16S rRNA Taxonomic Profiling using QIIME

Introduction

With decreases in sequencing costs and massively parallel approaches to library production 16S sequencing has become a popular technique used to investigate taxonomic composition of microbiomes. In contrast to previous molecular techniques, high throughput 16S rRNA sequence produces an enormous amount of data which has necessitated the development of computational solutions for analyzing this mountain of data. Two of the most popular methods for analyzing 16S data have been Quantitative Insights into Molecular Ecology (QIIME; pronounced chime), and mothur. Each of these software packages have strengths and weakness, for simplicity we will use QIIME in this module.

The previous datasets we have worked with were all publicly available and previous quality controlled. As we generated this data ourselves we will need to do some quality control (QC) steps before the data is ready to go into the main workflows of QIIME. It is common for sequencing cores to provide some demultiplexing and initial data filtering steps before the provide the end user (us) with the data. We have requested the CGRB demultiplex the data for us. They also provide us with some information about the quality of our data. Though they have conducted this quality control step for us we will start by running the analysis on some test data so we have experience with this important step. We will then perform more quality control steps using QIIME, and finally we will annotate our reads in OTU space using QIIME.

Objectives

1. Gain experience in the use of quality control softare used to preprocess 16S data
2. Learn how QIIME QC's data and how to modify its behavior
3. Gain experience in the operation of QIIME
4. Increase competency working with open source data
5. Preprocess, QC, and cluster our data using QIIME

Protocol

Quality Control and Split Libraries

1. Get set to go.

```
$ bash
$ source /nfs1/Teaching/CGRB/525_f15/local/bin/.qiime
```

All of our data from our 16S sequencing run is in fastq format and as mentioned above is not quality controlled. To start let's have a closer look at the anatomy of the .fastq file.

2. Start by opening the any fastq file in your ~/raw_metagenomes directory

```
$ less ~/raw_metagenomes/<input.fastq>
```

You should see something like the output below.

```
@HWUSI-EAS712_102317177:7:100:10000:10052/1
TCTCCGCTGTCTTAACCTTATGGAAAAGCCGACCTCAGGTGAT
+HWUSI-EAS712_102317177:7:100:10000:10052/1
GGGGGGGFGGGGGGGGGGGGGGGFGFGGEGGGGEEG?DFBGGE
```

Each line of the file has a specific purpose.

1. Line 1: A header line that starts with an '@' character and is followed by a sequence identifier and an optional description of the read.
2. Line 2: The sequence corresponding to this read.
3. Line 3: A second (optional) header line that starts with an '+' symbol. Note that although the description is optional here this line is required and at minimun a '+' symbol must be present.
4. Line 4: This line contains quality score for each nucleotide in line 2 (Phred score).

The Phred score reflects the confidence that we have in the base that the score belongs to.

Phred Score	Probability of Incorrect Call	Accuracy
10	1 in 10	99%
20	1 in 100	99%
30	1 in 1,000	99.9%
40	1 in 10,000	99.99%
50	1 in 100,000	99.999%
60	1 in 1,000,000	99.9999%

Next it is generally desirable to gather some summary statistics and other information about our sequence file before we spend hours on downstream analysis. We probably would not want to analyze these reads by hand (there are millions of reads per file!) so we use more free software to do it. The software that we will be using is called FastQC (http://www.bioinformatics.bbsrc.ac.uk/projects/fastqc/) . This software has a command line interface as well as a graph user interface (GUI), you will use the command line version. Since our data has already been run through FastQC we will be running a sample file.

3. Take a look at the fastqc help file

```
$ fastqc -h
```

   One unfortunate drawback of using free software is that sometimes the documentation is sparse and the only way to know what to do is to access help files like you just have done (note fastqc actually has pretty good documentation). Passing the `-h` (or the `--help`) option will often get you a help file like this that will at the very minimum give you a little info on how to use the software called a usage statement.

