An analysis of bacterial genomes relating to amphibian skin microbiomes

# July – Sept 2024

## Summary

* Recording the metadata found in the .fasta files, located on SCW
* Using the R package Checkm2 to find the completion and contamination of each sample
* Testing kegg and cog on flies, eggnogmapper
* 2 week project - Using the GBDT-tk package in R to make phylogenetic trees of the bacterial genomes

## Methods

### Getting on to SCW and getting the metadata

I started the summer part of the project by accessing the original project files, stored on the Hawk Supercomputer of SuperComputing Wales. They are stored in the directory: /scratch/scw2160/02\_outputs/flye\_asm

In there there are many directories, roughly 20, 10 for samples run through the flye assembler and 10 for the files I ran through checkm2, discussed below. Inside these directories there are many files, however, the ones I used were the “flye.log” files, with which I constructed a table of data about the samples.

A screenshot of a computer screen

Description automatically generated and creating an Excel table from the data at the bottom of the files marked “flye.log”. each sample had its own associated directory containing this file, data pictured below;

A close-up of a white background

Description automatically generated

This table had columns for the length, fragments and coverage, which I then added the checkum2 scores to later.

### Creating slurm scripts

Slurm is a job scheduling program used on SCW in order to use resources most efficiently. Slurm decides when a job will run based on how many jobs are running and the requirements specified in the slurm script. I did this by creating the script inside a batch file. Progress on the job can be seen with the commands squeue and scontrol show job [and the job number].

<https://github.com/aptemus/tnunn_research/blob/bb69990d0be95bbb32cc5700841897a7a5b3f663/2024-25%20Research/frog%20bacteria/03%20data_analysis/allgtdbtk.sh>

### Generating the completeness and contamination with checkm2

This part was contained to SCW, I created a directory in the project workspace:

/scratch/scw2160/02\_outputs/flye\_asm

using the mkdir command, then followed the steps in the framework file, supplied by Aaron Comeault, and stored here:

/scratch/scw2160/00\_scripts

to create a bash script to run Checkm2 over all the flye\_asm files;

A screenshot of a computer

Description automatically generated

This produced some folders, one for each flye folder, which I then opened in turn and opened the “quality report.tsv” file, which contained the completeness and contamination of each sample by adding them to the aforementioned Excel table



### Generating the Kegg and Cog heat maps and clustered bars for fly examples [more detail]

This process begins with the flye folders, created by the “flye assembler” module. Which, as I understand it, is a de novo assembler that takes fragments of DNA and turns it into a potential complete genome. These were supplied to me by Aaron Comeault. Then I take the “.fasta” files from the folders, download them to my machine, and run them through “eggNOG-mapper”, a website that can take the genetic sequence(from the fasta file) and annotate the genes(this was doneon August 19th 2024). It outputted several .xlsx and .gff files, which I collated into a folder. [I believe] Aaron Comeault then did something[?] to turn those files into a few files called “cog\_counts\_raw.txt”, “cog\_enriched\_categories.txt” and “kegg\_enriched\_pathways.txt”, which were given to me, by him, on the 20th. However, the files could be unrelated, in which case, I have no idea why I ran eggNOG-mapper. He then supplied me with R scripts with which I

inputted the aforementioned files and produced some heatmaps and clustered column charts to compare sites between species where genes were significantly different. This then lead onto creating another Excel table outlining these kegg pathways of note and their function.

## Outputs

A blue squares with black lines

Description automatically generated

* *Figure 1 – This is a heat map showing which pathways were distinct between the four species of bacteria sampled, brighter blue signifying significant difference. I then looked up online what these pathways represented and expanded on them in a table. This is what I plan to do for the bacterial samples from Dendrobates tinctorius.*

<https://github.com/aptemus/tnunn_research/blob/e04ca565741ed1d1164beb601290e74e85020ad4/2024-25%20Research/fly%20bacteria/04%20outputs/fig%204%20description%20table.xlsx>

Table 1 – a description of the functions of distinct genes as identified through processing of KEGG pathways

|  |  |  |  |
| --- | --- | --- | --- |
| **pathway** | **encriched in** | **function name** | **formal description** |
| map00053 | Orbus | Ascorbate and aldarate metabolism | Ascorbate and aldarate metabolism is a carbohydrate metabolism pathway that protects bacteria from oxidative stress |
| map00261 | Gill\_dros | Monobactam biosynthesis | Monobactam biosynthesis is the production of monobactim, an antibiotic |
| map00450 | Gill\_dros | selenocompound metabolism | Selenocompound metabolism is the metabolism of compounds containing selenium, such as Selenocysteine, a seloamino acid |
| map00561 | Orbus | Glycerolipid metabolism | Glycerolipid metabolism is the creation and breakdown of compounds containing lipids and sugars |
| map00564 | Orbus | Glycerophospholipid metabolism | Glycerophospholipid metabolism is the creation of Glycerophospholipids used in the cell membrane of gram negative bacteria |
| map00630 | Orbus, Gill\_apis | Glyoxylate and dicarboxylate metabolism | Glyoxylate and dicarboxylate metabolism is the breakdown of said molecules in order to be converted into glucose and be used in respiration respectively. |
| map01501 | Gill\_dros | beta-Lactam resistance | beta-Lactam resistance is a bacterial ability to resist effects from this specific group of antibiotics. |
| map02025 | Gill\_Bombus | Biofilm formation - Pseudomonas aeruginosa | Biofilm formation - Pseudomonas aeruginosa - the bacteria produce a matrix of proteins and polysaccharides that bind them to a location and together for protection. This is odd as it contradicts motility. We believe the reason for the mention of this species is that the pathway was first studied under it. |
| map02026 | Gill\_apis, Gill\_Bombus | Biofilm formation - Escherichia coli | Biofilm formation - Escherichia coli - the bacteria produce a matrix of proteins and polysaccharides that bind them to a location and together for protection. This is odd as it contradicts motility. We believe the reason for the mention of *E coli* is that the pathway was first studied under it as a model organism for bacteria |
| map02030 | Gill\_apis, Gill\_Bombus | bacterial chemotaxis | Bacterial chemotaxis is the ability of a bacterium to move towards favourable chemical conditions. This pairs well with the COG N value of motility, also shared between G*illiamella* on both groups of bees |
| map02040 | Gill\_apis, Gill\_Bombus | Flagellar assembly | Flagellar assembly is an organelle used for movement |

