bioRxiv pre-print LaTeX v0.4

Title:

Recentrifuge: robust comparative analysis and contamination removal for metagenomic data

Running title:

Recentrifuge

Jose Manuel Martí[1,#]

[1] *Institute for Integrative Systems Biology* (I^{2}SysBio), *Valencia, Spain.*

April 18, 2018

Abstract

Metagenomic sequencing is becoming more and more widespread in environmental and biomedical research, and the pace is increasing even more thanks to nanopore sequencing. With a rising number of samples and data per sample, the challenge of efficiently comparing results within a sample and between samples arises. Reagents, laboratory, and host related contaminants complicate such analysis, which demands of a reliable method for the removal of negative-control taxa from the rest of samples. Contamination is particularly critical in low-biomass body sites and environments, where it can comprise most of a sample if not all. With Recentrifuge, researchers can analyze results from Centrifuge and LMAT classifiers using interactive hierarchical pie charts with particular emphasis on the confidence level of the classifications. In addition to control-subtracted samples, shared and exclusive taxa per sample are provided with scores, thus enabling robust comparative analysis of multiple samples in any metagenomic study.

[#] Contact: jose.m.marti@uv.es

PREPRINT

bioRxiv pre-print LaTeX v0.4

Keywords— metagenomics, comparative analysis, contamination, low biomass, Centrifuge, LMAT

Introduction

Studies of microbial communities by metagenomics are becoming more and more popular in different biological arenas, like environmental, clinical, food and forensic studies (Miller, Montoya, et al. 2013; Ercolini 2013; Fricke et al. 2011). New DNA and RNA sequencing technologies are boosting these works by dramatically decreasing the cost per sequenced base. Scientists can now analyze sets of sequences belonging to microbial communities from different sources and times to unravel clinical or longitudinal patterns in the microbiota (see Supplemental Figure 6 for an example model). In shotgun metagenomic sequencing (SMS) studies, researchers extract and purify nucleic acids from each sample, sequence them, and analyze the sequences through a bioinformatics pipeline (see Supplemental Figures 7 and 8 for detailed examples). With the development of nanopore sequencing, portable and affordable real-time SMS is a reality (Edwards et al. 2016).

In case of low microbial biomass samples, there is very little native DNA from microbes; the library preparation and sequencing methods will return sequences whose principal source is contamination (Weiss et al. 2014; Kim, Hofstaedter, et al. 2017). Sequencing of RNA requiring additional steps introduces still further biases and artifacts (Kim, Song, et al. 2016). From the data science perspective, this is just another instance of the importance of keeping a good *signal-to-noise ratio* (Skolnik 2008). When the *signal* (inherent DNA/RNA, target of the sampling) approaches the order of magnitude of the *noise* (acquired DNA/RNA from contamination and artifacts), particular methods are required to try to tell them apart.

The roots of contaminating sequences are diverse, as they can be traced back to nucleic acid extraction kits (the *kitome*) (Salter et al. 2014), reagents and diluents (Olm et al. 2017; Kulakov et al. 2002), the host (Ames, Gardner, et al. 2015) and the post-sampling environment (Lusk 2014), where contamination arises from different origins such as airborne particles, crossovers between current samples or DNA remains from past sequencing runs (Gruber 2015). Variable amounts of DNA from these sources are sequenced simultaneously with native microbial DNA, which could lead to severe bias in magnitudes like abundance and coverage, particularly in low microbial biomass situations (Nayfach and Pollard 2016). Even the metagenomic reference databases have a non-negligible amount of cross-contamination (Ames, Gardner, et al. 2015; Gruber 2015; Lu and S Salzberg 2018).

bioRxiv pre-print LaTeX v0.4

Regarding the *kitome*, it varies even within different lots of the same products. For example, tthe DNeasy PowerSoil Kit (formerly known as PowerSoil DNA Isolation Kit), a product that usually provides significant amounts of DNA and has been widely used, including Earth Microbiome Project and Human Microbiome Project, often yields a background contamination by no means negligible (Kim, Hofstaedter, et al. 2017). The lower the biomass in the samples, the more essential it is to collect negative control samples to help in the contamination background assessment because without them it would be almost impossible to distinguish inherent microbiota in a specimen (signal) from contamination (noise).

Even if the native and contaminating DNA are accurately separated, the problem of performing a reliable comparison between samples remains. In general, the sequence reads are assigned to different taxonomic ranks, especially when using a more conservative approach like Lowest Common Ancestor (LCA) (Ames, Hysom, et al. 2013). While LCA drastically reduces the risk of false positives, it usually spreads the taxonomic level of the classifications from the more specific to the more general. Although the taxonomic classifier does not use the LCA strategy, each read is assigned a particular score or confidence level, which should be taken into account by any downstream application as a reliability estimator of the classification.

Despite these difficulties, it is still more challenging to compare samples with very different DNA yields, for instance, low microbial biomass samples versus high biomass ones, because of the different resolution in the taxonomic levels. This sort of problem also arises when the samples, even with DNA yields in the same order of magnitude, have an entirely different microbial structure so that the minority and majority microbes are fundamentally different between them (Nayfach and Pollard 2016). A closely related problem emerges in metagenomic bioforensic studies and environmental surveillance, where it is essential to have a method prepared to detect the slightest presence of a particular taxon (Bazinet et al. 2017; Doggett et al. 2016; Fricke et al. 2011) and provide quantitative results with both precision and accuracy.

From the beginning, the application of SMS to environmental samples provided biologists with insight of microbial communities not obtainable from sequencing of Bacterial Artificial Chromosome (BAC) clones or 16S rRNA (Allen et al. 2004; Venter et al. 2004). The need and challenges of comparative metagenomics were soon underlined (Schloss and Handelsman 2005; Tringe et al. 2005). MEGAN (Huson, Auch, et al. 2007), one of the first metagenomic data analysis tools, provided in its initial release a very basic comparative of samples that has improved with an interactive approach in more recent versions (Huson, Beier, et al. 2016). In general, metagenomic classification and assembly software is more intra- than inter-

bioRxiv pre-print LaTeX v0.4

sample oriented (Breitwieser, Lu, et al. 2017). Several tools have tried to fill the gap, starting with CoMet (Lingner et al. 2011), a web based tool for comparative functional profiling of metagenomes. Soon later, a different approach was presented with the discovery of the crAssphage thanks to the crAss software (Dutilh et al. 2012), which provides reference-independent comparative metagenomics using cross-assembly. The following year, a new tool for visually comparing microbial community structure across microbiomes, Community-analyzer (Kuntal et al. 2013) appeared. In 2014, it was released COMMET (Maillet et al. 2014), a tool that goes a step further by enabling the combination of multiple metagenomic datasets. A package for direct comparative metagenomics by using k-mers (Benoit et al. 2016) was published in 2016.

In 2015, a highly publicized report on the metagenomics of the New York subway suggested that the plague and anthrax pathogens were part of the normal subway microbiome. Soon afterwards, it was widely criticized (Ackelsberg et al. 2015) and, later, it was proved that reanalysis of the New York subway data with appropriate methods did not detect the pathogens (Tiffany et al. 2016). As a consequence of this and other similar problems involving metagenomic studies, a work directed by Rob Knight (A González et al. 2016) emphasized the importance of validation in metagenomic results and issued a tool based on BLAST (Platypus Conquistador) that confirms the presence or absence of a taxon of interest within SMS datasets by relying on two reference sequence databases: one for inclusions, with the sequences of interest, and the other for exclusions, with any known sequence background. Another BLAST-based method for validating the assignments made by less precise sequence classification programs has been recently announced (Bazinet et al. 2017).

Recentrifuge enables researchers to analyze results from the Centrifuge and the LMAT classifiers using Krona-like charts (Ondov et al. 2011), but placing emphasis on the score level of the classifications by the use of a scaled color-map. In addition to the scored plots for the raw samples, four different sets of scored plots are generated for each taxonomic level of interest: control-subtracted samples, shared taxa (with and without subtracting the control) and exclusive taxa per sample. This battery of analysis and plots permits robust comparative analysis of multiple samples in metagenomic studies, especially useful in cases of low biomass environments or body sites.

PREPRINT

bioRxiv pre-print LaTeX v0.4

Methods

Visualization and computing kernel

The visualization part of Recentrifuge is based on interactive hierarchical pie charts. The Krona (Ondov et al. 2011) 2.0 JavaScript (JS) library developed at the Battelle National Biodefense Institute was adapted and extended to fit the particular needs of the problem.

