# GATech Class 6720 Lab05

Author: Roth Conrad, April 01 2020.

The objective of this lab is to taxonomically identify the most abundant population in a target metagenome based on its metagenome-assembled genome (MAG) and check the quality of the recovered MAG. To do this, we will assemble an Illumina sequenced metagenomic sample from Pensacola beach sands collected during the Deep-Horizon (aka BP) oil spill. We will start with our metagenome in interleaved fasta format. The paired reads in this file have already been quality checked and trimmed. Our task will be to assemble the reads into contigs using the IDBA-UD assembler, cluster the contigs into MAGs using MaxBin2, find the most abundant MAG, and identify the nearest (named) taxonomic relative as well as the level of completeness, contamination, and quality for each MAG using the Microbial Genomes Atlas (MiGA).

A good overview of metagenomic sampling and analysis can be found [here](https://www.nature.com/articles/nbt.3935).

This lab will require at least 4000MB of base memory and around 10GB of disk space. You can change the base memory from the settings menu in the virtual box manager from the “System” tab while the VM is powered off. To ensure you have enough disk space, I recommend to create and mount a virtual harddisk drive to the VM following these instructions:

#### Add a second hard disk to your VM to increase disk space

Takes 5 min or less.

1. Make sure your VM is powered off.
2. Open the Oracle VM VirtualBox Manager
3. Select the MicrobiomeOS and click Settings
4. Navigate to the Storage tab
5. Select Controller: SATA inside the Storage Devices box on the left and click the disk drive icon with the green “+” (Not the CD icon)
6. Click Create
7. Select VHD (Virtual Hard Disk) and click continue
8. Select Fixed size and click continue
9. Select desired size (Need at least 10gb for lab 5) and click Create
10. Select the new drive we just created from beneath the Not Attached arrow and click Choose (It should be something like MicrobiomeOS\_1.vhd)
11. We have now created and added a second virtual harddrive to our machine. Click Ok.
12. Start the MicrobeOS virtual machine and let it load.
13. From the left bar at the top select search machine and search for the word “Disks”. You should find an application called Disks. Open this application.
14. In the Disks application you will see two disks and cd/dvd drive in the left column. The top disk should be the main disk and the second disk (or third etc) will be the new disk we added. Select the new disk. In the right frame at the bottom, the Device should read /dev/sdb (sdb depends on the number of disks on the system. sda is the main disk, sdb is the second, sdc is third and etc.)
15. Underneath Volumes select the gears icon and click format partition. Choose a name for the new disk such as Lab5, for Type select the Ext4 file system, and click format
16. If it asks you if you are sure, click format again. If it asks for a password, the password for the microbiomeOS is microbiome.
17. You should now see the play symbol next to the gear icon. You must now click the play play button. This will mount the partition.
18. Below the play button you will see a Contents label showing the path where your new disk is Mounted at. It should be something like /media/microbiome/Lab5. This is the new folder to install and store files for Lab5.

## Step 00: Required tools :: Conda with Python 3.6+, IDBA-UD, and MaxBin2.

### Conda with Python 3.6+

From the [mini conda installation page](https://docs.conda.io/en/latest/miniconda.html), select the appropriate installer and download it. For the VM choose Linux installers Python 3.7 (most modern computers are 64-bit) and save the file. Open a terminal window and navigate to the Miniconda installer (try the Downloads folder). Once you locate the file, you can run it like this (change the file name if yours is different):

bash Miniconda3-latest-Linux-x86\_64.sh

1. Press enter and then use the spacebar to scroll through the terms.
2. Enter yes to accept the terms.
3. We want to install miniconda3 to our Lab5 disk. Input /media/microbiome/Lab5/miniconda3 and type enter.
4. We want the installer to initialize Miniconda3 by running conda init. type yes.
5. For changes to take effect we need to close the terminal session and open a new one.
6. When you reopen the terminal you will see (base) next to the microbiome command prompt. This lets us know which conda environment we are currently in.

You can learn more about Conda environments [here](https://docs.conda.io/projects/conda/en/latest/user-guide/index.html).

