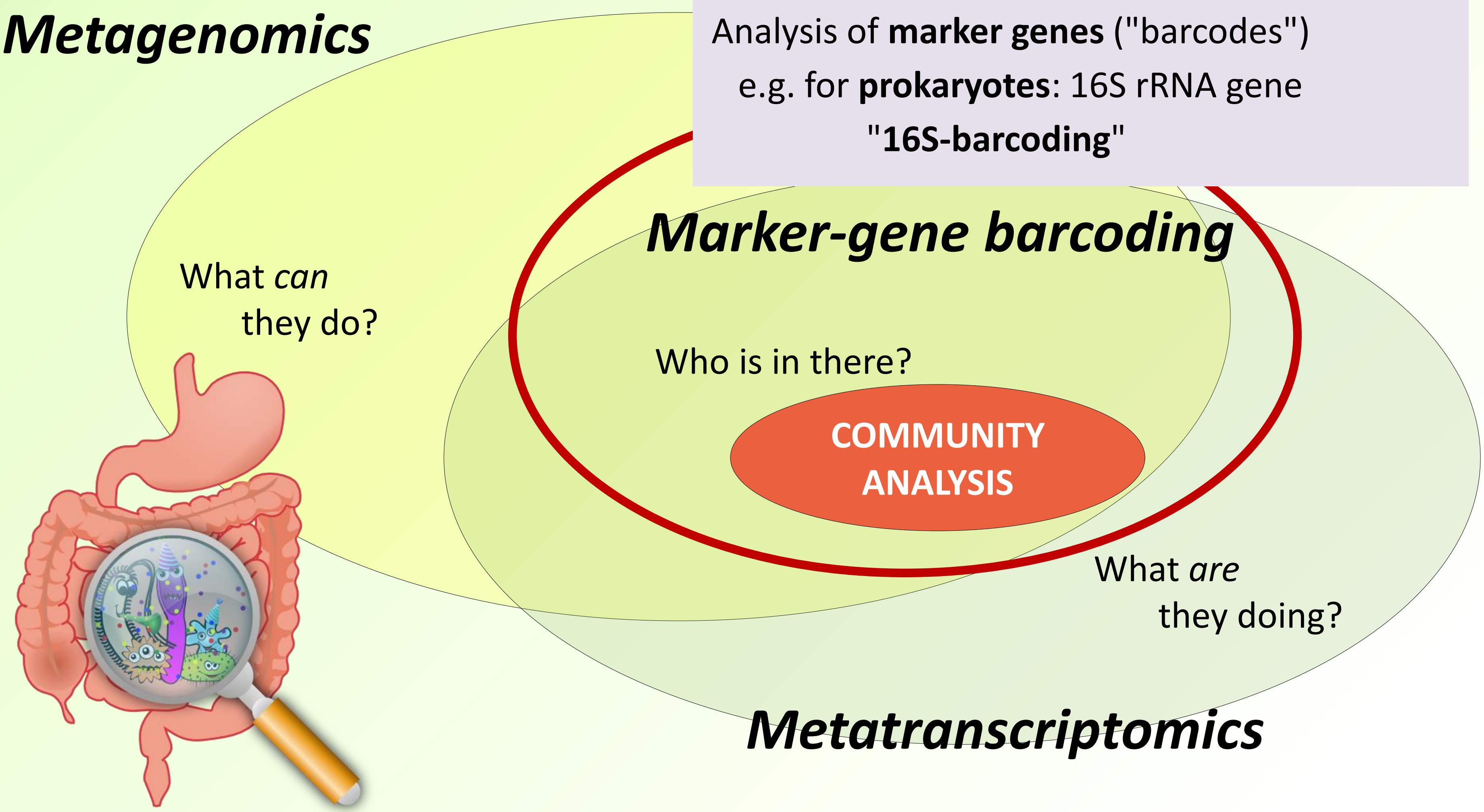
- Part 1: 27/1/2017
  - "Biological and Experimental Stuff that a microbiome bioinformatician needs to know"
  - Overview of marker gene sequencing for community analysis
- Part 2: 10/2/2017
  - Overview of whole-metagenome sequencing
- Part 3: 24/2/2017
  - Focus on metatranscriptomics
- Part 4: 10/3/2017
  - Different bioinformatics approaches to processing 16S read data
- Part 5: 24/3/2017
  - *De novo* OTU clustering: sequence identities and how thresholds have been determined historically; relationships to taxonomic levels
- Slideshows
  - http://ghfs1.ifr.ac.uk/ghfs/

To be confirmed…

- 21st April Rollesby
- 5th May Barton
- **NO SESSION ON 19th MAY**
  - as Student Showcase takes place
- 2nd June Barton
- 16th June Barton

A brief recap

- **Who is in there?**
  - In what amounts?



Analysis of **marker genes** ("barcodes")

e.g. for **prokaryotes**: 16S rRNA gene

"**16S-barcoding**"

Metagenomics

What *can* they do?

Marker-gene barcoding

Who is in there?

COMMUNITY ANALYSIS

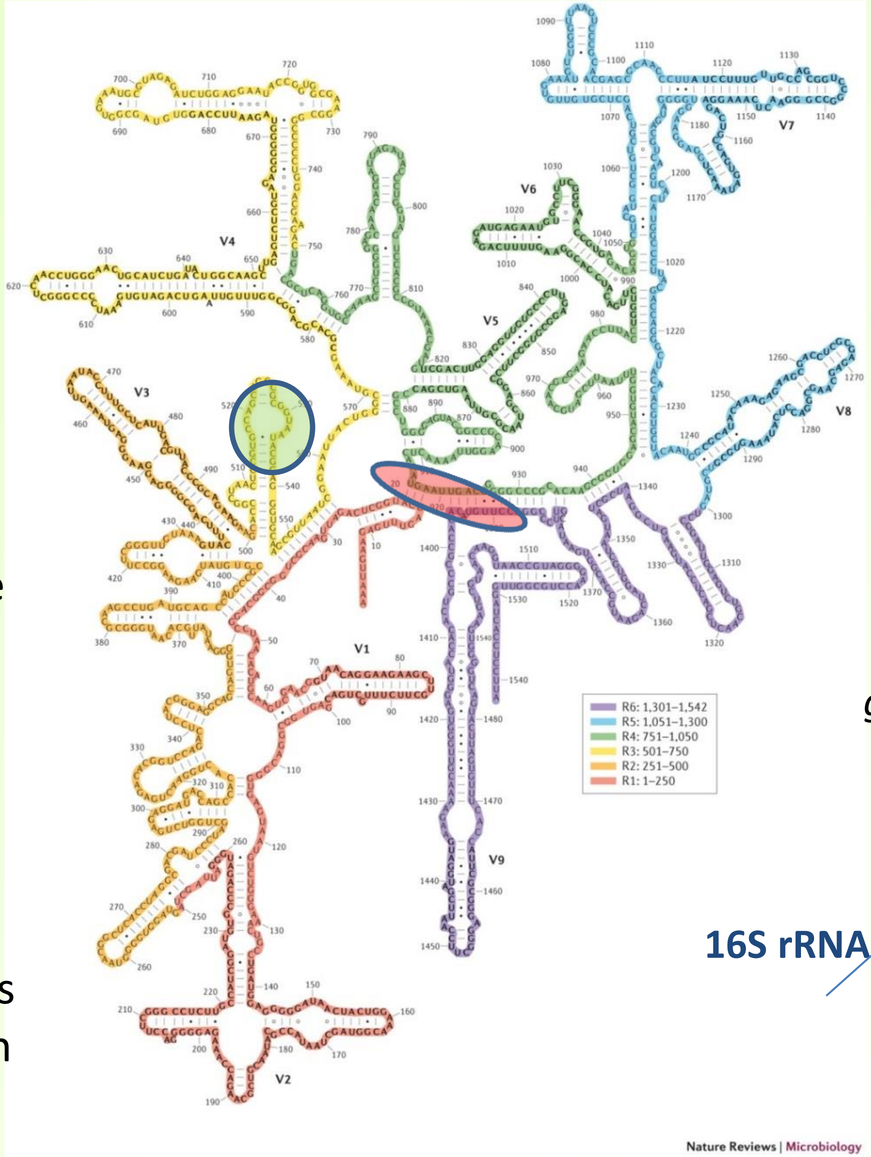
What *are* they doing?

Metatranscriptomics

Amplification of a **segment** of the gene which codes for a **variable** region of the 16S rRNA molecule

→Primers

The variable region is chosen to distinguish between taxa

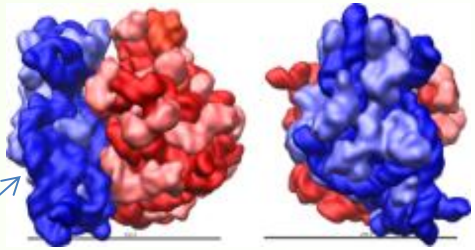


Nature Reviews | Microbiology

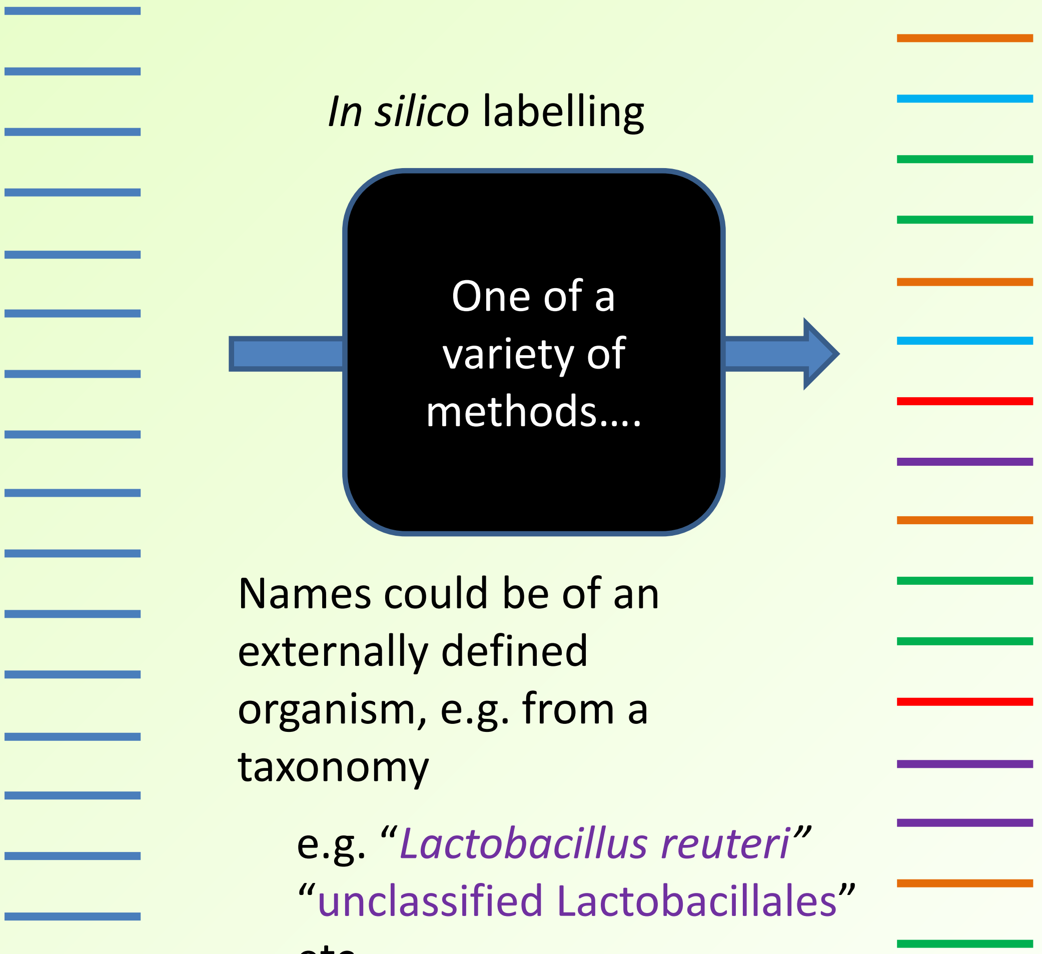
marker gene
("barcode")
for ***phylotypes***

gene which codes for…

16S rRNA



Community analysis by marker-gene sequencing



Raw, unlabelled reads

In silico labelling

One of a variety of methods….

Label to indicate bug of origin

Name1

Name2

Name3

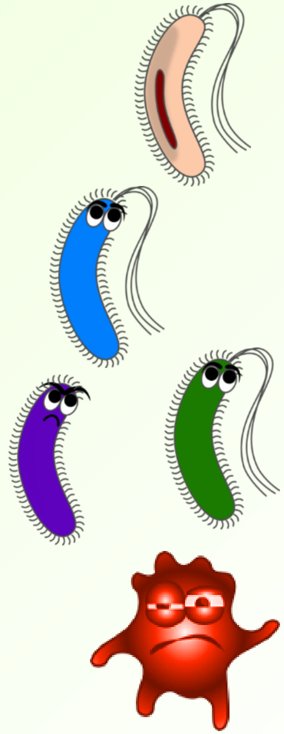
Name3

Name1

Name2

Name4

…etc..