# August - October 2024

## Summary

[put short summary here]

## Methods

### Generating phylogenetic trees

Firstly, I found the package GTDB-tk. I believe it was first suggested to me by my supervisor Aaron Comeault, one can find modules like this on SCW using the command “module avail”. The files at the beginning of the process had been run through a process by Aaron Comeault(I believe it is called flye assembler). They were located on Super Computing Wales as files in the type .fasta. I created a slurm batch file to execute GTDB-tk against my .fasta files. Firstly, I started with the “test” function, then moving onto a single proper run just to see if it would work, after this I added more lines to the batch script so that all the files would be run at once to produce one output directory. SCW then ran each of the files through the processes in GTDB-TK and output a directory full of .tree files.

Several .tree files were created. I then pulled them down to my machine using MobaXterm using its in-built multiple window function. This led to an initial error as one of the files “gtdbtk.bac120.summary.tsv” is a development version of the actual tree files which has some inconsistencies. However, I then moved onto the other three .tree files that grouped samples with related public samples, these were “gtdbtk.bac120.classify.tree.1.tree”, “gtdbtk.bac120.classify.tree.4.tree” and “gtdbtk.bac120.classify.tree.5.tree”.

#### Dendroscope

I downloaded the windows application Dendroscope(<https://uni-tuebingen.de/fakultaeten/mathematisch-naturwissenschaftliche-fakultaet/fachbereiche/informatik/lehrstuehle/algorithms-in-bioinformatics/software/dendroscope/> )( accessed between July and September 2024), I viewed each tree in Dendroscope and identified a common ancestor for the purpose of pruning so that groups remained monophyletic, this resulted in 6 .tree files. I saved these also as .nexml files as this allowed them to be processed by Power Query in Excel. The pruning is done by selecting a node label, right clicking and selecting the option that says “Select Subnetwork”, then going to the “Select” pane and selecting “Invert Selection”. Finally, go to the “Edit” pane and select “Delete Taxa”.

Here is the document I used to do that.

#### Power Query

I created a parameter in Excel so I could specify the name of the .nexml file and have Power Query perform its steps against the specified file. This allowed me to run the same operation over all 6 files, and I can repeat this in the future when needed.

Power Query:

* extracts the list of tip labels from the tree
* removes the local “flye\_asm samples”
* reformats the remaining labels to remove the “GB\_” and “RS\_” where applicable
* calls the NCBI datasetsv2 REST API, specifically the /genome/accession/datasetreport API <https://www.ncbi.nlm.nih.gov/datasets/docs/v2/api/rest-api/#get-/genome/accession/-accessions-/dataset_report>
  + to be able to use the API, I created an account on the NIH website and was then able to generate a unique API key.
  + Because I knew the maximum number of results I could get was less than 40 I hard coded the page\_size parameter in the API call to 40. To make this code more flexible I need to learn how to manage pagination in API calls
* Process the .json objects returned from the API call to extract the following:
  + Species name
  + Whether the sample is withdrawn
* The reference samples were recombined with the local samples so that the tip labels and colours could be defined
  + Tip label is accession number / assembly ID followed by new line character followed by organism name
  + Tip label colour is red for withdrawn, blue for local sample and black for all others.
* Returns a table back into Excel

#### Excel

In Excel formulas are used to construct all of the necessary R code:

* Vectors for the tip labels and tip label colours
* Code to read in the .tree file and apply the tip labels and tip label colours
* Output a phylogenetic tree and save it as a .png

https://github.com/aptemus/tnunn\_research/blob/6312e7efcbb7b05a8cea82fecee706d546d5c29c/2024-25%20Research/frog%20bacteria/03%20data\_analysis/scriptmakerdatasets%20version%201.xlsx

#### R Studio

I manually copy the generated code into R studio and run it, creating this file.

https://github.com/aptemus/tnunn\_research/blob/0c487ca21d0c783b7075446d688f88ed1d7ce77a/2024-25%20Research/frog%20bacteria/03%20data\_analysis/change%20tip%20labels.R

note: process has subsequently been revised to only include R, this is discussed below in the “Week 43 – 21st of October to 27th of October” section.

#### API call

In order to do any API call for the ncbi website, one needs an “API key”, which just means an account identification so they know you are a trusted person. In order to generate this I went to this site:

<https://www.ncbi.nlm.nih.gov/home/develop/api/>

logged in with my personal Microsoft account, using the “log in” button in the top right corner of the screen. From there one can go to their account's dashboard here: <https://account.ncbi.nlm.nih.gov/settings/>

and go to the section called “API Key Management” to generate a key.

