GigaScience

Libra: scalable k-mer based tool for massive all-vs-all metagenome comparisons

--Manuscript Draft--

Manuscript Number:	GIGA-D-18-00324R1
Full Title:	Libra: scalable k-mer based tool for massive all-vs-all metagenome comparisons
Article Type:	Research
Funding Information:	Directorate for Computer and Information Science and Engineering (1640775) — Prof. Bonnie L Hurwitz
Abstract:	Background: Shotgun metagenomics provides powerful insights into microbial community biodiversity and function. Yet, inferences from metagenomic studies are often limited by dataset size and complexity, and are restricted by the availability and completeness of existing databases. De novo comparative metagenomics enables the comparison of metagenomes based on their total genetic content. Results: We developed a tool called Libra that performs all-vs-all comparison of metagenomes for precise clustering based on their k-mer content. This tool presents three main innovations: the use of a scalable Hadoop framework for massive metagenome comparisons, Cosine Similarity for calculating the distance using sequence composition and abundance while normalizing for sequencing depth, and a web-based tool in iMicrobe (http://imicrobe.us) that uses the CyVerse advanced cyberinfrastructure to promote broad use of the tool by the scientific community. Conclusions: A comparison of Libra to equivalent tools using both simulated and real metagenomic datasets, ranging from 80 million to 4.2 billion reads, reveals that methods commonly implemented to reduce compute time for large datasets—such as data reduction, read count normalization, and presence/absence distance metrics—greatly diminish the resolution of large-scale comparative analyses. In contrast, Libra uses all of the reads to calculate k-mer abundance in a Hadoop architecture that can scale to any size dataset to enable global-scale analyses and link microbial signatures to biological processes.
Corresponding Author:	Bonnie Hurwitz UNITED STATES
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	
Corresponding Author's Secondary Institution:	
First Author:	Illyoung Choi, MS
First Author Secondary Information:	
Order of Authors:	Illyoung Choi, MS Alise J. Ponsero, PhD Matthew Bomhoff, BS Ken Youens-Clark, BA John H. Hartman, PhD Bonnie L Hurwitz, PhD
Order of Authors Secondary Information:	
Response to Reviewers:	As part of your revisions, it would be great if you can include performance evaluation in the case of long reads from Oxford Nanopore, PacBio, or Illumina sequencers. Reviewer #2 suggests to use some real nanopore datasets (available in e.g.,https://github.com/nanopore-wgs-consortium/NA12878/blob/master/Genome.md) for testing and evaluating Libra against other tools.

RESPONSE: We thank the reviewer and the editor for this excellent suggestion. We performed additional experiments using long read data (for the mock community and HMP datasets) per the reviewer's suggestion to evaluate Libra in comparison to other tools. The results show that Libra performs equally well on long and short read datasets. These data have been included in the manuscript, and as a detailed response to the reviewer below. We also go one step further, to show that Illumina and 454 short read technologies produce consistent results.

In addition, please register any new software application in the SciCrunch.org database to receive a RRID (Research Resource Identification Initiative ID) number, and include this in your manuscript. This will facilitate tracking, reproducibility and re-use of your tool.

RESPONSE: Thank you for the excellent recommendation. We have now registered Libra as a tool in SciCrunch.org and have added the RRID (SCR_016608) to the manuscript for tracking and re-use of our tool.

Response to Reviewers

Reviewer reports:

Reviewer #1: Title: Libra: robust biological inferences of global datasets using scalable k-mer based all-vs- all metagenome comparisons

Summary:

The authors present Libra, a software system for metagenomics sequence data analysis. Libra is "the first step in implementing a cloud-based resource." The authors claim 3 innovations: (1) Libra uses Hadoop, (2) Libra use of distance metrics, (3) Libra runs on CyVerse. The manuscript presents a software system that bundles known techniques into an integrated platform that should scale well to large datasets and is freely available on an existing cloud resource.

Commentary:

The software appears to be useful and well architected. The comparison to other tools is extensive. The manuscript says this was the first step of a system in development. The manuscript may be better presented as an application note or a progress report published elsewhere rather than a Research article for GigaScience. A paper with similar scope and similar format, published in GigaScience and referenced in this manuscript, appeared as a Review article not a Research article (Guo R, Zhao Y, Zou Q, Fang X, Peng S. Bioinformatics applications on Apache Spark. Gigascience. 2018).

