**Understanding Core Global Families**

In order to assess genome quality, it is useful to compare your genome to a reference set of proteins that you would expect to find within your genome. Some programs do this by comparing to a set of universally conserved proteins. In this instance, we have chosen to assess genome quality by comparing a genome to a core set of PATRIC global families (PGFs). The overall objective for computing core PGFs is to find a set of protein families that are held in common across all members of a genus. This is advantageous because it is fast and allows us to potentially assess a broader swath of proteins than we would be able to assess by just looking at universal proteins.

In practice finding core PGFs that span all members of a genus can be challenging for a variety of reasons. Here are a few:

1. Each genus differs in how much evolutionary divergence exists between its members. For instance, there is little evolutionary difference between members of the Escherichia, but the distance between members of the Mycoplasma (in terms of 16S percent identity) can be as large as the threshold used for declaring different phyla. This means some taxa of Mycoplasma will have different PGFs, for core proteins, so finding a spanning set is requires adding flexible thresholds.
2. Each genus has a different number of sequenced genomes for the species. For instance, “Mycobacterium” is currently 11,708 genomes, 80% of which are tuberculosis (9,373 genomes), 3% are bovis (344 genomes) and 0.1% are leprae (13 genomes). This makes finding a set of PGFs that is held in common across all members can be challenging and the final set may not evenly represent all members.
3. The MCL algorithm will make splits in PGFs that are not based on percent identity and can be difficult to predict.

These considerations have influenced the design of the core PGF algorithm. There are also several other points of consideration. First, we don’t worry about whether the family is found in other genera; that is, they are not genus specific. Second, we do not worry about whether a core family contains duplicates. The software reports duplicates core PGFs found within a genome (see below), but these need to be considered in the greater context of what should be expected for that genus. Finally, we only compute core global families for genera with more than 4 members.

**Core PGF Algorithm**

We start by reading Bob’s most recent file of pattyfams (/vol/patric3/fams/2017-0701/merge1/prop.out/merged.families.1.1). Before the algorithm is run, we compute the average and standard deviation of the number of proteins in each genome for the members of the genus. If a genome in the genus differs from the mean by 2 standard deviations, it is thrown out. At this point, if the genus still contains at least four members we run the algorithm (Figure 2).

The algorithm attempts to gather a set of core PGFs that is roughly ≥ 10% of the average genome size for the members of that genus, and for which the family is found in as many members of the genus as possible. For any PGF, if the fraction of genomes in the genus having that PGF drops below 0.75, it cannot be considered to be a core family.

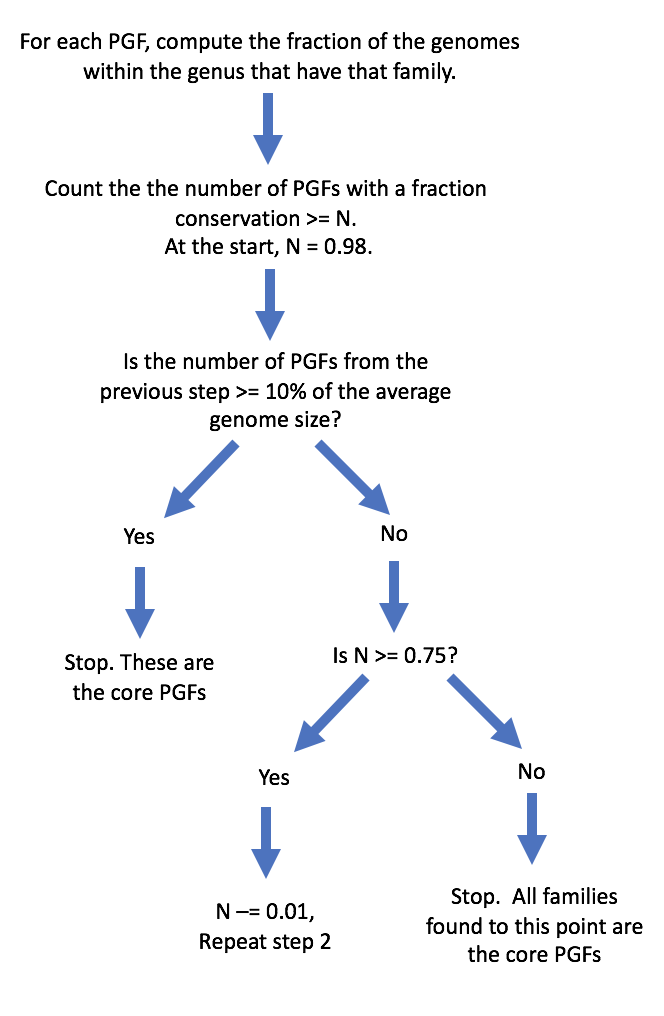


Figure 2. Core PGF algorithm.

The program *“get\_core\_PGFs.pl”* generates a directory that contains a file of Core PGFs for each genome. For each genus, the output the output file contains 5-columns, which are:

1. PGF
2. Genus
3. Average gene length for the proteins in the PGF
4. Standard deviation of lengths for the proteins in the PGF
5. Annotation

The *“get\_core\_PGFs.pl* program has a help menu that can be accessed with the –h flag.

**Genome quality code**

For a new genome, genome quality statistics are generated using the program “*core\_PGF\_genome\_quality.pl*”. The program takes as input, the genus, a fasta file of proteins, for the genome, and the directory of core PGFs generated by “*get\_core\_PGFs.pl”*. It looks up the PGFs using “*place\_proteins\_into\_pattyfams”* and returns an 8-column tab-delimited file to standard output. Columns are:

1. The total number of core PGFs for the genus
2. The total number of core PGFs found in your genome
3. The fraction of core PGFs found in your genome
4. The fraction of core PGFs that are duplicated in your genome
5. The fraction of core PGFs that were missing in your genome
6. The Fraction of core PGFs with lengths that are too short, P-value < 0.01
7. The Fraction of core PGFs with lengths that are too long, P-value < 0.01
8. The Ratio of all PGFs found in your genome to the total number of proteins in your genome (This will tell you how many novel or weird proteins you had. It will be high in bad genomes and in genomes that are less common in PATRIC)

The program also returns a file enumerating the proteins that are duplicated, too short, too long or the core PGFs that are missing. More detailed information can be found using the –h option on the program.