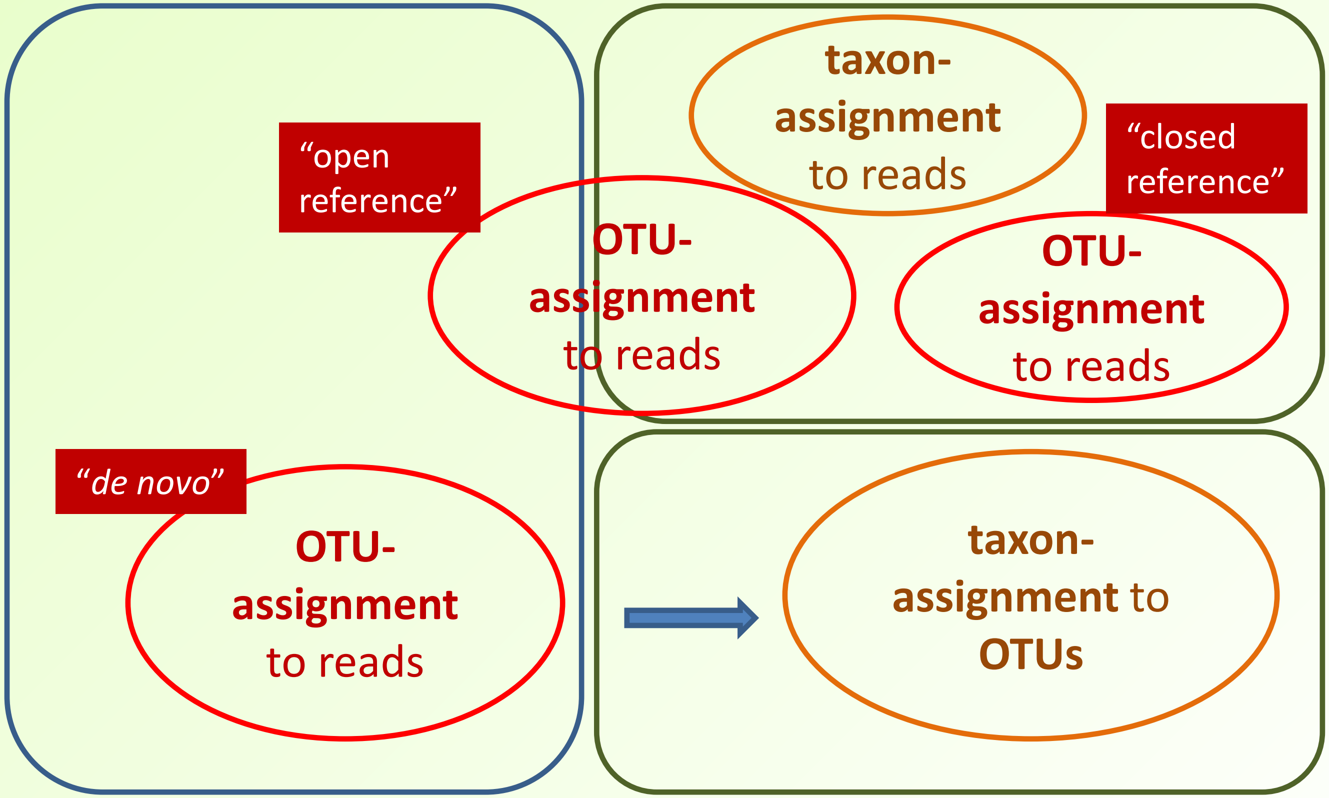
Names could be of an externally defined organism, e.g. from a taxonomy

e.g. “*Lactobacillus reuteri*”
“unclassified Lactobacillales”
etc

Or could be **completely anonymous**, a name existing only within your data e.g. “OTU5432”
- Diversity studies still possible

Clustering : comparing reads with each other ("self-referential")

Using a reference database



"open reference"

OTU-assignment to reads

taxon-assignment to reads

"closed reference"

OTU-assignment to reads

"*de novo*"

OTU-assignment to reads

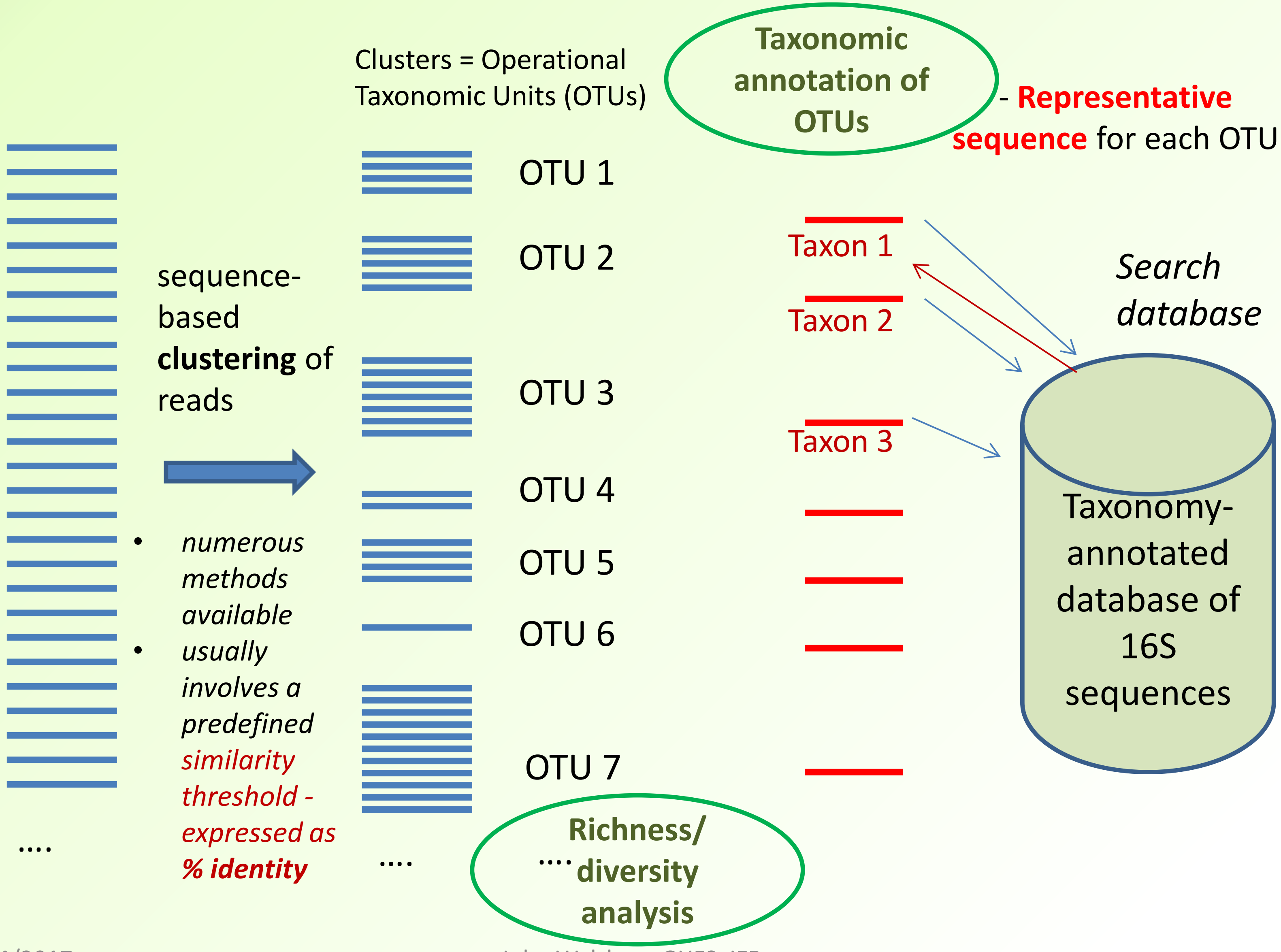
taxon-assignment to **OTUs**

processing each **read** independently

processing representative sequences of each OTU

Different *Operational Taxonomic Units* (OTU) approaches and non-OTU approaches

OTUs by *de novo* clustering (not the only way)



Clusters = Operational Taxonomic Units (OTUs)

Taxonomic annotation of OTUs

- **Representative sequence** for each OTU

sequence-based **clustering** of reads

- *numerous methods available*
- *usually involves a predefined similarity threshold - expressed as **% identity***

OTU 1

OTU 2

OTU 3

OTU 4

OTU 5

OTU 6

OTU 7

....

....

....

Taxon 1

Taxon 2

Taxon 3

Search database

Taxonomy-annotated database of 16S sequences

Richness/ diversity analysis

Previous session….*97*

- **97% sequence identity** is often used as a threshold when comparing 16S sequences
  - Including for assigning 16S reads to **OTUs**
- This is due to its correlation to a threshold in **chemotaxonomic methods** which have long been established in determining differences between **species**
- On that basis, if two 16S gene sequences are <97% identical, it can usually be concluded that they do not originate from the same species
- It **does not follow** that two sequences with ≥ 97% identity belong to the same species

Previous session….*97*

- If two 16S gene sequences are ≥97% identical
  - they might originate from the same species
  - they might not
- there are plenty of examples of **two different species** whose 16S genes are > 97% identical
  - And that's for the **whole gene** sequence
  - The situation for an amplified region might be worse (or better)

Various degrees of [sequence identity] in stretches of 200 nucleotides along the primary structure of pairs of 16S rRNAs from organisms with different degrees of relatedness (after Stackebrandt & Goebel, 1994)

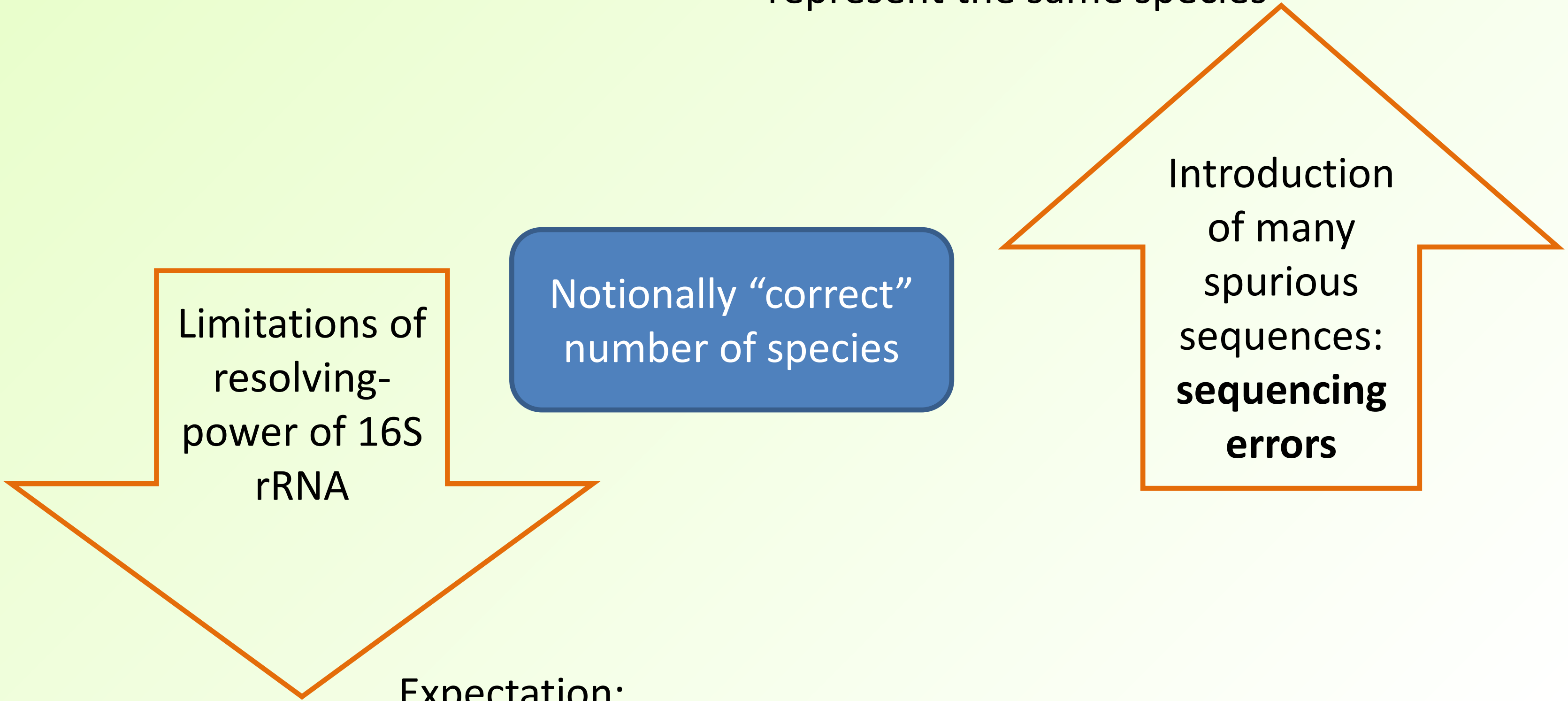
Position	**16S rRNA sequence identity (%) between:**		
	Streptomyces ambofaciens and *Streptomyces violaceoruber*	*Mycobacterium phlei* and *Mycobacterium tuberculosis*	*Aeromicrobium erythreum* and *Rhodococcus fascians*
Overall	98.8	96.4	90.9
0-200	**96.3**	94.1	**80.7**
201-400	98.4	97.8	94.6
401-600	100.0	93.1	94.6
601-800	99.0	97.9	85.7
801-1000	100.0	100.0	94.0
1001-1200	98.9	**92.8**	90.0
1201-1400	99.5	100.0	94.0

Approx. position of V4-V5 ampl-icons

Previous session….*97*

- Often in the literature, there is an implicit assumption that OTUs represent species
- But given the relationships described, one should expect many instances of different species being put into the same OTU
  - This is not an "error" in the methodology
  - Simply a limitation of the 16S gene sequence – especially shorter segments of it – to resolve different taxonomic groups
- And yet….
  - Many 16S data sets resolve to a very high number of OTUs
    - (but can depend very much on how the OTU-assignment is done)
  - A much higher number than might be expected for the number of species
  - This seems to contradict the above expectation – why is this so?