RESPONSE: We sincerely thank the reviewer for understanding and recognizing the merit of the work. We decided to pursue a Research Article rather than a Data Note given that in addition to performing extensive analyses to compare and contrast the Libra to other tools based on synthetic data and mock communities, we also re-analyzed the Tara Oceans Virome data to reveal new biological insights that were missed in the original 2015 Science article. Specifically, we show for the first time that viral communities in the ocean are similar across temperature gradients, irrespective of their location in the ocean. We feel that this finding provides additional scientific insight into viruses in the ocean and therefore merits publication as a GigaScience Research article, rather than Data Note which would be constrained to just technical advances.

As a Research article, the manuscript makes three claims to innovation. One claimed innovation is Libra's use of sophisticated distance metrics. Libra gives users a choice of three metrics. The manuscript says two of those metrics are "widely used" and the other is "a new distance metric … using Cosine Similarity" (line 140). This is not the first use of cosine similarity in metagenomics (e.g., Virtual metagenome reconstruction from 16S rRNA gene sequences. Okuda et al. Nature Communications 2012). The manuscript does not distinguish this usage from prior ones. The authors say cosine similarity was demonstrated here only because it had the shortest runtime (line 235). The other two claims to innovation specify the use of Hadoop and CyVerse but both are widely used already. Thus, the claims seem unproven.

RESPONSE: We appreciate the reviewers' comments. Distance metrics have been widely used in metagenomics for a variety of purposes. In the paper the reviewer cites, cosine similarity was used as a metric to evaluate the accuracy of reconstructed genomes from "virtual metagenomes" based on the number of KEGG Orthologous Genes in common. The "virtual metagenomes" were derived based on species present in a 16S rRNA dataset obtained from gel electrophoresis (amplicon data), and are technically not from metagenomes which would consist of WGS data from microbes in a sample. Therefore the analysis is based on gene counts in genomes, and not on metagenomic sequence data. Our approach uses cosine similarity as a distance metric for comparing complete metagenomic sequence signatures, that has not been applied in this capacity before (in comparable tools Mash and Simka). As suggested, we updated the paper to cite this reference and describe its use in an alternative capacity in genome analysis. Similarly, no other tool for comparing sequence signatures from metagenomes uses Hadoop for massive analytics, or has been imbedded in the CyVerse cyberinfrastructure. Thus, these innovations remain novel for our use-case and stated applications.

Some claims would be easier to assess if the language were more precise. For example: (1) The Title claims the new tool provides robust inference and the Abstract claims that other tools diminish the robustness of analysis. The manuscript also says Hadoop is robust. "Robust" is not defined or discussed further. (2) The Abstract describes Libra's three distance metrics as "complex" and the Innovations section refers to them as "sophisticated" but neither word gets defined or defended.

RESPONSE: We thank the reviewer for pointing out the need for further clarification of these terms. In the "Libra Implementation" section we define robust in the following way: "Hadoop allows robust parallel computation over distributed computing resources via its simple programming interface called MapReduce, while hiding much of the complexity of distributed computing (e.g. node failures)." The term robust refers to the ability to handle error without the need to restart analyses which is vital as the scale of data increases. We have updated the text to explicitly define this and have also removed the word "robust" from the title.
We define complex distance metrics in the introduction in the following way "simple distances scale linearly and complex distances metrics scale quadratically as additional samples are added". We define "complex distance" as a distance metric with a high complexity in terms of compute time. This is an important point, we have removed the term from the text to avoid confusion.

We agree with the reviewer that sophisticated is not a precise word choice and have removed the term from the Innovations section to be consistent with the abstract.

The referencing could use more rigor. For example: (1) Cosine similarity is introduced with an off-topic reference [34] (line 140) to a conference talk that compares several similarity metrics within the domain of document clustering. (2) A seemingly relevant review of prior art is not referenced (Web Resources for Metagenomics Studies. Dudhadara et al. GPB 2015). A seemingly relevant claim to prior art, found right in the CyVerse online documentation, is not noted (Scalable metagenomic analysis using iPlant. Vaughn. CyVerse Wiki 2013). (3) The Introduction says one existing tool is the fastest (line 72) without reference or explanation. The same paragraph states that abundance is a critical and previously ignored factor "central to microbial ecology" without providing a reference or sufficient evidence.

RESPONSE: Thank you for your careful review and drawing our attention to issues with the references. We have carefully reviewed the references and updated according to the reviewer's suggestions. We removed the reference for cosine similarity given that other publications in the field do not reference any papers, given that it is a commonly used similarity metric.