The computing kernel of Recentrifuge is an object-oriented parallel code written in Python. For each sample a logical taxonomic tree is populated according to the NCBI Taxonomy (NCBI 1988b), with the leaves usually belonging to the lower taxonomic levels like varietas, subspecies or species. The methods involving trees were implemented as recursive functions for compactness and robustness, making the code less error-prone. One of such methods is essential for understanding the way Recentrifuge prepares samples before any comparison or operation, like control subtraction. It recursively "folds the tree" for any sample if the number of assigned reads to a taxon is under the `mintaxa` setting (minimum reads assigned to a taxon to exist in its own), or because the taxonomic level of interest is over the assigned taxid (taxonomic identifier). See Figure 1A for a working example of the method in action for two samples. The same procedure is applied to the trees of every sample in the dataset. This method does not just "prune the tree", on the contrary, it accumulates the counts n_i of a taxon in the parent ones n_p and recalculates the parent score σ_p as a weighted average taking into account the counts and score of both. In general:

$$\sigma'_p \text{ (parent taxa new score)} = \frac{1}{n_p + \sum_i^D n_i} \left(\sigma_p n_p + \sum_i^D \sigma_i n_i \right) \forall (\sigma_i, n_i) : 0 < n_i < \texttt{mintaxa}$$

where D is the number of descendant taxa that are to be accumulated in the parent one and σ_i their respective scores. This is done recursively until the desired conditions are met. This method is applied, at a given taxonomic level, to the trees of every sample before being compared in search for the shared and exclusive taxa. The `mintaxa` parameter defaults to 10 as in LMAT, but this value can be modified and set independently for control and real samples.

For implementation details of the Recentrifuge computing kernel please see Supplementary Section 2.

PREPRINT

bioRxiv pre-print LaTeX v0.4

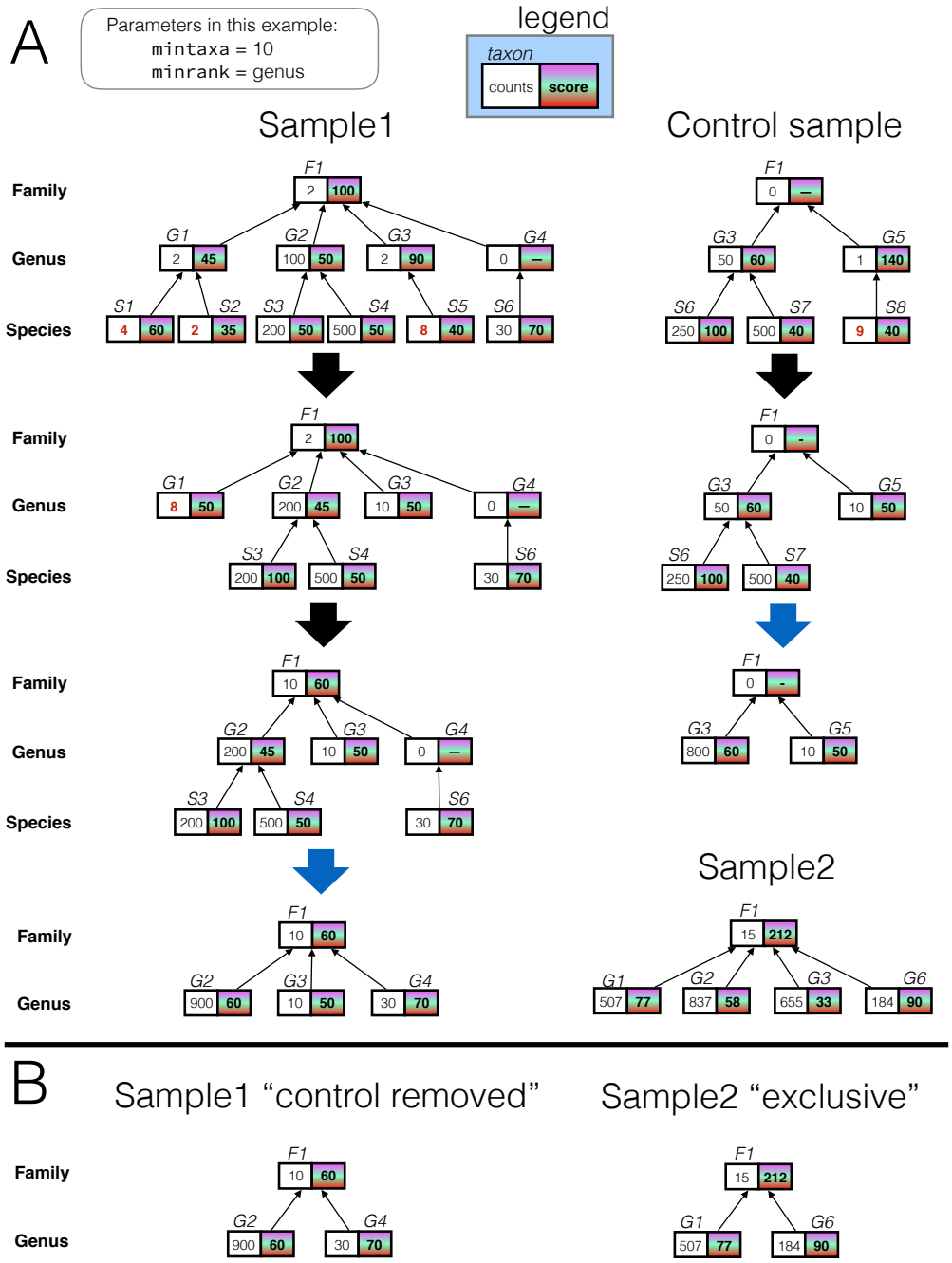

Figure 1. Operating with taxonomic trees. (**A**) Example of the recursive function which "folds the tree" to prepare the taxonomic trees for further operations, with the parameter `mintaxa` set to 10 (default) and the minimum rank of interest `minrank` set to 'genus'. Initially, their trees show the direct taxonomic classification results. Then, recursively, the leaves of the tree are accumulated in the parent node if their number of assigned reads is under `mintaxa` (shown in red and bold counts) or if their corresponding taxonomic rank is below `minrank`. In this 'folding', the parent score is updated with a weighted average of its own score and the ones of the descendants that are being accumulated. E.g., after the 1st step, the $G1$ taxon at the sample is updated with $n_p = 2 + 4 + 2 = 8$ counts and score of $\sigma_p = \frac{1}{8}(60 * 4 + 35 * 2 + 45 * 2) = 50$. As the counts for $G1$ are still under `mintaxa`, in the 2nd step they are accumulated in $F1$ and its score updated to $\frac{1}{10}(50 * 8 + 100 * 2) = 60$. (**B**) Continuing with the example in (A), at genus level, a couple of derived samples are shown: the right one with the control removed from *Sample1*, the left one with the exclusive taxa of *Sample2*, those taxa not present in the rest of samples, in this case, the control and *Sample1*.

bioRxiv pre-print LaTeX v0.4

Derived samples

In addition to the input samples, Recentrifuge includes some sets of derived samples in its output. After parallel calculations for each taxonomic level of interest, it includes hierarchical pie plots for CTRL (control subtracted samples), but also for EXCLUSIVE, SHARED and SHARED_CONTROL samples, defined below.

Let $\mathbb{T}$ mean the set of taxids in the NCBI Taxonomy (NCBI 1988b) and T_s the collection of taxids present in a sample s. If R_s stands for the set of reads of a sample s and C_s for the group of them classifiable, then the taxonomic classification c is a function from C_s to $\mathbb{T}$, i.e., $C_s \xrightarrow{c} \mathbb{T}$, where $C_s \subseteq R_s$ and $c[C_s] = T_s \subseteq \mathbb{T}$. The set L of the 30 different taxonomic levels used in the NCBI is ordered in accordance with the taxonomy, so $(L, <)$ is a strictly ordered set, since *forma* $<$ *varietas* $<$ *subspecies* $< \cdots <$ *domain*. Then, $T_s = T_s^{forma} \bigcup \cdots \bigcup T_s^{domain} = \bigcup^L T_s^l$, where T_s^l represents the collection of taxa belonging to a sample s for a particular taxonomic rank or level l. Related with this, we can write as $T_s^{\rightarrow l}$ the taxa of the sample s for a taxonomic level l once we have applied the 'tree folding' to such level l detailed in the previous subsection (and Figure 1A).