I did this on the 27th of September

## Outputs

This is an example of the original phylogenetic trees:

A white screen with black text

Description automatically generated

# Week 42 – 14th of October to 20th of October

I started the school year by finishing off my work from the summer, by generating completed phylogenetic trees for the proper identification of the samples, this is seen further down on pages 11-18 of this document. I also began background reading into the subject with “**Composition of the North American wood frog (Rana sylvatica) skin microbiome and seasonal variation in community structure**” by (Douglas et al. 2020).

They hypothesize on the microbiome changing with season and the effect it has on pathogen relationships, this may be an interesting point for our samples **“when were they collected?”**

they mention Pseudomonas spp. and Rhizobium spp. As common skin bacteria that are antifungal, **“are there any common genes between these and ours?”**

I did notice that they only mention two sample sites and one sample year, so the data set could be small

They talk about environmental factors affecting the microbiome, not many tests I can run on that, but maybe it would be good to note the environment they came from if this gets formally written up.

They talked of chytridiomycosis and Batrachochytrium dendrobatidis – which I think is relevant to this project, probiotic uses of bacteria.

# Week 43 – 21st of October to 27th of October

## Summary

I revised the previous code used to make the phylogenetic trees.

* To perform all operations in R
* To make use of the ggtree package to store additional attributes about each tip in order to apply aesthetics to the tree

## Methods

The new R code:

* Imports a .tree file from the specified directory
* Takes the tip labels which refer to NCBI samples and constructs an API call to get a dataset per sample, this uses the httr package
* Parses the dataset to pull out the pieces of information of most interest to us:
  + - species name
  + - sample status, i.e. reference, suppressed
* Combines this with the information about the samples taken in the Bangor lab to produce a phylo tree plot with appropriate colours and symbols
* Saves this to a png in the same directory

https://github.com/aptemus/tnunn\_research/blob/0c487ca21d0c783b7075446d688f88ed1d7ce77a/2024-25%20Research/frog%20bacteria/03%20data\_analysis/call%20API%20and%20process%20json%20-%20with%20legend.R

for future enhancements:

* add error handling for API call to ensure there is a status 200
* for loop in so it will do for each tree file automatically
* add processing to calculate the right width and height based on the number of taxa
* calculate the right xlim value by using the distance attributes
* iterate over the directory with the tree files and do all of them

I also started working on file organization on the 27th so that files were more accessible. This includes setting up a space on Github where my data files should be