Expectation:
OTU-count > true species count; many OTUs may represent the same species



Limitations of resolving-power of 16S rRNA

Notionally “correct” number of species

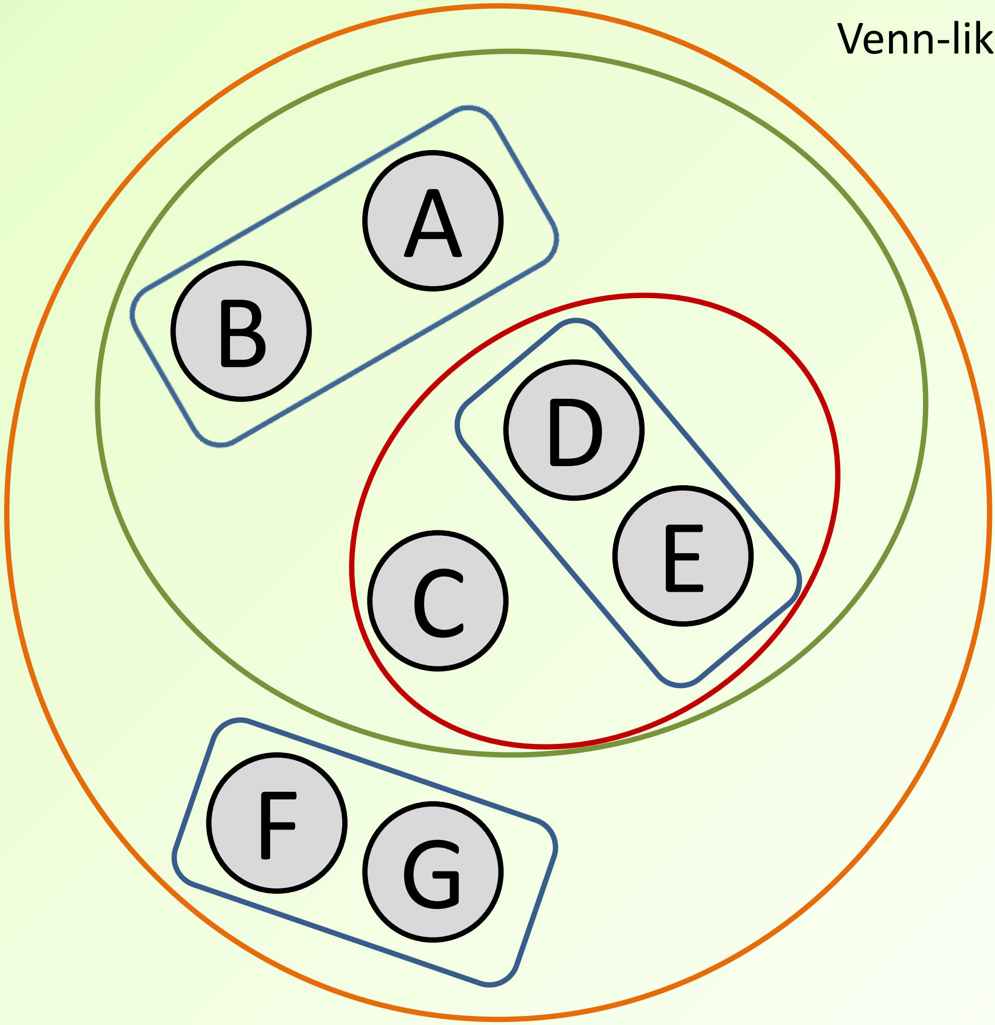
Introduction of many spurious sequences: **sequencing errors**

Expectation:
OTU-count < true species count;
an OTU may represent > 1 species

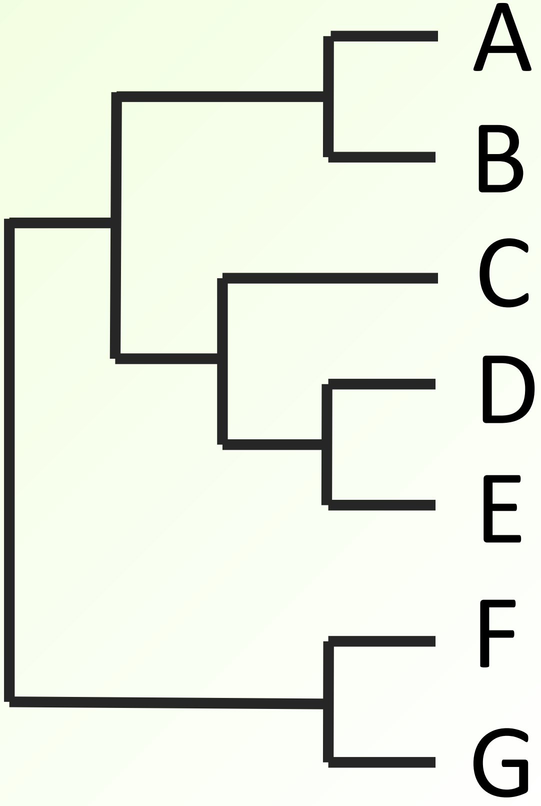
Clustering

- Grouping of any groups of items based on similarities/differences
- Many different methods
- Some are **hierarchical**
  - These may be involved in some downstream analyses that you perform in 16S analysis
- Some are not hierarchical
  - These include **methods you are likely to use for OTU-assignment**

