METAGENOMICS ANALYSIS WITH NGLESS

Luis Pedro Coelho
coelho@embl.de
@luispedrocoelho

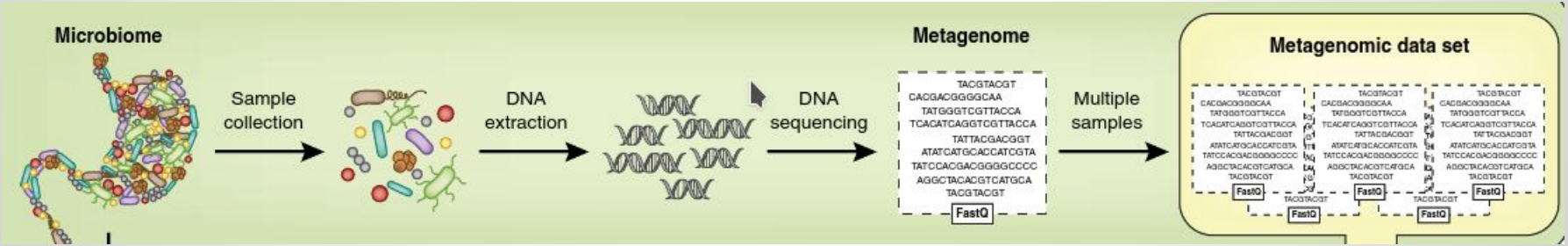
EMBL

OVERVIEW

1. What is metagenomics? How can it be useful?
2. NGLess as a tool for metagenomics analysis

What is metagenomics?

- Shotgun metagenomics is the sequencing of all the genetic material in a community (mixed) sample.



From Quince et al., 2017

Typical problems for which metagenomics is employed

- Human gut microbiome analysis
  - Disease influence
    - Colorectal cancer
    - Mental health
  - Diet effects
- Other mammals (model organisms/livestock)
- Ocean/Aquatic
  - Tara Oceans project
  - Limnology studies
- Soil
  - Agricultural productivity

What is metagenomics (II) ?

- With current technology, you get many (millions) of short reads (this is the vast majority of the data out there). What to do with this?
- *Show of hands:* who knows what a FASTQ file is?
- Ideas on how to process these?

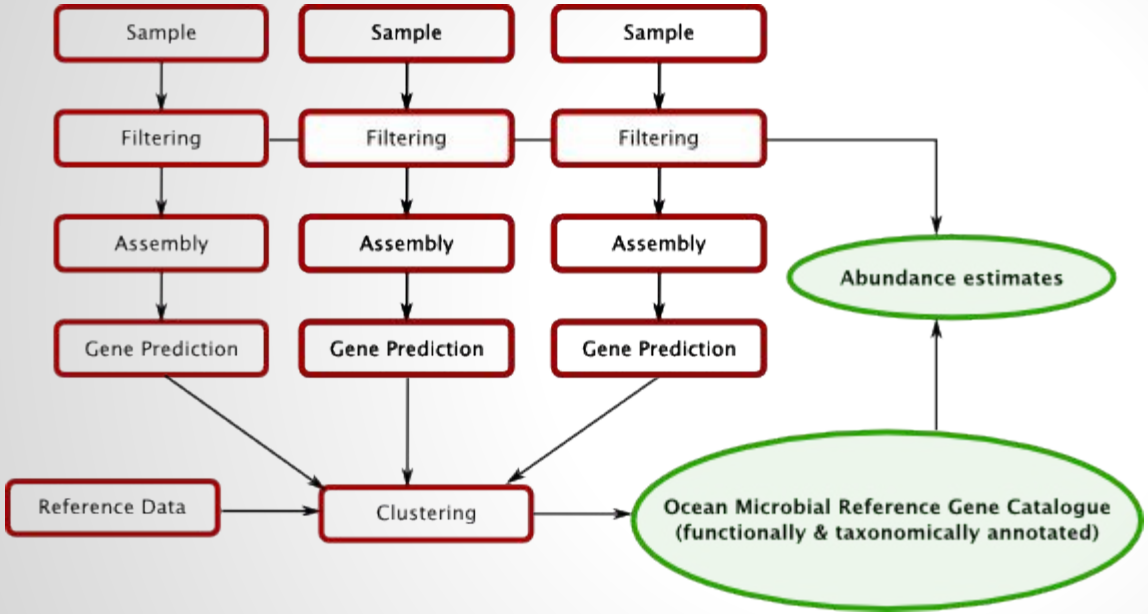
The gene catalogue approach

1. Preprocessing:
    a. Technical quality control
    b. Read trimming
    c. Filtering of contaminants (adapters, human DNA, &c)
2. Assemble contigs from the metagenomes
3. Call genes (ORFs) on the contigs
4. Build a **non-redundant** set of genes

What you can do with a gene catalogue

- Discovery of new gene families
- Discovery of variation within gene families
- Binning into (inferred) genomes
- Species/strain inference
- Profiling of metagenomes
  - Gene-level
  - Functional (after annotation)
  - Taxonomic
- ...