## Outputs

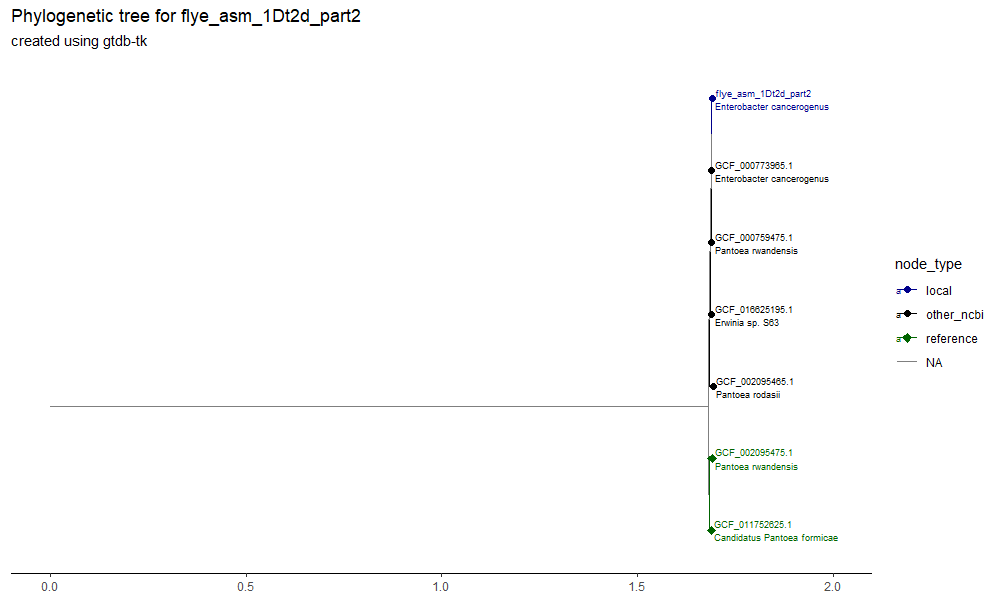


Figure . 1Dt2d

A screen shot of a computer

Description automatically generated

Figure . 3Dt2h

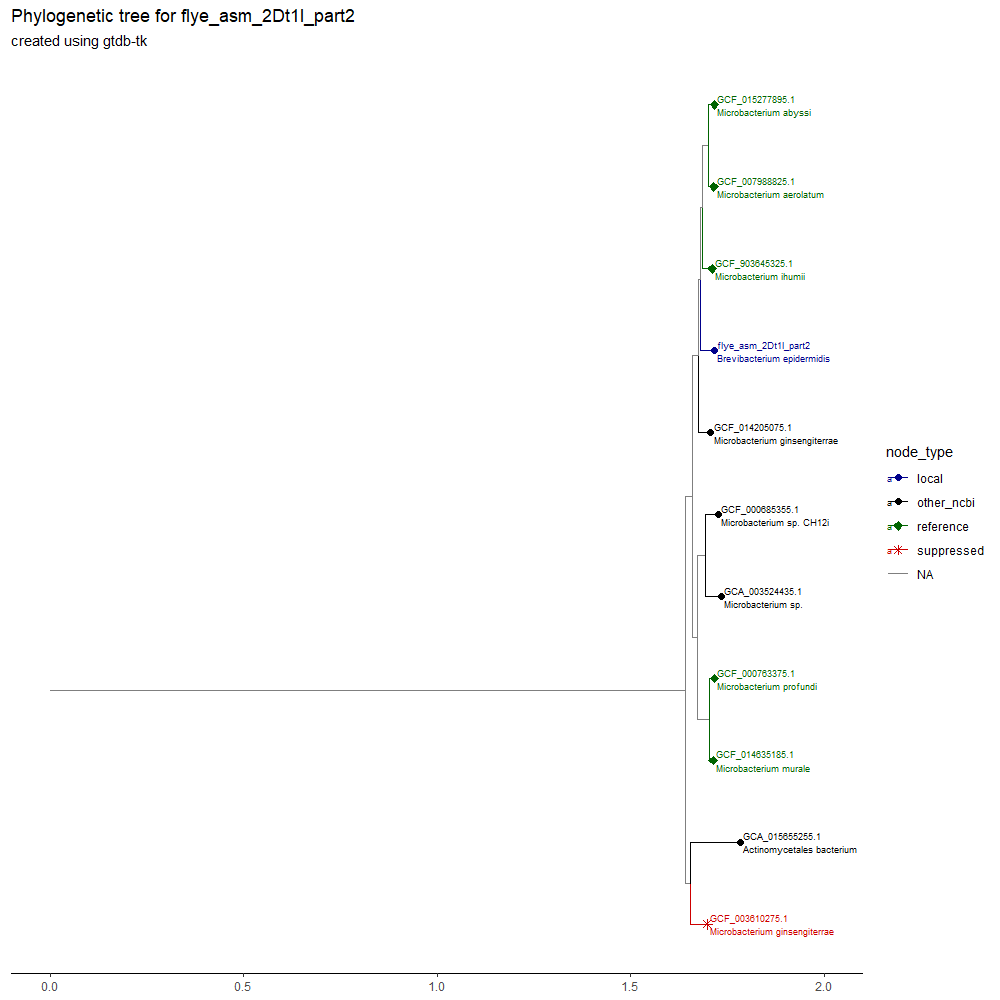


Figure . 2Dt1l

A screenshot of a computer

Description automatically generated

Figure . 1Dt100g, 3Dt1c, 3Dt2c and 3Dt2j

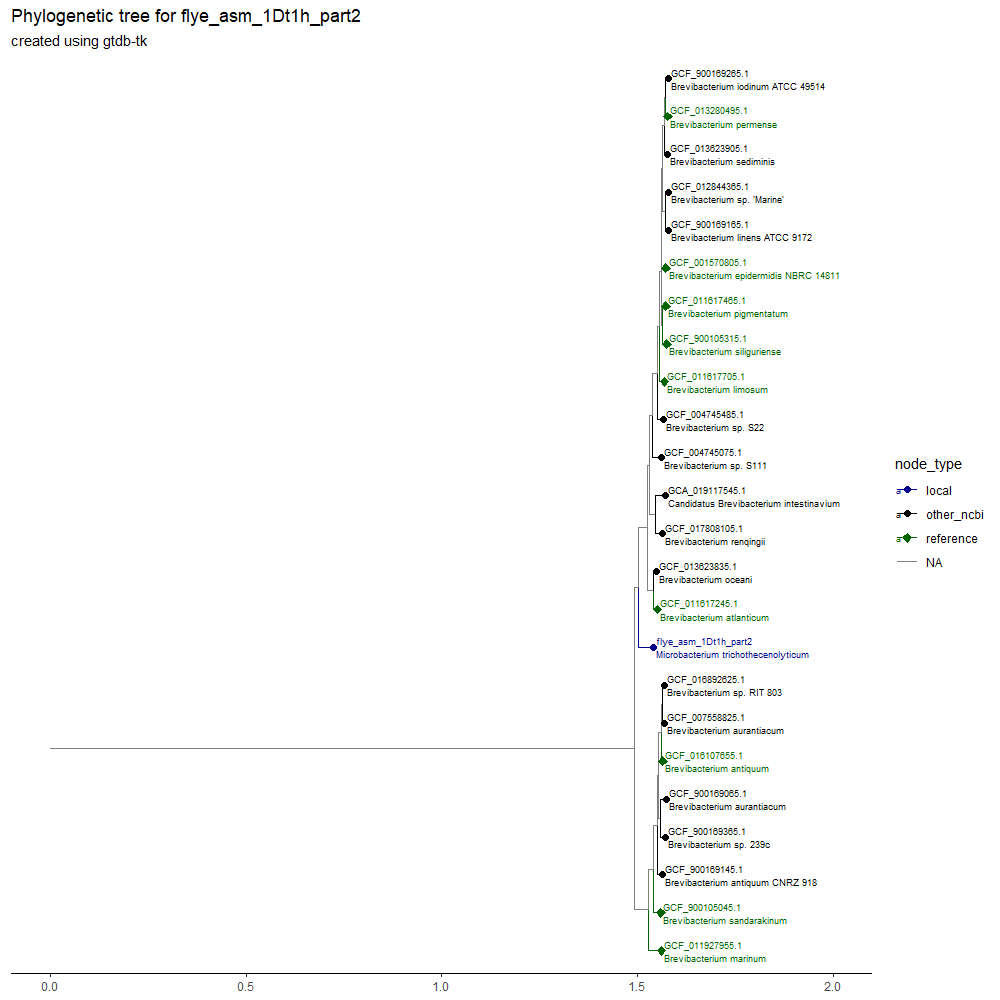


Figure . 1Dt1h

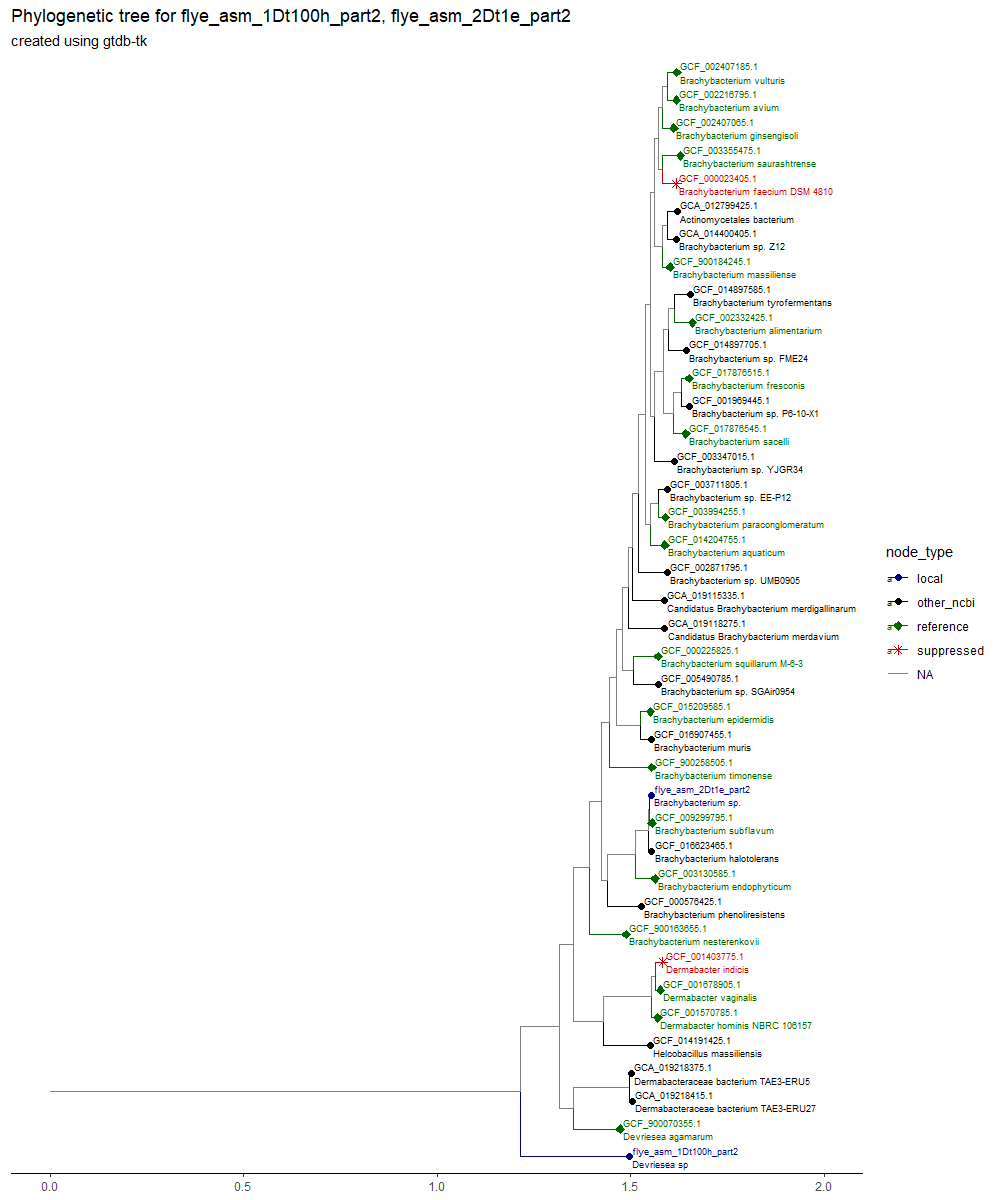


Figure . 1Dt100h and 2Dt1e

### Summary Table

|  |  |  |
| --- | --- | --- |
| Accession | Reference figure | Conclusion |
| 1Dt2d | Figure 1. 1Dt2d | Likely correct classification as the sister sample is of the same species, however it is not a reference so confidence is lower. |
| 3Dt2h | Figure 2. 3Dt2h | This is an odd one, it isn’t matched with a “microbacterium testaceum”, but the one it is matched has no species ID so it could be, the weird thing is there are “testaceum” samples all over the tree that aren’t next to each other, but they are all “microbacterium” so ID is probably accurate |
| 2Dt1l | Figure 3. 2Dt1l | Warning alarms on this one, it is a “Brevibacterium”, surrounded by “Microbacterium”, so very likely miscategorized somewhere. |
| 1Dt100g | Figure 4. 1Dt100g, 3Dt1c, 3Dt2c and 3Dt2j | Odd sample, very distant to everything else, no species name means we don’t know if that is correct, but it is still in the right genus at least |
| 3Dt1c | Figure 4. 1Dt100g, 3Dt1c, 3Dt2c and 3Dt2j | No species, matched up with another without species ID, so likely the same but unhelpful for our purposes |
| 3Dt2c | Figure 4. 1Dt100g, 3Dt1c, 3Dt2c and 3Dt2j | Matched with a speciesless sample, there are an odd number of those in this tree, maybe something to look into, so it might be correct, there are no other samples on the tree under “panni”, maybe should check NIH for “Sphingomonas panni” as if there is one and it isn’t on the tree, our one could be miscategorized |