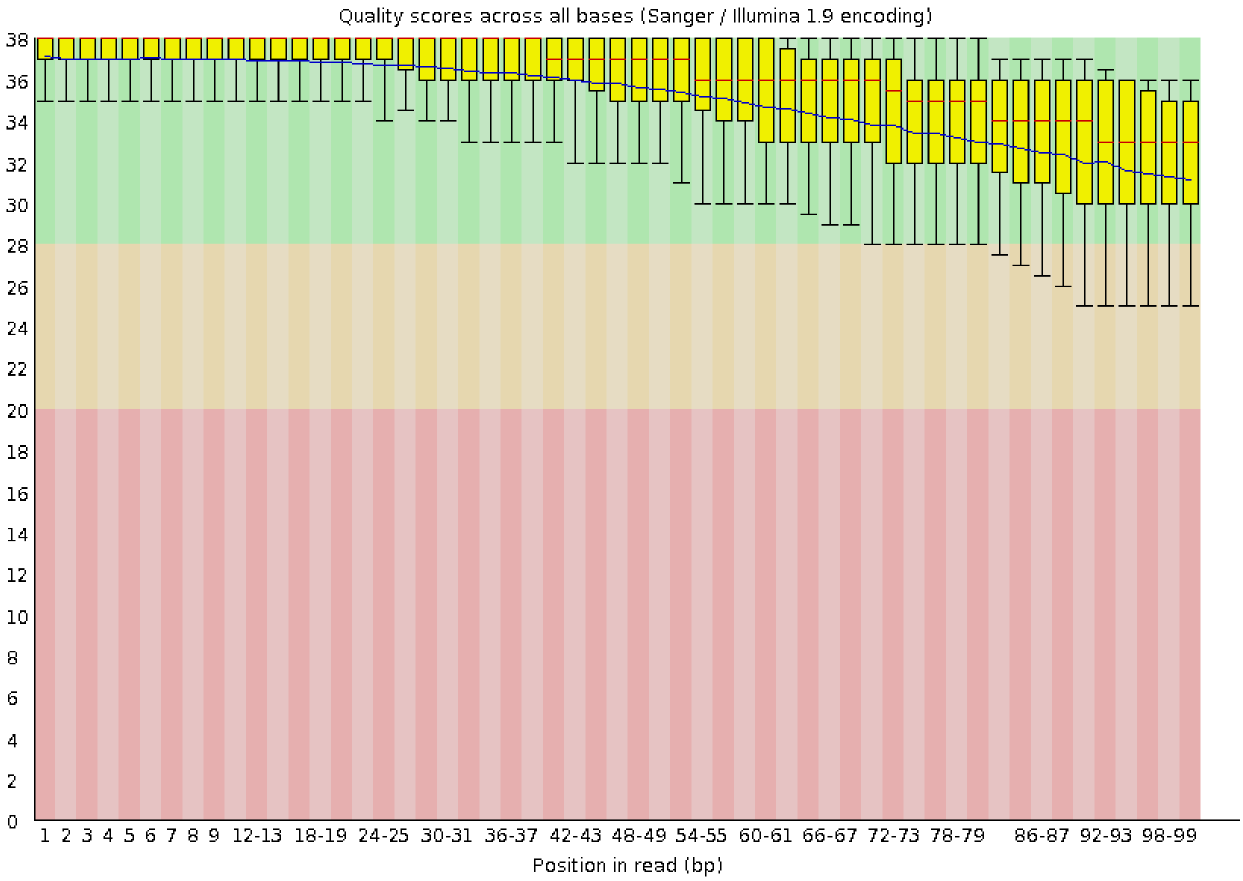
4. Using the information from the fastq usage statement try to run this software using the SRS011405.fastq file in raw_metagenome as your test file. If you run into issues try looking through the documentation on the website above. **Hint:** Don't worry too much about the options except `-o.`

5. After you are finished you should have two files one ends in `.zip` the other should end in `.html`. Using the skills you have gained in previous labs transfer the file ending in `.html`.

6. Let's open this output file. It should open in your web browser.
   The first thing that you should see is some summary statistics of the input data. After that there will be a series of images illustrating various parameters of the data. A description of each plot can be found below.