Visualising hierarchical clusters

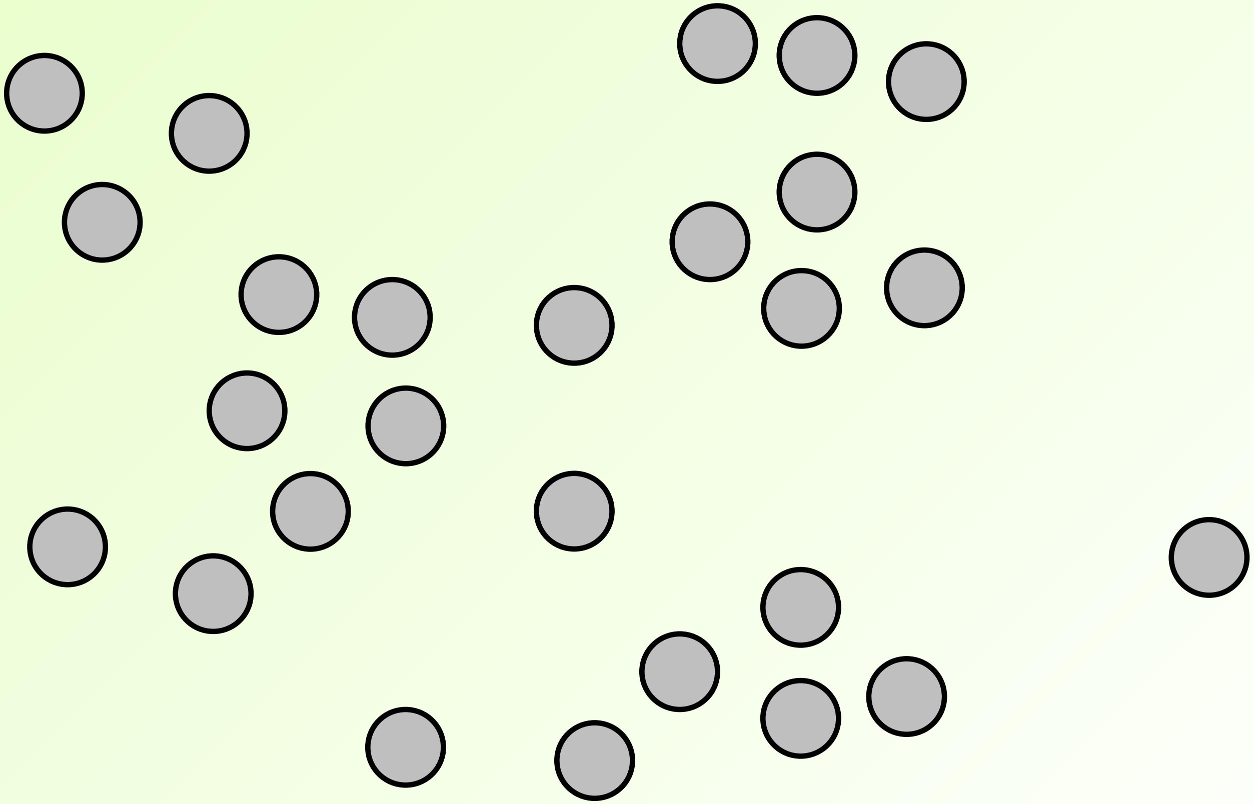


Venn-like

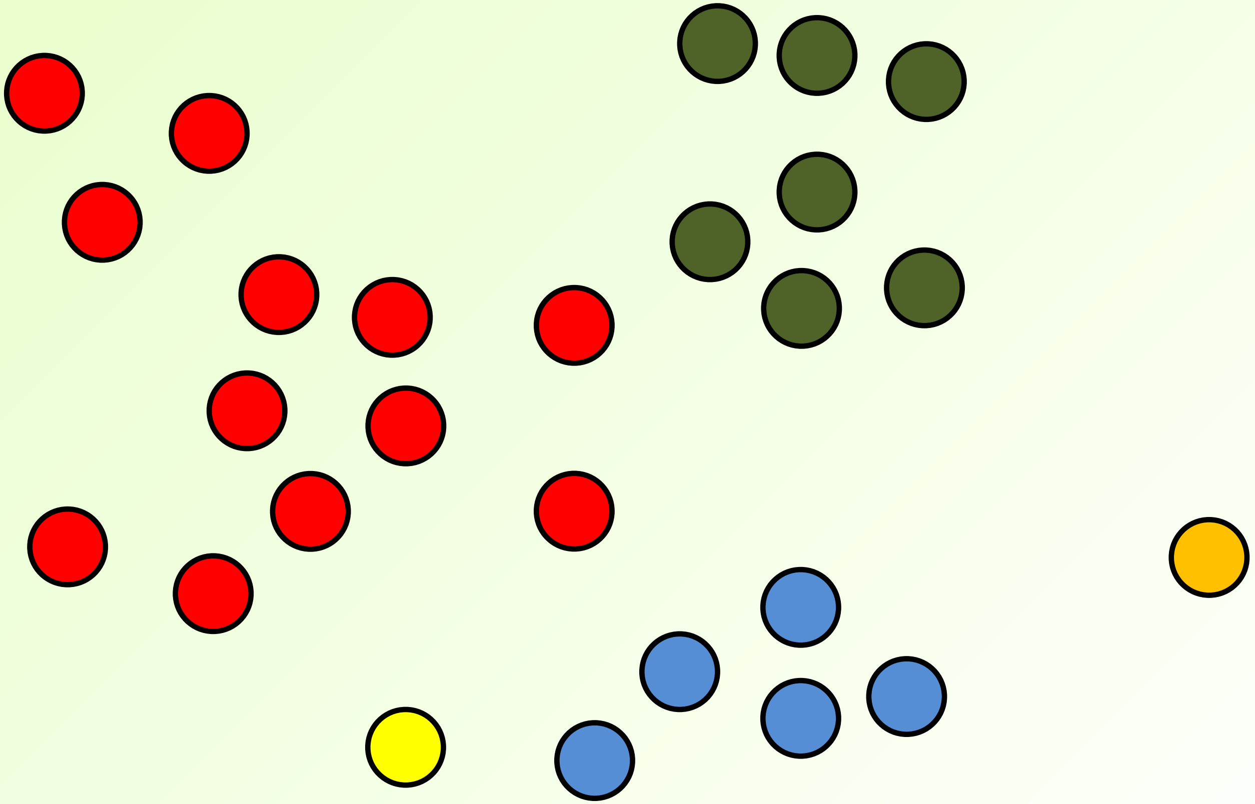


Tree-like

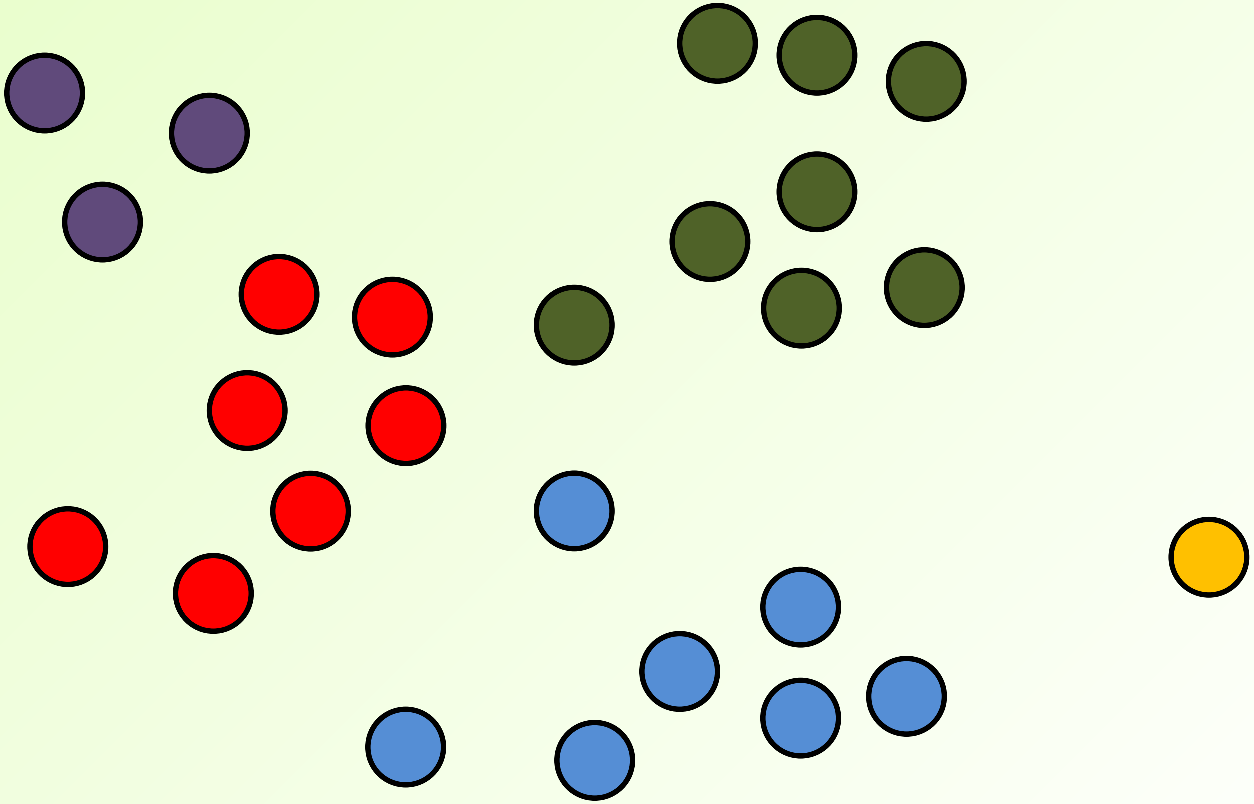
Non-hierarchical (“one level only”)



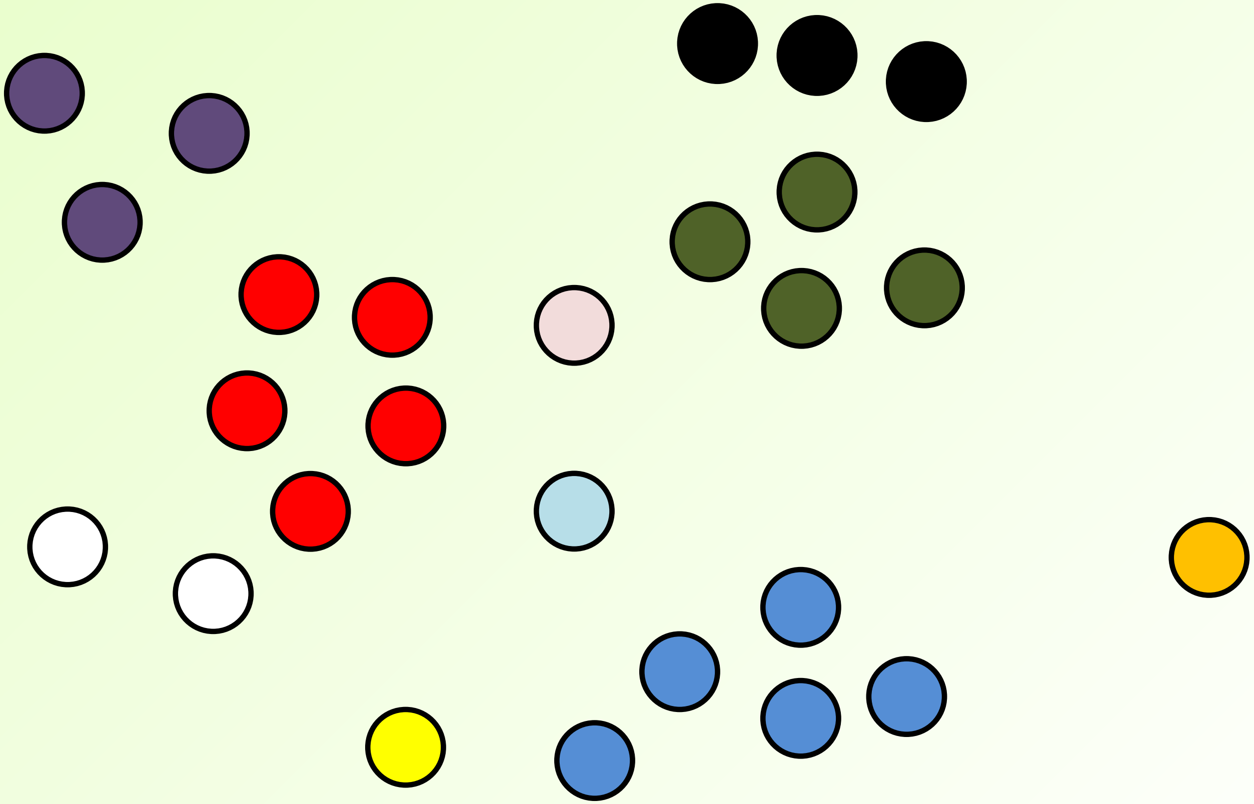
There is no single “correct” solution….



….that's true of typical 16S data too…

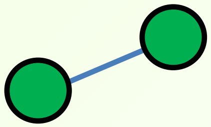


…different algorithms and parameters will give different answers

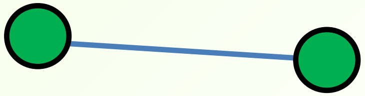


Depicting differences between DNA sequences in 2D….

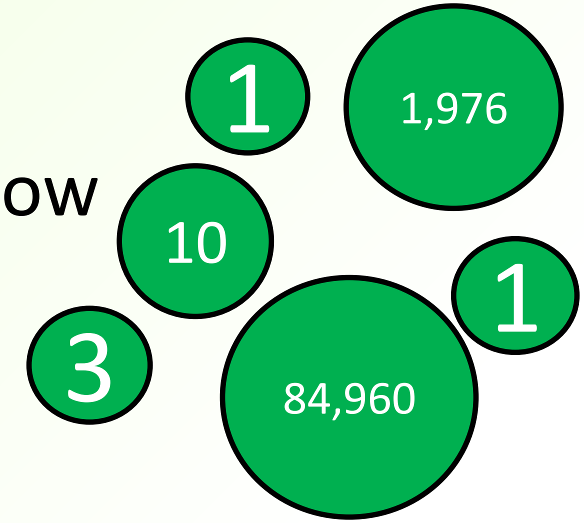
- (or RNA or protein sequences)
- Simple enough - use sequence *differences* as a measure of 'distance'



- Greater distance = lower % sequence identity



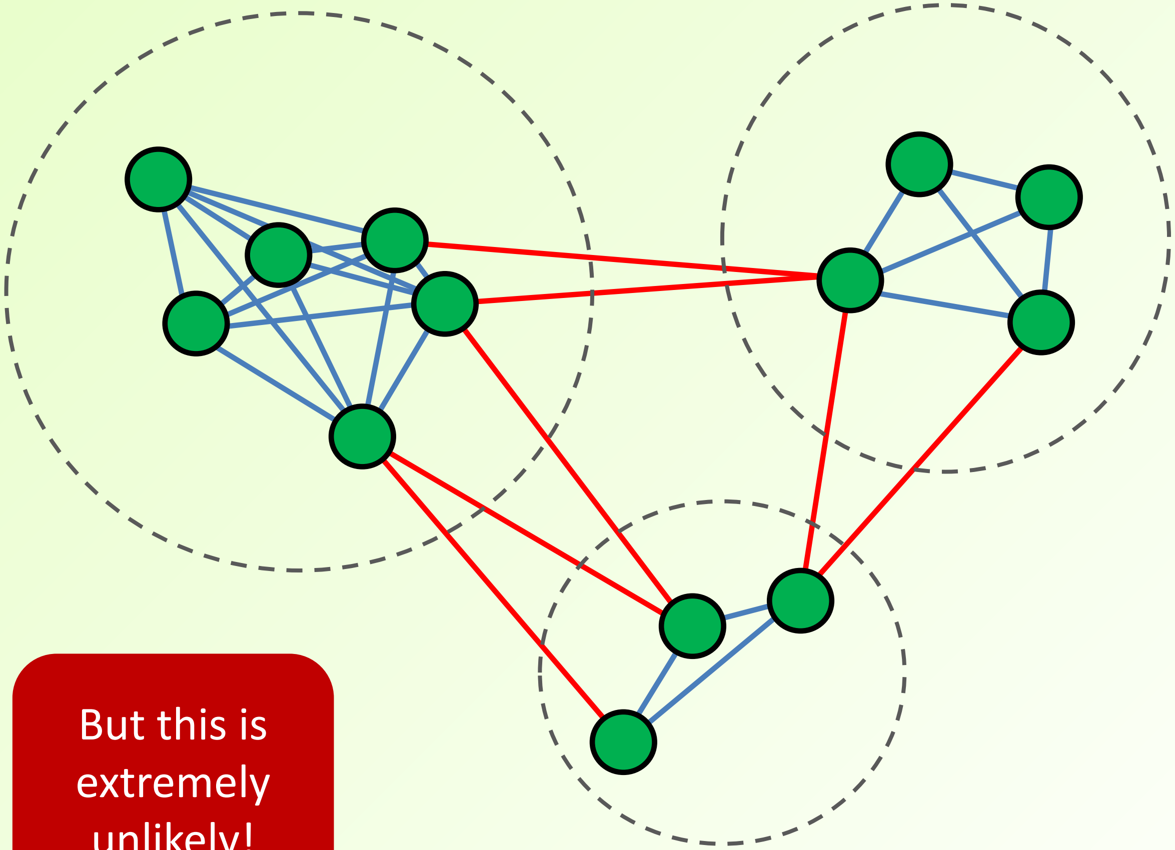
- Each blob is a sequence read
  - **or > 1 identical reads**
- So following figures could also show numbers of **100% identical** reads there are



Clustering reads into OTUs

- Goals:
- 1) Put every read into a cluster (= OTU)
  - Thus: 1 ≤ number of clusters ≤ number of reads
- 2) For each OTU, select a single sequence to be the representative
  - In practice, this is always one of the actual sequences in the cluster (OTU)
    - In some papers/algorithms, this is referred to as the 'centroid'
    - (But in many of these algorithms, isn't the centroid in the strict sense)
  - An alternative would be to use a **consensus** sequence – which may or may not be the same as one of the actual sequences
    - (and may or may not be a real centroid)
    - I'm not aware that this is used in marker-gene analysis; could have some dangers
    - Consensus sequences **are** used in some other completely different types of sequence analysis however

Hypothetical perfect scenario

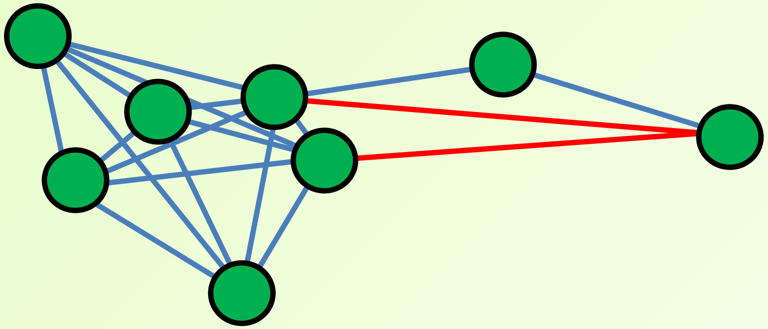


🟢 = one sequence

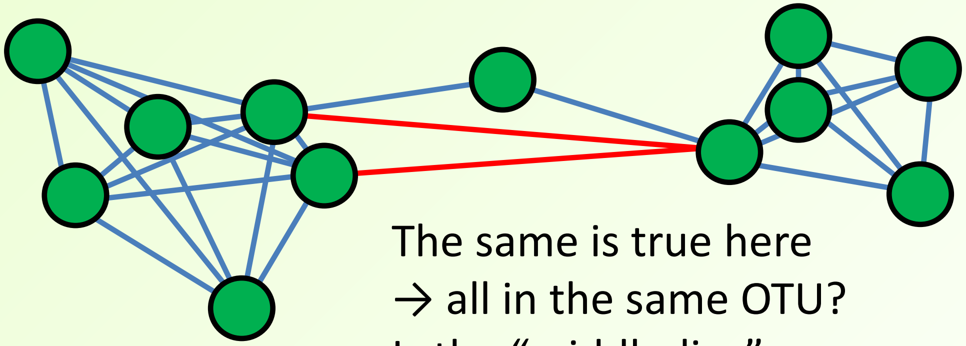
🟦 = ‘distance’ of ≤ 3% difference in sequence identity
All connected sequences are ≥ 97% identical

🟥 = ‘distance’ of > 3% difference in sequence identity
All connected sequences are < 97% identical

But this is extremely unlikely!



