sourmash taxonomy: LCA summarization of genome-resolved taxonomic profiling

This manuscript (permalink) was automatically generated from bluegenes/2022-paper-sourmash-taxonomy@dc67327 on November 14, 2022.

Authors

- **N. Tessa Pierce-Ward**
  0000-0002-2942-5331 · bluegenes · saltyscientist
  Department of Population Health and Reproduction, UC Davis · Funded by Grant 1711984 from the NSF; Grant GBMF4551 from the Gordon and Betty Moore Foundation; Grant 2018911 from the NSF

- **Luiz Irber**
  0000-0003-4371-9659 · luizirber · luizirber
  Graduate Group in Computer Science, UC Davis; Department of Population Health and Reproduction, UC Davis · Funded by Grant GBMF4551 from the Gordon and Betty Moore Foundation; Grant R01HG007513 from the NIH NHGRI

- **Taylor Reiter**
  0000-0002-7388-421X · taylorreiter · ReiterTaylor
  Graduate Group in Food Science, UC Davis; Department of Population Health and Reproduction, UC Davis · Funded by Grant GBMF4551 from the Gordon and Betty Moore Foundation; Grant R03OD030596 from the NIH Common Fund

- **C. Titus Brown**
  0000-0001-6001-2677 · ctb
  Department of Population Health and Reproduction, UC Davis · Funded by Grant GBMF4551 from the Gordon and Betty Moore Foundation; Grant R01HG007513 from the NIH NHGRI; Grant 2018911 from the NSF; Grant R03OD030596 from the NIH Common Fund

✉ — Correspondence possible via GitHub Issues

Abstract

Background

Taxonomic profiling intro…

Sourmash gather is a method to use combinatorial observations of k-mers to find the minimum set of reference genomes that contain all k-mers of the query dataset. Sourmash gather results are non-overlapping: each k-mer is assigned to one and only one genome match. Gather results are to specific genomes, but many biological applications work at the species-level, rather than strain level. This is especially important as these strain matches may not be the best/ideal results. In many cases, the specific strain in your dataset may not be available in the database, meaning the results end up matching to a few suboptimal reference genomes. Sourmash taxonomy is a sourmash module designed to ingest sourmash gather results, integrate taxonomic information, and optionally aggregate the results using a lowest-common-ancestor approach.

Implementation

Sourmash taxonomy conducts LCA-style taxonomic summarization of the genomic profiling results from sourmash gather. It was introduced in sourmash v4.2, and all commands and outputs described here are available as of sourmash v4.6.

LCA-Style Lineage summarization

Sourmash gather uses a minimum set cover approach to identify the smallest set of reference genomes that contain all query information (k-mers) [1]. These matches are non-overlapping; that is, the sum of the query fraction assigned to each genome will be at most 100% (entire query matched to reference genomes).

Sourmash taxonomy LCA methods apply the taxonomic information from these reference genomes to their assigned query fraction and sum matches that correspond to the same taxonomic rank. For example, if the `sourmash gather` results for a metagenome include matches to 10 different strains of a given species, `sourmash tax` LCA can sum the fraction uniquely matched to each strain to obtain the total fraction uniquely matched to this species.

Because this approach relies upon non-overlapping reference assignments, separate `sourmash gather` results for the same query cannot be combined. However, `sourmash gather` can be run with any number of desired reference databases at once to produce a single set of non-overlapping assignments.

Two `sourmash tax` commands use this LCA-Style summarization: `metagenome` and `genome`.

sourmash tax metagenome

“sourmash tax metagenome” is designed to conduct LCA aggregation for metagenomes to build a taxonomic profile. `tax metagenome` ingests sourmash gather results from one or more metagenome queries and summarize the results for each metagenome at each taxonomic rank. `tax metagenome` provides several output file options, including some that are designed to facilitate input into downstream analysis tools.

Output Formats

csv_summary This output file reports a lineage summarization for each query at each taxonomic rank. Enable this output with `-F csv_summary`.

krona When used with `-F krona --rank RANK`, `sourmash tax metagenome` optionally produces a tab-separated list of results at a specific rank, which can be directly used to generate a `krona` plot (cite krona; add krona figure to results). This format is minimal, containing fraction of the query matched to the reported rank and lineage, with columns for each taxonomic rank down to the rank used for summarization.

lineage_summary The lineage summary format is a way to compare taxonomy results over multiple metagenome queries. It can be generated with `-F lineage_summary --rank RANK`, and will consist of one row per summarized lineage, with columns for the fraction matched in each metagenome sample.

kreport The kreport output reports kraken-style kreport output, with tab-separated columns. While this format typically records the percent of number of reads assigned to taxa, `sourmash taxonomy` creates comparable output by reporting the percent of k-mers matched to each taxon and the estimated number of base pairs that these k-mers represent. To best represent the percent of all reads, we use k-mer abundance information in this output. To generate this properly, query FracMinHash sketches should be generated with abundance information (`-p abund`), which will yield gather results with abundance weighting information.

sourmash tax genome

`sourmash tax genome` is designed to aggregate sourmash gather results run on genome assemblies. Rather than summarizing at each taxonomic rank, sourmash `tax genome` summarizes gather results starting from the lowest rank (species) and will classify the genome as soon as a user-modifiable criterion is reached. There are two classification strategies: classify the query once a match threshold is reached (e.g. 10% containment or 95% cANI), or classify the query once a rank is reached, regardless of percent match. The first strategy is recommended for more robust classification; the second strategy is required for downstream tools requiring all inputs at the same rank.

Output Formats

csv_summary This outputs a csv with taxonomic classification for each query genome. This output currently consists of six columns, query_name,rank,fraction,lineage,query_md5,query_filename, where fraction is the fraction of the query matched to the reported rank and lineage. The status column provides additional information on the classification:

krona When used with `-F krona --rank RANK`, `sourmash tax genome` optionally produces a tab-separated list of results at a specific rank, which can be directly used to generate a `krona` plot (cite krona; add krona figure to results). This format is minimal, containing fraction of the query matched to the reported rank and lineage, with columns for each taxonomic rank down to the rank used for summarization.

Utility commands

sourmash tax annotate

tax annotate annotates gather results with taxonomic information, without doing any LCA summarization.

sourmash tax prepare

tax prepare is a method for converting a csv of taxonomic lineage information into an sqlite database to enable faster loading and lineage assignment. It can also be used to combine lineage information for more that one database (e.g. GTDB, NCBI).

sourmash tax summarize

Summarize the lineage information in a human-reaedable summary

sourmash tax crosscheck

sourmash tax grep

select genomes entries by lineage; most useful for selecting subsets of results or reference genomes for sourmash analyses.

Results

Tax Metagenome

Reads

Contigs

Tax Genome

Discussion

- Database recommendations

- gather thresholding discussion/recommendations

- Limitations

  - K-mer size – > specificity/ sensitivity
  - K21 will give you species assignments, but you might not want to use them...

- Improvements:

  - Tax db inside of zip database?
  - Additional plotting?
  - CAMI output (need taxid)
  - genbank vs gtdb taxonomy translation?

References

1. **Lightweight compositional analysis of metagenomes with FracMinHash and minimum metagenome covers**
Luiz Irber, Phillip T Brooks, Taylor Reiter, NTessa Pierce-Ward, Mahmudur Rahman Hera, David Koslicki, CTitus Brown
Cold Spring Harbor Laboratory (2022-01-12) https://doi.org/gn34zt
DOI: 10.1101/2022.01.11.475838