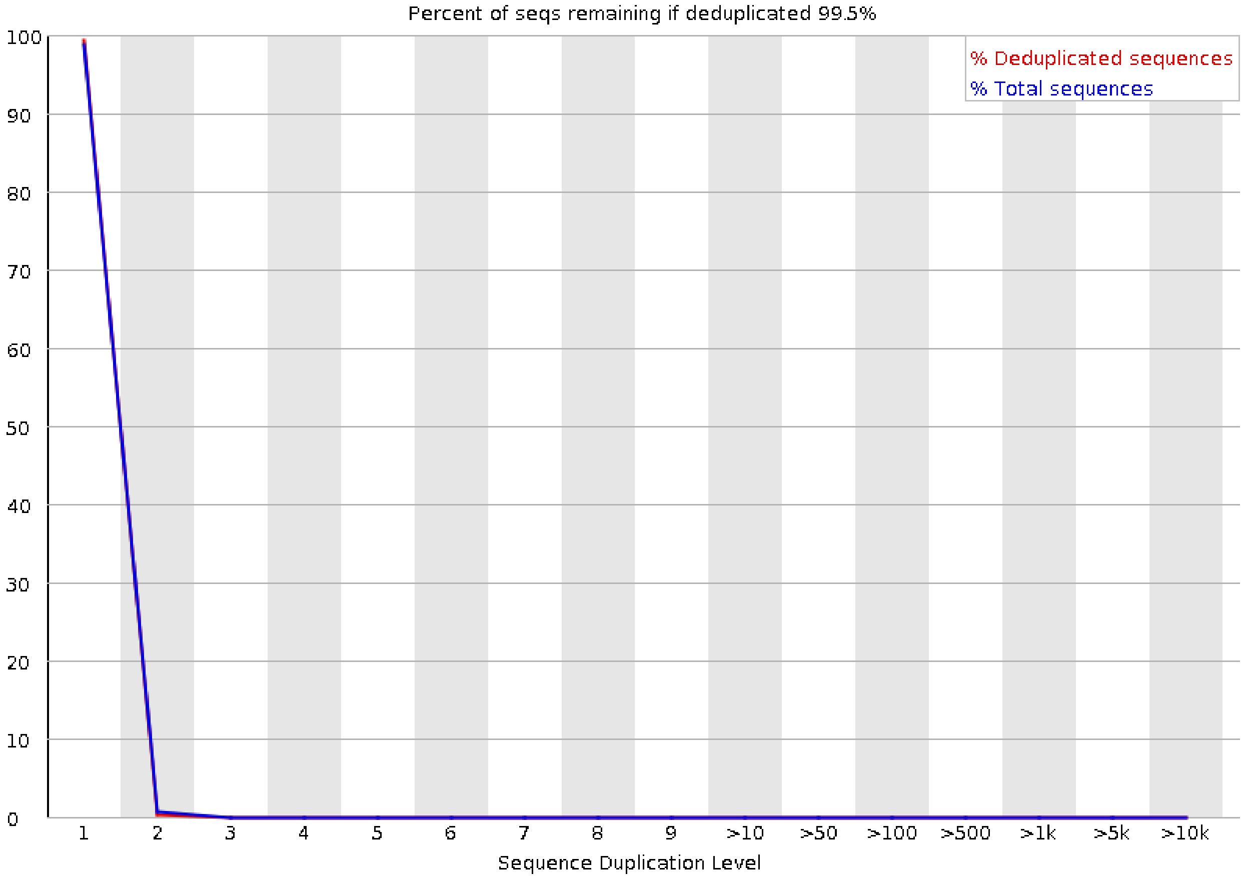
   1. Per base sequence quality: A per position average Phred score
   2. Per tile sequence quality: Sequence quality per tile of the flow cell. This wil help identify problems in quality associated with the sequencing equipment.
   3. Per sequence quality score: Average quality score across sequences.
   4. Per base sequence content: The proportion of each base at each position.
   5. Per sequence GC content: The average GC content per sequence.
   6. Per base N content: N bases occur when the base caller cannot call a base with sufficient confidence. This image summarizes N base content per position.
   7. Sequence length distribution: A distribution plot of sequence length for the library.
   8. Sequence duplication levels: The percent sequences duplicated at a specific level. The title will include an estimated percentage of reads retained after de-duplication.
   9. Over represented sequences: Sequences the make up more than 0.1% of the library.
   10. Adaptor Content: Displays the cumulative percentage of reads that have adaptor content at a given percentage.
   11. Kmer Content: Plots the relative abundance of Kmers at every position in the library.

7. Lets take a closure look at a few of these plots. Navigate to the per base sequence quality plot.



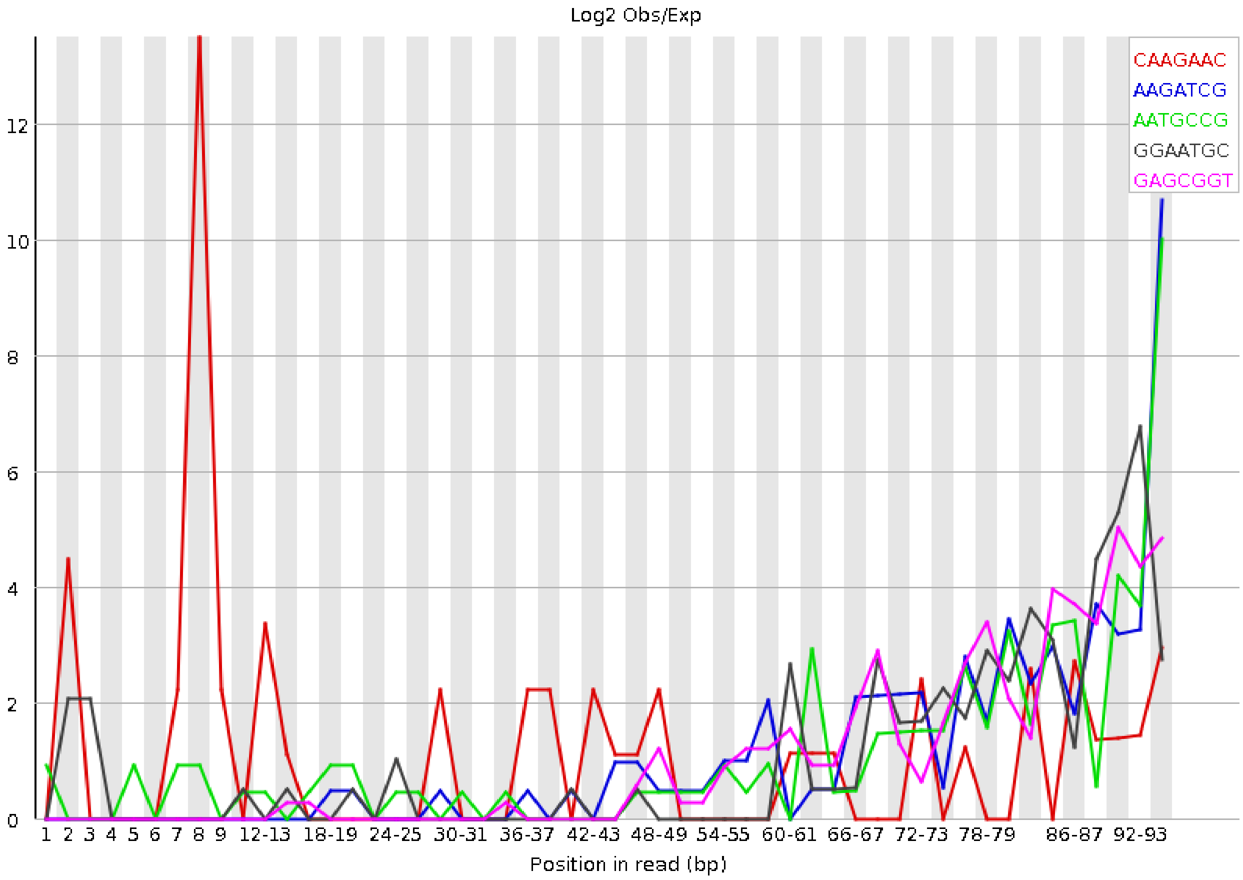
It looks like the quality of this test library is pretty good (i.e., all of the average Phred scores are > 20). Next move onto the sequence duplication levels plot.