For a taxonomic rank k of interest, in a series of S samples where there are $N < S$ negative controls, the sets of taxa in the derived samples CTRL $\left({}^{\text{CTRL}}T_s^k\right)$, EXCLUSIVE $\left({}^{\text{EXCL}}T_s^k\right)$, SHARED $\left({}^{\text{SHARED}}T^k\right)$ and SHARED_CONTROL $\left({}^{\text{SHRD_CTRL}}T^k\right)$ are computed by Recentrifuge as follows:

$$
\begin{aligned}
{}^{\text{CTRL}}T_s^k &= T_s^{\rightarrow k} \setminus \bigcup_n^N T_n^{\rightarrow k} \\
{}^{\text{EXCLUSIVE}}T_s^k &= T_s^{\rightarrow k} \setminus \bigcup_{m \neq s}^S T_m^{\rightarrow k} \\
{}^{\text{SHARED}}T^k &= \bigcap_m^S T_m^{\rightarrow k} \\
{}^{\text{SHARED_CONTROL}}T^k &= \bigcap_{m>N}^S T_m^{\rightarrow k} \setminus \bigcup_n^N T_n^{\rightarrow k}
\end{aligned}
$$

Please see Figure 1B for examples. Finally, in parallel, Recentrifuge generates a set of SUMMARY samples condensing the results for all the taxonomic levels of interest.

Robust contamination removal

For a taxonomic rank k, after the 'tree folding' procedure detailed above, the contamination removal algorithm retrieves the set of candidates $\bar{T}_s^{\rightarrow k}$ to contaminant taxa from the $N < S$ control samples. Depending on the relative frequency $(f_i = n_i / \sum_i n_i)$ of these taxa in the control samples and if they are also present in other samples, the algorithm classifies them in contamination level groups: critical, severe, mild and other. Except for the latter group, the contaminants are removed from non-control samples. Then, Recentrifuge checks any taxon in the other contam-

bioRxiv pre-print LaTeX v0.4

inants group for crossover contamination so that it removes any taxon marked as crossover from every sample except the one or ones selected as source of the contamination, i.e., the algorithm removes any taxon $t_s^k \in \bar{T}_s^{\to k}$ from a normal sample unless it passes the crossover robust check: a statistical test screening for overall outliers and an order of magnitude test against the control samples. See Figure 2 for an example.

The crossover robust tests are defined as follows:

$$\begin{aligned} \text{Outliers statistic test } (t_s^k) \quad &: \quad f_{t_s^k} \quad > \quad \text{median}\left\{f_{t_1^k}, \ldots, f_{t_S^k}\right\} \; + \; \delta\, Q_n \\ \text{Order of magnitude test } (t_s^k) \quad &: \quad f_{t_s^k} \quad > \quad 10^{\xi} \max\left\{f_{t_1^k}, f_{t_2^k}, \ldots, f_{t_N^k}\right\} \end{aligned}$$

where Q_n (Rousseeuw and Croux 1993) is a scale estimator to be discussed below, and δ and ξ are constant parameters of the robust contamination removal algorithm. The parameter δ (3 to 5 by default) is an outliers cutoff factor, while ξ (2 to 3 by default) is setting the difference in order of magnitude between the relative frequency of the candidate to crossover contaminator in the sample s and the greatest of such values among the control samples.

Q_n is the chosen scale estimator for screening the data for outliers because of his remarkably general robustness and other advantages compared to other estimators (Rousseeuw and Croux 1993; Rousseeuw and Croux 1994) , like the MAD (median absolute deviation) or the k-step M-estimators: it has a 50% breakpoint point, a smooth influence function, very high asymptotic efficiency at Gaussian distributions and is suitable for asymmetric distributions, which is our case, all at a reasonable computational complexity, as low as $O(n)$ for space and $O(n \log n)$ for time. So, here:

$$Q_n \;=\; d\left\{\left|f_{t_i^k} - f_{t_j^k}\right|_{i<j\leq S}\right\}_{(m)} \quad : \quad m = \binom{\frac{S}{2}+1}{2} = \frac{\Gamma\left(\frac{S}{2}+2\right)}{2\,\Gamma\left(\frac{S}{2}\right)} = \frac{S}{4}\left(\frac{S}{2}+1\right)$$

where $d = 3.4760$ is a constant selected for assymetric non-gaussian models similar to the negative exponential distribution, m refers to the mth order statistics of the pairwise distances and Γ is the Gamma function.

Results

Recentrifuge is a metagenomics analysis software with two different main parts: the computing kernel, implemented and parallelized from scratch using Python, and the interactive interface, written in JavaScript as an extension of the Krona (Ondov et al. 2011) JS library to take full advantage of the classification confidence level. Figure 3 summarizes the package context and data flows. Recentrifuge straightforwardly accepts both Centrifuge (Kim, Song, et al. 2016) and LMAT (Ames, Hysom, et al. 2013) output files, thus enabling scored oriented analysis and visualization for these tools. LMAT plasmids assignment system (Ames, Gardner, et al. 2015) is also supported by Recentrifuge.

bioRxiv pre-print LaTeX v0.4

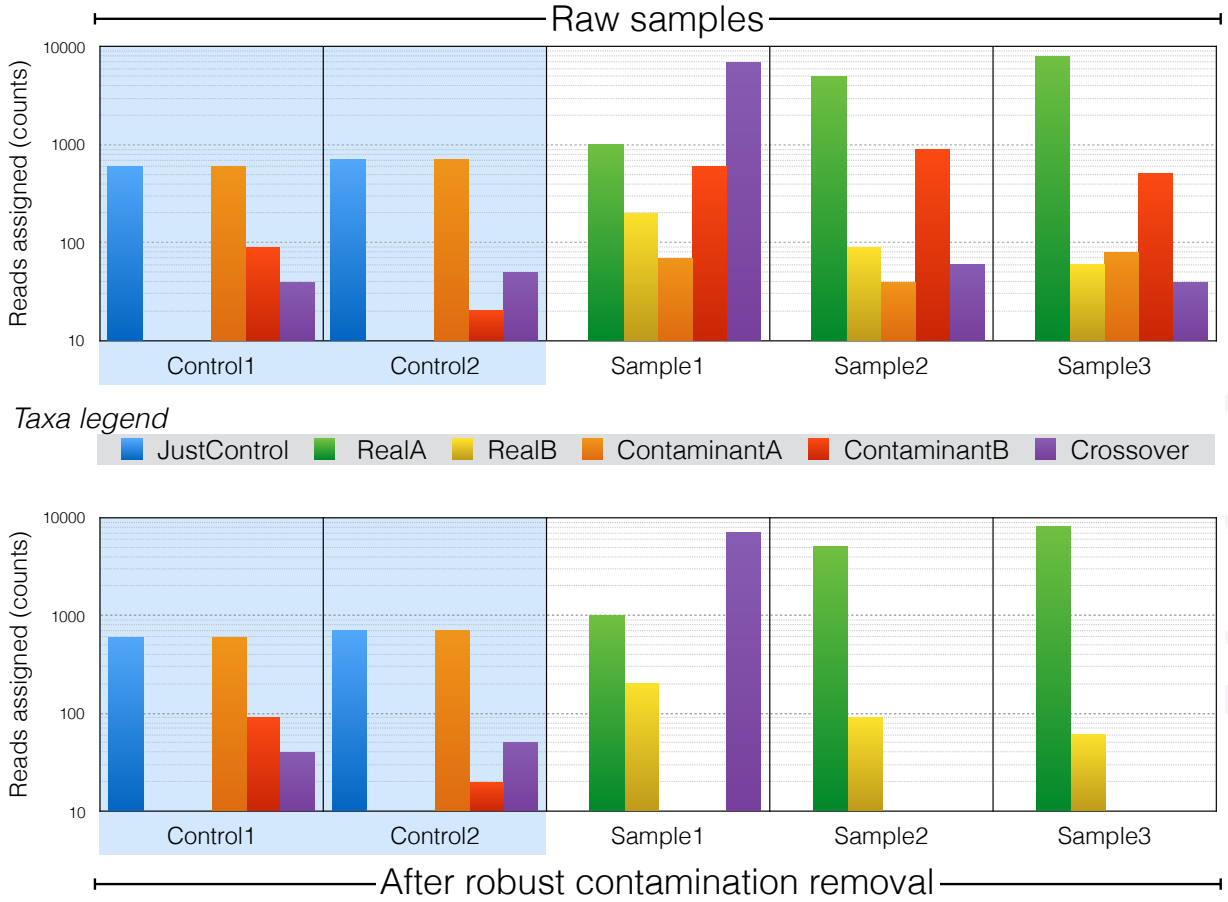

Figure 2. Robust contamination removal. This is a hypothetic example with 5 samples and 6 dominant taxa to illustrate how the algorithm works. The top and bottom part of the Figure show the absolute frequency of reads assigned to the taxa before and after the contamination removal, respectively. There are 2 control samples, which are not modified through the process. In the rest of samples, the general contaminants (those taxa present in the controls and other samples, like *ContaminantA* and *B*) are removed, except in case of crossover contamination: those taxa are kept in the source(s) sample(s) –`Sample1` here– and removed from other real samples (`Sample2` and `3` in this example).

