Branch: **master** ▾

Find file **Copy path**

ESPM_112L / Week_6 / **Week_6_Walkthrough.md**

jwestrob Update Week_6_Walkthrough.md

ea4a702 on Feb 27

1 contributor

Raw **Blame** **History**

117 lines (60 sloc) 7.2 KB

Welcome to week 6 of metagenomics data analysis lab!

This week we're going to be finishing up the process of binning, using automated binning software (MaxBin) to generate another set of bins, and then consolidating them using DASTool.

Outline:

- Generate scaffold2bin files for the ESOM bins you generated last week
- Use bins from Maxbin, ggKbase, and ESOM together in DASTool to generate a final set of bins
- Upload these new bins to ggKbase and further refine using taxonomy/GC/coverage tools

IMPORTANT THING TO DO BEFORE PROCEEDING:

The python install on this system is a little messy right now, so we're not going to use the system-wide python for today's lab. Instead, we're going to use one that I've installed in my own directory.

Please, if you would, enter the following commands in your system (you can just copy-paste):

```
echo "alias python=/home/jwestrob/.pyenv/shims/python" >> ~/.bashrc
```

```
source ~/.bashrc
```

This will make it so that you're using the python I've installed, not the one on the system. Any questions, feel free to ask.

A clarifying point before we begin: I'm going to start referring to the identifier for each baby as BABY_ID from now on. For example, if your baby is baby 2, your baby ID is `S3_002_000X1` . For baby 3, it's `S3_003_000X1` , and so forth. Keep this in mind.

DASTool is a program that takes the results of multiple binning methods and aggregates them in order to refine, combine, and filter the bins for quality.

In order to use this, we're going to need a `scaffold2bin` file for each of your binning methods (ggKbase, ESOM, maxbin). You have one for ggKbase already, which you used to generate your `cls` file before doing ESOMs last week. If you can't find it, go back to the week 5 tutorial to find out how to download it again!

Section 1: generating scaffold2bin files from your ESOM `.cls` files

So last week you ended by saving a file with an extension ".cls" that contains the membership information for bins you made with the ESOM software. This week you're going to use those files to generate a set of FASTA files containing the DNA corresponding to those bins.

Find your `cls` file from last week, and run the following:

```
python /class_data/cls_s2b.py [YOUR CLS FILE] [YOUR BABY ID]
```

An example command, saying for example that you are working with baby 10 (which none of you are):

```
python /class_data/cls_s2b.py my_esombins.cls S3_010_000X1
```

important: it's going to throw an error at you : `/home/jwestrob/.pyenv/versions/3.6.4/lib/python3.6/site-packages/pandas/compat/__init__.py:117: UserWarning: Could not import the lzma module. Your installed Python is incomplete. Attempting to use lzma compression will result in a RuntimeError.`

Don't worry about it!!! You're fine! Keep going!

Now you'll see a file, `[BABY_ID]_ESOMbins.scaffold2bin.tsv` , in the directory where you just ran this program. Type `ls` and you'll see it. (If you don't, call me over!)

Section 2: generating scaffold2bin files from your maxbin results

I've run Maxbin already for each of your files, since it takes a long time (and a lot of resources) to complete. Each group can find the output of Maxbin in `/class_files/[YOUR_BABY_ID]/maxbin_output` .

Take a look at these files! Go to `/class_data/[BABY_ID]/maxbin_output` and do `ls -thora` . What do you see?

- How many fasta files are there? How does this number compare to the number of bins you got out of ggKbase/ESOM?
- How big are they? These are rough indications of the length of the bacterial genomes you're binning out of your samples.