| 3Dt2j | Figure 4. 1Dt100g, 3Dt1c, 3Dt2c and 3Dt2j | Good sample, sister to a reference sample of the same species |
| 1Dt1h | Figure 5. 1Dt1h | Far away from all other groups on its own and in the wrong genus cluster, it’s a “Microbacterium” surrounded by “Brevibacterium” so very likely wrong |
| 1Dt100h | Figure 6. 1Dt100h and 2Dt1e | Sister to the entire rest of the tree, so quite distinct, but the “closest” reference is of the same genus, this is another without species ID, so it could loosely be the same, I think its enough to confirm genus anyway |
| 2Dt1e | Figure 6. 1Dt100h and 2Dt1e | No species ID, but sister group to a reference sample under “Brachybacterium subflavum”, very likely could be the same species. |

Note: I could add checkm2 stuff as seems relevant / might explain why things don’t match.

I just now noticed that 1Dt1h is a Microbacterium surrounded by Brevibacterium

And

2Dt1l is a Brevibacterium surrounded by Microbacterium

Is it wholly possible that these samples got their names flipped somewhere

“I believe that the phylogenetic trees show that the sample IDs of these two samples got switched somewhere in the process of data collection and manipulation”

# November

## 1st – 22nd November:

I started November with a cluster of field trips which meant that I did not have much time for this project, I however have started on my presentation for this. Which has lead to a greater appreciation of why this work is important. I will next sequence the frog microbiome samples through the R script for identifying KEGG pathways, stored here:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\01 inputs

I plan on doing more background reading for this project as this is the final planned section.

I found a very interesting article on Chytridiomycosis