To ensure the widest portability for the interactive visualization of the results, the central outcome of Recentrifuge is a stand-alone HTML file which can be loaded by any JavaScript-enabled browser. Figure 4 shows a labeled screenshot of the corresponding Recentrifuge web interface for an example of SMS study (see Supplemental Figure 6). The package also provides statistics about the reads and their classification performance. Another Recentrifuge output is an Excel spreadsheet detailing all the classification results in a compact way. This format is useful for further data mining on the data, for example, as input for applications such as longitudinal (time or space) series analyzers like *cmplxCruncher* (in development). In addition, the user can choose between different score visualization algorithms, some of which are more interesting for datasets containing variable length reads, for example, the ones generated by Oxford Nanopore sequencers.

bioRxiv pre-print LaTeX v0.4

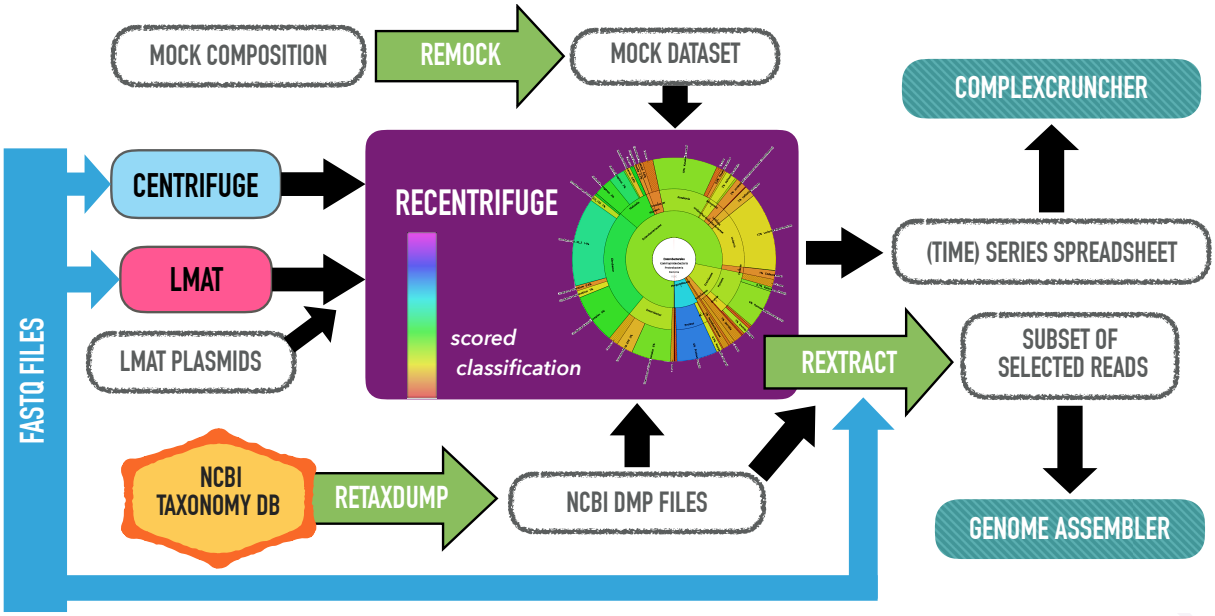

Figure 3. Outline of the Recentrifuge package with its context and main data flows. The green arrows show fully implemented pipelines. Recentrifuge accepts both Centrifuge (Kim, Song, et al. 2016) and LMAT (Ames, Hysom, et al. 2013) direct output files, thus enabling a scored oriented visualization for any of these codes. Recentrifuge is also supporting LMAT plasmids assignment system (Ames, Gardner, et al. 2015). The additional output of Recentrifuge to different text field formats enable further longitudinal (time or space) series analysis, for example, using *cmplxCruncher* (in development). The NCBI Taxonomy (NCBI 1988b) dump databases are easily retrieved using *Retaxdump* script. *Rextract* utility extracts a subset of reads of interest from single or paired-ends FASTQ input files, which can be used in any downstream application, like genome assembling and visualization.

Finally, some filters are available, like the minimum score threshold. This filter can be used to generate different output sets from a single run of the classifier with a low minimum hit length (MHL) setting. That would not only save time and computational resources, but also would make compatible, on the one hand, the advantage of a low MHL in low quality reads with, on the other hand, the benefit of a high MHL to avoid false positives.

Further products are obtained using auxiliary tools in the package, like *Rextract*, a script which helps in extracting a subset of classified reads of interest from the single or paired-ends FASTQ input files. These reads can be used in any downstream application, such as genome visualization and assembling. *Remock* is a script for easily creating mock Centrifuge samples, which is useful not only for testing and validation purposes, but also for introducing a list of previously known contaminants to be taken into account by the robust contamination removal algorithm.