what error rate does a q of 20 correspond to?



This plot shows that this library is nearly completely composed of unique reads. However, there does seem to be some overrepresented Kmers. Check out the Kmer content plot.

What is your duplication rate (i.e., the percent of reads that would be filtered)? If you donâ€™t have any (or enough) sequences partner up with someone else in your group.



Are there any positions in your sequences that appear to be biased? Consider both the Kmer and per base sequence content plots. What do each tell you?

8. Now that we know how perform fastqc analysis let's have a look at the actual plots that the CGRB generated for our data. Using your web browser navigate to the link provided for your fastqc data. How are these plots differ from the test data? Why might this be?

9. After looking over each of you fastqc files let's get the data ready for entry into qiime. First copy your files to your home directory. You will need to get the location of these files from your instructor/TA.

```
$ cp <file_path_to_cgrb_files> ~/raw_16S
```

10. Next we need to prep the files for QIIME. Since these files have been demultiplexed and trimmed of barcodes we need to concatenate them back together and create a file that will correspond to barcodes for each library. To do this we will use some custom software called bfm2.0.pl. Then we will concatenate the sequence files together.

```
$ cd ~/raw_16S
$ perl
/nfs1/Teaching/CGRB/525_f15/local/bin/bfm2.0.pl ./
$ cat *fastq.gz > <groupX_all_seqs.fastq.gz>
```

A little more info on the above.

- `perl`: This is the language that bfm2.0.pl was written in and this command tells the computer to execute the commands in this script using the perl interpreter.
- `bfm2.0.pl`: This software will take zipped fastq files and create a qiime ready barcode file. It does this by grabbing the appropriate barcode file from the file headers in the fastq file. It then prints these headers and barcodes to a barcode file. Importantly, this occurs in a specific order (QIIME breaks if it is out of order).
- `./`: This is an example of a relative path, and it points to the directory we are currently in.
- `cat`: This concatenates the input. In our case this is all the .fastq.gz files in the current directory.

Be sure to change the name of the output file.

11. Now we will perform quality control and library splitting with QIIME. Recall that you made the mapping file earlier and barcode_file.fastq is the output of bfm2.0.pl

```
# Be sure to change the name of the input file to
reflect what the output was above.
$ split_libraries_fastq.py -m <map_file.txt> -i
<groupX_all_seqs.fastq.gz> -q 19 -b
barcode_file.fastq -o ~/qiime_out/Qiime_spl_libs_out
&
```

This step might take a few minutes. While you wait let's take a look at some of what is happening under the hood. QIIME uses a popular QC algorithm to trim and demultiplex reads. This is a free software package called ea-utils. Specifically the algorithm used by QIIME is fastq-mcf. Lets look at the help file.

```
$ fastq-mcf -h
```

QIIME sets most of these options for use, but we did set one manually by passing the `-q 19` option. This set the quality threshold to Phred >19, that means that nucleotides below this are in danger of being trimmed from the reads. Our reads had some quality issues at the end of the reads and these low-quality runs of nucleotides are trimmed during the split libraries step. Primer/adaptor sequences are also trimmed from the ends of reads at this step.

12. The next step is to cluster the sequences into OTUs.

```
# Note: $reference_seqs and $reference_tax should be
written as is.
$ cd ~/qiime_out/Qiime_spl_libs_out/ $ nohup
pick_closed_reference_otus.py -i seqs.fna -r
$reference_seqs -t $reference_tax -o
~/qiime_out/Qiime_spl_libs_out/Closed_ref_otus -p
/nfs1/Teaching/CGRB/525_f15/local/bin/qiime_params.tx
t -a -O 5 -f &> nohup.log.out &
```

This command might warrant a bit more explanation. I have included an explanation of each part of this command below.