Every read is ≤ 97% identical to at least one other read
→ all in the same OTU?



The same is true here
→ all in the same OTU?
Is the “middle-lier” suspicious?
How should an algorithm treat this?

The resulting number and membership of clusters depends on the algorithm used

Greedy and non-greedy algorithms

- Clustering – a step-by-step process
- Greediness versus non-greediness applies to very many types of algorithms
- **Greedy:** for each step (e.g. the next read sequence) make a decision immediately, based on the information known so far
- **Non-greedy:** decisions may be delayed until more (perhaps all) data has been assessed
  - Then use the total information to compute the best decision
  - N.B.: a non-greedy algorithm isn't necessarily the absolute "best" (globally optimum)

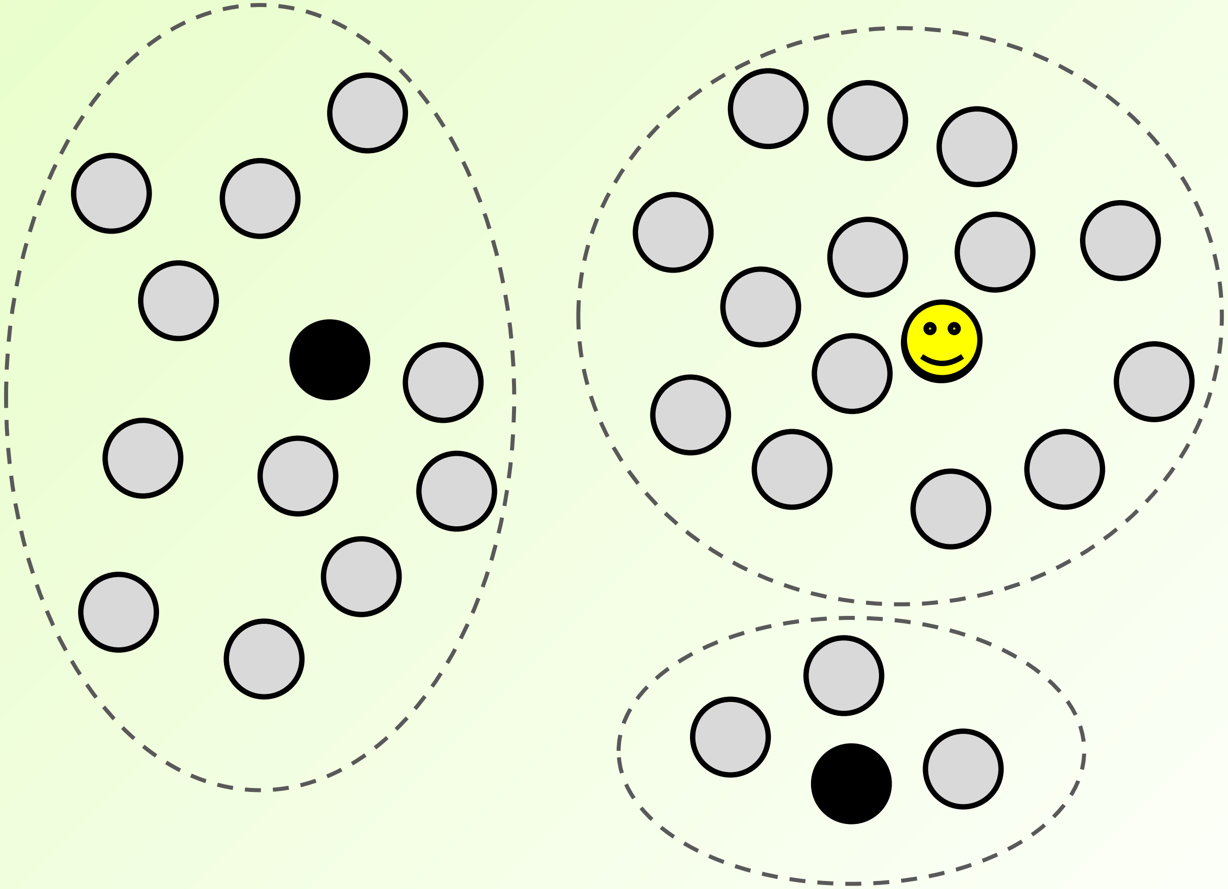
Example non-greedy approach to *de novo* OTU-clustering

- Compare every pair of reads to compute all read ↔ read distances
- Then build optimal set of clusters from the resulting data
- You probably want to avoid this, if you have a total data set of say, 20 million 16S reads
  - (not uncommonly large these days)
  - That would require just under 2 x 10^{14} comparisons
  - And thus 2 x 10^{14} distances to build your clusters from
  - By coincidence, about the same number of prokaryote cells in/on the human body…(give or take an order of magnitude…)
  - You will benefit from *absolutely the best answer*
  - (in several years/decades depending on the hardware you run it on)

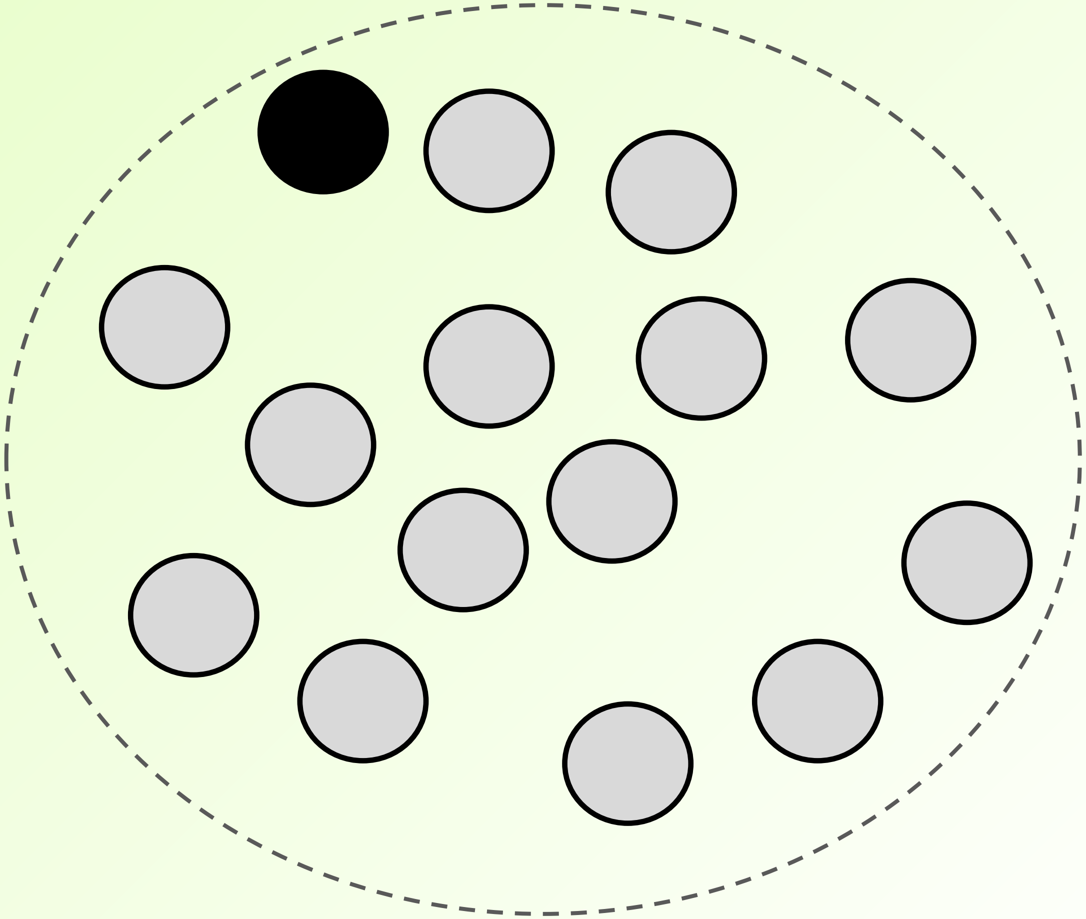
Example greedy approach to *de novo* OTU-clustering

- Read #1 forms the first cluster (Cluster #1), and is its centroid (Centroid #1)
- If Read #2 is similar enough (≥ *x*% identical) to Centroid #1, then add it to Cluster #1
  - Otherwise, Read #2 forms a new cluster (Cluster #2)
- Repeat this for all reads:
  - Compare read with Centroid #1
  - if match is good enough, add read to Cluster #1
  - If not, make same comparison with all other Centroids in turn, until a good enough match occurs; add the read to corresponding Cluster
  - If no matches occur, the read becomes the centroid of a new Cluster

Example: A comforting result



Order matters with greedy algorithms

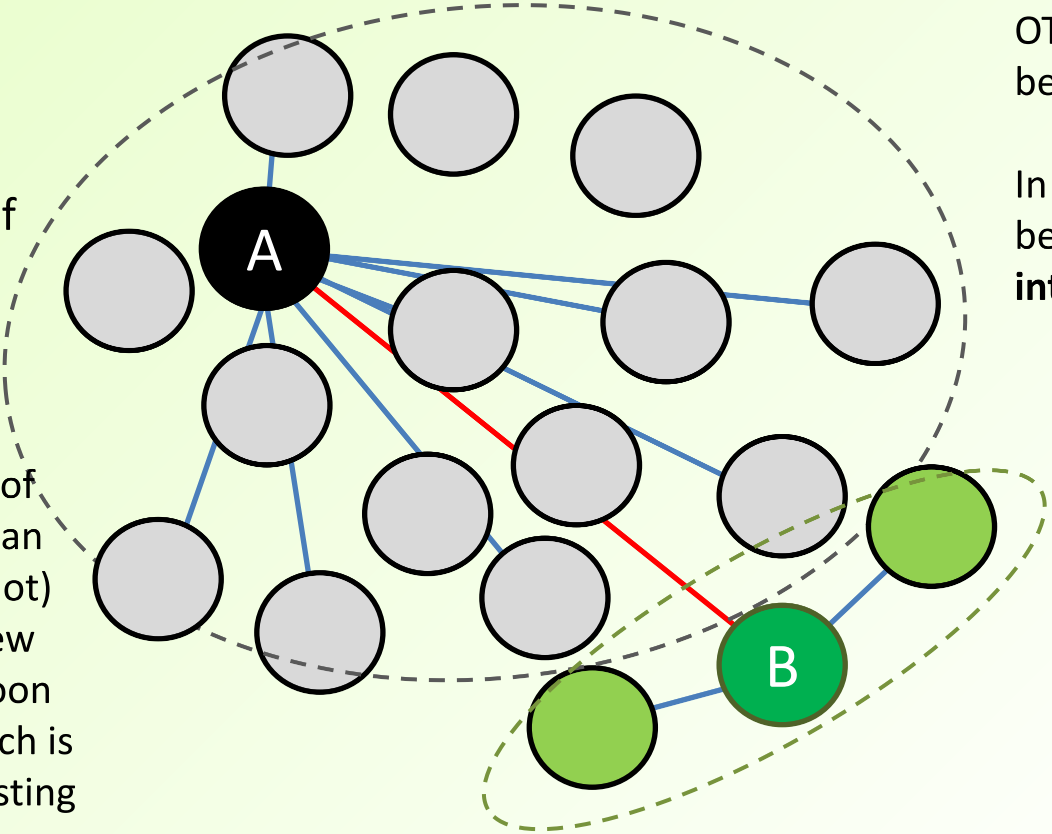


We might prefer to avoid this –
The selected sequence is one of the least "representative"

Order matters with greedy algorithms

First sequence A encountered
→ centroid of new OTU

Partial greediness…. Different parts of the algorithm can be greedy (or not) E.g. for each new read, stop as soon as a ≥ 97% match is made to an existing cluster?