bioRxiv pre-print LaTeX v0.4

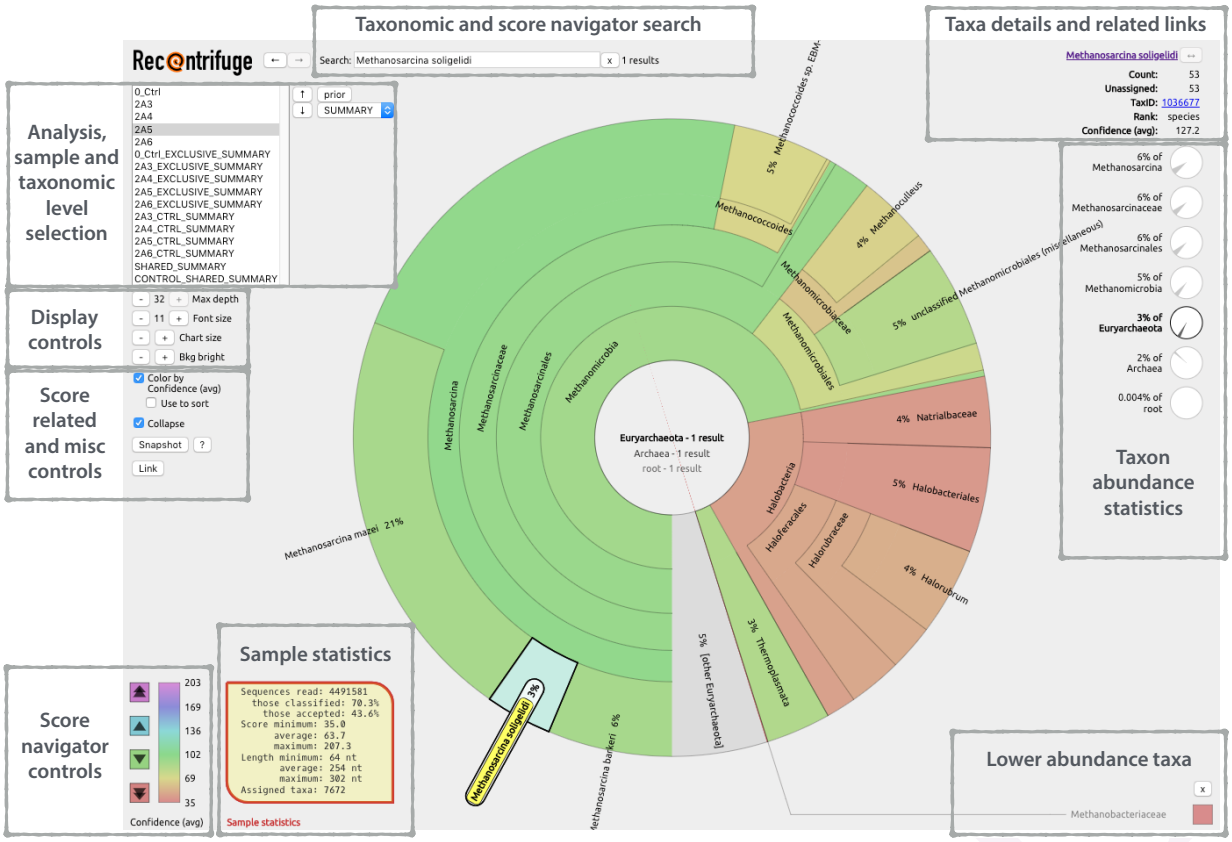

Figure 4. Screenshot of the Recentrifuge web interface for an SMS study (see Supplemental Figure 6 for details of the example). The main parts of the interface are highlighted and labeled. The sample 2A5 is selected (see the sample selection box in the top left under the Recentrifuge logo) and essential sample statistics are shown in the bottom left of the view. In the center, the corresponding hierarchical pie chart is drawn with zoom in the phylum *euryarchaeota*. For each taxon, the background color reflects the average confidence level of the taxonomic classification following the scale plotted in the bottom left of the figure, where the buttons for the score navigator are also found. Sorting of taxa is done attending to the average confidence level, as that option is enabled in the plot. In this particular case, the taxon *Methanosarcina soligelidi* is selected in the pie chart, thus prompting the display of related statistics and links in the top right of the figure. The current links are to Google Scholar and NCBI Taxonomic Browser (NCBI 1988b), while the statistics include the number of reads assigned to this or lower taxonomic levels —Counts— and their average confidence —Confidence (avg)—, the number of reads just assigned to this level —Unassigned—, the NCBI taxid —TaxID— and rank —Rank—, and some relative frequencies data.

Discussion

Recentrifuge enables score-oriented robust comparative analysis of multiple samples, especially in low-biomass metagenomic studies, where contamination removal is a must. Figure 5 summarizes the instant benefits of introducing Recentrifuge in an SMS study (see Supplemental Figure 6 for details about the example).

Just as it is important to accompany any physical measurement by a statement of the associated uncer-

bioRxiv pre-print LaTeX v0.4

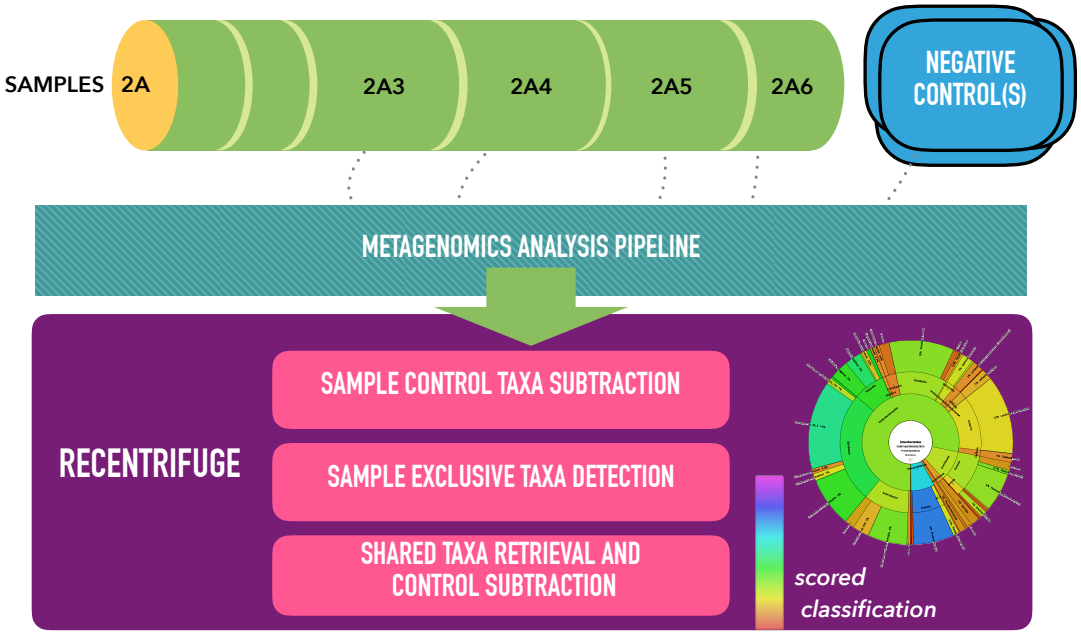

Figure 5. Summary of immediate advantages of applying Recentrifuge to a study involving SMS of different but related samples, including negative controls (see Supplemental Figure 6). Four different sets of scored charts are generated for each taxonomic level of interest in addition to the scored plots for the raw samples: samples with the control taxa subtracted, the exclusive taxa per sample and the shared taxa with and without control taxa subtracted. This battery of analysis and plots permits robust comparative analysis of multiple samples in low biomass metagenomic studies.

tainty, it is desirable to accompany any read classification by a confidence estimation of the taxon that has been finally assigned. Recentrifuge reads the score given by Centrifuge or LMAT to any read and uses this valuable information to calculate an averaged confidence level for each classification node. This value may be also a function of further parameters, such as read quality or length, which is especially useful in case of notable variations in the length of the reads, like in the datasets generated by nanopore sequencers.

Only a few codes, such as Krona (Ondov et al. 2011) and MetaTreeMap (Hebrard and Taylor 2016), are hitherto able to handle a score assigned to the classification nodes. In Recentrifuge, the calculated score is kept in all the downstream analysis and comparisons. That is an essential advantage of Recentrifuge over other metagenomic dataset analysis tools. It provides with an interactive framework for an easy assessment of the validity of the taxonomic assignments.

By enabling score-oriented robust comparative analysis of multiple samples in metagenomic studies, Recentrifuge could play a key role not only in the analysis of oligotrophic microbes in environmental samples, but also in a more reliable detection of minority organisms in clinical or forensic samples.

bioRxiv pre-print LaTeX v0.4

Data access

Recentrifuge is currently available in Github (https://github.com/khyox/recentrifuge) under different free software licenses. The wiki (https://github.com/khyox/recentrifuge/wiki) is the most extensive and updated source of documentation for Recentrifuge.

Acknowledgments

I would like to thank Jordi Burguet-Castell and Andrea Salvador-Pascual for critical comments on the manuscript. I would also like to thank Carlos P. Garay and Jose Fco. Martí for valuable conversations about Recentrifuge.

References

Ackelsberg J, Rakeman J, Hughes S, Petersen J, Mead P, Schriefer M, Kingry L, Hoffmaster A, and Gee JE. 2015. Lack of Evidence for Plague or Anthrax on the New York City Subway. *Cell Systems*. **1**: 4–5.

Allen EE, Richardson PM, Tyson GW, Chapman J, Banfield JF, Hugenholtz P, Rokhsar DS, Solovyev VV, Rubin EM, and Ram RJ. 2004. Community structure and metabolism through reconstruction of microbial genomes from the environment. *Nature*. **428**: 37–43.

Ames SK, Gardner SN, Marti JM, Slezak TR, Gokhale MB, and Allen JE. 2015. Using populations of human and microbial genomes for organism detection in metagenomes. *Genome research*. **25**: 1056–1067.

Ames SK, Hysom DA, Gardner SN, Lloyd GS, Gokhale MB, and Allen JE. 2013. Scalable metagenomic taxonomy classification using a reference genome database. *Bioinformatics (Oxford, England)*. **29**: 2253–2260.

Balvočiūtė M and Huson DH. 2017. SILVA, RDP, Greengenes, NCBI and OTT — how do these taxonomies compare? *BMC Genomics*. **18**: 114.

Bazinet AL, Ondov BD, Sommer DD, and Ratnayake S. 2017. BLAST-based validation of metagenomic sequence assignments. *BioRxiv*. DOI: 10.1101/181636.

Benoit G, Peterlongo P, Mariadassou M, Drezen E, Schbath S, Lavenier D, and Lemaitre C. 2016. Multiple comparative metagenomics using multiset k-mer counting. *PeerJ Computer Science*. **2**: e94.

bioRxiv pre-print LaTeX v0.4

Breitwieser FP, Lu J, and Salzberg SL. 2017. A review of methods and databases for metagenomic classification and assembly. *Briefings in Bioinformatics*. **0**: bbx120.

Breitwieser FP and Salzberg SL. 2016. Pavian: Interactive analysis of metagenomics data for microbiomics and pathogen identification. *BioRxiv*. DOI: 10.1101/084715.