What you're going to need to do is to create a scaffold2bin file now using these results. Luckily, a program to do this is included with the DASTool software package. You can run it like so:

```
bash /home/jwestrob/DAS_Tool/src/Fasta_to_Scaffolds2Bin.sh -i /class_data/[BABY_ID]/maxbin_output/ -e fasta > [SCAFFOLDS2BIN FILE]
```

Section 3: Using DAStool to consolidate/improve your bins

Now for the really cool bit. We're going to use DAStool to integrate the results from all three binning methods into a single set of high-quality bins. Here's what you need to do:

- Get all your scaffold2bin files in a directory. Doesn't much matter what you call them, but make sure they're in the same folder.
- Choose a basename: all the output files from DASTool are going to start with this, so make sure it's something easily recognizable and distinct from the files you already have in your directory. This is specified by the `-o` option in DASTool.
- Run a simple command to pull up the help menu: `/opt/bin/bio/dastool/DAS_Tool -h` . Look over the options. You're not required to use the options that I provide to you when you're running these commands. Do some seem useful? Do some seem like a bad idea? Try some different options out! (`-create_plots` won't work for now since we don't have R installed on the cluster. That'll be fixed by next week.)

Finally, now you're ready to run DASTool. Here's an example command:

```
/opt/bin/bio/dastool/DAS_Tool -i S3_010_000X1_maxbin.scaffolds2bin.tsv,S3_010_000X1_ESOMbins.scaffold2bin.tsv,S3_010_000X1.scaffolds_to_bin.tsv -c S3_010_000X1_scaffold_min1000.fasta -o baby10_dastool_test -t 3
```

See how I have a comma-separated list containing the filenames for each of my three scaffold2bin files? You're going to need to do that. (DON'T copy the ones I put in above! They're for baby 10, which I use to test the labs before class. None of you are using baby 10.)

Make sure you're only using 3 threads! (`-t 3`) We don't have enough resources for you to do any more than that.

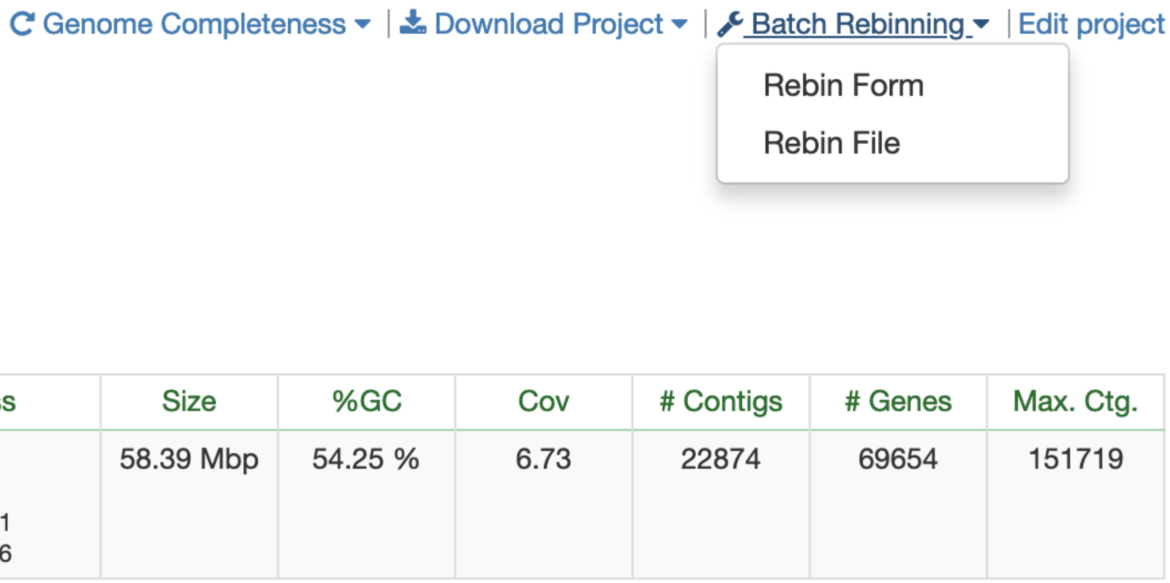
Now you've got a new scaffolds2bin file- `[basename]_DAStool_scaffolds2bin.txt` . We're gonna upload this to ggKbase and refine your bins now.

