DENOISING

WITH LULU/MUMU

Frédéric Mahé

April 9, 2025

CONTEXT

- metabarcoding data,
- post-clustering,
- occurrence table (clusters vs. samples),
- amplification and sequencing are noisy,
- denoise to shrink occurrence tables?

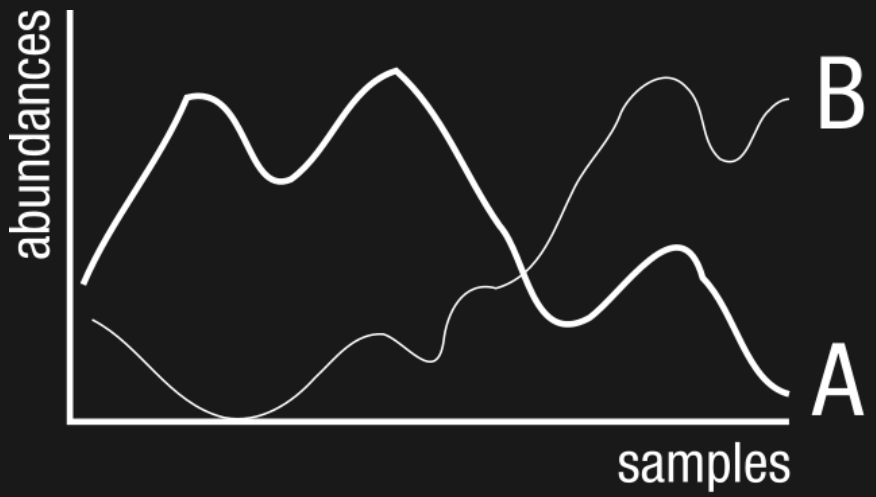
GENERAL IDEA

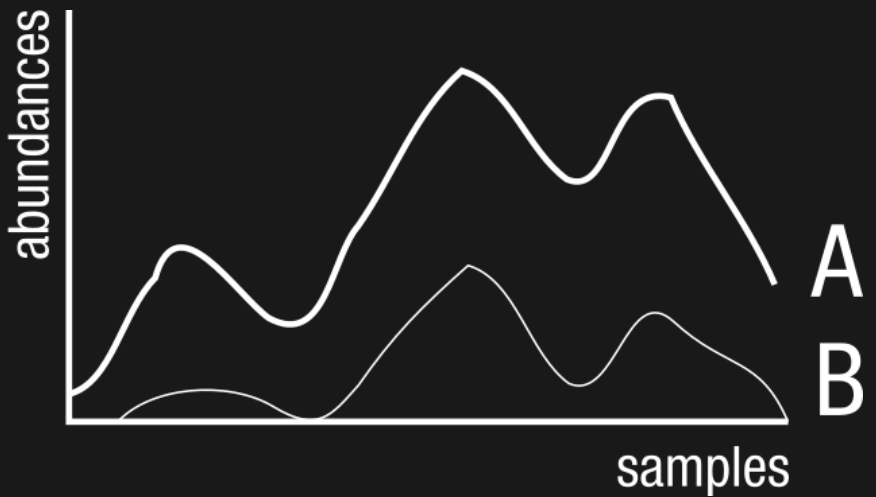
- eliminate clusters,
-

GENERAL IDEA

- eliminate clusters,
- group clusters

DISTRIBUTION PATTERNS

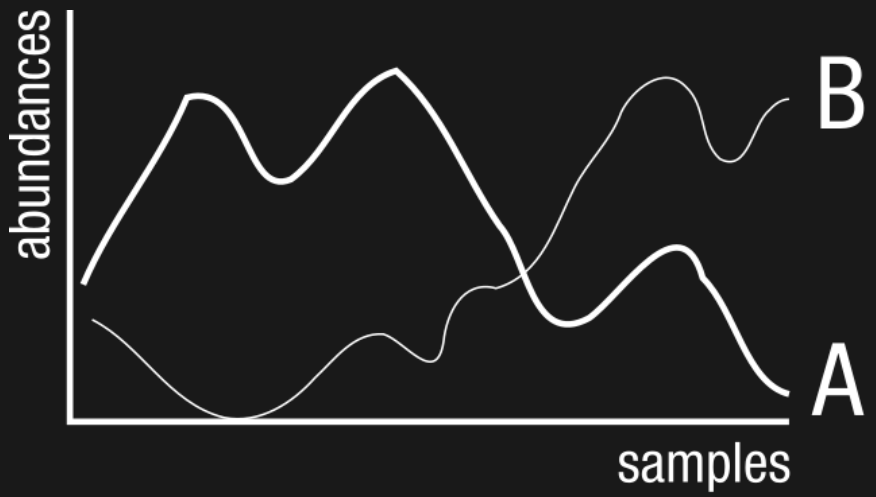


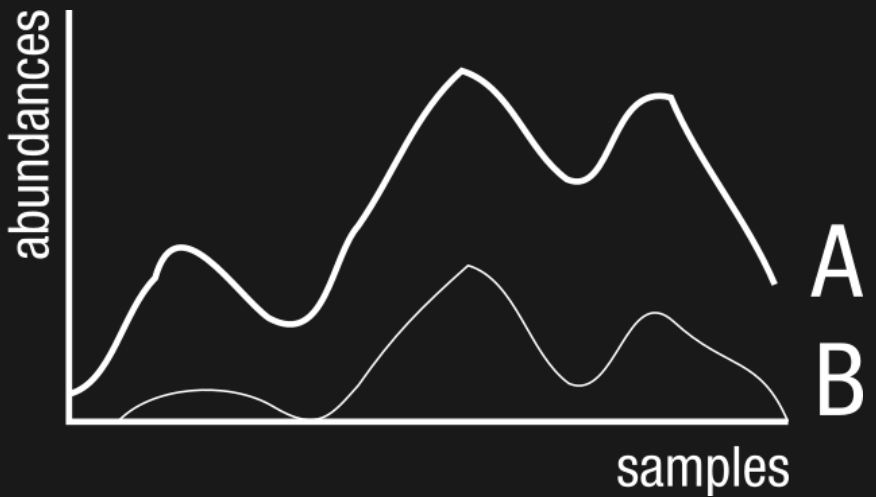


GENERAL IDEA

- eliminate clusters,
- group clusters,
- group co-varying clusters

DISTRIBUTION PATTERNS





GENERAL IDEA

- eliminate clusters,
- group clusters,
- group co-varying clusters,
- group similar, co-varying clusters!

IMPLEMENTATION

LULU

- Frøslev *et al.* (2017). Algorithm for post-clustering curation of DNA amplicon data yields reliable biodiversity estimates. *Nature Communications*, 8(1), 1188.
- R package,
- https://github.com/tobiasgf/lulu

LULU'S ASSUMPTIONS (1)

- occurrence tables often have more clusters than expected from biological knowledge,
- occurrence tables often contain low-abundance clusters, which are taxonomically redundant,
- taxonomically redundant, low-abundance clusters often have lower sequence similarity with reference sequence than more abundant clusters with the same taxonomic assignment,
- taxonomically redundant, low-abundance clusters often consistently co-occur with more abundant, well-assigned clusters,

LULU'S ASSUMPTIONS (2)

- it can be assumed that the majority of these low-abundant clusters are in fact methodological and/or analytical errors, or rare (intragenomic) variants, which will cause inflated diversity metrics.

IMPLEMENTATION

MUMU

- Mahé *et al.* (in prep.),
- C++ program,