- `nohup`: This is a very special command that prevents out job from aborting even if the connection to the server is severed. This allows us to log out and have the job run until completion. **Note:** The & at the end of the command will cause this job to be run in the background, but if you log out the job will still terminate. Nohup is one of the best ways to ensure this does not happen.
- `-i`: These are the input sequences for the OTU picking step and where the output of the split libraries step.
- `-r`: Path to the reference OTU sequences. Recall from the discussion today that closed reference OTU picking occurs against a reference database. We point to this reference using the `-r` option.
- `-t`: Path to the reference taxonomy. Taxonomy is assigned to each of the resulting clusters using a pre-constructed lookup table of OTUs to taxonomy (think of this like a dictionary where the OTU is the word and the taxonomy is the definition). The path to the taxonomy file is passed with the `-t option.`
- `-o`: The output directory.
- `-p`: The path to the parameters file. This file contains options passed to scripts that are run inside of the workflow
- `-a`: Tells the computer we want to run this in parallel (i.e., with several processors).
- `-O`: Number of processors to be used (integer).
- `-f`: Specifies that we want to overwrite the output directory if it already exists (i.e., the script broke but still created the directory).
- `&>`: Redirects the STDERR to the log file.
- `&`: Runs the job in the background.

This will likely take several hours so we will stop here for the day.

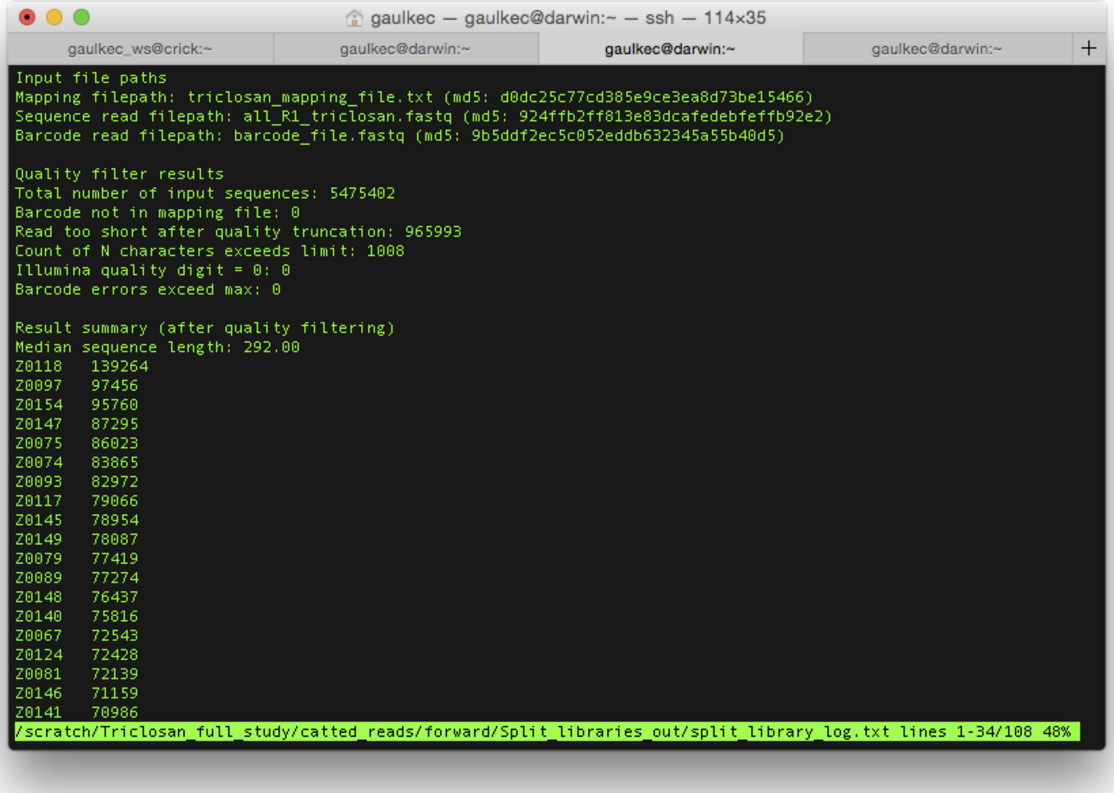
QIIME Core Diversity Analysis

QIIME provides set of tools for analysis of alpha- and beta- diversity as well as taxonomic structure. In this module we will use these tools to examine the differences between the microbiomes we sampled.

1. First, we need to take a look at the number of sequences generated in the split libraries step of the protocol to determine some parameters for this analysis.

```
$ less
~/qiime_out/Qiime_spl_libs_out/split_library_log.txt
```

The file should look something like the image below.



```
Input file paths
Mapping filepath: triclosan_mapping_file.txt (md5: d0dc25c77cd385e9ce3ea8d73be15466)
Sequence read filepath: all_R1_triclosan.fastq (md5: 924ffb2ff813e83dcafedebfeffb92e2)
Barcode read filepath: barcode_file.fastq (md5: 9b5ddf2ec5c052eddb632345a55b40d5)

Quality filter results
Total number of input sequences: 5475402
Barcode not in mapping file: 0
Read too short after quality truncation: 965993
Count of N characters exceeds limit: 1008
Illumina quality digit = 0: 0
Barcode errors exceed max: 0

Result summary (after quality filtering)
Median sequence length: 292.00
Z0118   139264
Z0097   97456
Z0154   95760
Z0147   87295
Z0075   86023
Z0074   83865
Z0093   82972
Z0117   79066
Z0145   78954
Z0149   78087
Z0079   77419
Z0089   77274
Z0148   76437
Z0140   75816
Z0067   72543
Z0124   72428
Z0081   72139
Z0146   71159
Z0141   70986
/scratch/Triclosan_full_study/catted_reads/forward/Split_libraries_out/split_library_log.txt lines 1-34/108 48%
```

This file will provide a brief summary of the quality filter conducted above in the first several lines. After the summary there is a two column list of the sample IDs and reads per library. Most of the libraries in this particular run generated a large number of reads (>50,000), however, as in most sequencing runs, there were also some failures (<1000 reads; not shown). Depending on the quality of you libraries (and how lucky you are) you should have several thousand to several tens of thousands of reads. If you have some bad libraries it is ok, but we will probably filter those libraries out in the next step.

