Dead Man's Teeth. Introduction to metagenomics analysis.

New bioinformatics skills covered: 16S vs whole metagenome, alignment, ASV clustering.

In 1990, archaeologists working on a monastic site in Dunheim, Germany dug out several human skulls. However, they weren't analyzing cranial measurements: they were looking at teeth covered by **dental calculus** – a cover on the dental surface caused by periodontitis. Since periodontitis is the second most common dental disease worldwide, you may think they wanted to prove that our ancestors suffered from the same dental disease as 750 million people today. However, they were actually interested in the time capsules that dental calculus provides: they can preserve DNA for 1000s of years!

We can refer to these samples as "Microbial Pompeii". Like the citizens of the ancient city that was buried by ash, this microbial community was buried by layers of dental calculus and was kept intact for more than 1000 years. Scientists extracted DNA from the material underneath the dental calculus.

In this project, we will explore these samples and study the history of oral diseases in humans.

Figure: Dalheim human dental material

Researchers sequenced the DNA samples they extracted and submitted the raw data and sequences to public databases. There are lots of samples, and we will start with the results of sequencing portions of V5 16S ribosomal RNA obtained by an instrument Roche GS Junior (454).

We will be using samples from dental calculus, and will compare them with the samples from the ancient tooth roots.
I uploaded raw data on the Figshare repository, along with some additional files.

All data from the original research are available in the NCBI Short Read Archive (SRA) under number SRP029257 (BioProject PRJNA216965). In addition, all the data was uploaded to the server MG-RAST, an open source web application server that provides automatic phylogenetic and functional analysis of metagenomes. We are interested in data stored there in Project 365 - "Ancient Oral Metagenome".

Part 1. Amplicon sequencing.

The output of the sequencing machine is a lot of fragments, each corresponding to a different molecule from the different species you are studying. Some of them are more abundant, some less; we don't know which yet, but now we're going to try to separate these fragments into several piles and try to show that sets of similar fragments originate from the same organism. DNA sequences can be conservative: the same sequence can be found without too many changes in several species. That's why we cannot tell in advance whether a certain sequence we've amplified corresponds to a particular species or genus. For a particularly slow-evolving gene, we may not even be able to tell if it's in the same family! In this case, we can call each of these piles an "Operational Taxonomical Unit" or OTU, and then try to attribute it to one branch on the Tree of Life.

1. QIIME2 installation

We will use the package QIIME2 ("Quantitative Insights Into Microbial Ecology"), an open-source bioinformatics pipeline for performing microbiome analysis from raw DNA sequencing data, developed in the laboratory of Dr. Rob Knight at UC San Diego.

Install Qiime2 according to the instructions (I hope you do conda).

2. Importing data.

Oh boy, that's tricky. We will be mostly following the main QIIME2 tutorial ("Moving Pictures", check it out - it covers way more stuff that we actually need for the project, explore it and have some fun).
In the tutorial they have input files in very common EMP format - multiplexed file with reads and file with barcodes for demultiplexing. To import the regular FASTQ files, we need an additional "manifest file" (it's in the same repository, next to reads).

Note: you can edit the manifest file (change absolute paths to read files), if you want to reorganise the way you store the data.

Ok, let's import. We have single-end reads in FASTQ format with Phred33 encoding.

```
 qiime tools import   --type 'SampleData[SequencesWithQuality]'
--input-path manifest.tsv   --output-path sequences.qza
--input-format SingleEndFastqManifestPhred33V2
```

Output: **sequences.qza**

You can check correctness of any QIIME artifact with qiime validate

```
qiime tools validate sequences.qza
```

3. Demultiplexing and QC

Our reads are already demultiplexed (1 sample per file). Now it is useful to explore how many sequences were obtained per sample, and to get a summary of the distribution of sequence qualities.

```
qiime demux summarize --i-data sequences.qza --o-visualization
sequences.qzv
```

Output: **sequences.qzv**

Note: you can visualize any .qzv file at https://view.qiime2.org/

4. Feature table construction (and more QC)

For the subsequent analysis we will need the metadata table (you can find it in the same repository). There is a barcode - a unique sequence in the beginning of each sample, and primer sequence, that was used to amplify the V5 region of the rRNA. We need to strip it out, and filter chimeric sequences. Here we will use the DADA2 pipeline, as follows:

```
qiime dada2 denoise-single   --i-demultiplexed-seqs sequences.qza
--p-trim-left m --p-trunc-len n --o-representative-sequences
rep-seqs.qza --o-table table.qza --o-denoising-stats stats.qza
```

Output: stats.qza
table.qza
rep-seqs.qza

Here we need to select value **m** for `--p-trim-left` as a total length of the artificial sequences (barcodes), and value **n** for `--p-trunc-len` according to Interactive Quality Plot tab in the sequences.qzv (it's going to be a total length of the sequence after the trimming).
In our case we know that the primer + adapter length is about 35 bp, and amplicon size is about 145 bp. m = 35 and n = 140 can be a reasonable choice in this case.