I will work on better understanding API calls.

## 23rd – 30th November:

I started stepping through the transformation file “analyse\_eggnogmapper\_excel\_annotations.R”. found here: C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\01 inputs/

This lead me to the conclusion that it would be best to rewrite the code as a learning exercise. Im unclear whether the metadata file should only contain genus as the metadata file I produced contained genus and species. The metadata file that Aaron produced seemed to contain custom groupings made up of host and bacterium genera. I will produce a method for the creation of the new pipeline going from the .fasta files to the final heatmaps.

#### 26th November

After confirmation from Aaron, I was given permission to flip the names for accessions, 1Dt1h and 2Dt1l, due to likely label swapping before I got the data. I will then use these phylogenetic trees for two purposes:

* downloading of sequence data for annotation, possibly using an API call, this will lead into the pipeline for KEGG enrichment, possibly using the file I made, and produce heat maps to differentiate by genus
* the creation of a table based on the metadata for the online samples downloaded presenting the: accession ID, taxonomy broken into two columns for filtering and the origin/ host for each

I started by creating the blank excel file for the creation of this table, found here:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\04 outputs\metadata for online reference samples.xlsx

I used excel formulas to automatically take the full name and split it into genus and species so that the genus can be used to filter the table.

Then I completed the rework of the two mislabeled trees, which was trivial due to the automation in place in the files here:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\02 data\scriptmakerdatasets version 1.xlsx

