QC-Datasets: Normalizing the process of filtering NCBI derived Genomes

Alan Shteyman[1], Colin Price[1], Phillip Davis[1],Brian Clark[2] & Joseph A. Russell[1]

1. *MRIGlobal, 65 West Watkins Mill Road, Gaithersburg, MD 20878*
2. *Formerly at MRIGlobal, 65 West Watkins Mill Road, Gaithersburg, MD 20878*

PIP-MON-141

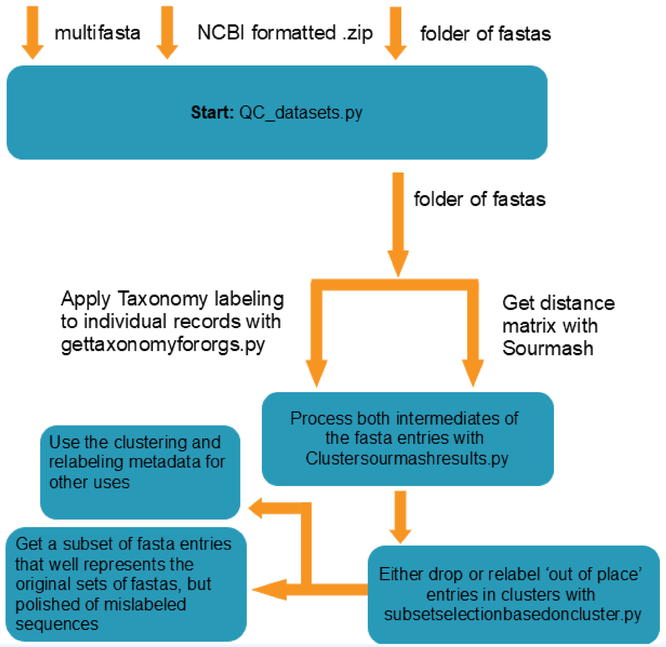
Abstract

The National Center for Biotechnology Information (NCBI) has provided a central hub for various omics datasets for decades, offering invaluable, widely-used databases[1]. These resources have facilitated numerous research projects by enabling easy access to quality-controlled omics data. However, with the exponential increase in data submissions, maintaining the quality of these datasets has become increasingly challenging. This issue is compounded by the introduction of data from diverse sources, including corporate research outputs, synthetically derived sequences, and other non-traditional submissions, which can vary significantly in quality.

To address these challenges, our group has developed a suite of scripts designed to enhance the quality control of genome sequences from the NCBI's nucleotide database. Our workflow can process multifasta files, directories containing individual fasta files, or zipped directories provided by NCBI. The initial script filters genomes based on the presence of degenerate nucleotide characters, significant deviations in genome length compared to typical entries, outlier GC content, or specific metadata keywords. Successful entries are then analyzed using the sourmash[2] tool to calculate distances for clustering. Simultaneously, each sequence is assigned a taxonomic ID (taxid) at a specified level using the MRItaxonomy package for Python. This facilitates further refinement based on taxonomic information, merged with the sourmash distances, to produce hierarchical clusters. The final script clusters genomes, excluding or relabeling outliers within taxid-majority clusters based on a predefined distance threshold. The resulting high-quality genome sets are then randomly subsampled without replacement, but accounting for variability, ensuring a representative subset selected for each taxid.

The tool composed of these constituent scripts allows researchers to rapidly and efficiently refine their datasets, minimizing time and effort while maximizing usability for various applications. The goal is to provide end-users with a robust method for quickly enhancing the quality of genomes they download from NCBI and integrate into their bioinformatics analyses.

QC-datasets Workflow



Detailed Workflow Overview

QC-datasets.py:

- Accepts a multifasta, a folder of fastas or a zipped folder of fasta files formatted, in the same way the NCBI datasets[3] command line tool will provide collections of records in it
- Can interconvert between all three forms
- Can filter out entries that have:
  - a specified amount of degenerate characters
  - unexpectedly extreme GC content
  - are non-refseq
  - have substrings to be filtered out
  - do not have substrings reference entries to be kept
  - are outliers in terms of sequence length

```
(qc_tool_dev_env_streamlit) [ashteyman@d010 14:44:28 workforposter]$ python ../data_utils/QC_dataset.py
 -i francisella_ncbi.zip -bm median -bp .1 -n 3 -d - -gc_lower .3 -gc_upper .35 -se plasmid -on f_datas
et_forqc_workflow -cf directory -sd
Filtering N character sequences
Before filtering: 155
After filtering: 143
Filtering degenerate characters
Before filtering: 143
After filtering: 139
Filtering GC characters
Before filtering: 139
After filtering: 135
Filtering exclude strings
Before filtering: 135
After filtering: 131
Filtering by binning seq lengths
Median: 1892850
Upper bound: 2082140.5000000002
Lower bound: 1703559.5
Before filtering: 131
After filtering: 131
```

- Does initial filtering in a self-contained way such that, if the above issues are all that need to be screened for, can be the beginning and end of workflow

Sourmash:

- Well validated tool to take collections of sequence data, hash them (using something like sourmash sketch dna -p k=31,scaled=100,noabund to generate sketches) and then compare those sequences (using sourmash compare) to get distances to easily do hierarchical clustering

gettaxonomyfororgs.py :

- Uses MRItaxonomy, an in-house python package leveraging the marisa-trie[4,5] data structure, to very quickly assign taxonomic metadata, using very little hard drive space and RAM
- Assigns NCBI derived taxids to fasta entries to eventually cluster sequences using taxonomic relationships

clustersourmashresults.py :

