Using GAST – Global Assignment of Sequence Taxonomy for SSU rRNA

The GAST algorithm assigns taxonomy to SSU sequences by comparing them to a set of reference sequences with known taxonomy. In the simplest terms, taxonomy is assigned to each query sequence based on the consensus of the nearest references sequences. The alignment distance between the query and references is also reported as a measure of how similar the query and reference sequences really are.

Installation Instructions

GAST is a perl script for linux that calls upon one other perl module, Taxonomy.pm, and two other public domain bioinformatics packages (USEARCH and mothur). It can be run through a terminal window on the Macintosh, but has not been adapted for Windows. You will need to install mothur and usearch prior to running GAST.

To install the GAST perl code, simply copy the gast file into an appropriate directory in your path. The Taxonomy.pm perl module should be either copied into the same directory as the gast file, or if you have a specific location on your system for perl modules you can place it there.

If the command usearch is not already in your path, then you can either update your path variables to include it, or edit the gast script to provide the explicit path. The gast perl script is a text file, simply open the file in a text editor, find the first instance of usearch (my $usearch_cmd = "usearch";), and update with the complete path.

GAST compares sequences in your data to a set of reference sequences. Several sets of reference sequences are available on VAMPS. The sequences include two files, a fasta file of the reference sequences themselves and the list of all taxonomic assignments to those sequences. The definition line of the fasta file consists of a unique identifier. The taxonomy file is a tab-delimited file with the first field is the unique identifier from the fasta file, the second field is the full taxonomic string, using semicolons to separate ranks, and then third column is the number of instances of that sequence with that taxonomy. If a sequence appears in the reference database more than once and has different taxonomic assignments, then it will have more than one line in the taxonomy file, but only one fasta entry.

Copy the appropriate fasta and taxonomy files for the section of SSU rRNA gene you are comparing against, or you can create your own file pairs.

How GAST actually works

The essence of GAST is simply to check each sequence in your sample data against a reference set of known taxa, and then assign taxonomy based on the consensus of the nearest reference sequences.

First, the input data are dereplicated – exact copies of the same sequence will have the same matches and therefore be given the same taxonomic assignment. It is obviously more efficient to run on unique copies of each sequence. The input fasta file is

dereplicated into a unique fasta and a mothur-formatted names file, which includes information on all copies of each unique sequence.

The dereplicted sequences are run through USearch to find the nearest sequence or sequences in the reference fasta file. USearch outputs a *.uc file containing the list of hits. Each hit corresponding to the minimum distance between the query and any reference sequence is then looked up in the reference taxonomy table and the consensus taxonomy calculated. If there is more than one taxonomic name in the reference set corresponding to the nearest reference sequences, then GAST performs a comparison. Starting at the lowest taxonomic rank found (usually genus or species), if the majority agree (default requires 66% for a majority), then the taxonomic name is given at that level. If, however, a majority is not found, either because assignments disagree or because not enough references were assigned a taxonomic name to that level (i.e., only a few were assigned to genus, the rest only down to family), gast goes up one level and seeks consensus at that level. This continues until a consensus is found, or if no consensus is achieved at the domain level, then the taxonomic name "Unknown" is assigned.

Running GAST

GAST is simple to run, using a commandline syntax. Typing "gast" alone on the command line will return a complete usage statement.

In its simplest usage:
gast -in input_fasta -ref ref_fasta -rtax ref_taxonomy -out output_file

- -in your sequence data in fasta file format
- -ref the fasta file of the reference data as described above
- -rtax the tab-delimited text file of the taxonomy of the reference sequences, as described above.
- -out the filename for your results

GAST includes other options as well:

- -host database host name
- -db database name
- -table if you have a database table set up in advance, the data can be uploaded directly into the database
- -minp minimum percent identity. If the percent identity between the data sequence and the nearest reference sequence is less than minp, then do not count it as a match at all. The default value is 0.70 (30% difference).
- -maxa usearch –max_accepts value. The default value used by gast is 15.
- -maxr usearch –max_rejects value. The default value used by gast is 0
- -maj percent majority required for taxonomic consensus. The default value is 66, which means that at least two thirds of the taxonomic assignments hit have to agree for the final assignment to be made at that level. If a majority of reference taxa do not agree at that taxonomic level, the assignment goes up one level (e.g., from genus to family) and tries again until a consensus is reached.

GAST output file format

The output file is a simple tab-delimited text file with a single line for each input fasta sequence. The data fields are described below. If run with the –terse option, only the read_id, taxonomy, distance and rank are included.

read_id	identifier for the input sequence taken from the fasta file
taxonomy	taxonomic assignment made. The taxonomy is always in the order of Domain;Phylum;Class;Order;Family;Genus;Species;subspecies. No intermediate ranks are skipped or inserted, but any unassigned ranks at the bottom are not reported (e.g., if the assignment is to family, then the taxonomic string ends at family).
distance	distance from the input sequence to the nearest sequence in the reference database, expressed as a decimal (e.g., 0.0100 which corresponds to a 1% difference)
rank	taxonomic rank of the assignment (e.g., family, genus, species)
refssu_count	the number of reference sequences used for the assignment
vote	percentage of reference sequences in agreement at the taxonomic rank assigned.
minrank	the lowest taxonomic rank of the reference sequences hit
taxa_counts	at each taxonomic level, how many different taxa were found in the reference sequences. For example 1;1;1;1;1;1;4;0 means that at the domain, phylum, class, order, family and genus levels all reference sequences agreed to the same single taxon. At the species level, there were four different species in the reference sequences.
max_pcts	at each taxonomic level, what was the percentage of the most common taxon. For example 100;100;100;100;100;100;19;0, means that at the domain, phylum, class, order, family and genus levels the most common taxonomic name was represented by 100% of the references. But at the species level the most common name was represented by only 19% of the reference sequences. No taxon represented a 66% majority.
na_pcts	at each taxonomic level, what percentage of the reference sequences had no taxonomic assignment at all (i.e., taxonomy assigned as “NA”). For example, 0;0;0;0;0;0;79;100 explains why the most common of the four species assignments was less than 25% -- because 79% of the references did not have an assignment at the species level at all. And none of them had an assignment below species.
ref_ids	a list of all the reference sequences used for the taxonomic assignment.

Reference

Huse SM, Dethlefsen L, Huber JA, Mark Welch D, Relman DA, et al. (2008) Exploring Microbial Diversity and Taxonomy Using SSU rRNA Hypervariable Tag Sequencing. PLoS Genetics 4: e1000255 [http://www.plosgenetics.org/article/info%3Adoi%2F10.1371%2Fjournal.pgen.1000255]