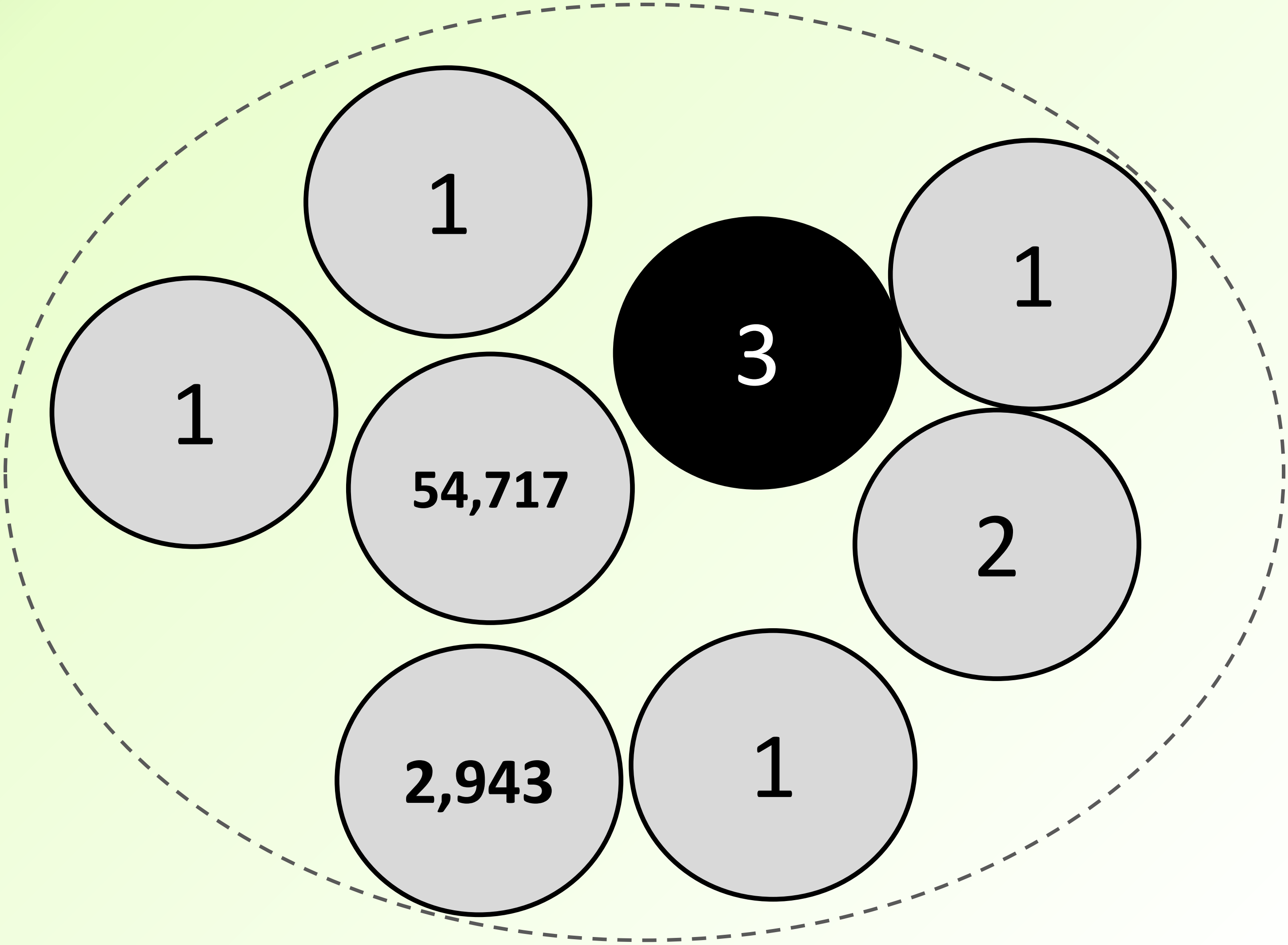


Many sequences might be equally well assigned to either OTU; but have not been (=OTU size bias)

In any case: might be better **clustering all into a single OTU**

Later sequence B encountered: < 97% identical to centroid A→ B becomes centroid of new OTU

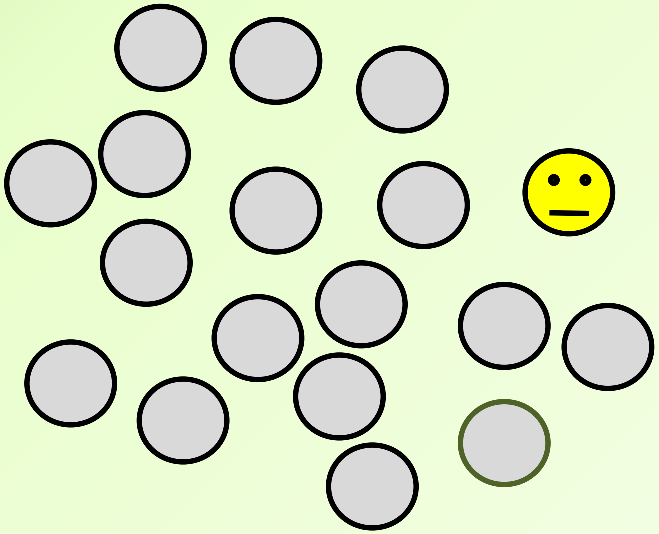
Order matters with greedy algorithms

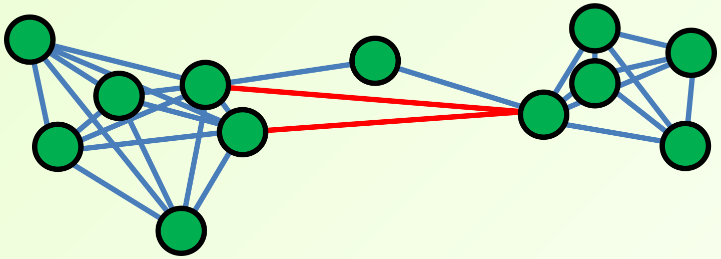


One to avoid: It is of course preferable to use a very commonly-occurring sequence as the representative

Process the reads in the right order

- A lot of the problems with greediness can be improved:
- **Pre-sort** the reads in a meaningful order; e.g:
  - Most abundant sequences first
  - Or
  - Longest reads first
    - (shorter reads are however likely to be less abundant; and in some approaches they may simply be discarded)
- This pre-sorting can be achieved relatively quickly even with a huge dataset
  - Including by use of greedy algorithms
    - (which work perfectly for this particular purpose)





Are these examples a bit contrived?

Do we really get such a spread of sequence reads?

If so, why?

Causes

- Genuine biological variation
  - Between species
  - Between strains
  - Other biological variation within the population
- Experimental artefacts generating sequences which were not in the sample
  - Mainly: chimeras caused by amplification
- Sequencing errors

Sequencing errors

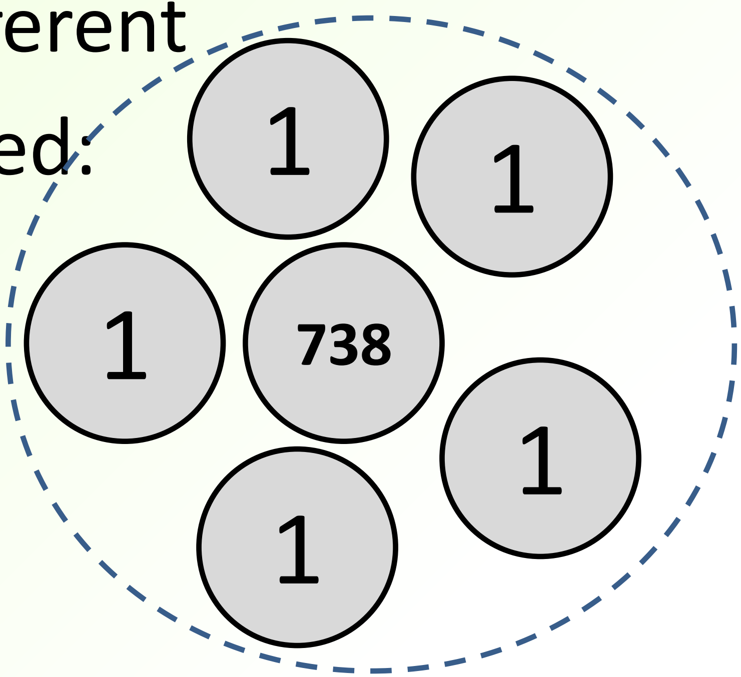
N.B.: everything in the preceding slides assumes that the reads have **already been quality-screened** (because that's a pretty fundamental thing to do with any set of read data)

Post quality-screening, loads* of your base calls are still wrong

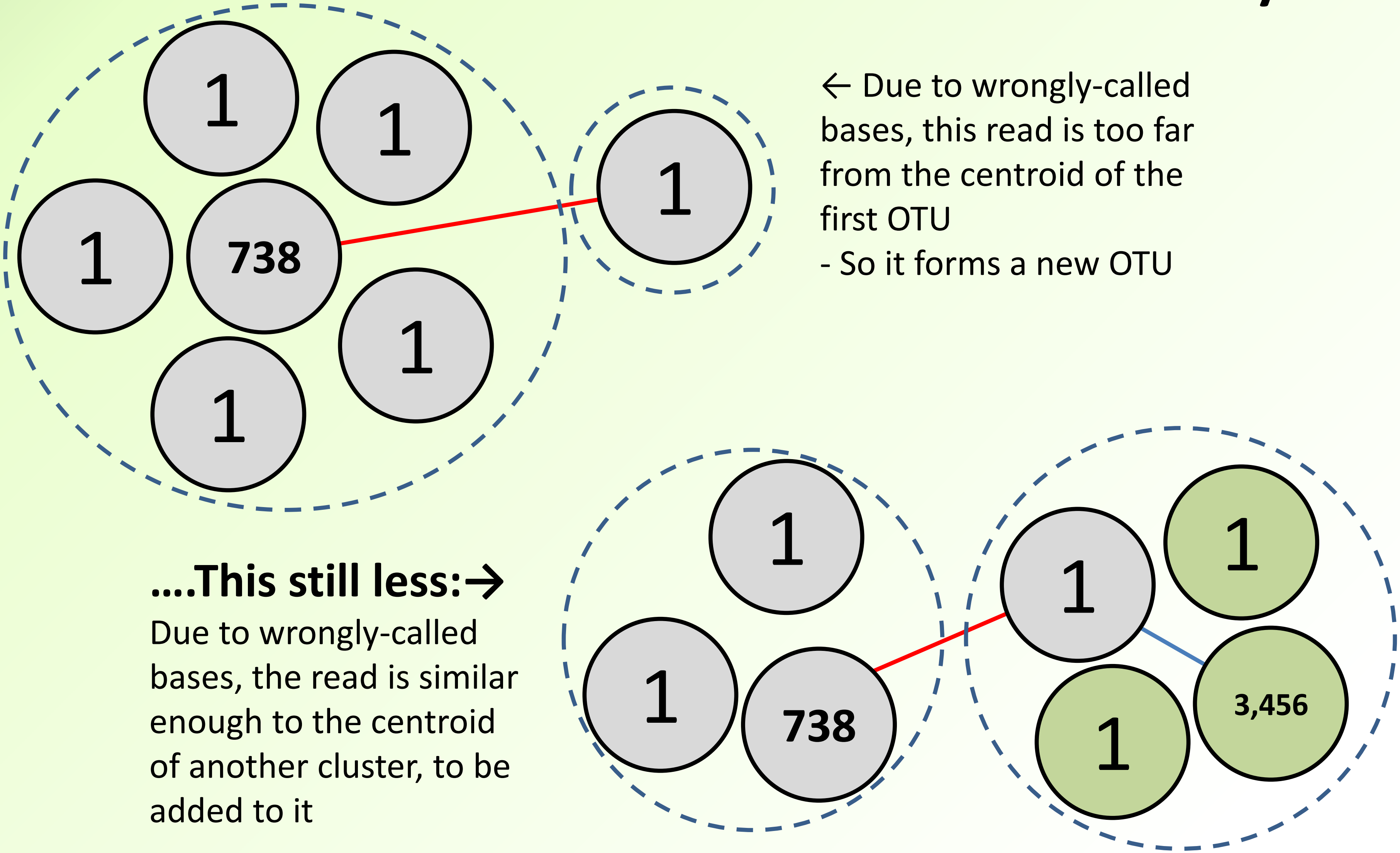
- *In absolute terms, in a very large data set
- On average:
  - 1 in 10,000 of the bases with a quality score of 40
  - 1 in 1,000 of the bases with a quality score of 30
  - 1 in 316 of the bases with a quality score of 25
- E.g. 250 b.p. reads:
  - if (hypothetically) all base calls had Q=30, that’s one wrong base for every 4 reads - ***on average***
  - in a large data set, numerous reads will have 1, 2, 3, 4… miscalled bases

Let’s assume 250 b.p. reads

- (Usually, sequenced as paired-end; joined reads typically could be slightly longer)
- For a sequence identity of ≤ 97%, two 250-b.p. reads must have ≥ 8 b.p. different
- So lots of this can be expected:
- (recall that the numbers are **instances** of an **identical** sequence) ① = ‘singleton’



Whereas this should be less likely…



← Due to wrongly-called bases, this read is too far from the centroid of the first OTU

- So it forms a new OTU

….This still less:→

Due to wrongly-called bases, the read is similar enough to the centroid of another cluster, to be added to it

