CGPFinder. A comparative genomics approach to gene clusters

Release structure

Python scripts

The software package is organized as a python3 run script (`cgp_exec.py`), which runs the homology search (via blast), process coordinates information, and finds clusters of gene paralogs (CGP). The functions needed for the computations are organized as a python module under the directory cgpFinder/
In addition to the running script and its library, CGPFinder expects the following files and directories under a database directory (a sample "mini" database is included).
* **all_seqs.fa**, **all_seqs.fa.phr**, **all_seqs.pin**, and **all_seqs.fa.psq**: Blast database used to find potential paralogs.
* **blasts/*.json**: The blasts directory contains the results of an all vs. all blast search. Each of the JSON files contain all the queries from each species and the corresponding blast results.
* **chromosomes.csv**: Table with chromosome annotations (species, chromosome name, taxID, GI, accession, assembly, length).
* **genes_parsed.csv**: Table with gene annotation for all species (accession, start coordinate, end coordinate, strand, chromosome, species, gene symbol, name, length).
IMPORTANT: If a custom database is going to be used, every single annotation table, blast database and sequence file (all_seqs.fa) should have identical accession numbers, chromosome locations, etc. In addition, the json-parsed all vs. all blast and the blast database should be computed from `all_seqs.fa`, and `all_seqs.fa` should have a header for each sequence of the form:
>species_name|chromsome|accession|symbol|start|end|strand. Refer to `preprocessing.pdf` for a description of the procedure by which all these files were generated

Installation

Before installing CGPFinder, the following python modules should be installed:
* numpy
* pandas
* numexpr * scipy
* scikit-learn
* termcolor
* biopython
* Aditionally, blastp (from the NCBI Blast suite) should be installed (Version 2.2.30+ was used in release tests and in the publication)

In order to install the CGPFinder libraries, the directory containing the subdirectory cgpFinder, as well as the directory cgpFinder itself should be included in PYTHONPATH. One way to do it is to simply link (using ln -s in linux) the cgpFinder directory to the directory where the python interpreter usually looks for modules.
If the python interpreter is located in /usr/bin/python, python should be looking for modules in /usr/lib/python_version/site-packages. In general, if the python interpreter is installed in /path/path/bin/python_version, the modules "should" be under /path/path/lib/python_version/site-packages. Using the last example path, the following bash commands should make the CGPFinder module usable:

$ cd /path/path/lib/python_version/ (this might need sudo priviliges if the modules are not installed under the user directory tree)
$ ln -s /path/to/cgpFinder
A bash script `path.sh` is also included. Running it on linux (bash) will include the necessary PYTHONPATH paths.

Running instructions

CGPFinder is run as a python script (`python cgp_exec.py`)
The following are the minimal options that should be specified

- --(n)ame_family: name of the gene family
- --(r)ef_seq: path to the fasta file used as query