Reviewer #2: The authors developed a new k-mer based method called Libra that enables large scaled metagenomics samples comparison. The authors introduced the advanced method MapReduce to the area of comparative metagenomics and designed a pipeline for counting k-mers and computing distances using MapReduce. The new method was extensively evaluated on simulations and real datasets. The authors also made the software available on iMicrobe, which is easily accessible by biologists in the community. Overall the manuscript is well written and the datasets are publicly available. More details and discussions can be added in order to make the paper more comprehensive. Here are some comments:

RESPONSE: We thank the reviewer for recognizing the value of the work and providing valuable suggestions for enhancing the work.

1. In Figure 2A, it seems that the distances defined by Mash and Libra decrease as the sequencing depth increases. However, the authors claim that "sequencing depth has little effect on the distance between samples in Mash and Libra (natural weighting)", which is confusing. Ideally, since the four artificial metagenomes were generated from the same community as the original sample, the distance between the artificial sample and the original sample should be small. The figure shows that as sample size is as large as 5M, the distance of Libra is close to 0. The large distance for small sample size may due to the variation in the sampling. The authors could elaborate more on the results.

RESPONSE: If the communities were sampled at their exact ratios we would theoretically get a distance of zero irrespective of the sample size. However, similar to real-world sequencing, random sampling selects more sequences from dominant organisms than rare (based on a higher probability of sampling a dominant organism over a rare one). This means that decreasing the sequencing depth removes the rare community component. Simka does not see this effect, because they normalize all samples to the lowest read count. Whereas Mash and Libra are taking into account all of the reads in the metagenomes, therefore they measure a larger difference when you compare the smallest (0.5M read sample) and largest (10 million read sample). We have updated the text to better describe this important point.

2. The authors claimed that "the Mash algorithm shows lower overall resolution (Figure 3A) as compared to Libra (Figure 3B)". Could the authors explain more how they defined "resolution"? From Figure 2B, it seems to me that the range of Mash distance is relatively smaller compared with that of other measures. So plotting heatmaps under the same range (0-1) may lead to the unclear patterns for Mash as what we see in Figure 3A.

RESPONSE: Thank you for your comment. This is indeed an important clarification. Mash, Simka, and Libra all report distance in the same range (0-1), and therefore we plot the data according to the reported results from each tool. The distance between metagenomes that Mash is able to detect based on the sketching algorithm (that uses a subset of reads) is small, leading to lower resolution in the graph compared to Simka and Libra that use 100% of the reads. We have updated the legend for the Figure to better describe this important point.

3. The author claimed from Figure 4 that "these differences reflect the effect of using all of the read data (Libra) rather than a subset (Mash)." It is true that Mash estimate the distance based on a subset of data. On the other hand, Mash and Libra use different measures. So the difference in clustering may also come from the different measures. The authors could add a discussion for this.

RESPONSE: We agree with the reviewer’s comment. Distance metrics are fundamental to comparative metagenomic analyses, but also add clarification on the importance of using abundance in the distance calculation. In Figure 4, Mash (Fig 4A) and Simka (Fig 4C) both use Jaccard distance, however Simka achieves better clustering by using all of the reads and including the abundance in the distance calculation. We have updated the text to clarify this point and also reference the Simka paper which shows a careful analysis of the effects of sketching compared to using all of the k-mers.

4. Have the authors compared the running time of Libra with other methods? It would

be great to see if Libra can have high accuracy and at the meanwhile reduces the running time or is within the similar running time with other methods.

RESPONSE: A direct comparison of the runtime of the tools is not possible given that each tool runs on a different computational architecture with a different number of servers and total CPU/memory (Mash runs on a single server; Simka runs on an HPC; and Libra on Hadoop). When running the HMP dataset we found that Mash runs in minutes, Simka in 2-3 hours, and Libra in ~12 hours. Because Libra uses a Hadoop framework, staging the data into HDFS takes significant run time, although the calculations are fast. Libra is developed as a method to scale to large datasets and be fault tolerant, whereas smaller datasets will run faster and with equal resolution using Simka. Thus, the major innovation Libra provides is analyses at scale. This important point was added into the discussion.

Reviewer #3: Choi et al propose a new tool called Libra for computing pairwise comparisons of samples in the case of large set of samples that is scalable (via cloud-based resources),
fast and as accurate as (or better) than standard methods.
Several major and minor issues were detected:

RESPONSE: We thank the reviewer for their time and excellent suggestions.