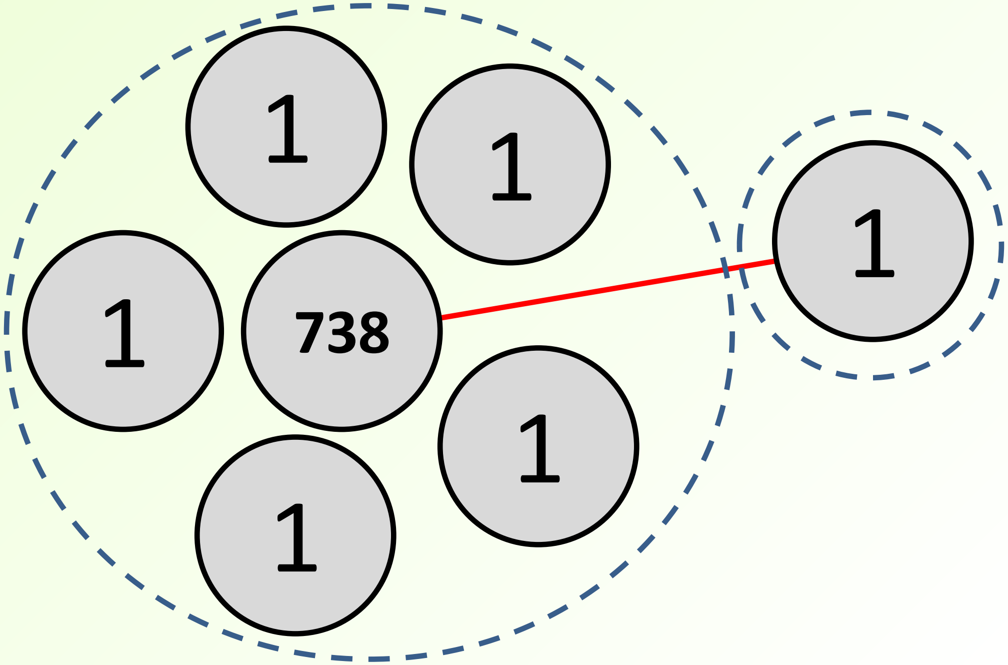
- Miscalled bases are relatively unlikely to “transform” many 16S sequences (amplicon thereof) into the 16S sequence of another organism
- But it can happen
- Recall that it’s possible to change the 16S segment of one organism into another…
  - …by making **zero** changes
    - (cases of 2 different species with some of their variable regions identical)
  - Making 1 b.p. change, or 2 b.p. etc, can also do this
    - If they are in the right places ; usually won’t be
  - But in most cases, these ‘transformations’ **won’t change the OTU assignment** of these reads (difference still < 3%)

However – sequencing errors are not random

- Evaluating the consequences of error probabilities alone (from quality scores) ignores this problem:
- Errors are more likely to occur in some places than others, due to local sequence context
- E.g. 454 platform:
  - More likely to be erroneous extra bases in homopolymers
- Illumina:
  - Poly-G and other G-rich regions can have a higher frequency of miscalls (e.g. Minoche *et al.* (2011) *Genome Biology* 12:R112)
  - A different problem is sequencing bias favouring (higher coverage of) GC-rich regions

This non-randomness
Results in some reads having a higher concentrations of miscalled bases than would be expected by chance

So this sort of thing is **more likely**: →

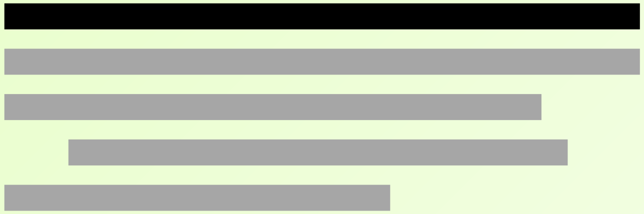


Quality-trimming shortens many reads

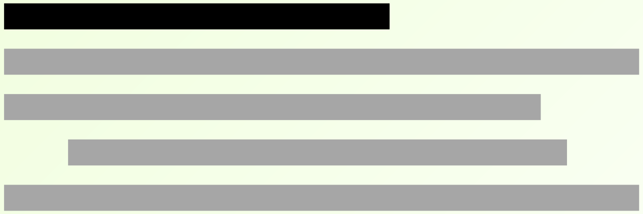
Different ways of dealing with this

Some consequences

Different-length reads in clusters



Longest read selected as representative ✓



✗

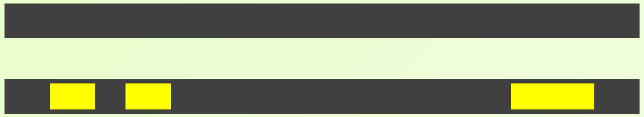
Sorting all reads prior to OTU-clustering helps to avoid problems

E.g. sorting by length

Or sorting in order of abundance – as full-length reads should be more common

Some algorithms require the reads to be pre-sorted

- Or for all reads to **be trimmed to the same length**, and **shorter reads to be discarded**



Centroid of cluster 1

Centroid of cluster 2

- differs significantly from centroid 1

🟨 = differences

Shorter reads could exactly match both centroids

- And so be equally well placed into both

Some greedy algorithms would assign **all** of the shorter reads to whichever of (1) or (2) was encountered first

- Which is why they may insist on using reads **trimmed to the same length**, with **shorter reads discarded**

A far worse problem still

[IMAGE: “cut-and-shut”
see https://firstcar.co.uk/news/two-for-one]

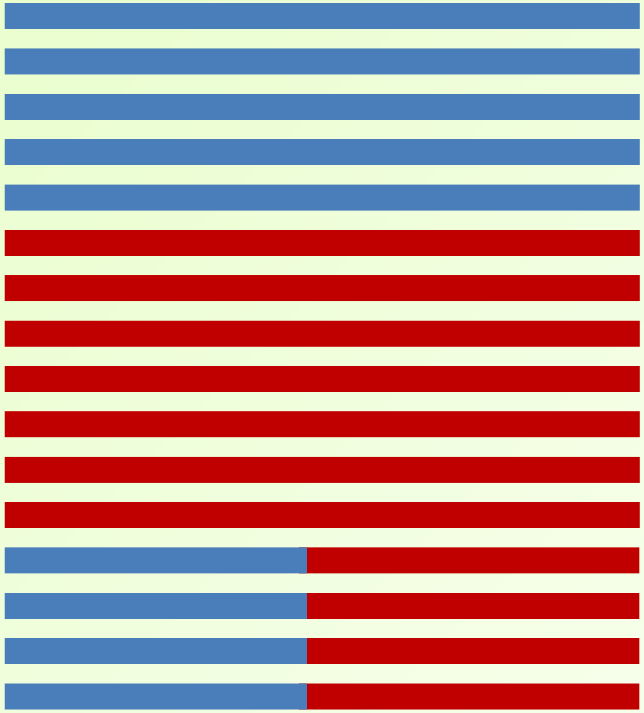
Chimeric artefacts

- In this context, **chimeric sequences** are artefacts of the **amplification process** (PCR)
- A chimera usually consists of two halves of the real, biological sequences joined together
- Chimeras formed of segments from > 2 original sequences also occur
- Chimeras can themselves be amplified

Chimera frequencies

- Chimera frequencies can be platform-dependent
- - including screening procedures in the sequencing software
- Sanger and 454 platforms: considerable variation in frequencies in 16S datasets
  - A few % of reads, up to almost 50%
  - e.g. Haas *et al.* (2011) *Genome Res.* **21**, 494-504
- 16S on Illumina platforms: frequencies much lower
  - current datasets – evidence for chimera in << 1% of reads
  - Still potentially a big problem for a large dataset
  - Many (not all) chimeras will create additional OTUs

De novo chimera detection



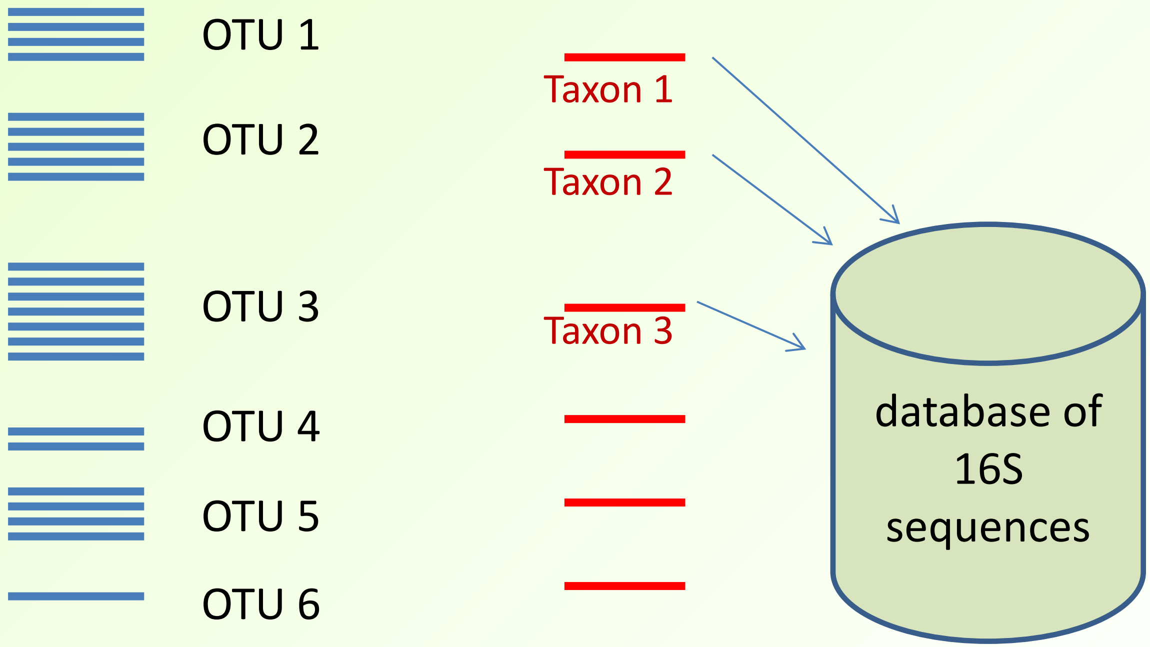
- 2 distinguishable groups of read sequences :
- Within groups, reads are highly similar (or identical)
- Larger differences between groups
- Sequences which are identical or near-identical to part, and only part, of other reads in the data set

→ chimeras

Chimera detection using reference sequences

Clusters = Operational Taxonomic Units (OTUs)