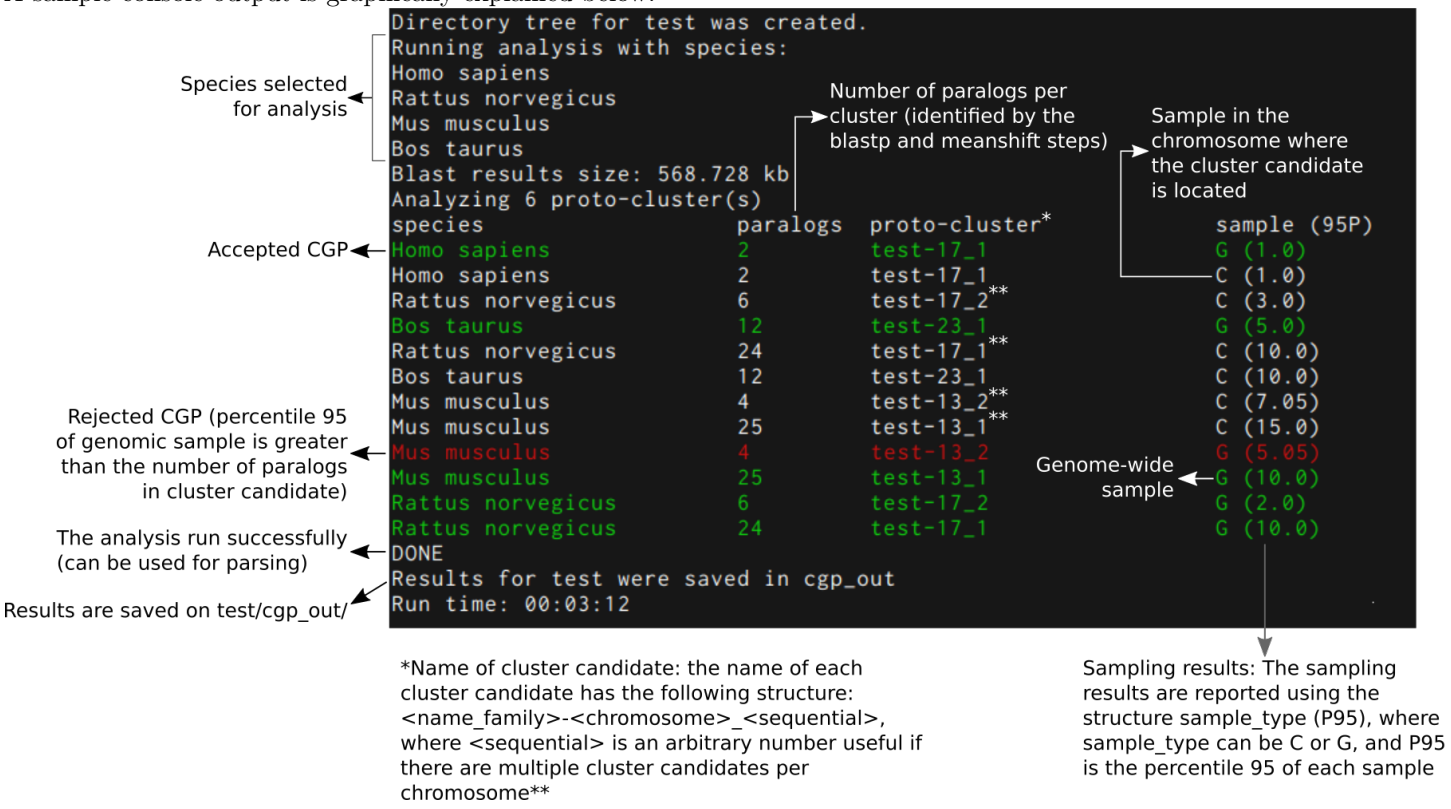
- --(b)last_samples: number of samples to build the empirical distribution of paralogs.
- --(d)b: database: directory whith the required blast database and annotation tables (all_seqs.fa, genes_parsed.csv, etc) Importantly, the python script should be run from the installation directory. A minimal run will look somethin like this: $ python cgp_exec.py --ref-seq path/to/reference.fa --name_family gene_family_name --blast_samples 1000 –db mini.

Optional parameters

- --(s)p: If specified, the analysis will be performed only with the specified species. Note: the species name should be typed in qotes ("Homo sapiens"). If more than one species is going to be specified, an independent –sp option should be used (e.g. --sp "Homo sapiens" --sp "Mus musculus"). If not specified, the analysis wll be run with all the available 23 species.
- --(o)ut_dir: directory where the results of the run are going to be stored (default: cgp_out/).
- --(c)pu: Number of cores used (default: 1 core).

Output

A sample console output is graphically explained below:



Under the cgp_out directory (or the directory specified by --out_dir), a directory for each gene family (specified by --name_family) (name_family) will be created. This is useful in case several analyses are going to be performed. In case an analysis is going to be run for a gene family that already exists in the output direcrory, CGPFinder will terminate.
Under the gene family directory, the results from the run are found in the "report" directory. The "report" directory contain the following files:
* `name_family.blast`: raw blast result using the reference sequence(s) (specified by –ref_seq) and the blast database. * `name_family.blast_filtered`, `name_family.blast_out`: parsed blast outputs in csv format. `name_family.blast_filtered` is a blast result, in which only subjects with more than 30% of sequence length overlap with their queries. `name_family.blast_out` contains the same results as `name_family.blast_filtered`, but it is enriched with genomic annotation. This is the file that is going to be used to define the cluster candidates * `blast_samples.tgz`: compressed tar.gz file that contains the sampled coordinates for each species and cluster candidate. Under the name_family directory, genome-wide (blast_samples_gw) and in-chromosome (blast_samples) samples set are contained. In each of the sample sets, .coords files containing the species name, chromosome name, and start and stop coordinates describe the sample coordinates used in the analysis * `name_family_genes.csv`: genes found in the analysis. Most of the columns are self-explanatory. The "cluster" column contains the name of the cluster where the gene was found, and the "order" column describes the

order (according to the genomic coordinates) of the gene in the CGP. If the "cluster" and "order" columns have 0 (zero), the gene was found as single copy in that chromosome. If the value of "cluster" and "order" is "na", the gene was found in a chromosome with cluster candidates, but were discarded by either the meanshift or the statistical sampling steps. * `name_family_numbers.csv`: Coordinates of the cluster candidates. The values of the "cluster" column follow the convention for the name_family_genes.csv file. The column "perc95_chrom" and "perc95_gw" columns describe the 95th percentile of the paralogs empirical distribution that was sampled in the analysis. * `name_family_numbers_clean.csv`: A version of the `name_family_numbers.csv` file witout rejected clusters ("na" value in the "cluster" column). * `name_family.samples`: Table that describes the maximum number of paralogs in each cluster candidate. These are the distributions used to calculate the above mentioned percentiles.