Major issues:
- Unlike authors of Mash, authors of Libra do not provide any performance evaluation in case of long reads from Oxford Nanopore, PacBio, or Illumina sequencers. It seems Libra was only tested for short reads.
If this is the case and given the fact that long reads (10kbp or more) are becoming standard size for metagenomics, genomics (cf. numerous paper published in Nature methods, and Nature Biotechnology dealing with Nanopore reads) then authors should explicitly mention in the manuscript as well as in the title of the manuscript that Libra works only for short reads.
Otherwise, if Libra can be used for Nanopore sequencing for example then authors should create synthetic datasets with NanoSim (Yang et al, GigaScience. 2017.doi:10.1093/gigascience/gix010) and show the performance of it.

Also several real datasets of nanopore data are available (e.g., https://github.com/nanopore-wgs-consortium/NA12878/blob/master/Genome.md) for testing and should be used for evaluating Libra against the other tools.

RESPONSE: We thank the reviewer for their excellent and timely suggestion, we have added new experiments that demonstrate the utility of each of the tools (Mash, Simka, and Libra) on long read data. Specifically, we show that simulated data long read data for the mock community shows a similar stepwise distance pattern between each of the mock communities (as expected), but has a higher overall distance between each of the mock communities likely due to the high simulated random error rate compared to short read data. We added this analysis to the results, and included a new supplemental figure to show the results. Thus, all of the tools can distinguish differences in long read and short read data alike. Please note that we chose to use SimLoRD for the simulated metagenomic data given that Nanosim is constrained to simulated genomic data. The same supplemental also includes the simulated data for the mock community based on Illumina data (per the reviewer’s suggestion below).

Per the reviewer’s suggestion, we have also added an analysis of the CAMI HMP “toy dataset” with simulated long reads from PacBio, to complement the analyses we already ran on real short read Illumina data from the Human Microbiome Project. This analysis shows that each of the tools is able to cluster the samples broadly by body site, however there are small misclassifications shared across all tools. These data suggest that increased error rate of the technology could have a limited impact on k-mer based analytics.

- The supplemental document, in docx format, containing information about methods has formulas that are not readable. Please correct and update this document, compile it in PDF, and also include as much as possible of it in the main text.

RESPONSE: We thank the reviewer for drawing our attention to this issue. We integrated the supplemental methods document into a comprehensive and refined methods section in the main article. All formulas have been checked and fixed.

Minor issues:
- Please provide a reference related to the microbial dark matter in for the claim in introduction:"k-mer based classifiers that rapidly assign metagenomic reads to known microbes
miss the microbial dark matter". Then, please discuss/explain how well/bad is Libra to deal with "the microbial dark matter" that these taxonomic classifiers miss?

RESPONSE: Thank you for pointing out the missing reference, we have updated the text to include a reference. A detailed discussion of how comparative metagenomic approaches in general (employed by Mash, Libra, and Simka) elucidate the unknown fraction of metagenomes is included in the section titled “De novo comparative metagenomics offers a path forward.”

- Table 1: This big table provides a long list of tools and yet the list is not exhaustive. Since this list is not exhaustive, and it is not clear how the tools were selected or even ordered, I'd recommend to explain better or put in supplement.
I'd also include a recent paper surveying these tools of your choice in case the readers want to know more and to simplify the reading.

RESPONSE: Thank you for this suggestion. The main point of the table was to show that tools have been developed to compare genomes using Hadoop (which are much smaller in terms of total bytes), but none compare metagenomes to-date. Moreover, none of these Hadoop-based tools are not available in an easy to use web-interface and accessible to the general user. We also show that metagenomic tools extensively use k-mer based analytics, most of these perform comparisons to known reference databases for taxonomic classification, and some have been developed to compare reads between metagenomes (however most cannot scale). We also point out that there are a number of tools for k-mer based comparisons, but none of these calculate the distance between metagenomes. We agree with the reviewer and have moved the table to supplemental.

- For Figure 2, authors created "synthetic" or "simulated" datasets and called them "artificial". Why? Authors should rather call these datasets "synthetic" or "simulated" to be consistent
with the language used by authors of GemSIM and generally language used in studies using synthetic datasets built with known profile.

RESPONSE: Thank you for pointing this out, we have updated the figures, figure legends, and text throughout the manuscript to consistently using the word “simulated”.

- Authors do show tests with 454 reads, however since this technology is not supported any more, I am afraid this evaluation brings limited value.

RESPONSE: We agree with the reviewer that 454 technology is not used as often these days, but have chosen to include 454 in addition to Illumina/Pacbio data (added in Supplemental Figure X) for the mock community analysis to show that the methodology works irrespective of the sequencing platform. This point is important for users who wish to compare new datasets with older datasets derived from 454 technologies.