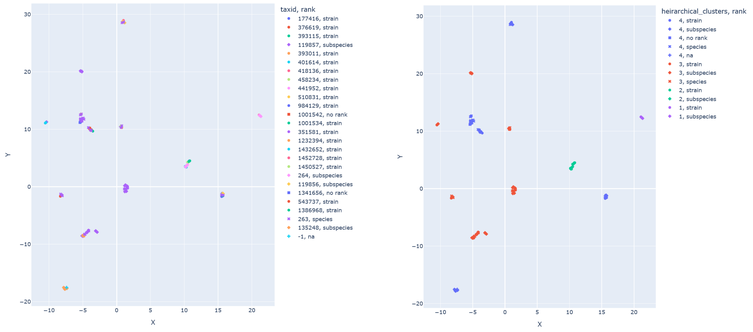
- Combines the information from the last 2 steps to produce flat clusters labeled with taxids and other information
- Has the capability to output a umap[6] representation of the clusters to allow manual inspection of the clustering
- Saves all this metadata for each sequence in .tsv file

subsetselectionbasedoncluster.py:

- Leverages the cluster metadata to go cluster by cluster and determine an identity for that cluster using the majority taxid for that cluster
- Then either drops sequence entries or relabels their taxid, for sequences that started with a taxid contrary to the cluster identity
- If desired, due to an overabundance of sequences present in the dataset, random subsampling without replacement is used to filter out records
- This ensures that each cluster is subsampled, with selected entries picked to capture as much variability on average, within the cluster as possible, while striving for broad representation of the key variation and minimizing duplicate sequences and otherwise repetitive sequences
- In essence, the 'variability' is retained as much as possible while the raw sequence amount is minimized for the subset of the original dataset, that have not been filtered out yet
- The result is output as a tsv that retains the cluster and taxonomic information
- Can be used to filter the original sequences to
  - have a small data footprint while retaining as much variability and sequence information as possible
  - Be free of sequences that have suspect properties like very high GC or too many degenerate characters, as well as minimize misannotated sequences
  - Allow downstream analyses to be carried out in a more tractable manner

Representative example - Francisella tularensis[7]

- Pulled down Francisella tularensis using NCBI datasets tool
- Did initial filtering using qc-datasets to remove genome sequences:
  - with too extreme a GC content (GC under 30% or over 35%)[8]
  - Too many degenerate characters (in this case 1 or more degenerate characters)
  - a sequence length more than 10% off the median sequence length, which can indicate assemblies with a lot of missing regions or inflated 'over-assembled' entries
  - that reference 'plasmid'
- Reduced 155 initial entries down to 131 with reasons listed for various groups output as a file and summarized as shown to the left
- Next Sourmash and gettaxonomyfororgs.py is applied to the remaining fasta entries
- Clustersourmashresults.py then combine this information, clusters the records and give an initial overview of how the clusters look (on the left), with a threshold of t=1, and then a final taxonomy labeling of the cluster (on the right) shown here:



- Then subsetslectionbasedoncluster.py, uses this information, along with a goal to have no more than 10 representatives for each taxid,
- For each of the 4 clusters, drops entries that have taxids that clash with their majority taxid (though they could also be relabeled instead), where each cluster belongs to a sub-clade of Francisella tularensis
- Final result is a raw set of genomes (with 155 entries) is filtered to down to a very compact set of 71 records that has been selected to retain as much variability as possible, allowing further down stream analysis to be much more tractable

References

1. Sayers EW, Beck J, Bolton EE, Bourexis D, Brister JR, Canese K, Comeau DC, Funk K, Kim S, Klimke W, Marchler-Bauer A, Landrum M, Lathrop S, Lu Z, Madden TL, O'Leary N, Phan L, Rangwala SH, Schneider VA, Skripchenko Y, Wang J, Ye J, Trawick BW, Pruitt KD, Sherry ST. Database resources of the National Center for Biotechnology Information. Nucleic Acids Res. 2021 Jan 8;49(D1):D10-D17. doi: 10.1093/nar/gkaa892. PMID: 33095870; PMCID: PMC7778943.
2. Pierce NT, Irber L, Reiter T *et al.* Large-scale sequence comparisons with *sourmash* [version 1; peer review: 2 approved]. *F1000Research* 2019, **8**:1006 (https://doi.org/10.12688/f1000research.19675.1)
3. O'Leary NA, Cox E, Holmes JB, Anderson WR, Falk R, Hem V, Tsuchiya MTN, Schuler GD, Zhang X, Torcivia J, Ketter A, Breen L, Cothran J, Bajwa H, Tinne J, Meric PA, Hlavina W, Schneider VA. Exploring and retrieving sequence and metadata for species across the tree of life with NCBI Datasets. Sci Data. 2024 Jul 5;11(1):732. doi: 10.1038/s41597-024-03571-y. PMID: 38969627; PMCID: PMC11226681.
4. Mori Y. marisa-trie: A space-efficient, simple and fast trie-based dictionary [version 1.0; peer review: not peer-reviewed]. GitHub Repository 2021, https://github.com/s-yata/marisa-trie
5. Korobov M. marisa-trie: A space-efficient, simple and fast trie-based dictionary [version 1.2; documentation]. ReadTheDocs 2018, https://marisa-trie.readthedocs.io/, https://github.com/pytries/marisa-trie
6. McInnes, L., Healy, J. and Melville, J., 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*
7. Accessed Feb. 8, 2024
8. Neubert, K., Zuchantke, E., Leidenfrost, R.M. *et al.* Testing assembly strategies of *Francisella tularensis* genomes to infer an evolutionary conservation analysis of genomic structures. *BMC Genomics* **22**, 822 (2021). https://doi.org/10.1186/s12864-021-08115-x

Contact Information

Alan Shteyman
T.# (240)-361-4054
E. ashteyman@mriglobal.org

MRIGlobal
65 West Watkins Mill rd.
Gaithersburg, MD