This file is useful as it provides us an the number of sequences per library after filter, but what if we want to know how many reads were actually recruited into clusters? Recall that closed OTU picking is done against a reference database, this means that some reads will not be recruited into an OTU since we expect that some reads will not share enough sequence identity with OTUs in the reference database. To determine how many reads were recruited into OTUs we will need to run another script.

2. We will use part of the biom software package to produce a summary of the biom table we created yesterday. A biom table is just a specifically formatted table type that is easy and fast to parse.

```
# remember to get into bash and source the .qiime
file that we sourced yesterday.
$ cd ~/qiime_out/Qiime_split_libs_out/Closed_ref_out/
$ biom summarize-table -i
~/qiime_out/Qiime_split_libs_out/Closed_ref_out/otu_t
able.biom -o biom_sum.txt
$ less
```

3. Record the values in this file and note the lowest and highest values. With the distribution of the data present in in mind pick a number of reads which the majority of libraries you have constructed is above. For example, if you have eight libraries and the distribution of read counts was [1,500, 10,000 ,30,000, 23,000, 12,000, 100,000, 15,000, 0] a good value to pick would be 10,000. If you have any questions about your choice ask an instructor or TA

4. Record the number you picked you will need it later. We will us this value to rarify our data so that all diversity and statistical measurements are done on the same size of library. Rarefaction accounts for uneven samling of data due to experimental factors.

   What was your number?

5. Next before we proceed to the beta-diversity analysis let's look at one more file.

```
$ less ~/qiime_out/Qiime_spl_libs_out/seqs.fna
```

   Recall from yesterday that we gave QIIME a set of fastq files and used it to do quality trimming and filtering. Also recall that from our fastq reports we know that there is a large amount of sequence duplication in our data. This file contains the unique set of quality control reads for our libraries. Note that each fasta header now contains a sample ID, this format is important for QIIME and if you file is not in this format QIIME will break and you will likely have to write some code to push your file into QIIME.

6. Lets now get to work on the beta diversity analysis.

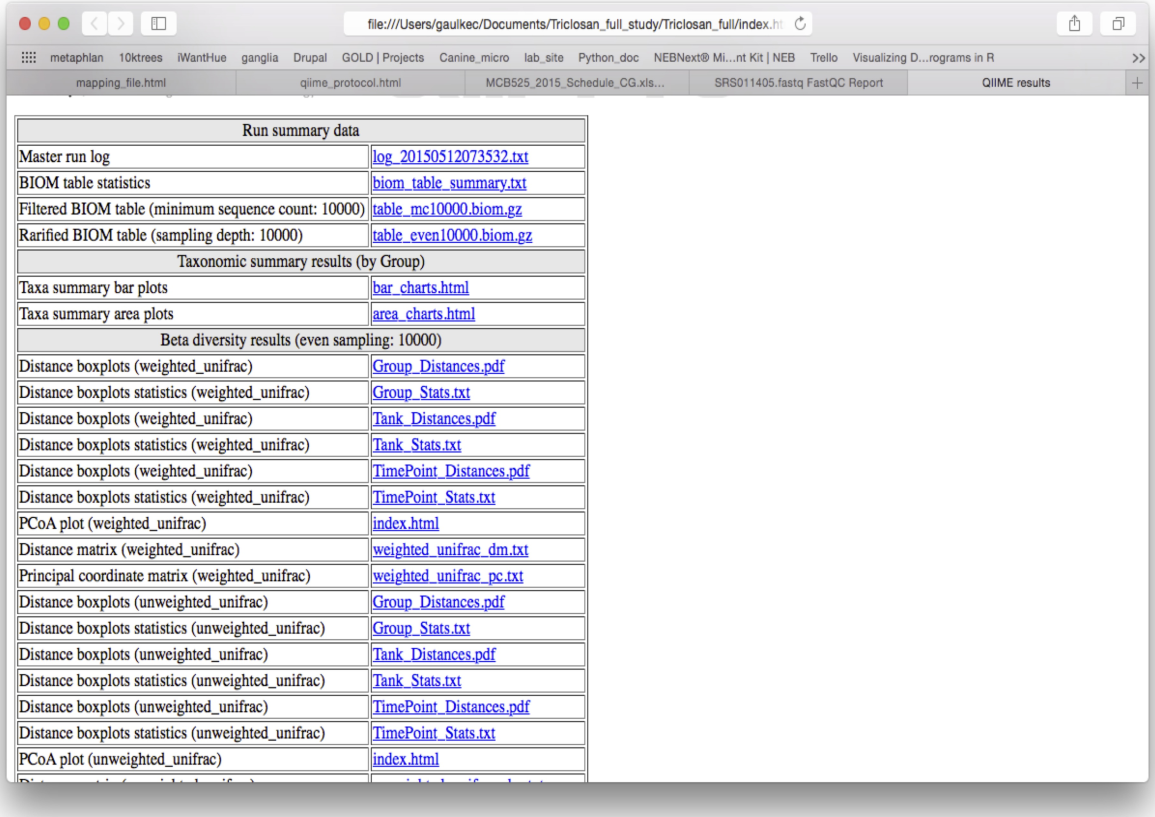
```
$ cd ~/qiime_out/Qiime_spl_libs_out/Closed_ref_otus/
$ nohup core_diversity_analyses.py -i otu_table.biom
-o Core_diversity_analysis -m <your_mapping_file> -e
<your_value_from_above> -t
/nfs1/Teaching/CGRB/525_f15/data/greengenes/gg_13_8_o
tus/trees/97_otus.tree -c Site &> nohup.log.out &
```