Doggett NA, Mukundan H, Lefkowitz EJ, Slezak TR, Chain PS, Morse S, Anderson K, Hodge DR, and Pillai S. 2016. Culture-Independent Diagnostics for Health Security. *Health security*. **14**: 122–142.

Dutilh BE, Schmieder R, Nulton J, Felts B, Salamon P, Edwards RA, and Mokili JL. 2012. Reference-independent comparative metagenomics using cross-assembly: crAss. *Bioinformatics*. **28**: 3225–3231.

Edwards A, Debbonaire AR, Sattler B, Mur LA, and Hodson AJ. 2016. Extreme metagenomics using nanopore DNA sequencing: a field report from Svalbard, 78 N. *BioRxiv*. DOI: 10.1101/073965.

Ercolini D. 2013. High-Throughput Sequencing and Metagenomics: Moving Forward in the Culture-Independent Analysis of Food Microbial Ecology. *Applied and Environmental Microbiology*. **79**: 3148–3155.

Ewels P, Magnusson M, Lundin S, and Käller M. 2016. MultiQC: summarize analysis results for multiple tools and samples in a single report. *Bioinformatics (Oxford, England)*. **32**: 3047–3048.

Fricke WF, Cebula TA, and Ravel J. 2011. Chapter 28 (Genomics). In *Microbial Forensics*. (Ed. by B Budowle, SE Schutzer, RG Breeze, PS Keim, and SA Morse), pp. 479–492. Academic Press, San Diego.

González A, Vázquez-Baeza Y, Pettengill JB, Ottesen A, McDonald D, and Knight R. 2016. Avoiding Pandemic Fears in the Subway and Conquering the Platypus. *mSystems*. **1**: e00050.

González R, House P, Levkivskyi I, Roach L, and Rosum GV 2016. PEP-526: Syntax for Variable Annotations. [Internet]. Python Software Foundation.

Gruber K. 2015. Here, there, and everywhere. *EMBO reports*. **16**: 898–901.

Hebrard M and Taylor TD. 2016. MetaTreeMap: An Alternative Visualization Method for Displaying Metagenomic Phylogenic Trees. *PloS one*. **11**: e0158261.

Huson DH, Auch AF, Qi J, and Schuster SC. 2007. MEGAN analysis of metagenomic data. *Genome research*. **17**: 377–386.

Huson DH, Beier S, Flade I, Górska A, El-Hadidi M, Mitra S, Ruscheweyh HJ, and Tappu R. 2016. MEGAN Community Edition - Interactive Exploration and Analysis of Large-Scale Microbiome Sequencing Data. *PLoS computational biology*. **12**: e1004957.

bioRxiv pre-print LaTeX v0.4

ISO/IEC 2012. BS ISO/IEC 19505-1:2012: Information technology. Object Management Group Unified Modeling Language (OMG UML). Infrastructure. English.

Kim D, Song L, Breitwieser FP, and Salzberg SL. 2016. Centrifuge: rapid and sensitive classification of metagenomic sequences. *Genome Research*. **26**: 1721–1729.

Kim D, Hofstaedter CE, Zhao C, Mattei L, Tanes C, Clarke E, Lauder A, Sherrill-Mix S, Chehoud C, and Kelsen J. 2017. Optimizing methods and dodging pitfalls in microbiome research. *Microbiome*. **5**: 52.

Kulakov LA, McAlister MB, Ogden KL, Larkin MJ, and O'Hanlon JF. 2002. Analysis of Bacteria Contaminating Ultrapure Water in Industrial Systems. *Applied and Environmental Microbiology*. **68**: 1548–1555.

Kuntal BK, Ghosh TS, and Mande SS. 2013. Community-analyzer: a platform for visualizing and comparing microbial community structure across microbiomes. *Genomics*. 409–418.

Lingner T, Aßhauer KP, Schreiber F, and Meinicke P. 2011. CoMet-a web server for comparative functional profiling of metagenomes. *Nucleic Acids Research*. **39**: W523.

Lu J and Salzberg S. 2018. Removing Contaminants from Metagenomic Databases. *BioRxiv*. DOI: 10.1101/261859.

Lusk RW. 2014. Diverse and widespread contamination evident in the unmapped depths of high throughput sequencing data. *PloS one*. **9**: e110808.

Maillet N, Collet G, Vannier T, Lavenier D, and Peterlongo P 2014. Commet: Comparing and combining multiple metagenomic datasets. English. In: *2014 IEEE Int Conf on BIBM*. IEEE, pp. 94–98.

Miller RR, Montoya V, Gardy JL, Patrick DM, and Tang P. 2013. Metagenomics for pathogen detection in public health. *Genome medicine*. **5**: 81.

Miller RR, Uyaguari-Diaz M, et al. 2016. Metagenomic Investigation of Plasma in Individuals with ME/CFS Highlights the Importance of Technical Controls to Elucidate Contamination and Batch Effects. *PLoS One*. **11**: e0165691.

National Center for Biotechnology Information 1988a. Nucleotide. [Internet]. National Library of Medicine (Bethesda, MD, USA). URL: https://www.ncbi.nlm.nih.gov/nucleotide/.

National Center for Biotechnology Information 1988b. Taxonomy. [Internet]. National Library of Medicine (Bethesda, MD, USA). URL: https://www.ncbi.nlm.nih.gov/taxonomy/.

Nayfach S and Pollard KS. 2016. Toward Accurate and Quantitative Comparative Metagenomics. *Cell*. **166**: 1103–1116.

PREPRINT

bioRxiv pre-print LaTeX v0.4

Olm MR, Butterfield CN, Copeland A, Boles TC, Thomas BC, and Banfield JF. 2017. The Source and Evolutionary History of a Microbial Contaminant Identified Through Soil Metagenomic Analysis. *mBio*. **8**: 1969.

Ondov BD, Bergman NH, and Phillippy AM. 2011. Interactive metagenomic visualization in a Web browser. *BMC bioinformatics*. **12**: 385.

Patel A, Picard A, Jhong E, Hylton J, Smart M, and Shields M 2017. Google Python Style Guide. [Internet]. Version 2.59. Google. URL: `https://google.github.io/styleguide/pyguide.html`.

Rossum GV, Warsaw B, and Coghlan N 2013. PEP-8: Style Guide for Python Code. [Internet]. Python Software Foundation.

Rosum GV, Lehtosalo J, and Langa L 2015. PEP-484: Type Hints. [Internet]. Python Software Foundation.

Rousseeuw PJ and Croux C. 1993. Alternatives to the Median Absolute Deviation. *Journal of the American Statistical Association*. **88**: 1273–1283.

Rousseeuw PJ and Croux C. 1994. The bias of k-step M-estimators. *Statistics & Probability Letters*. **20**: 411–420.

Salter SJ, Cox MJ, Turek EM, Calus ST, Cookson WO, Moffatt MF, Turner P, Parkhill J, Loman NJ, and Walker AW. 2014. Reagent and laboratory contamination can critically impact sequence-based microbiome analyses. *BMC Biology*. **12**: 87.

Schloss PD and Handelsman J. 2005. Metagenomics for studying unculturable microorganisms: cutting the Gordian knot. *Genome biology*. **6**: 229.

Skolnik MI. 2008. Radar handbook. In. McGraw-Hill, New York.

Tiffany H et al. 2016. Urban Transit System Microbial Communities Differ by Surface Type and Interaction with Humans and the Environment. *mSystems*. **1**: e00018.

Tringe SG et al. 2005. Comparative Metagenomics of Microbial Communities. *Science*. **308**: 554–557.

Venter JC et al. 2004. Environmental Genome Shotgun Sequencing of the Sargasso Sea. *Science*. **304**: 66–74.

Weiss S, Amir A, Hyde ER, Metcalf JL, Song SJ, and Knight R. 2014. Tracking down the sources of experimental contamination in microbiome studies. *Genome biology*. **15**: 564.

Wood DE and Salzberg SL. 2014. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biology*. **15**: R46.

PREPRINT

bioRxiv pre-print LaTeX v0.4

Supplemental Material of Recentrifuge: robust comparative analysis and contamination removal for metagenomic data

1 Longitudinal metagenomics

In longitudinal metagenomics, sets of sequences belonging to microbial communities from different sources and times can be retrieved and analyzed to unravel spatial and temporal patterns in the microbiota (see Supplemental Figure 6 for an example). In those studies involving SMS, DNA is extracted from each sample using a commercial kit or an optimized custom protocol, the purified DNA is then sequenced and the sequences analyzed by a bioinformatics pipeline. This entire procedure is summarized in Supplemental Figure 7 but the core of the analysis is detailed in Supplemental Figure 8.