- https://github.com/frederic-mahe/mumu

EXPERIMENTAL BIOINFORMATICS

1. **observation** & **hypothesis**: formulate a testable and falsifiable prediction.
2. **experimental design**: plan an experiment that can test the hypothesis.
3. **experiment** & **data collection**.
4. **analysis** & **conclusion**: does the data support the hypothesis?
5. **repetition**: with modified parameters, or as is to check if program behavior remains the same.

SET-UP

To install *lulu* you need R, with the package *devtools*.
To run *lulu* you also need *dplyr*:

```
## install devtools and dplyr
packages <- c("dplyr", "devtools")
for (package in packages){
    if(! package %in% installed.packages()){
        install.packages(package, dependencies = TRUE)
    }
}

## install lulu
if(! "lulu" %in% installed.packages()){
    require(devtools)
    install_github("tobiasgf/lulu")
}
```

SMALL EXAMPLE

The required input of *lulu* is an occurrence table, and a corresponding matchlist with pairwise sequence similarity values. Let's try with a 2x2 table:

clusters	s1	s2
A	9	9
B	1	1

- cluster **B** is 99% similar to cluster **A**

SMALL EXAMPLE

In *R*, we can build this 2x2 dataset as such:

```
library(dplyr)
library(lulu)

data.frame(row.names = c("A", "B"),
           s1 = c(9, 1),
           s2 = c(9, 1)) -> occurrencetable

data.frame(x = "B",
           y = "A",
           z = 99.0) -> matchlist

lulu::lulu(occurrencetable, matchlist)$curated_table
```

```
  s1 s2
A 10 10
```

SOFTWARE TESTING

writing small, readable and replicable tests allows to:

- probe a program,
- check an assumption,
- modify a program,
- maintain a program,
- develop a new program

This is what I've done to understand what *lulu* does (or does not), and what I've used to develop and maintain *mumu*.