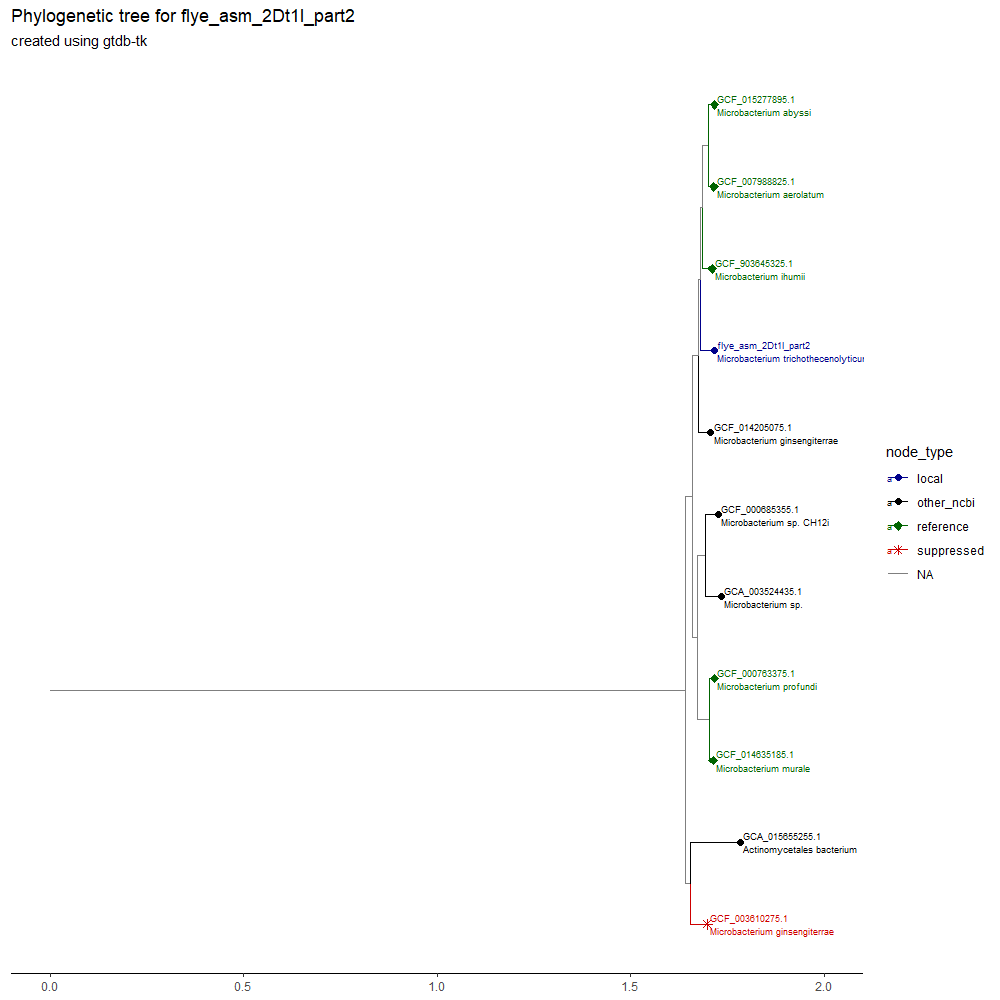
And

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\03 data\_analysis\call API and process json - with legend.R

As I only needed to swap the position of the two names, producing these two trees:

A screen shot of a computer

Description automatically generated



## Conclusions

This study lead to the revelation that 2 samples (\_\_\_ and \_\_\_) had their associated “blastn” names switched at some point. However, they were otherwise correctly associated. This was a good exercise in using R code in a novel scenario, as I could not rely on much previous experience in graph generation, as these use a different syntax for creation. I made many mistakes and so reran these multiple times, it was a good learning experience. The data that created these will go on to be used for the next analyses.

The day ended with the creation of my pipeline of work for the next few weeks: pictured on the Trello board below(just a website that is good for organisation)

A blue frog with black spots

Description automatically generated

### 27th-30th November

I chose PostgreSQL as a database tool because I had heard it was the go-to system for other academics. It is in this that I created several tables for data storage. However, the manipulation can still be done in R. Firstly, I created the seeding script, stored here:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\03 data\_analysis\databaseseedingscript.R

This file uses two packages, “RPostgres” and “DBI” to make a connection and manipulate tables in my PostgreSQL database, on my machine, from the R code itself. The operations in the aforementioned packages makes it possible to write SQL code in R, and as long as it is between speech marks, it will be passed into SQL and the database will act as if I am using the code inside its user interface.

The first thing I did was initiate the connection using the dbconnect() command. Then, I created a table for the local samples in the database, and populated it, all from inside this script. This table contains colums for: an internal id, the accession id, the taxonomy split between genus and species, the host and the tree id(another unique identifier, but this is because there are two trees with multiple accessions in them). The second table created was only an intermediary for the online samples, containing the file name and the path to it. This data was inserted using a for loop iterating over the list of names, relatively simple compared with what is to come.

The third table is the important one, containing several identifiers for each online sample(unique id, accession id and tree id again), as well as a field for host, taxonomy and a special field for containing the full JSON file I get from the first API call, which was very difficult to do and thus, happens in the next script. Also populated using a for loop. The final step of this script updated the first table so that it contained a tree id, something I didn’t know how to do, but knew I wanted when I first created the table.

This was done in late-November 2024, in January 2025 ncbi are changing the rules of the API call so that only 10 requests can be issued per second.

# December

## 1st – 8th December

### 1st december

This leads into my second script:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\03 data\_analysis\queryforncbi.R

Wherein I call the first API for the ncbi to get just the JSON files containing data about the samples. I will later call another one to get the .fna (fasta) files for processing in eggnog, but not now. This uses yet another for loop to construct a command that is passed to the ncbi website by the API call, using the httr package, to get the .JSON format file, then store it as text and add it to my online samples table. After this point I then found out how to use this file to populate the other columns of the table using some very sophisticated code. This leaves me primed to pass in the names of these files again to the ncbi database to get the accessions for annotating. This step has given me the host and taxonomy required to later filter what I get to fit Aaron’s specifications(two analyses, one by genus of bacteria, matching the ones on our samples, one by amphibian hosts, as amphibian hosts are rare hes given me permission to extend the scope to the family of bacteria, which is possible thanks to viewing the unpruned trees in dendroscope, which I should have described far above here, in “\_\_\_”). Then I will pass them into a final R script for discovering the KEGG pathways and generating the final heatmaps.

### 2nd - 4th December

After consultation with Aaron Comeault, I went back to my .tree files generated from GTDB-TK, here:

C:\Users\nancy\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\99 archive\gtdb\_tk\_all\classify\gtdbtk.bac120.classify.tree.4.tree

I decided to only do one tree for now as a proof of concept to present at a meeting on the 8th, the rest will be done later if this is a good path to follow. This tree is pruned much further back at the order (Sphingomonedales). This is beneficial as it casts a wide net for filtering by amphibious hosts and can be recut later to just the specific genus required for that filter.