GENERATION OF AN **NON REDUNDANT GENE CATALOG**



Marker gene techniques provide taxonomic profiles.

A gene catalog is a parts list.

Annotations provide context for interpretation.

Useful for analysis of marine metagenomics datasets.

Taxonomic profiling
Logares, Sunagawa, et al., Env Microbio 2014
Sunagawa et al., Nature Methods 2013

QUESTIONS

1. Why did we not pool all our data?
2. What cutoff did we use for clustering? Why?
3. Do we always need to build a new gene catalog?
4. What exactly do we get out of this? I mean, what do the outputs of the pipeline look like?

OVERVIEW

1. What is metagenomics? How can it be useful?
2. NGLess as a tool for metagenomics analysis

http://ngless.embl.de/

NGLess is built around a *domain-specific language* for genomics

- Flexible system with built-in support for many basic operations
- Designed for **reproducibility from the ground-up**.
- Can integrate with Common Workflow Language based systems (basic operations are already available as CWL modules, more complex ones will soon be).
- Can scale up to 10,000s of samples (including cluster integration and robustness to node failures)

We will work on a very small example

- Real world examples would need at least a few hours of CPU time. A big gene catalog takes years of CPU time to build (weeks on a cluster).
- We will work on
  - Sampled data
  - Only some basic operations

What we will be doing:

1. Loading a FastQ file
2. Preprocessing it
3. Assembling sample

Step 0: Download the data

Ngless --download-demo ocean-short

This will download data and script for a short demo based on metagenomes from the ocean, in particular from Tara Oceans

http://ngless.readthedocs.io/en/latest/tutorial-ocean-metagenomics.html

(Sunagawa, Coelho*, Chaffron* et al, Science 2015)*

Step 1: Declare version

First line of script:

```
ngless "0.0"
```

Scripts are 100% reproducible and future proof

No more "since we updated *libc-mthread* from 0.9.2 to 0.9.3, now the clustering finds 4 clusters, instead of 5" or "This runs fine on Ubuntu 14.10, but on RocksOS 5.1.4v2, you have to re-install boost 2.1.3-pre3."

Step 2: Load data

```
input = paired('SAMEA2621033.sampled/’
                +’ERR594391_1.fastq.gz.short.fq.gz',
                'SAMEA2621033.sampled/’
                +’ERR594391_2.fastq.gz.short.fq.gz')
```

We are simply loading the data, which is paired-end data.

Q: What is paired-end data again?

Step 3: Preprocess data

```
preprocess(input, keep_singles=False) using |read|:
    read = substrim(read, min_quality=25)
    if len(read) < 45:
        discard
```

Step 4: assembling the sample

```
write(assemble(input),
      ofile='assembled.fna')
```

This is pretty trivial: call the `assemble` function and `write` the results out.

What we did not see

- `map()` to a reference, such as a gene catalog or a genome.
- Manipulating the mapped files
- Profiling abundances
- Handling many samples
- Taxonomic profiling modules
- ....

http://ngless.readthedocs.io

NGLess can be run inside Python

The *Python* script on the right will run an NGLess pipeline from Python.

Thus, your pipeline can be dynamically defined.

THIS IS VERY EXPERIMENTAL

```
from ngless import NGLess

sc = NGLess.NGLess('0.0')
e = sc.env

e.input = sc.paired_('SAMEA2621033.sampled/ERR59439
                     'SAMEA2621033.sampled/ERR594391

@sc.preprocess_(e.sample, using='r')
def proc(bk):
    bk.r = sc.substrim_(bk.r, min_quality=25)
    sc.if_(sc.len_(e.r) < 45, sc.discard_)

sc.write_(sc.assemble_(e.input),
    ofile='assembled.fna')

sc.run()
```

Thank You

95% Identity corresponds to a species-level cutoff

Using a pangenomic dataset, we verified that between strains of the same species we observe genes >95% ID, which rarely happens within a genus: