UniteM v1.0.0

By Donovan Parks (donovan.parks [at] gmail.com) December 14, 2021

Introduction

UniteM is a software toolkit implementing different ensemble binning strategies which combine the output produced by multiple binning methods. It implements three ensemble binning strategies. UniteM is designed for metagenome-assembled genomes (MAGs) recovered from a single assembly. This could be an assembly from a single metagenome or a co-assembly of multiple metagenomes. If you have MAGs from several different assemblies each assembly must be processed independently by UniteM.

UniteM is distributed under the GNU General Public License v3. For software updates or to report a bug please visit the UniteM GitHub page at https://github.com/dparks1134/UniteM. Support and other inquiries may be sent to Donovan Parks (donovan.parks [at] gmail.com).

Installation

The 'bin' command externally calls several popular binning methods and these must be installed independently and placed on your system path. UniteM currently supports the following programs and versions (later versions will likely work, but have not been tested):

- MaxBin v2.2.7: https://downloads.jbei.org/data/microbial_communities/MaxBin/MaxBin.html
- MetaBAT v2.12.2: https://bitbucket.org/berkeleylab/metabat
- GroopM v2: https://github.com/Ecogenomics/GroopM

MetaBAT v2 contain executables for both v1 and v2 of this binning method and both can be called by the 'bin' command provided by UniteM.

Quick Start

The functionality provided by UniteM can be accessed through the help menu:

```
> unitem -h
```

Usage information about specific commands can also be accessed through the help menu, e.g.:

```
> unitem bin –h
```

UniteM works by first (optionally) producing MAGs with a number of different binning methods using the 'bin' command, e.g.:

```
> unitem bin –c 16 --mb2 --max40 my_assembly.fna my_bins --bam_files ./mappings/*.bam
```

This command will run MetaBAT v2 (mb2) and MaxBin (--max40) using 16 CPUs, the assembled contigs in *my_assembly.fna*, and all BAM files in the *mappings* directory. Results will be placed in the directory *my_bins*. Any assembler (e.g., MEGAHIT, metaSPAdes) and mapping program (e.g., CoverM, BWA, Bowtie2) can be used to create the required assembly and mapping files.

Estimates of the completeness and contamination of MAGs must then be determined using the 'profile' command, e.g.:

```
> unitem profile –c 16 –f ./my_bins/bin_dirs.tsv my_profile
```

This command processes all MAGs in the directories specified in the file *bin_dirs.tsv*. For convenience, the 'bin' command produces this file as output for use with the 'profile' command. Results will be placed in the *my_profile* directory.

Finally, one or more of the three ensemble binning strategies can be applied, e.g.:

```
> unitem greedy –f ./my_bins/bin_dirs.tsv my_profile greedy_results
```

This command operates on the output of the 'profile' command (i.e. the *my_profile* directory) and writes results to the *greedy_results* directory. The 'consensus' and 'unanimous' ensemble strategies are run in an identical manner.

General Workflow

Ensemble binning strategies produce a refined set of bins by integrating the results of multiple binning methods (**Fig. 1**). UniteM works with the assembly of a single metagenome or a co-assembly produced by assembling multiple metagenomes together. The UniteM 'bin' command can be used to produce MAGs from multiple popular binning methods. This is an optional quality-of-life command and UniteM can be run with MAGs produced by any binning method. The UniteM 'profile' command uses the presence and absence of ubiquitous, single-copy archaeal and bacterial marker genes to evaluate the quality of the MAGs produced by each binning method. UniteM implements three ensemble binning strategies: greedy, consensus, and unanimous. The greedy and consensus strategies aim to produce large sets of high-quality MAGs, while the unanimous strategy aims to provide MAGs with minimal contamination and additional information for assessing the stability of specific MAGs across binning methods. The draft UniteM manuscript discusses these strategies in detail.

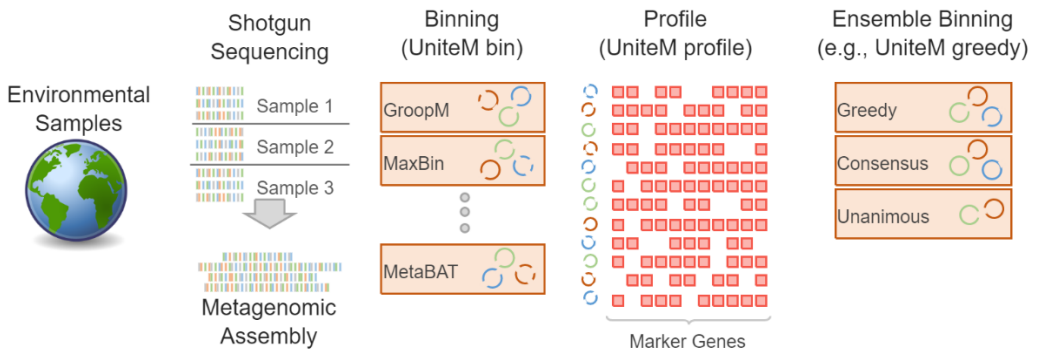

Figure 1. **General workflow for UniteM.** One or more metagenomic samples are used to create assembled contigs. These contigs are then binned together into metagenome-assembled genomes (MAGs) using multiple binning methods. The quality of recovered MAGs is assessed using the presence and absence of ubiquitous, single-copy marker genes. Finally, an ensemble binning method is used to establish a non-redundant set of MAGs using the binning information from the binning methods.

Bin Command

UniteM combines the results of different binning methods. For convenience, the 'bin' command can be used to run several popular binning methods. This command aims to reduce computation where possible (namely, by calculating a single coverage profile). This command is provided purely as a quality-of-life function and MAGs produced by any binning method can be used by UniteM (see 'profile' command). The 'bin' command is executed as follows:

```
> unitem bin --<binning_method> <assembly_file> <output_dir> --bam_files <bam_files>
```

where *binning_method* is one or more of the binning methods given below, *assembly_file* is a FASTA file of assembled contigs, *output_dir* is the desired output directory, and *bam_files* is one or more BAM files to be used to determine coverage information used by the binning methods. For example,

```
> unitem bin --gm2 --mb2 my_assembly.fna my_bins --bam_files sample1.bam sample2.bam
```

Or, more concisely:

```
> unitem bin --gm2 --mb2 my_assembly.fna my_bins --bam_files sample*.bam
```

The 'bin' command places the MAGs produced by each binning method in their own directory and creates two additional output files:

- *bin_dirs.tsv*: file indicating the path to the MAGs produced by each method. This file is created for convenience as it can be directly used as input to the 'profile' and ensemble binning commands.
- *coverage.tsv*: coverage information for each contig across the provided BAM files. This file can be manually inspected to determine the coverage of specific contigs and provided to the *cov_file* argument of the 'bin' command in order to skip coverage calculations should this command need to be run again.

In addition, the output produced by each binning method is written to the *output.log* file contained in each binning methods output directory.

One or more of the following binning methods can be specified:

Argument	Binning Method	Reference
--gm2	GroopM v2	Imelfort et al., 2014
--max40	MaxBin v2 with the 40 marker gene set	Wu et al., 2016
--max107	MaxBin v2 with the 107 marker gene set	Wu et al., 2016
--mb2	MetaBAT v2	Kang et al., 2015
--mb_verysensitive	MetaBAT v1 with the verysensitive preset settings	Kang et al., 2015
--mb_sensitive	MetaBAT v1 with the sensitive preset settings	Kang et al., 2015
--mb_specific	MetaBAT v1 with the specific preset settings	Kang et al., 2015
--mb_veryspecific	MetaBAT v1 with the veryspecific preset settings	Kang et al., 2015
--mb_superspecific	MetaBAT v1 with the superspecific preset settings	Kang et al., 2015

The 'bin' command also accepts the following optional arguments:

Optional Argument	Operation
-m, --min_contig_len	Minimum length of contig to bin (default: 2500)
-c, --cpus	Maximum number of CPUs to use during binning

Coverage information can be calculated from BAM files or by explicitly providing a file with this information:

Required Arguments (one of)	Operation
--bam_files	BAM file(s) to parse for coverage profile
--cov_file	File indicating coverage information

The provided coverage file must have the format produced by the *jgi_summarize_bam_contig_depths* script provided with MetaBAT or by running CoverM to estimate contig coverage using the *metabat* coverage method. This format is as follows and must have the specified headers:

```
contigName<\t>contigLen<\t>totalAvgDepth<\t><sample1>.bam<\t><sample1>.bam-var …
```

where <\t> indicates a tab character. Below are the first 4 lines of a valid coverage file with two samples (cck10 and arg13). Any number of samples can be specified.

```
contigName    contigLen   totalAvgDepth   cck10.bam   cck10.bam-var   arg13.bam    arg13.bam-var
contig1       468121      435.27          78.50       2362.7          356.7        52616.3
contig2       398118      27.714          0.69        69.2            27.0         142.153
contig3       373078      506.378         88.5431     2533.29         417.8        60828.1
```

Profile Command

The UniteM 'profile' command uses Prodigal (Hyatt et al., 2010) and HMMER (Eddy, 2011) to identify domain-specific marker genes across all MAGs (see *UniteM Marker Genes*). The presence and absence of these marker genes are used to estimate the completeness and contamination of each MAG and assembly statistics used as a secondary indication of MAG quality. This information is used by the different ensemble binning strategies in UniteM to determine how MAGs should be selected and refined.

In order to profile MAGs, you need to specify the directory containing the MAGs produced by each binning method. This can be done either directly on the command line or using a file. For example, assume you have the following directory structure:

```
./binning_methods/method1

./binning_methods/method2

./binning_methods/method3
```

This layout can be specified in a tab-separated file (<\t> is a tab) as follows:

```
first_method<\t>./binning_methods/method1

second_method<\t>./binning_methods/method2

third_method<\t>./binning_methods/method3
```

The first column is a descriptive label for the binning method and the second column is the path to the MAGs produced by this method. MAGs must be in FASTA format and end with the extension '.fa', '.fna', or '.fasta'. All MAGs from a specific binning method must have the same extension.

Profiling can be performed using:

```
> unitem profile –c 16 -f bin_dirs.tsv <output_dir>
```

where *bin_dirs.tsv* is the tab-separated file described above, profiling will be performed with 16 CPUs, and the results written to the directory specified by *output_dir*. Profiling is the most computationally intensive step of UniteM and will run more quickly with additional CPUs.

Alternatively, the bin directories can be specified on the command line:

```
> unitem profile -c 16 <output_dir> -b ./binning_methods/method1
./binning_methods/method2 ./binning_methods/method3
```

or, simply:

```
> unitem profile -c 16 <output_dir> -b ./binning_methods/method*
```

The bin directory (*-b*) argument must be the last one specified on the command line.

The 'profile' command has the following arguments:

Positional Arguments	Operation
output_dir	Specifies desired output directory

Required Arguments (one of)	Operation
-b, --bin_dirs	Directories with MAGs from different binning methods
-f, --bin_file	File indicating directories with MAGs from different binning methods

Optional Argument	Operation
-c, --cpus	Maximum number of CPUs to use during binning

Greedy, Consensus, and Unanimous Commands

The UniteM ensemble binning strategies create a non-redundant set of MAGs across multiple binning methods. Details on these different strategies can be found in the draft UniteM manuscript, but briefly the strategies work as follows:

- *greedy*: iteratively selects the highest-quality MAG across all binning methods. Contigs in the selected MAG are then removed from all remaining MAGs and their quality recalculated. This process is repeated until there are no remaining MAGs above a specified quality threshold.
- *consensus*: the consensus strategy aims to improve upon the 'greedy' selection process. It works by identifying 'matched sets' of MAGs that represent the same microbial population produced by different binning methods. Contigs across the MAGs in a matched set are then examined in order to produce a single refined MAG. Similar to the greedy strategy, this strategy proceeds in an iterative fashion selecting the highest-quality matched set at each iteration, removing contigs from the produced MAG for all remaining MAGs, and then recalculating the quality of the remaining MAGs.
- *unanimous*: this is a conservative strategy that identifies subsets of contigs that are consistently binned together across different methods. The goal is to produce MAGs with minimal contamination as they are comprised of contigs that are stable across binning methods. These MAGs can be viewed with more confidence than those provided by any individual binning method and thus strengthen arguments regarding novel findings. The strategy works in a similar manner to the consensus method, except a contig is only retained if it is present in all MAGs comprising the highest-quality matched set.

All three commands operate in a similar fashion:

```
> unitem greedy -f bin_dirs.tsv <profile_dir> <output_dir>
> unitem consensus -f bin_dirs.tsv <profile_dir> <output_dir>
> unitem unanimous -f bin_dirs.tsv <profile_dir> <output_dir>
```

where *profile_dir* is the output directory of the 'profile' command and *output_dir* will contain the output files produced by the ensemble binning strategy. Identical to the 'profile' command the directories containing the MAGs from each binning method can either be specified in a file (*-f* argument) or on the command line (*-b* argument).

MAGs produced by the ensemble binning commands are placed in the 'bin' subdirectory. Four addition summary output files are also created:

- *bin_info.tsv*: provides information about each of the MAGs produced by the ensemble binning strategy. This includes estimates of completeness and contamination (though, see the note below about obtaining more accurate quality estimates) and genome statistics such as size, number of contigs, and N50. For the consensus and unanimous strategies, the number of contigs added or removed by the refinement process of these methods is also indicated.
- *bin_quality_summary.tsv*: provides summary information about the estimated quality of MAGs produced by the ensemble binning strategy. The number of MAGs meeting specific quality criteria (e.g., ≥90% complete, ≤5% contamination) is given. This is provided in order to give an overview of the produced MAGs and allow an initial comparison between different ensemble binning strategies and/or parameter settings. The results in this file should not be reported as these quality estimates can be improved (see the note below about obtaining more accurate genome quality estimates).
- *matched_set_info.tsv*: provides information about each of the 'matched sets' used by the consensus and unanimous strategies to refine MAGs.
- *contig_info_initial*.tsv: provides information about individual contigs as they appear across the different binning methods before the ensemble binning strategy is applied. For each binning method, the MAG containing the contig is indicated along with its estimated completeness and contamination. Contigs not contained in a MAG are marked as *unbinned*.
- *contig_info.tsv*: provides information about individual contigs binned by the ensemble strategy including which ensemble MAG they are contained in and the state of this contig across the different binning methods. For each binning method, a contig is classified as either:
    - *matched:* indicating it is in the 'matched set' used to produce the ensemble MAG
    - *unmatched:* indicating it is in a MAG not within the 'matched set' used by the ensemble binning strategy
    - *degenerate:* indicating it is in a MAG produced by the binning method of sufficiently poor quality that it was not considered by the ensemble binning strategy
    - *unbinned:* indicating it is not contained in a MAG produced by the binning method

  The greedy method does not calculate 'matched sets' so no methods are reported as *matched*. For each binning method, the estimated completeness and contamination of the MAG is also provided. This quality estimate is for the state of the MAG as is appears during the point in the ensemble binning process at which it is being considered. This is an important distinction as MAGs can be modified during the ensemble binning process so these estimates may not reflect the quality of the MAG produced by the binning method.

UniteM decorates the header lines of the FASTA files produced for each bin with information about the distribution of a contig across the different binning methods. Specifically, it provides the number of binning methods that are *matched*, *unmatched*, *degenerate*, or *unbinned* (see *contig_info.tsv* file information above). The header line also indicates a contigs as 'added by consensus' if the consensus strategy has added it to a bin based on the consensus criteria.

Note on Genome Quality Estimates: UniteM estimates MAG quality using ubiquitous, single copy archaeal and bacterial marker genes (see *UniteM Marker Genes*). These are suitable for guiding the ensemble binning process, but more accurate estimates can be obtained using lineage-specific marker sets. We recommend estimating the quality of MAGs produced by UniteM with the lineage-specific marker set workflow provided by CheckM (Parks et al., 2015).

The ensemble binning commands accept the following arguments:

Positional Arguments	**Operation**
profile_dir	Directory with MAG profiles created by the 'profile' command
output_dir	Desired output directory

Required Arguments (one of)	**Operation**
-b, --bin_dirs	Directories with MAGs from different binning methods
-f, --bin_file	File indicating directories with MAGs from different binning methods

Optional Argument	**Operation**
-w, --weight	Weight given to contamination for assessing MAG quality (default: 2)
-q, --sel_min_quality	Minimum quality of MAG to consider during MAG selection process (default: 50)
-x, --sel_min_comp	Minimum completeness of MAG to consider during MAG selection process (default: 50)
-y, --sel_max_cont	Maximum contamination of MAG to consider during MAG selection process (default: 10)
--report_min_quality	Minimum quality of MAGs to report (default: 10)
--simple_headers	Do not added additional information to FASTA headers.
-p, --bin_prefix	Prefix to append to produced MAGs (default: bin)

Optional Argument (consensus only)	**Operation**
-r, --remove_perc	Minimum percentage of MAGs required to remove contigs from highest-quality MAG (default: 50)
-a, --add_perc	Minimum percentage of matched MAGs required to add contigs to highest-quality MAG (default: 50)
-m, --add_matches	Minimum number of matched MAGs required to add contigs (default: 3)

UniteM Marker Genes

UniteM estimates the completeness and contamination of MAGs using archaeal and bacterial genes identified as being ubiquitous and single copy. These genes were identified by considering all GTDB (Parks et al., 2021) species representative genomes with an estimated completeness ≥95% and contamination ≤5% as determined using CheckM v2 (*in development*). Genes from the set of 120 bacterial and 122 archaeal marker genes used to infer the GTDB reference trees were considered and any gene identified as single-copy in ≥95% of the quality filtered GTDB representative genomes retained as a potential marker gene. This set of marker genes was then further filtered to remove genes that were not predominately single copy in the majority of phyla. Specifically, a gene was removed if it was not single copy in ≥75% of the genomes in ≥75% of phyla.

For GTDB R202, there were 26,287 bacterial and 836 archaeal representative genomes with an estimated completeness ≥95% and contamination ≤5%. A total of 86 bacterial and 85 archaeal genes were identified as being single-copy in ≥95% of these genomes. A single bacterial gene and 3 archaeal genes were further filtered as they were not found in ≥75% of the genomes in ≥75% of phyla. Filtering of archaeal genes was found to be sensitive to this latter filtering since there were only 12 archaeal phyla with genomes passing QC and 7 of these phyla contain <10 genomes.

A table indicating the phylum-level distribution of these genes is given in the *markers* directory. As expected, some phyla (e.g. Patescibacteria and Nanoarchaeota) are systematically missing some of these otherwise ubiquitous, single-copy genes. While this results in reduced completeness estimates, this bias is systematic for all binning methods so should not unduly compromise the ensemble binning strategies.

Reference

Eddy SR. 2011. Accelerated profile HMM searches. *PLoS Comput Biol* **7**, e1002195.

Graham ED, Heidelberg JF, Tully BJ. 2017. BinSanity: unsupervised clustering of environmental microbial assemblies using coverage and affinity propagation. *PeerJ* **5**, doi: 10.7717/peerj.3035.

Kang DD, Froula J, Egan E, Wang Z. 2015. MetaBAT, an efficient tool for accurately reconstructing single genomes from complex microbial communities. *PeerJ* **3**, https://dx.doi.org/10.7717/peerj.1165.

Hyatt D, et al. 2010. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics* **11**, https://doi.org/10.1186/1471-2105-11-119.

Imelfort M, Parks DH, Woodcroft BJ, Dennis P, Hugenholtz P, Tyson GW. 2014. GroopM: an automated tool for the recovery of population genomes from related metagenomes. *PeerJ* **2**, http://dx.doi.org/10.7717/peerj.603.

Parks DH, Imelfort M, Skennerton CT, Hugenholtz P, Tyson GW. 2015. CheckM: assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. *Genome Res* **25**, 1043-1055.

Parks DH, et al. 2018. GTDB: an ongoing census of bacterial and archaeal diversity through a phylogenetically consistent, rank normalized and complete genome-based taxonomy. *Nucleic Acids Res*, https://doi.org/10.1093/nar/gkab776.

Wu YW, Simmons BA, Singer SW. 2016. MaxBin 2.0: an automated binning algorithm to recover genomes from multiple metagenomic datasets. *Bioinformatics* **32**, 605-607.