*Conda is currently my preferred way to install and manage software. A quick way to check if a tool is avaible through conda is to google “conda install tool name”*

### IDBA and MaxBin2

Utilizing the Conda system makes installing programs and their dependencies much easier. For instance, looking at the [MaxBin2 README.txt file](https://sourceforge.net/projects/maxbin2/files/) you’ll notice that installation requires some prerequisites and auxiliary software packages. But, thanks to the [BioConda Project](https://bioconda.github.io/) most commonly used bioinformatics software (along with all dependencies) can be quickly installed. You’ll also notice that IDBA-UD is an auxilliary package for MaxBin2. This means that with one conda install command for MaxBin2 we get IDBA-UD as well.

# Create a conda environment for lab 5  
# The default location should be /media/microbiome/Lab5/miniconda3/envs/EnveomicsLab5  
conda create -n EnveomicsLab5  
# Activate the lab 5 environment  
conda activate EnveomicsLab5  
# install MaxBin2 with all dependies - appreciate while conda does all the work for you.  
conda install -c bioconda maxbin2

*You’ve now installed MaxBin2 and IDBA-UD. Just remember you can activate and deactivate conda environments. When you first open a terminal session you will need to activate the EnveomicsLab5 environment before you can run IDBA-UD and MaxBin2*

## Step 01: Retrieve the data

Navigate to the Lab5 disk, make a directory for the data, and download the data.

cd /media/microbiome/Lab5  
mkdir 00\_Reads\_QCed  
wget http://rothlab.com/Data/T4AerOil\_sbsmpl5.fa.gz  
gunzip T4AerOil\_sbsmpl5.fa.gz  
mv T4AerOil\_sbsmpl5.fa 00\_Reads\_QCed

## Step 02: Assemble the metagenome.

The file we just downloaded contains paired Illumina sequence reads in interleaved format. Interleaved simply means that the first read pair is always immediately followed by the second read pair on the next line. The next computational challenge here is to assemble reads together that represent the same species (metagenome assembly). We will use the IDBA-UD assembler for this task.

You can read more about genome and metagenome assembly [here](https://doi.org/10.1093/bib/bbw096) or [here](https://doi.org/10.1186/s40168-016-0154-5).

You can read more about the IDBA assembler [here](https://doi.org/10.1093/bioinformatics/bts174) and [here](https://github.com/loneknightpy/idba) or by typing idba\_ud at the command prompt in your terminal window with the EnveomicsLab5 conda environment activated.

*With 1 core and48GB of RAM allocated to the VM, assembly took 48 minutes. You need at least 8GB of ram allocated to your VM to assemble this metagenome. If you do not have enough RAM, you can skip this step and follow directions in Step 03 to download the files you need. You can change the amount of RAM your VM has access to under the “System” tab in the virtual box manager settings.*

# First type idba\_ud on its own to see all the options  
idba\_ud  
# Here is an example of one simple way to run it.  
# Change the input filenames to your file  
idba\_ud -r interleaved\_metagenome.fasta --min\_contig 1000 -o 01\_IDBA\_Assembly

*If your computer has multiple threads and you’ve configured your VM to use more than 1, look at the –num\_threads flag to reduce computation time. To increase the cores available to your VM, first power off your VM, then click settings from the Virtual Box Manager, select the System tab, then the Processor tab, choose up to half of your available CPUs then click ok and then turn your VM back on. You now have this number of threads available.*

Further reading (alternative, robust assemblers):

1. SPAdes assembler [publication](https://doi.org/10.1089/cmb.2012.0021), [website](http://cab.spbu.ru/software/spades/)
2. MegaHit publication, website

## Step 03: Cluster the assembly into bins (MAGs)

With the current state of technology, it is not typically possible to reconstruct complete genomes from the short Illumina sequenced reads based on assembly alone. What we end up with after the assembly process are hundreds or thousands of sections of contiguous sequences (contigs) representing the different species in the sample. The next computational challenge is to sort out which contigs belong to which genomes(species) and group them into population genome bins to obtain MAGs. We will use MaxBin2 for this task.