Check how many reads are passed the filter and were clustered:

```
qiime metadata tabulate   --m-input-file stats.qza
--o-visualization stats.qzv
```

Output: stats.qzv

5. FeatureTable and FeatureData summaries

One of the main results of the DADA2 step is a clustering into an amplicon **sequence variant (ASV)** - a higher-resolution analogue of the traditional OTUs. Thus, the Feature Table we just obtained is an equivalent of the OTU tables in other metagenomic pipelines.

*A feature is essentially any unit of observation, e.g., an OTU, a sequence variant, a gene, a metabolite, etc, and a **feature table** is a matrix of sample X feature abundances (the number of times each feature was observed in each sample). In this case we can think of features as OTUs.*

First, let's create visual summaries of the data - how many sequences are associated with each sample and with each feature, etc.

```
qiime feature-table summarize   --i-table table.qza
--o-visualization table.qzv   --m-sample-metadata-file
sample-metadata.tsv
```

Output: table.qzv

Then we can map feature IDs to sequences, to use these representative sequences in other applications, e.g. BLAST each sequence against the NCBI nt database.

```
qiime feature-table tabulate-seqs  --i-data rep-seqs.qza
--o-visualization rep-seqs.qzv
```

Output: rep-seqs.qzv

6. Taxonomic analysis

Now we can compare the representative sequences with the taxonomy database, and get the answer to our first question: who lives here?
First, we need the database itself - download it from the data resources page.
Database itself is just a fasta file of the 16S representatives, and QIIME2 uses Naive Bayes classifiers trained on this data.
https://data.qiime2.org/2022.2/common/silva-138-99-nb-classifier.qza

(if the link doesn't work, you may download it from here:
https://disk.yandex.ru/d/QxQWKV8x5ucxvw)

(515F/806R region doesn't work - it corresponds to V4 region of 16S, and we got V5).

```
qiime feature-classifier classify-sklearn   --i-classifier
silva-138-99-nb-classifier.qza   --i-reads rep-seqs.qza
--o-classification taxonomy.qza
```

Output: taxonomy.qza

And visualise:
```
qiime metadata tabulate --m-input-file taxonomy.qza
--o-visualization taxonomy.qzv
```

Next, we can view the taxonomic composition of our samples with interactive bar plots. Generate those plots with the following command and then open the visualization.

```
qiime taxa barplot \
  --i-table table.qza \
  --i-taxonomy taxonomy.qza \
  --m-metadata-file sample-metadata.tsv \
  --o-visualization taxa-bar-plots.qzv
```

Output: taxonomy.qzv
taxa-bar-plots.qzv

7. Bacterial Teamwork

Different bacterial species in a community can work together, and one can find a lot of these groups in the metagenomics environments. The human oral metagenome is no exception. One remarkable example is a group of bacteria that were found with high abundance in patients with severe forms of periodontal disease. These three bacterial species are called "the red complex", and they are usually found together in periodontal pockets. This suggests that they

may cause destruction of the periodontal tissue in a cooperative manner. We know this from modern studies, but what about these ancient samples?

There is osteological and proteomic evidence of periodontal disease in at least two of the subjects in study. Check taxa barplot, look for the red complex bacteria, and try to identify those who have been affected by the disease. Describe your findings in the report.

P.S.
There is an option to play with your data in the online version of MicrobiomeAnalyst. This requires exporting the data to an ASV table and taxonomy file, and making a few adjustments to the metadata file. To avoid the formatting hassle, it is better to use BIOM binary format. We only need a few commands:

Export ASV table to biom file:

```
qiime tools export --input-path table.qza --output-path export_biom
```

Export taxonomy table:

```
qiime tools export --input-path  taxonomy.qza --output-path export_biom
```

Add taxonomy information to biom file:

```
biom add-metadata -i export_biom/feature-table.biom -o
export_biom/feature-table-with-taxonomy.biom --observation-metadata-fp
export_biom/taxonomy.tsv --sc-separated taxonomy --observation-header
OTUID,taxonomy,confidence
```

Fix header of the metadata file:

```
sed  's/SampleID/NAME/g' sample-metadata.tsv   > sample-metadata.txt
```

Now you can use the “Marker data profiling” section of the MicrobiomeAnalyst to explore your data in detail.

Part 2. Shotgun sequencing.

Well, now we will be analyzing shotgun sequencing (whole genome) data. We will try to obtain the ancient sequence of these pathogens and compare them with modern one.

We realized that 1000 years ago periodontal disease was caused by the same bacteria that we can find now in our mouth. But we know that bacteria evolve very quickly, and we have a unique opportunity to explore how it happens.

To investigate it, an affected individual G12 was selected for a dental calculus whole metagenome shotgun sequencing, and reads were assembled into contigs. Metagenome assembly is based on the same principles as for prokaryotic assemblies - usually it is a modification of existing algorithms, taking into account coverage (we expect similar coverage for each OTU) and less aggressive error correction (because in metagenomic

samples it can be natural variants or close strain).