This will take a while to complete. While you wait take a look at the explanation of what each portion of the command does below.

- `nohup`: Prevents job from being terminated due to logout.
- `-i`: The input OTU table.
- `-o`: The output directory.
- `-m`: The mapping file you created earlier.
- `-e`: The rarefaction depth you selected above.
- `-t`: The path to the greengenes reference tree. This tree is used to calculate some of the diversity metrics produced by the script.
- `-c`: The category/categories you want statistics for.
- `&>`: Redirects the STDERR to the log file.
- `&`: Runs the job in the background.

7. Once this completes let's transfer the **whole directory** so we can examine the results using the file transfer client as with previous protocols.

8. Once you have transfered this directory look inside for a file called index.html. Open this file in your web browser. It should look similar to this.


file:///Users/gaulkec/Documents/Triclosan_full_study/Triclosan_full/index.ht
metaphlan
10ktrees
iWantHue
ganglia
Drupal
GOLD | Projects
Canine_micro
lab_site
Python_doc
NEBNext® Mi...nt Kit | NEB
Trello
Visualizing D...rograms in R
mapping_file.html
qiime_protocol.html
MCB525_2015_Schedule_CG.xls...
SRS011405.fastq FastQC Report
QIIME results

Run summary data	
Master run log	log_20150512073532.txt
BIOM table statistics	biom_table_summary.txt
Filtered BIOM table (minimum sequence count: 10000)	table_mc10000.biom.gz
Rarified BIOM table (sampling depth: 10000)	table_even10000.biom.gz
Taxonomic summary results (by Group)	
Taxa summary bar plots	bar_charts.html
Taxa summary area plots	area_charts.html
Beta diversity results (even sampling: 10000)	
Distance boxplots (weighted_unifrac)	Group_Distances.pdf
Distance boxplots statistics (weighted_unifrac)	Group_Stats.txt
Distance boxplots (weighted_unifrac)	Tank_Distances.pdf
Distance boxplots statistics (weighted_unifrac)	Tank_Stats.txt
Distance boxplots (weighted_unifrac)	TimePoint_Distances.pdf
Distance boxplots statistics (weighted_unifrac)	TimePoint_Stats.txt
PCoA plot (weighted_unifrac)	index.html
Distance matrix (weighted_unifrac)	weighted_unifrac_dm.txt
Principal coordinate matrix (weighted_unifrac)	weighted_unifrac_pc.txt
Distance boxplots (unweighted_unifrac)	Group_Distances.pdf
Distance boxplots statistics (unweighted_unifrac)	Group_Stats.txt
Distance boxplots (unweighted_unifrac)	Tank_Distances.pdf
Distance boxplots statistics (unweighted_unifrac)	Tank_Stats.txt
Distance boxplots (unweighted_unifrac)	TimePoint_Distances.pdf
Distance boxplots statistics (unweighted_unifrac)	TimePoint_Stats.txt
PCoA plot (unweighted_unifrac)	index.html

QIIME has provided us with a large amount of output to shift through here. Lets start by checking out the taxa plots.

9. Lets take a minute to go over some of these figures. Start with the taxa summary barplots (by group). These plots show the relative abundance of each taxonomic level from phyla to genus. The colors in these plots can be a little tough to pick out so if mouse over the slice in the plot a label will appear.

   What are the major differences in taxa abundance (by eye)?

10. We can see if the differences are significant by going back to the master index and opening the files under the group significance results heading. The header should look similar to the one below.

```
OTU Test-Statistic P FDR_P Bonferroni_P Buccal_mean
Ear_mean taxonomy
```

The fields of this file are:

1. OTU: The OTU ID.
2. P: The p-value associated with the test statistic for this OTU.
3. FDR_P: false discovery rate
4. Bonferroni_P: Bonferroni corrected p-value
5. Buccal_mean: Mean OTU abundance for the buccal samples.
6. Ear_mean: Mean OTU abundance for the ear samples.
7. taxonomy: taxonomy string associated with this OTU.

