Metagenomic analysis with anvi'o

Anvi'o is a powerful tool for analysis of metagenomic assemblies which integrates some pretty excellent visualizations. Today, we're going to use some previously-assembled metagenomes from the 2018 course: three Salt Pond metagenomes from varying depths.

Step 0: Get Ready

We'll start by using anvi'o on the server. Much like QIIME, we need to load a conda environment specific to anvi'o.

In [ ]:
```
conda activate anvio-7.1
```

Anvi'o requires two basic inputs to work: a FASTA file of your assembly contigs, and BAM files of your metagenome reads aligned to those contigs. We've prepared these for you in /datahaus /metagenomes. Each member of your group should pick one of the three metagenomes; ensure that all three are represented across your group.

In [ ]:
```
mkdir /YOUR_GROUP/YOUR_DIRECTORY/metagenomes
```

In [ ]:
```
cd /YOUR_GROUP/YOUR_DIRECTORY/metagenomes
```

In [ ]:
```
cp /datahaus/metagenomes/saltpond.zip .
```

Step 1: Load contigs into an anvi'o database.

Reads from each metagenome have been mapped to a single set of contigs generated from assemblies of the corresponding metagenomes; this file is called saltpond_contigs_top1k.fasta. For demonstration purposes, this file includes only the 1000 longest contigs.

Use the following command to generate a contig database (~4 mins):

In [ ]:
```
anvi-gen-contigs-database -f saltpond_contigs_top1k.fasta \
-o saltpond_contigs.db -n 'saltpond' -T 4
```

This command will do a few useful things:

- Calculate tetranucleotide frequencies for all contigs
- Split long contigs (>20kb) into smaller bits
- Predict ORFs on contigs using Prodigal
- Store all that in a contigs database called saltpond_contigs.db.

Step 2: Identify genes on contigs

This command will use HMMER to identify hits in the predicted genes on your contigs to ribosomal RNAs and known single-copy core genes from bacteria, archaea, and eukaryotes (~2 mins).

In [ ]:
```
anvi-run-hmms -c saltpond_contigs.db -T 4
```

The -c flag tells anvio which contig database to use, and the -T flag allows us to use multiple processors, because we're impatient people.

We can also use NCBI's Cluster of Orthologous Genes (COGs) to try and identify those ORFs (~4 mins):

In [ ]:
```
anvi-run-ncbi-cogs -c saltpond_contigs.db -T 4 \
--cog-data-dir /opt/anvio_cog_data
```

And use sequences from the GTDB to assign taxonomy. (~1 min):

In [ ]:
```
anvi-run-scg-taxonomy -c saltpond_contigs.db -T 4 \
 --min-percent-identity 80 --debug
```

We can now pick a specific marker gene and see what the taxonomic composition of our metagenome looks like based on solely that gene.

In [ ]:
```
anvi-estimate-scg-taxonomy -c saltpond_contigs.db \
--metagenome-mode -S Ribosomal_L16
```

- *Do taxonomy estimates change for different marker genes?*

Side Note: You can also import other useful pieces of information about your contigs, such as gene calls you've previously generated, into anvi'o. Check out the anvi'o documentation for more info.

Step 3: Examine your contigs

Let's see what we just did to those contigs. Download the saltpond_contigs.db contigs database to your computer via Cyberduck, WinSCP, scp, etc. and start up your **local copy** of anvi'o.

In [ ]:
```
anvi-display-contigs-stats saltpond_contigs.db
```

This command does what it says on the tin. A browser window should open with a histogram of single-copy core genes and other assorted stats.

- *How many eukaryotic, archaeal, and bacterial genomes do we expect to see?*

Step 4: Profiling .bam files

In addition to contig databases, anvi'o also uses profile databases to store information about individual samples. A separate profile database is generated for each sample. For now, you're just working with one sample.

These .bam files were generated by mapping reads from each individual sample to the top 1k contigs using bowtie2, just like the read mapping from our genome assembly workshop.

Replace XXX with your sample (350, 375, 400) and run this command to generate a profile database (~3 mins).

In [ ]:
```
anvi-profile -i SPXXX_top1k_sorted.bam -c saltpond_contigs.db \
-o SPXXX_profile -S SPXXX -T 4
```

This command processes all contigs over 1,000bp (you can change this value if you wish). As part of profiling, anvi'o compares reads from the sample to your contigs and calculates the mean and standard deviation of coverage, as well as identifying any positions with single nucleotide polymorphisms (SNPs).

Step 5: Merge profiles

While you can use anvi'o with a single sample, it's much more fun with multiple metagenomes. Now that you've gotten one metagenome ready, go copy the PROFILE.db file for the other two metagenomes from your group members. Make sure to keep them in separate directories!

In [ ]:
```
cp -r /YOUR_GROUP_DIR/SOMEONE_ELSES_DIR/SPXXX_profile . #x2
```

Now merge your samples into a single profile. This step will also attempt to hierarchically cluster your contigs.

In [ ]:
```
anvi-merge SP350_profile/PROFILE.db SP375_profile/PROFILE.db \
SP400_profile/PROFILE.db -o saltpond_merged \
-S saltpond_merged -c saltpond_contigs.db
```

Step 6: Binning

Binning allows us to take our metagenomic contigs and cluster them into bins, allowing us to create MAGs. You can invoke several popular binning algorithms from within anvi'o, but this is considered an experimental feature. You could also export your contigs from anvi'o, bin them as you prefer, and reimport them into anvi'o for visualization.

We'll be binning using CONCOCT, which uses differential COverage among metagenomes and base COmposition (tetranucleotide frequencies) to cluster contigs.

In [ ]:
```
anvi-cluster-contigs -p saltpond_merged/PROFILE.db \
-c saltpond_contigs.db -C concoct_bins --driver concoct --just-do-it
```

Step 7: Visualization and bin refinement

This is where we switch from anvi'o on the server to your locally-installed copy of anvi'o. Download your **contigs database (again)** and **merged profile database directory** using your preferred method (scp, Cyberduck, WinSCP, etc.). Now, on your computer, load the anvi'o visualizations.

In [ ]:
```
anvi-interactive -p saltpond_merged/PROFILE.db -c saltpond_contigs.db
```

Some things to try:

- Blast a contig ("split")
- Find a SNP and identify which gene it's in
- Load your CONCOCT bins and check out taxonomy estimates
- Evaluate bin quality using completion and redundancy
- Pick a high-completion, high-redundancy bin to refine

You can generate a static summary page of your bins through the interactive interface, or from the command line:

In [ ]:
```
anvi-summarize -p saltpond_merged/PROFILE.db \
-c saltpond_contigs.db -C concoct_bins -o concoct_bin_summary
```

Check out the .html file this command produces for a summary of your data. The other folders that are generated contain all kinds of goodies you may find useful, including fasta files of all the contigs in each bin.

Anvi'o also has an interactive bin refinement tool, `anvi-refine`. Replace `Bin_X` with the bin you chose to refine and open `anvi-refine`:

In [ ]:
```
anvi-refine -p saltpond_merged/PROFILE.db \
-c saltpond_contigs.db -C concoct_bins -b Bin_X
```

For guidelines on bin refinement, check out:

https://merenlab.org/2016/06/09/assessing-completion-and-contamination-of-MAGs/

https://merenlab.org/2015/05/11/anvi-refine/

https://merenlab.org/2017/05/11/anvi-refine-by-veronika/