EXPLORATION

- experimental (codeless) approach to explore *lulu*,
- check https://github.com/frederic-mahe/BIO9905MERG1_lulu_seminar for the actual tests (code),
- goal: understand *lulu*'s strengths and weaknesses

CONTEXT (REMINDER)

- post-clustering: sequences are grouped into clusters,
- cluster occurrences in samples stored in a table,
- inter-cluster sequence similarities (computed with vsearch)

A SIMPLE CASE

- only two clusters: A and B
- three samples

clusters	sample 1	sample 2	sample 3
A	9	5	1
B	3	2	0

co-variance? what is B?

WHAT IS B?

- B could be another species,
- B could be a genetic variant of A,
- B could be an error deriving from A

symbiont, predator, parasite, competitor, variant, error? We need to know how close A and B are to be able to decide.

SEQUENCE SIMILARITY

- sequence **B** is 99% similar to sequence **A**

clusters	sample 1	sample 2	sample 3
A	9	5	1
B	3	2	0

co-variance: **B** could be an error deriving from **A**

clusters	sample 1	sample 2	sample 3
A	12	7	1

SEQUENCE SIMILARITY: HOW LOW?

- sequence **B** is 80% similar to sequence **A**

clusters	sample 1	sample 2	sample 3
A	9	5	1
B	3	2	0

co-variance: is **B** still an error deriving from **A**?

SEQUENCE SIMILARITY: HOW LOW?

- sequence **B** is 80% similar to sequence **A**

clusters	sample 1	sample 2	sample 3
A	9	5	1
B	3	2	0

co-variance: is **B** still an error deriving from **A**?

lulu's default similarity threshold is set to 84%.

ABUNDANCE RATIO

HOW CLOSE TO 1 CAN IT BE?

- sequence B is 99% similar to sequence A

clusters	sample 1	sample 2	sample 3
A	10	1,000	1,000,000
B	9	999	999,999

B is almost as abundant as A: can it derive from A?

ABUNDANCE RATIO

HOW CLOSE TO 1 CAN IT BE?

- sequence B is 99% similar to sequence A

clusters	sample 1	sample 2	sample 3
A	10	1,000	1,000,000
B	9	999	999,999

B is almost as abundant as A: can it derive from A?

lulu's default abundance ratio threshold is set to 1.0

PARTIAL OVERLAP?

- sequence B is 99% similar to sequence A

clusters	sample 1	sample 2	sample 3
A	9	5	0
B	3	2	1

can the sample overlap be less than 100%?

(see lulu issue #8 and my lulu seminar I gave)

PARTIAL OVERLAP?

- sequence B is 99% similar to sequence A

clusters	sample 1	sample 2	sample 3
A	9	5	0
B	3	2	1

can the sample overlap be less than 100%?

lulu's default abundance cooccurence ratio is set to 0.95, but tests show that this is not applied!

(see lulu issue #8 and my lulu seminar I gave)

CHAIN MERGING?

- sequence **B** is 97% similar to sequence **A**
- sequence **C** is 99% similar to sequence **B**

clusters	sample 1	sample 2	sample 3
A	9	5	1
B	3	2	0
C	1	1	0

C -> B -> A? What do you think? Should it be allowed?

CHAIN MERGING?

- sequence **B** is 97% similar to sequence **A**
- sequence **C** is 99% similar to sequence **B**

clusters	sample 1	sample 2	sample 3
A	9	5	1
B	3	2	0
C	1	1	0

C -> B -> A? What do you think? Should it be allowed?

Not allowed in lulu, allowed in mumu.

SINGLE SAMPLE?

- sequence **B** is 99% similar to sequence **A**

clusters	sample 1
A	9
B	3

merge or not?

SINGLE SAMPLE?

- sequence B is 99% similar to sequence A

clusters	sample 1
A	9
B	3

merge or not?

Not allowed in lulu, allowed in mumu.

CONCLUSION

- *lulu*'s default parameters? (similarity threshold, minimal abundance ratio, overlap margin)
- some hidden assumptions:
  - samples are not duplicates,
  - samples represent a certain level of granularity (in space and time),
- *lulu* reduces along the cluster axis: what about the sample axis?

CONCLUSION

- in practice, expect a 20-30% dataset reduction (observed on 16S or ITS2 datasets)
- if you want to give it a try:
  - https://github.com/tobiasgf/lulu
  - https://github.com/frederic-mahe/mumu
- thank you!