This information provides us with a statistical framework to support our assertions that there are differences in taxonomic abundance between the two body sites sampled. Although these statistical test provide us with more evidence that these two groups are different the magnitude of the difference remains unclear. The alpha- and beta-diversity measurements can provide us with some idea of how different these communities actually are.

11. QIIME plots beta-diversity in two ways, boxplots and principal coordinate plots. Lets take a look at the principal coordinate analysis (PCoA) plots first. There are two types of PCoA plots, weight and unweighted. Start by looking at the unweighted plots (PCoA plot (unweighted_UniFrac)).

Unweighted UniFrac PCoA plots give us an idea about how different the communities are in terms of the presence and absence of OTUs or taxa. Unlike bray-curtis distance, this measure is phylogenetically informative (i.e., organisms that are more distantly related contribute more to the overall score than those that are closely related).This plot is interactive so you can rotate the axes by mousing over a point in the plot holding down the mouse button then moving the cursor. To recenter the plot click the `options` tab on the right hand side of the screen and select recenter camera. You can also change the axes by modifying the parameters in the `axes` tab. We can also change the color scheme by clicking on the `colors` tab. If you want to label individual points you can click the `Samples Label Visibility` in the `labels` tab. Finally, you can adjust the visibility and scaling of the data by click on the corresponding tabs.

12. Take some time to play with these parameters and get a feel for the data.

Thinking about these data try to answer the following questions:

- Do the two sample sites cluster together or are they intermixed?
- Are there other dimensions (Principal coordinate axes) of the data that show a different pattern, or does the clustering look the same across all dimensions?
- What do these differences mean in terms of community structure?

Weighted UniFrac PCoA plots give us an idea about how different the communities are in terms of the abundance of OTUs or taxa. As with the unweighted version this measure is phylogenetically informative.

13. Repeat the above process for the weighted UniFrac plots.

14. While the PCoA plots show slices of beta-diversity, the beta-diversity boxplots show the total difference in beta-diversity between and within groups. Lets have a look at these plots. Return to the QIIME index and open the Distance boxplots (weighted) for Site.

These boxplots represent the UniFrac distance between all pairs of samples in the indicated groups. We commonly refer to these distances as between group (i.e., all pairs of samples in two separate groups) and within group (i.e., all pairs of samples within a group) distances. With regard to the current plot, an example of within group distance is the ear vs ear box while ear vs buccal is an example of between group difference. You can think about distance as difference with a value of 1 being no overlap in the community and a value of 0 being completely identical communities. To determine if there are any statistically significant differences take a look at the Distance boxplots statistics file that corresponds to the Distance boxplot file you are viewing.

Thinking about these data answer the following questions:

- Are there differences in beta-diversity accros communities? Within communities?
- Are these differences statistically significant?
- What do these differences mean?
- What would increased within group beta-diversity indicate? Increased between group relative to within?

15. Now repeat the previous step with the unweighted distance boxplots.

16. There is one last thing to take a look at on the master index and that is alpha-diversity. We will concentrate on phylogenetic diversity (PD) for this section. Open the alpha-diversity boxplots for Site,PD_whole_tree. These boxplots represent the minimum branch length that spans all OTUs in the greengenes tree. So if one group had many more distantly related organisms it would potentially have greater phylogenetic diversity. Using these boxplots and the statistics associated with them answer the questions below

    - Are there any significant differences between the groups?
    - If so what do you think these differences might mean in terms of community composition
    - Why might alpha diversity be an important measure? Do you think it might be applicable to other areas such as transcriptome analysis?

Finally, let's take a step back and re-examine beta-diversity.

17. Time to get back to the command line now because we have a couple more statistical tests to run. As a final question we are going to ask if there are statistical differences in beta-diversity between the sites surveyed.

```
$ cd
<path_to_core_div_analysis>/bdiv_even<your_rarefactio
n_depth>
$ compare_categories.py --method adonis -i
unweighted_unifrac_dm.txt -m <ptah_to_mapping_file> -
c Site -o adonis_out -n 999
$ compare_categories.py --method anosim -i
unweighted_unifrac_dm.txt -m <ptah_to_mapping_file> -
c Site -o anosim_out -n 999
```

- `--method`: Statistical method to be used.
- `-i`: The input distance matrix (i.e., the UniFrac distance tables.

- `-o`: The output directory.
- `-m`: The mapping file you created earlier.
- `-c`: The category/categories you want statistics for.
- `-n`: The number of permutations to be conducted.

The output of these commands will be two directories containing each containing one file. This text file will contain information about the method used (i.e., ANOSIM or adonis), and the results of the test. The most informative results in these files is the R^2 value (R for ANOSIM) and the p-value. If the test is significant (i.e., $p < 0.05$) it means that the grouping of samples by site is significant. For adonis, R^2 value is related to the amount of variation explained by the grouping by site. For ANOMSIM, R indicates how strong the grouping is by site.

18. Now try the above commands with the weighted_unifrac_dm.txt files

Do you get different results (i.e., p-values and r/R^2 values)? Why might these values differ?

Now you have some statistics and plots that you can use for your presentations. In the next module we will explore some common data visualization techniques and compare our sequencing results to metagenomes generated by the human microbiome project.