We will skip the actual assembly process, because the raw data is too large and it will take a lot of time (but you can download raw reads for this project from the MG-RAST or from the SRA database to assemble it on your own). Assembly results can be downloaded here, or also from the corresponding MG-RAST page.

1. Shotgun sequence data profiling. **(NB: the current version of Metaphlan relies on very large databases, and it is just unfeasible to run it on a laptop. Thus, I made this and all other Metaphlan-related parts optional)**
(NB2: Just in case if you want to play around with Metaphlan output, I've uploaded precomputed results here)

MetaPhlAn (pronounced meta-flan) is short for METAgenomic PHYlogenetic ANalysis. It takes the set of microbiota reads in our file and uses the bowtie2 aligner to map them to a specialised reference database that contains a set of key markers from known human microbiome genomes.

You can manually install MetaPhlAn from source, or use a conda-based approach.

MetaPhlAn will align our sequencing reads to the microbiota database, then tabulate the abundance of each type of microbe that matched. Run it on your assembly
(e.g. `metaphlan <your_sample.fasta> --input_type fasta --nproc <# of cores> > output.txt`)

This command may take a few minutes to run. When it is complete, inspect the .txt output file. This file contains the final computed organism abundances, listed one clade per line, tab-separated from the clade's percent abundance.

2. Comparison with samples from HMP (optional)

Now we can compare our results with data from the Human Microbiome Project.
SRS014459-Stool.fasta
SRS014464-Anterior_nares.fasta
SRS014470-Tongue_dorsum.fasta
SRS014472-Buccal_mucosa.fasta
SRS014476-Supragingival_plaque.fasta
SRS014494-Posterior_fornix.fasta
Each of these files contains shotgun read data from different parts of the human body. They are subsampled for rapid analysis - the original files can be found at the HMP website.

Now we want to run metaphlan on these samples. Each MetaPhlAn execution processes

exactly one sample, but the resulting single-sample analyses can easily be combined into an abundance table spanning multiple samples.

Fortunately, it is easy to set this up so that you don't have to do it one at a time.
if you're familiar with bash shell syntax, you can loop over the entire sample set:

```
$ for f in *.fasta.gz
> do
>     metaphlan $f --input_type fasta --nproc 4 >
${f%.fasta.gz}_profile.txt
> done
```

This command can take about 10 minutes to run, so, feel free to take a break.

3. Visualization of the metaphlan results as a Sankey diagram (optional)

You can use the Pavian web application (or install it locally) to visualise the classification results obtained with Metaphlan. It can also accept input from other commonly used classifiers such as Kraken and Centrifuge**.**
https://fbreitwieser.shinyapps.io/pavian/

Results from all other samples can also be added there, and under the "Sample" tab for any taxon you can see its distribution among the samples.

(optional) Visualization of the metaphlan results with a heat map

You can compare the files by looking at the profiled.txt results, but it will be easier to see what is
going on if we make a figure. First, you have to merge the metaphlan profile files into a single abundance table. Use the metaphlan script `merge_metaphlan_tables.py`.

Now you are ready to make a heat map. The command below will make a basic heat map
It can be easily done with `metaphlan_hclust_heatmap.py`. It calculates distance between samples based on species composition. Use a log scale for assigning heatmap colors (`-s log`) and use top 50 most abundant taxa (--top 50).
Try to give an interpretation to this heatmap - is there any difference, which samples are closer to each other and why.
You may try to play with parameters to build a new, more informative heatmap (or maps).

4. Comparison with ancient Tannerella forsythia genome

Our shotgun assembly is still pretty fragmented, so we will have to align our contigs to reference. Download data for the *T. forsythia* strain (there is only one complete genome in GenBank so far) - we will need the genome itself (fasta) and annotation (GFF3).

Align contigs on the downloaded reference (as we did in the very first week). You may visualize it in IGV, to get the idea what the coverage looks like. We can see that some of the regions in the modern strain have zero coverage, and probably were obtained during the strain evolution.

We definitely don't want to eyeball these regions. To do this automatically, we can use the bedtools package (maybe some of you already tried it in the first two projects).
First, we need to turn obtained alignment file (BAM) to BED with bedtools bamtobed, and then intersect with annotation file (GFF3) using bedtools intersect to keep only new regions in the modern strain (actually, we are doing subtraction, `-v`).
Explore the resulting file - based on the gene content, describe mechanisms of the bacterial evolution, and try to find any evidence of obtained adaptation.

For your lab report:

Intro: Provide biological motivation for metagenomics analysis and briefly mention two main approaches (16S/shotgun sequencing).

Methods: Briefly describe the raw data, tools, and databases you used.

Results: Provide information you were able to pull out. Trees, heatmaps etc.

Discussion: Describe the difference between microbiome content in different samples, and possible reasons behind. Discuss possible mechanisms of pathogen evolution based on your results.