You can read more about metagenome binning [here](https://www.nature.com/articles/nbt.2579) or [here](https://doi.org/10.1186/gb-2009-10-8-r85).

You can read more about MaxBin2 [here](https://doi.org/10.1093/bioinformatics/btv638) or by typing run\_MaxBin.pl at EnveomicsLab5 environment command prompt.

*If you didn’t have enough RAM to assemble the metagenome, follow the instructions below to download the assembly.*

# Download the file  
wget wget http://rothlab.com/Data/01\_IDBA\_Assembly.tar.gz  
# uncompress the file  
tar -xzvf 01\_IDBA\_Assembly.tar.gz

Once you have the assembly, follow the directions below to bin your contigs. This step only takes about 2 minutes with a single core and doesn’t use much RAM.

# Make a directory for the output files  
mkdir 02\_MaxBin\_MAGs  
# First run run\_MaxBin.pl on its own to see all the options  
run\_MaxBin.pl  
# Here is an example of one simple way to run it  
# Change the input filenames to your file  
run\_MaxBin.pl -contig metagenome\_assembly.fasta -reads interleaved\_metagenome.fasta -out 02\_MaxBin\_MAGs/Lab5\_MAG

*If your computer has multiple threads and you’ve configured your VM to use more than 1 look at the –thread flag to reduce computation time.*

Further reading (alternative binning tools):

1. MetaBat [publication](https://peerj.com/articles/7359/), [website](https://bitbucket.org/berkeleylab/metabat/src/master/)
2. Concoct [publication](https://doi.org/10.1038/nmeth.3103), [website](https://github.com/BinPro/CONCOCT)
3. BinSanity [publication](https://peerj.com/articles/3035/), [website](https://github.com/edgraham/BinSanity)
4. DasTool [publication](https://doi.org/10.1038/s41564-018-0171-1), [website](https://github.com/cmks/DAS_Tool)
5. CAMI challenge [publication](https://doi.org/10.1038/nmeth.4458), [website](https://data.cami-challenge.org/)

## Step 04: Evaluate the recovered MAGs

Now that we have clustered our assembled contigs in MAGs, we want to learn something them. In theory, each MAG should represent a single sequence-discrete population (species) living in the environment where where the metagenomic sample was collected. In practice, the automated (or even manual) clustering process is filled with noise and uncertainty. Furthermore, we would like to know something about the taxonomic assignments for the bins we’ve recovered. We will use MiGA to evaluate some common genomic metrics and to identify the closest taxonomic assignments of our MAGs.

You can read more about MiGA and watch the video tutorials [here](http://microbial-genomes.org/). The MiGA publication is [here](https://doi.org/10.1093/nar/gky467)

You can read about how to evaluate MAGs [here](https://doi.org/10.1038/nbt.3893)

*If you had issues binning your contigs with MaxBin2, you can download the bins you need to complete this assignment with the instructions below.*

# Download the file  
wget http://rothlab.com/Data/02\_MaxBin\_MAGs.tar.gz  
# uncompress the file  
tar -xzvf 02\_MaxBin\_MAGs.tar.gz

Further reading (alternative genome assessment):

1. CheckM [publication](http://www.genome.org/cgi/doi/10.1101/gr.186072.114), [website](https://ecogenomics.github.io/CheckM/)
2. BUSCO [publication](https://doi.org/10.1093/bioinformatics/btv351), [website](https://busco.ezlab.org/)
3. QUAST [publication](https://doi.org/10.1093/bioinformatics/btt086), [website](http://cab.spbu.ru/software/quast/)
4. MetaQUAST [publication](https://doi.org/10.1093/bioinformatics/btv697), [website](http://cab.spbu.ru/software/metaquast/)
5. Anvi’o [publication](https://peerj.com/articles/1319/), [website](http://merenlab.org/software/anvio/)

## Step 05: Build recruitment plots with Enveomics RecPlot2

Recruitment plots are used to visualize the distribution of metagenomic reads to a reference genome such as a MAG. Based on this distribution it is possible to infer if a population is heterogeneous or clonal, if there is another closely related population in the metagenome, where the sequence-discrete threshold is, and if any genes or genomic regions from the reference appear to be missing in the metagenome population. We will use the [BlastTab.recplot2.R](http://enve-omics.ce.gatech.edu/enveomics/docs?t=BlastTab.recplot2.R) script from the [Enveomics collection](http://enve-omics.ce.gatech.edu/enveomics/docs) for this task.