Download this `scaffolds2bin.txt` file to your computer! We'll be uploading it to ggKbase momentarily.

Section 4: Uploading the new scaffold2bin file to ggKbase

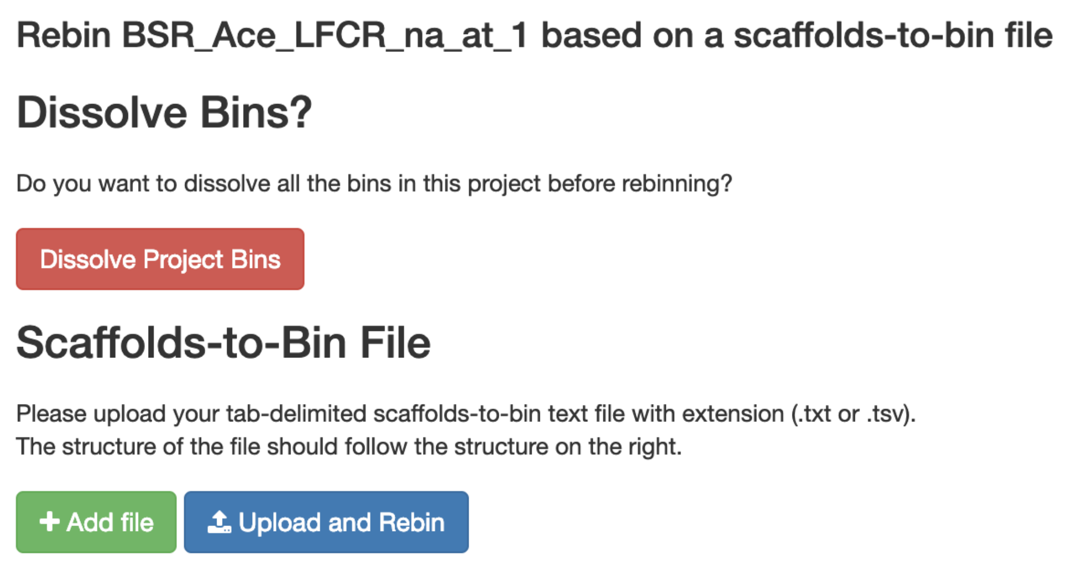
We're going to override the bins you made on ggKbase earlier with the new DASTool bins. If you want to save the bins you made earlier, you can just use the scaffold2bin file you downloaded last week, so don't fret!

Go to class.ggkbase.berkeley.edu, log in, and navigate to the project page for your baby. You're going to rebin this project now using your DASTool-consolidated bins. Navigate to 'Batch Rebinning' and click 'Rebin File', as so:



eteness	Size	%GC	Cov	# Contigs	# Genes	Max. Ctg.
C: 53 MC: 51 MC: 16	58.39 Mbp	54.25 %	6.73	22874	69654	151719

Now you're going to see a menu that looks like this:



Rebin BSR_Ace_LFCR_na_at_1 based on a scaffolds-to-bin file

Dissolve Bins?

Do you want to dissolve all the bins in this project before rebinning?

Scaffolds-to-Bin File

Please upload your tab-delimited scaffolds-to-bin text file with extension (.txt or .tsv).
The structure of the file should follow the structure on the right.

Select "Add File", and upload your `scaffolds2bin.txt` file from DASTool. Now, click "Upload and Rebin" to upload the binning information.

Now you've got your bins up

Now, go and look at these new bins - check their quality, and name them according to the taxonomy ggKbase has provided for you. If you see any mistakes (anomalous GC content/coverage/taxonomy), feel free to edit these bins!

We'll be using these bins for the rest of the semester so you want to make sure they're solid.

You did it! Feel free to explore more and look through these bins for interesting features. You can see full annotations for these genomes in ggKbase, which is super cool, so definitely take advantage of that and read through one or two!

YOU DID IT!