As a test of Eggnogmapper’s abilities, I am going to take 5 random accessions and, for now, manually download them. Then I will place them in a zip file for running in eggnogmapper.

### 5th December

I tried again today to access hawk and was immediately successful, thus I concluded the ssh box must have been down yesterday. I used the “module avail” command to find that “eggnog-mapper/2.0.1” is a module I can run, this fixes one of my issues as I am assuming many files can be processed at once, given hawk is a supercomputer.

**As of the 10th of December I had a talk with Aaron Comeault about how best to structure the document and the method decided below was preferred, however as this document is a reflection of my development I am not going to retroactively change what I have done.**

# Generating Heatmaps for KEGG Pathways

## Summary

This section is a follow on from the previous section on Phylogenetic trees. Using the relationships in the trees, i can conduct analyses on the differences between bacteria. This section has two analyses; firstly, a comparison of genera inside of their family, this is only possible with genera containing multiple of our samples(Sphingomonas and Microbacterium). This will help us to understand how our samples from the dying poison dart frog differ from their close relatives. Secondly, an analyses of the genera all our samples fall in will be carried out, this will tell us the different functions of each and what role they may have in the microbiome.

## Methods

This section involves the use of many R scripts with complicated mechanics, such as API calls and for loops, as well as a package “PostgreSQL” used to connect up to a SQL server that will be used to store large quantities of files. I will explain all of these when relevant.

Firstly, I created the preparation file, called “databaseseedingscript”, stored here:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\03 data\_analysis\databaseseedingscript.R

This script largely creates and populates some preparatory tables so that there is a place to send files I get from the ncbi database.

Firstly, a single fasta file from the local samples was taken to test annotation by eggnogmapper version 2.0.1 on hawk SCW, this was done on the 15th of December 2024. Using a file called “testnog”, here:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\03 data\_analysis

This file has some missing elements but did generate an output that could be turned into an excel file and thus be used by an R script. This will be modified in future after a meeting with Aaron Comeault. But acts as a firm base for making a proper script to run many files.

## Results

## Conclusions

# Protein heat maps

## Introduction

As a part of troubleshooting for the Kegg Pathway heat maps project (above) I found that there are already annotated sequences already on the ncbi database. To see this you can go to an accession / sample, and under the introductory table with the metadata in it is a button with the words “View annotated genes” next to a little graphic in a blue circle. This takes you to a page with a lot of information about genes found in each sample. This is similar to what we want with the Kegg Pathways, and is much easier as I can bypass eggnogmapper. However, I don’t know how they were annotated so I cannot be sure if Aaron will accept it. Thusly, I am going to do them as it will be much faster due to several snags and pivots in the Kegg method, this should both mean that I can get an end-to-end pipeline working and hopefully it will help inspire methods to help with the Kegg stuff. Either way I am learning R and SQL so it isn’t a waste of time and will produce something hopefully relevant.

## methods

This begins at the same stage in the pipeline as Kegg analysis. Using Dendroscope, an app I downloaded to my machine sometime in mid-late 2024 I pruned the trees outputted from GTDB-TK (above, link when fucking organise). This was done in two groups, the same two analyses I was told to run for Kegg:

* a genera-genera comparison inside of a family, when a genera within that family contains more than 1 bangor-made sample (so Sphingomonas and Microbacterium) (clarify better and link to trees above). This will produce 1 heatmap
* a genera-genera comparison for only the genera containing bangor-made samples

these trees are stored here:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\02 data\trees for heatmapping(family)

These I created on the 16th of December 2024

And here:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\02 data\trees for heat mapping(genera)

Created on the 18th of December 2024

[replace with github links when that gets up and running]

These tree files will then inform a list of accessions in the SQL server (outlined above?). from this I can run an API on the ncbi website:

<https://www.ncbi.nlm.nih.gov/datasets/docs/v2/api/rest-api/#post-/genome/annotation_report/download>

that will download the information about the annotated genes, bypassing the troubles with eggnogmapper completely, into another table in the SQL server. From this I can develop an R script to take this output and turn it into a set of heatmaps:

C:\Users\tobyn\OneDrive\Work\tnunn\_research\2024-25 Research\frog bacteria\03 data\_analysis\proteinheatmapmaker.R

## results

## conclusion

# serious issue:

18th December 2024: I was taking another look at the phylogenetic trees and they are seriously suspect at not being correct, the tree for 1Dt2d has samples in the wrong genera, 2 of the 10 samples cant be used because of this so that means half of the remaining are sphingomonas and Aaron isn’t going to weigh in over Christmas so I might be stuck / have to go back to square one if this cant be resolved.

# Final report

Here is a “write up” of studies conducted as part of my research as a link to an R markdown version uploaded to the github server, as R is a very large part of what I have done, I will also export it to word or powerpoint as instructed.

I have to admit that this got a little bit away from me, both making sure to do this as I go along and keeping all my directories clean. I thought a bit too much about churning out results, but not making things good. its too much so im going to start again in 2025, knowing what I know now it will be very easy and quick to build back up to here, but better, so this is the rough notes of where I was in 2024, but the next one for 2025 will hopefully be much cleaner and connected and will give me more of an idea so I fully know what is going on and generally so things can not be so confusing. So over Christmas im going to take a “pause and reset”, basically burn it all down and start again but better as I know a lot more about how to do a lot of things now than where I was in the summer.