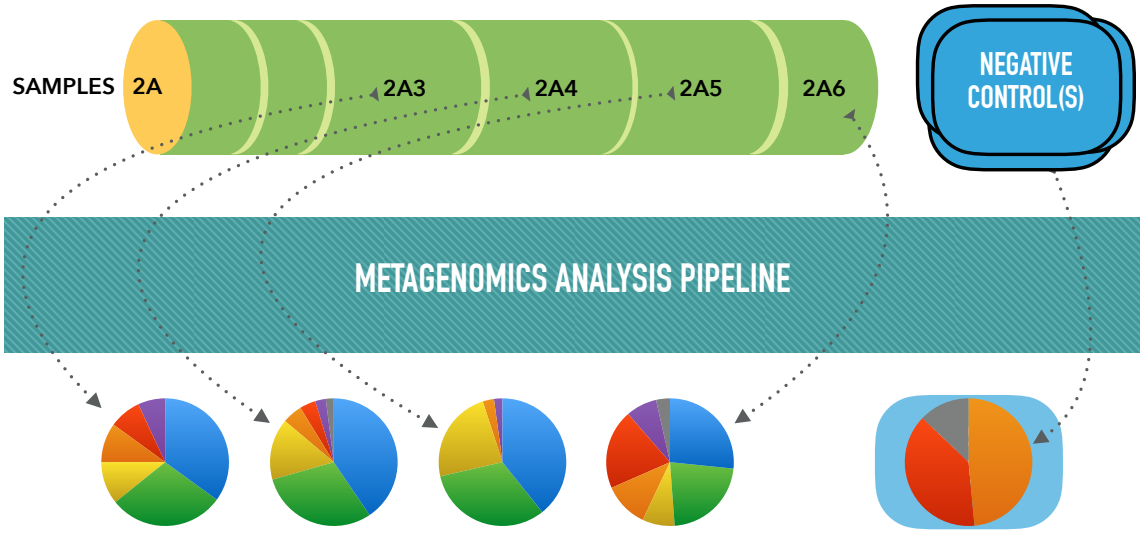

Figure 6. (Supplemental) This is an example outlining the problem of comparing different but related samples in a SMS study. The sample named 2A is subdivided longitudinally in six subsamples whose DNA is extracted along with a negative control sample. The purified DNA is then sequenced and the generated sequencing reads are processed through a metagenomics analysis pipeline (as the one detailed in Supplemental Figure 7). A collection of different datasets are finally generated, which should be adequately compared in order to elucidate lengthwise patterns in the microbiota within the 2A sample.

bioRxiv pre-print LaTeX v0.4

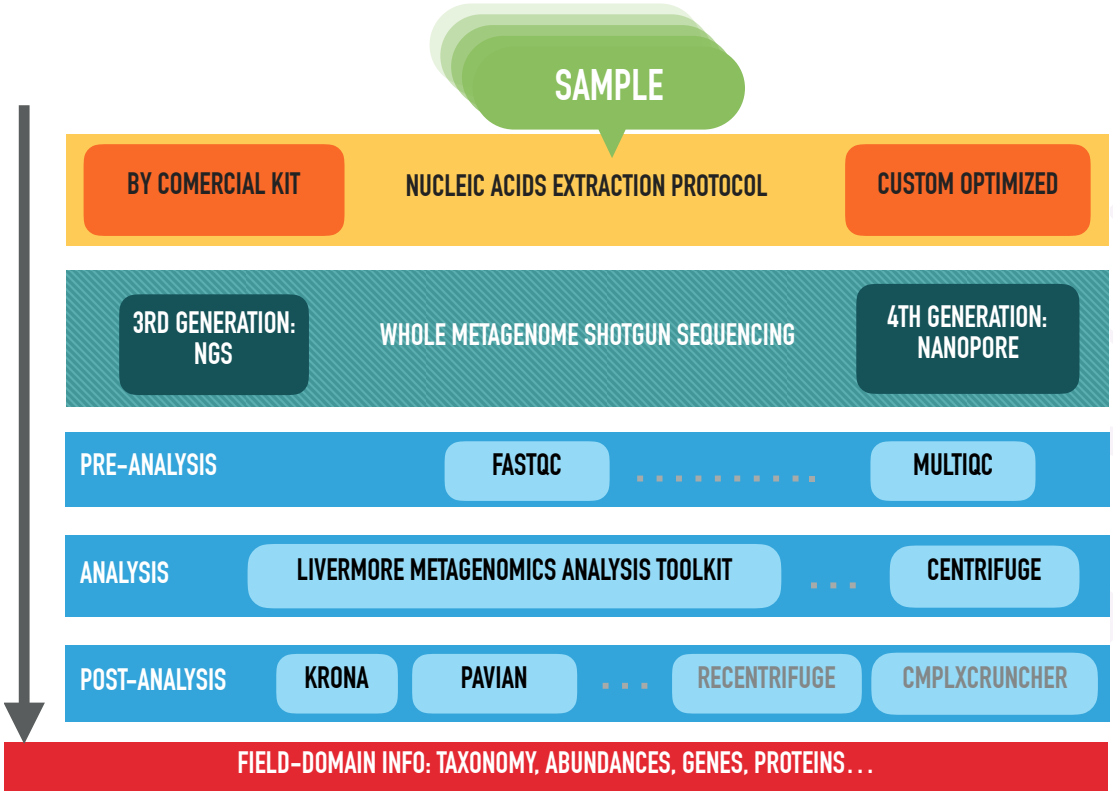

Figure 7. (Supplemental) A study involving SMS, like the one illustrated by Supplemental Figure 6, spans in a number of stages to extract valuable field-domain information starting from the original samples. For each sample, the DNA is extracted using a commercial kit, a custom protocol optimized for the type of sample or a combination of both. Next, a library matching the sequencing technology to be used is prepared with the purified DNA, which is then sequenced. The reads provided by the sequencer are processed through a bioinformatics pipeline that we could roughly separate in three consecutive steps: in the pre-analysis the reads are quality checked using codes like FastQC (Babraham Bioinformatics, 2016) and MultiQC (Ewels et al. 2016); in the analysis stage, the most computationally intensive, the reads are taxonomically or functionally classified using software packages like LMAT (Ames, Hysom, et al. 2013) and Centrifuge (Kim, Song, et al. 2016) (see Supplemental Figure 8 for details); finally, in the post-analysis step, the results are further processed to enable deeper analysis and improved visualization by using different tools like Krona (Ondov et al. 2011), Pavian (Breitwieser and SL Salzberg 2016) and others.