*With 1 core, the blastn step takes about 10 minutes for each MAG, and the BlastTab.recplot2.R step takes about an hour.*

# We're going to use some custom scripts for this section.  
# Download the scripts  
wget http://rothlab.com/Data/00\_Scripts.tar.gz  
# uncompress the file  
tar -xzvf 00\_Scripts.tar.gz  
# next we need to make a blast databases (repeat for each MAG fasta file)  
makeblastdb -dbtype nucl -in MAGname.fasta  
# Make new output directory  
mkdir 03\_RecPlot  
# map metagenomic reads to the MAG and get the output in tabular blast format  
# (repeat for each MAG fasta file)  
blastn -db MAGname.fasta -query interleaved\_metagenome.fasta -out 03\_RecPlot/blastoutput\_filename.blast -outfmt '6 qseqid sseqid pident length mismatch gapopen qstart qend sstart send evalue bitscore qlen slen'  
# filter the blast output for best hits  
# remember the discussion and doing this manually for Lab 1 and Lab4?  
# let's use a script this time  
python 00\_Scripts/BlastTab\_Filter.py -h  
# run the script with default settings  
# (repeat for each MAG tabular blast file)  
python 00\_Scripts/BlastTab\_Filter.py -i blastoutput\_filename.blast  
# We are finished with our conda environment for MaxBin2. We want deactivate it to access the native R copy and Enveomics package on the VM.  
conda deactivate  
# prepare blast output for recplot2 script  
# (repeat for each MAG filtered tabular blast file)  
BlastTab.catsbj.pl MAGname.fasta filtered\_blastoutput\_filename.blst  
# need to update the enveomics.R package in your R environment for RecPlot2 script.  
conda install –c conda-forge r-fitdistrplus r-sn r-investr  
wget https://cran.r-project.org/src/contrib/Archive/enveomics.R/enveomics.R\_1.5.0.tar.gz   
sudo R CMD INSTALL ./enveomics.R\_1.5.0.tar.gz  
# The password is microbiome  
# run the recplot2 script  
# (repeat for each MAG filtered tabular blast file)  
BlastTab.recplot2.R --prefix filtered\_blastoutput\_filename recplotOutput.Rdata recplotOutput.pdf  
# You can explore the recruitment plot by viewing the PDF file.  
# with a little bit of R code we can extract some statistics from the Rdata file.  
# (repeat for each MAG Rdata file)  
rscript 00\_Scripts/Recplot2\_Summary\_Stats.R recplotOutput.Rdata

## Questions:

1. How many contigs did the assembly step produce?
2. How long is longest contig? How is contig length measured (in what unit)?
3. What is the N50 value and how is this metric defined?
4. How many bins did the MaxBin step create?
5. Which bin is the most abundant?
6. Which bin is the longest?
7. Which bin has the most contigs?
8. What is the closest taxonmic affiliation of **the most abundant** MAG?
9. Do any of your MAGs contain a 16S rRNA gene sequence?
10. Which bin has the most contamination?
11. What are the completeness and contamination estimates based upon and how reliable they are? (Tip, you may want to read the “Learn more” boxes of MiGA)
12. Which bin has the greatest G+C% content?
13. Build recruitment plots for each of your MAGs. What can you infer about the intra-population diversity based on its recruitment plot?

## Challenge Questions:

1. We mentioned that Bowtie2 is a dependency for MaxBin2 and we installed it as part of our MaxBin2 conda environement but we didn’t directly run Bowtie2. What is Bowtie2 used for? Can you find the Bowtie2 manual? How would you install Bowtie2 if the conda recipe didn’t do it for you?
2. What percentage of the Illumina reads map to your high-quality draft MAGs?