Checking Genome Contamination (Redundancy) and Completeness with CheckM

When you sequence a genome, and especially when you construct metagenome assembled genomes,you would like to know whether the genome is complete (i.e. it has all the genes you would expect to be there), and whether there is data from more than one organism in the sequence.

One approach you can take to explore your genome is to use GenomePeek, which will show you contamination, but will not show you completeness.

An alternative is to use CheckM that does several computations.

CheckM uses prodigal to identify the genes in your sequences,

First, it places your genome on a tree, and then it uses that phylogenetic placement to identify sets of genes that should be in your genome.

Next, it looks for those genes using hmmer to search your genome. The completeness is an estimate of the fraction of genes that are expected to be there which were actually found. The contamination is based on identifying the number of single copy genes, that should only be there once.

There are lots of different workflows that you can complete using CheckM and you should check out their manual for more commands and workflows.

The simplest workflow that we use is:

```
checkm lineage_wf GenomeBins/ CheckMOut
```

Note: If you have more than one thread on your computer you can make this run a lot faster by using them. For example,

```
checkm lineage_wf -t 16 GenomeBins/ CheckMOut
```

will run `checkm` with 16 threads.

`GenomeBins` is the name of a directory that has the genome bins that you want to analyze.

You can find a detailed description of the steps that CheckM takes in the lineage_wf on the CheckM wiki.