bioRxiv pre-print LaTeX v0.4

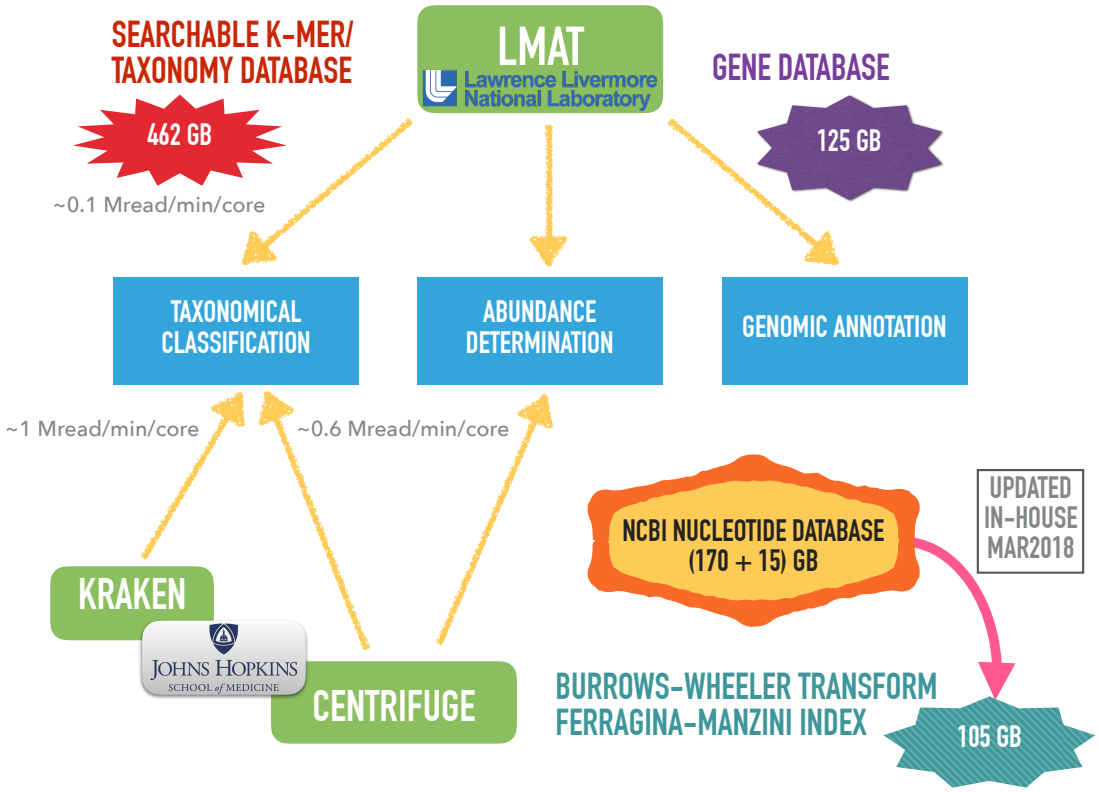

Figure 8. (Supplemental) The core phase of a metagenomics analysis pipeline (see Supplemental Figure 6 and 7 for the outline of the bioinformatics phases) is carried out by high performance computing software. These are intensive codes in both CPU and memory (sometimes, they are input/output intensive too), like LMAT (Ames, Hysom, et al. 2013), Kraken (Wood and SL Salzberg 2014) and, more recently, Centrifuge (Kim, Song, et al. 2016). All these tools are performing taxonomic classification and abundance estimation, whereas LMAT is also able to annotate genes. For the taxonomic classification, both LMAT and Kraken use an exact k-mer matching algorithm with large databases ($\sim$ 100 GiB) while Centrifuge use compression algorithms to reduce the databases size ($\sim$ 10 GiB) but at some speed expense. The most complete LMAT database is approaching half terabyte of required memory while the Centrifuge database generated in-house in March 2018 from the NCBI Nucleotide (NCBI 1988a) *nt* database (170 GiB) is occupying just 105 GiB.

bioRxiv pre-print LaTeX v0.4

2 Computing kernel implementation details

The following list abridges some implementation details concerning the Recentrifuge computing kernel:

1. Coded using python multi-platform parallelization (both by input file and by taxonomic level) in order to reduce the elapsed time when dealing with huge datasets.
2. Intensive use of recursive methods to cope with tree-arithmetics, which enables robust comparison between taxonomic trees at any rank. The code is informed of the 30 different taxonomic levels used in the NCBI taxonomy (NCBI 1988b). Recentrifuge can properly deal with arbitrary tax levels (`NO_RANK`) in between the named NCBI taxonomic ranks, as happens with some complex eukaryotic taxa.
3. It is a full statically annotated Python 3.6 code. PEP-8 (Rossum et al. 2013), PEP-484 (Rosum et al. 2015) and PEP-526 (R González et al. 2016) compliant. Written following the Google python style guide (Patel et al. 2017). Pylint and mypy checked.
4. Implemented under a object-oriented paradigm to ease future extensions targeting new or improved uses; it is prepared for additional input formats and other taxonomies different from NCBI —by direct support extending the base class or indirectly by using a mapping software like CrossClassify (Balvočiūtė and Huson 2017)—. A summarized UML (Unified Modeling Language) (ISO/IEC 2012) class diagram of Recentrifuge is shown in Supplemental Figure 9.

It is worth to mention that the code allows that a different parameter set could be applied to the control samples, including `mintaxa` and `minscore`. This feature is very useful when the control samples are too different from the ordinary ones and thus require special values for the parameters which define how Recentrifuge treats the samples data.

The table 1 shows basic code layout statistics about Recentrifuge (source files, lines, and number of code lines equivalent in third generation languages) provided by cloc (v.1.76):

PREPRINT

bioRxiv pre-print LaTeX v0.4

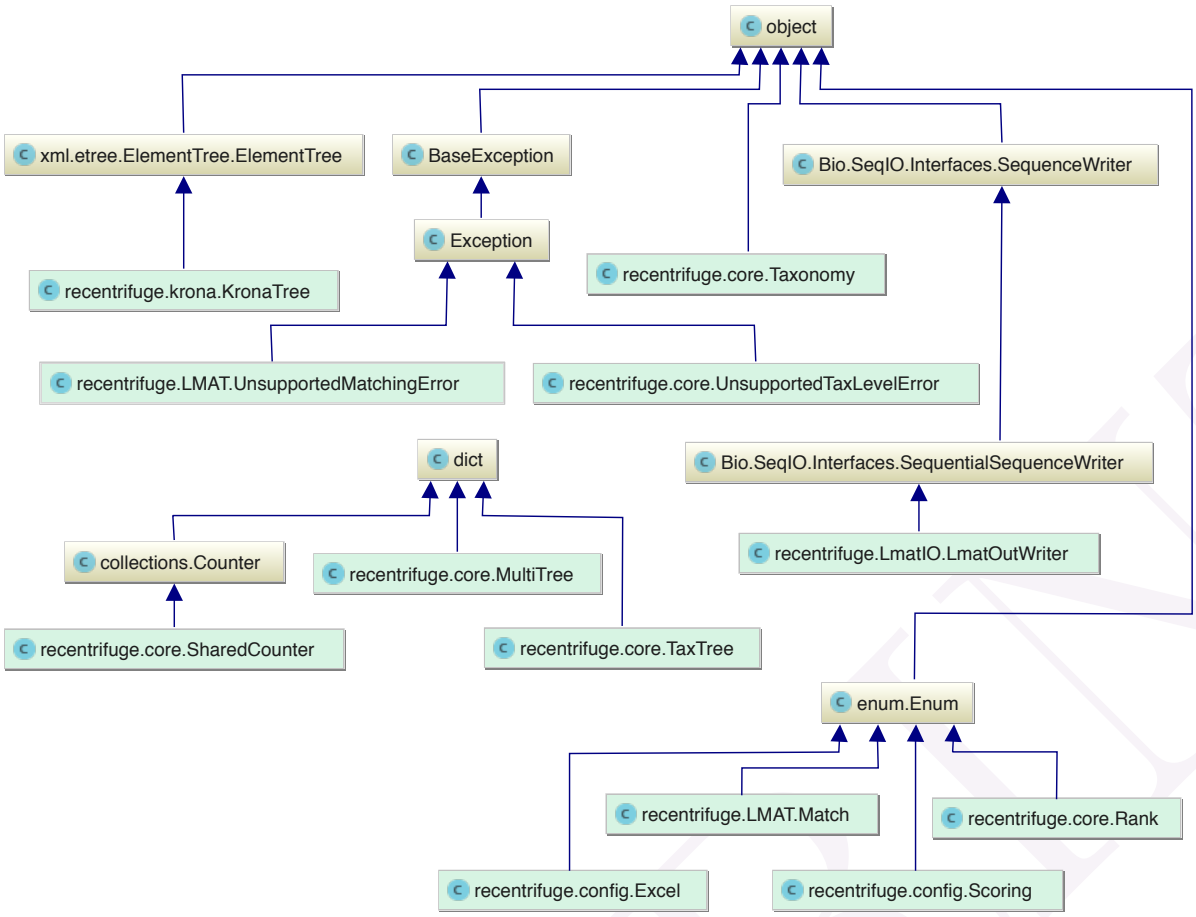

Figure 9. (Supplemental) UML class diagram summarizing relationships between classes in the Recentrifuge package. The green background denotes developed classes currently used in the package. The rest of classes in the figure, the parent classes from which the Recentrifuge ones are derived, belong to the Python Standard Library and BioPython.

Table 1. (Supplemental) Basic code layout statistics about Recentrifuge (source files, lines, and number of code lines equivalent in third generation languages) provided by cloc (v.1.76)

Language	files	blank	comment	code	×	scale	=	3rd gen. equiv
Python	16	435	686	3328	×	4.20	=	13977.60
JavaScript	1	927	537	5026	×	1.48	=	7438.48
SUM	17	1362	1223	8354	×	2.56	=	21416.08