- **Representative sequence** for each OTU



OTU 1

OTU 2

OTU 3

OTU 4

OTU 5

OTU 6

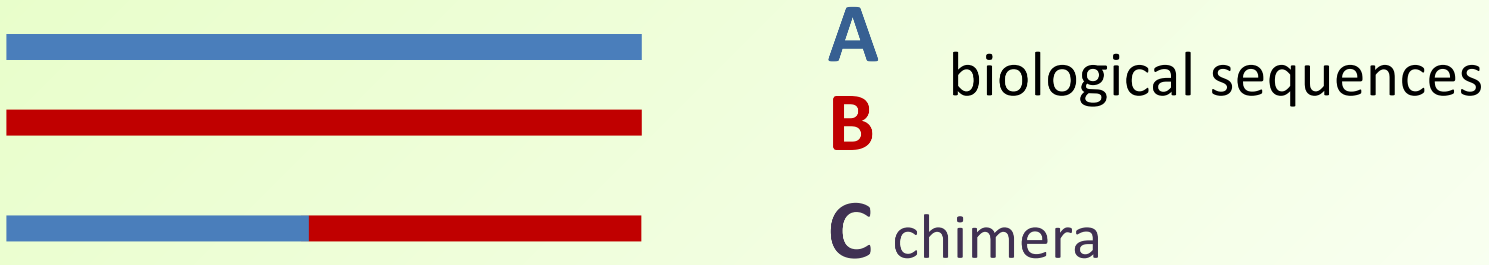
Taxon 1

Taxon 2

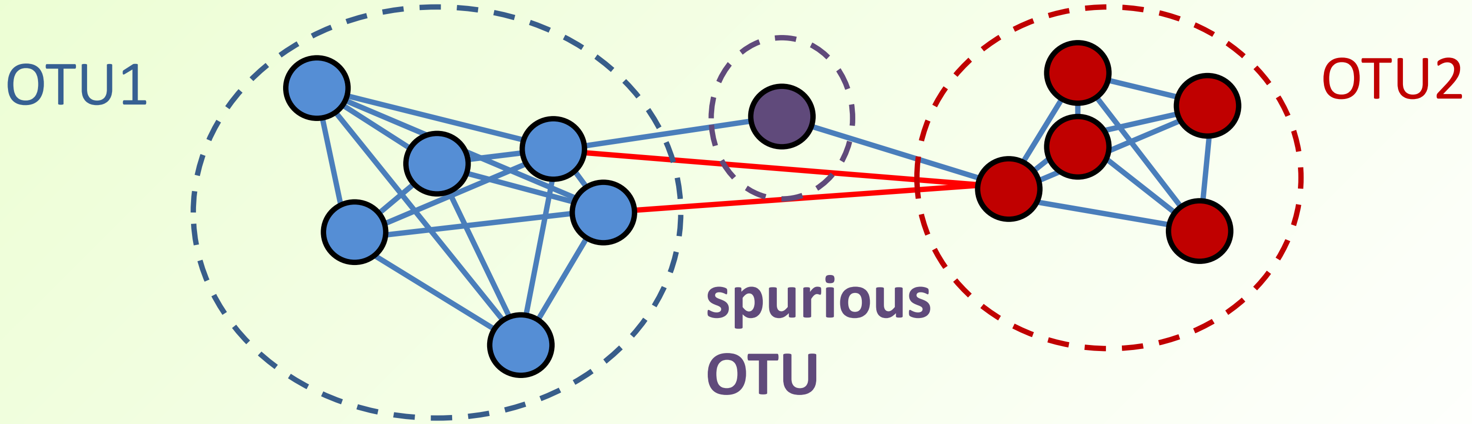
Taxon 3

database of 16S sequences

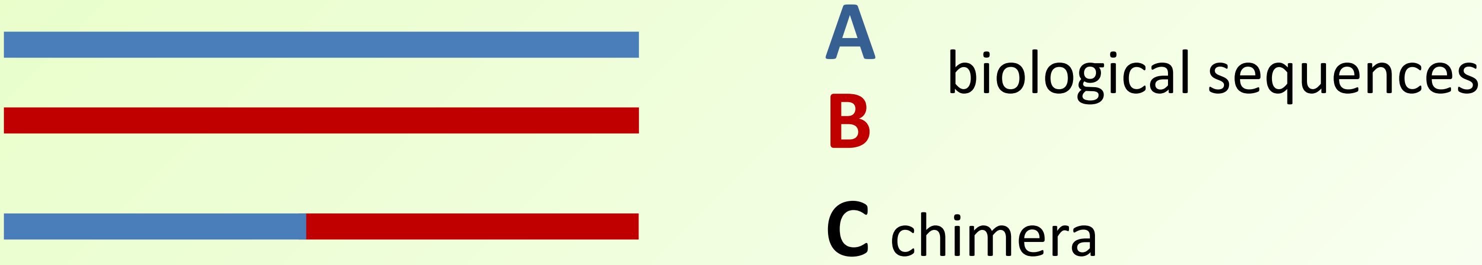
These differences are often **large**



- - clearly can give rise to these situations



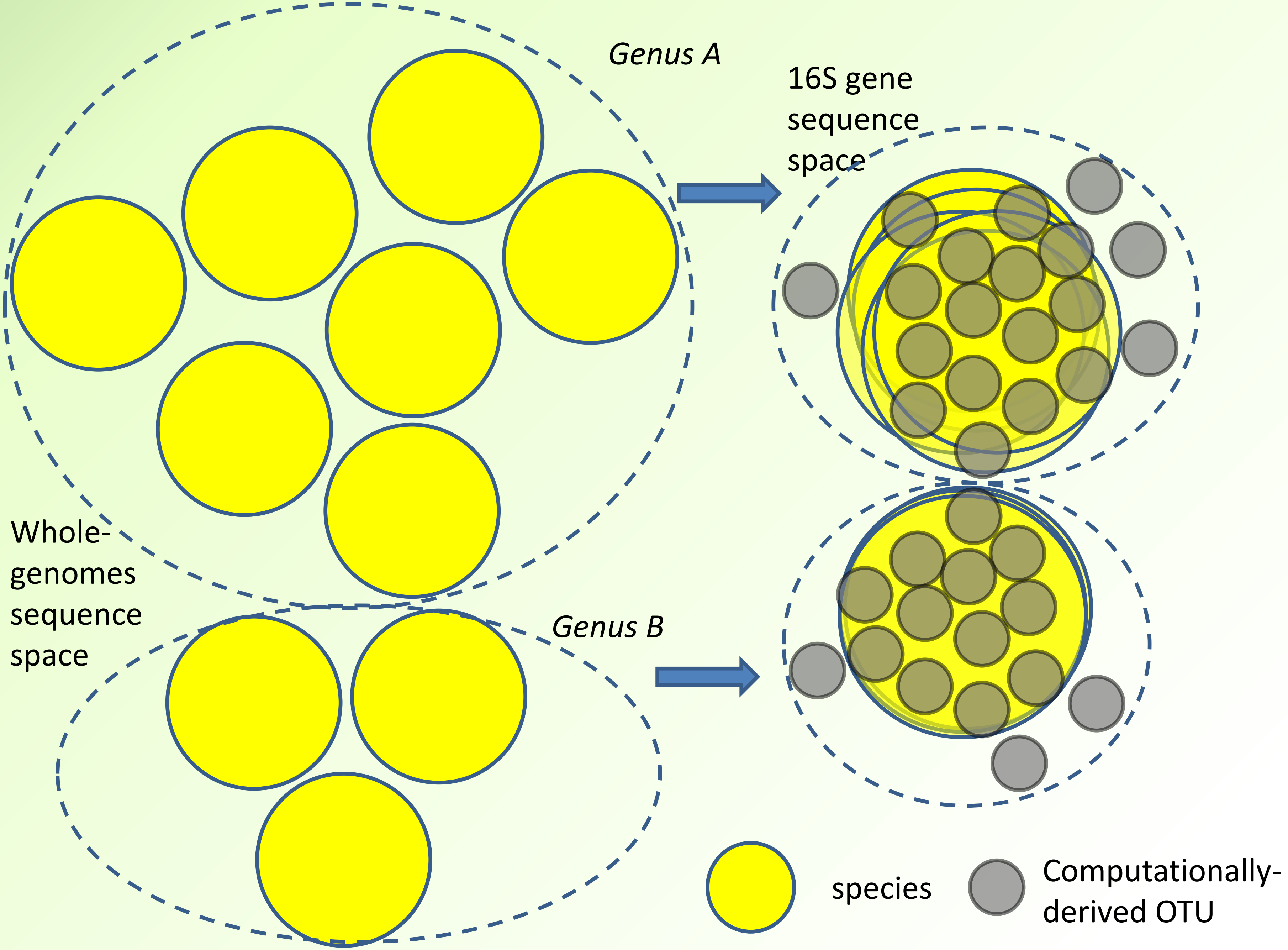
- but can be small



A
B biological sequences

C chimera

- If A and B are very similar, e.g. ≥ 97%, then **it does not matter** if C is not detected **(*false negative*)**
- as A, B, C will all be assigned **to the same OTU** in any case
  - This is fine for the purpose of OTU counts
  - Could be more minor implications for abundance
- Detection methods need to be optimised to find problematic cases (large difference A ↔ B)
- E.g. UPARSE (Edgar (2013) *Nature Methods* **10** (10) 996-8)



Genus A

16S gene sequence space

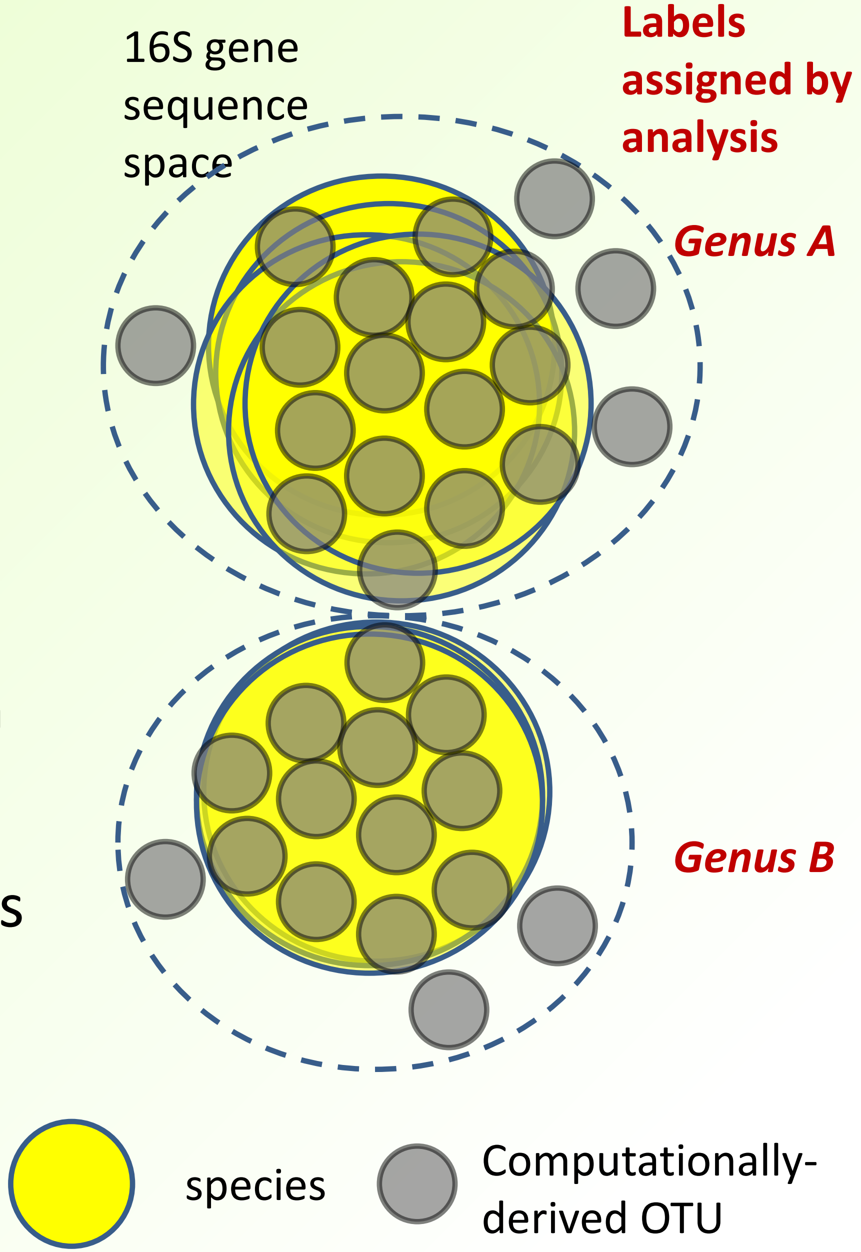
Whole-genomes sequence space

Genus B

species

Computationally-derived OTU

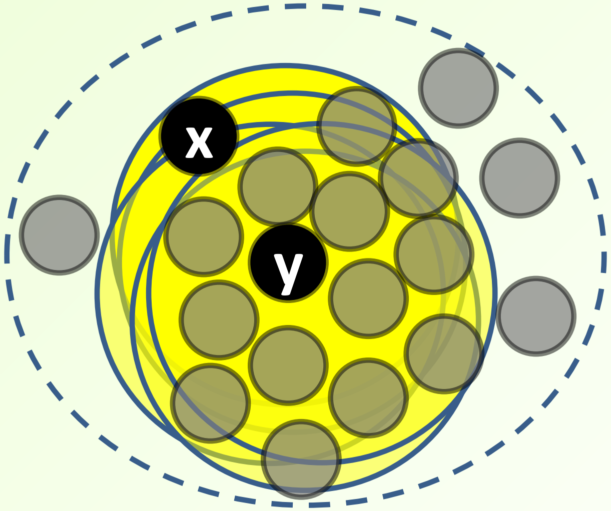
- Sequence differences between the OTUs are observable
- **In general,** these do not correlate with differences between species
- **Some** differences may reflect genuine biological variation
  - Between species
  - Between strains
  - Other biological variation within the population
- **But many differences** are due to experimental artefacts/errors
  - Amplification (including generation of chimeras)
  - Sequencing errors



- How interested are you in up/down changes in individual OTUs between samples?
- It’s possible *x* and *y* do represent biological differences
- But they might be there because of artefacts/errors

Genus A

SAMPLE 1

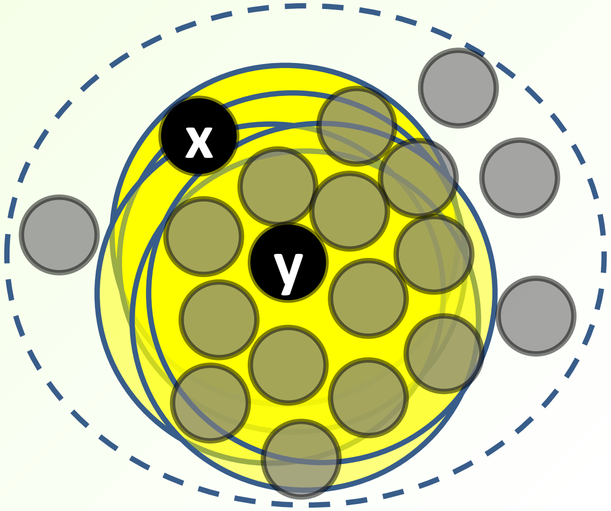


Genus A

SAMPLE 2

e.g. Compared to Sample 1:

x ↑ y↓



Summary

- Sequence differences between the OTUs are observable
- **In general,** these do not correlate with differences between species
- **Some** differences may reflect genuine biological variation
  - Between species
  - Between strains
  - Other biological variation within the population
- **But many differences** are due to experimental artefacts/errors
  - Amplification (including generation of chimeras)
  - Sequencing errors