- Please detail what are all the parameters for Libra's settings (for example, is the k-mer length variable ? is k equal to 21 like MASH's index ?...).

RESPONSE: We thank the reviewer for pointing this out. We have updated the methods to include information about the k-mer size and settings for Libra.

Additional Information:	
Question	**Response**
Are you submitting this manuscript to a	No

special series or article collection?	
Experimental design and statistics Full details of the experimental design and statistical methods used should be given in the Methods section, as detailed in our Minimum Standards Reporting Checklist. Information essential to interpreting the data presented should be made available in the figure legends. Have you included all the information requested in your manuscript?	Yes
Resources A description of all resources used, including antibodies, cell lines, animals and software tools, with enough information to allow them to be uniquely identified, should be included in the Methods section. Authors are strongly encouraged to cite Research Resource Identifiers (RRIDs) for antibodies, model organisms and tools, where possible. Have you included the information requested as detailed in our Minimum Standards Reporting Checklist?	Yes
Availability of data and materials All datasets and code on which the conclusions of the paper rely must be either included in your submission or deposited in publicly available repositories (where available and ethically appropriate), referencing such data using a unique identifier in the references and in the “Availability of Data and Materials” section of your manuscript. Have you have met the above requirement as detailed in our Minimum Standards Reporting Checklist?	Yes

Title: Libra: scalable k-mer based tool for massive all-vs-all metagenome comparisons.

Authors: Illyoung Choi[1], Alise J. Ponsero[2], Matthew Bomhoff[2], Ken Youens-Clark[2], John H. Hartman[1*], and Bonnie L. Hurwitz[2,3*]

Affiliations:

[1]Department of Computer Science, University of Arizona, Tucson, Arizona

[2]Department of Biosystems Engineering, University of Arizona, Tucson, Arizona

[3]BIO5 Institute, University of Arizona, Tucson, Arizona

Corresponding Author:

Bonnie L. Hurwitz bhurwitz@email.arizona.edu

ABSTRACT

Background: Shotgun metagenomics provides powerful insights into microbial community biodiversity and function. Yet, inferences from metagenomic studies are often limited by dataset size and complexity, and are restricted by the availability and completeness of existing databases. *De novo* comparative metagenomics enables the comparison of metagenomes based on their total genetic content.

Results: We developed a tool called Libra that performs all-vs-all comparison of metagenomes for precise clustering based on their k-mer content. This tool presents three main innovations: the use of a scalable Hadoop framework for massive metagenome comparisons, Cosine Similarity for calculating the distance using sequence composition and abundance while normalizing for sequencing depth, and a web-based tool in iMicrobe (http://imicrobe.us) that uses the CyVerse advanced cyberinfrastructure to promote broad use of the tool by the scientific community.

Conclusions: A comparison of Libra to equivalent tools using both simulated and real metagenomic datasets, ranging from 80 million to 4.2 billion reads, reveals that methods commonly implemented to reduce compute time for large datasets—such as data reduction, read count normalization, and presence/absence distance metrics—greatly diminish the resolution of large-scale comparative analyses. In contrast, Libra uses all of the reads to calculate k-mer abundance in a Hadoop architecture that can scale to any size dataset to enable global-scale analyses and link microbial signatures to biological processes.

Keywords: metagenomics, Hadoop, k-mer, distance metrics, clustering

INTRODUCTION

Over the last decade, scientists have generated petabytes of genomic data to uncover the role of microbes in dynamic living systems. Yet to understand the underlying biological principles that guide the distribution of microbial communities, massive ‘omics datasets need to be compared with environmental factors to find linkages across space and time. One of the greatest challenges in these endeavors has been in documenting and analyzing unexplored genetic diversity in wild microbial communities. For example, fewer than 60% of 40 million non-redundant genes from the Global Ocean Survey (GOS) and the Tara Oceans Expeditions match known proteins in bacteria [1,2]. Other microorganisms such as viruses or pico- eukaryotes that are important to ocean ecosystems are even less well defined (e.g. < 7% of reads from viromes match known proteins [3]). This is largely due to the fact that these organisms are unculturable and reference genomes do not exist in public data repositories. Thus, genome-sequences from metagenomic data await better taxonomic and functional definition. Consequently, even advanced tools such as k-mer based classifiers that rapidly assign metagenomic reads to known microbes (Supplemental Table 1) miss “microbial dark matter” that comprises a significant proportion of metagenomes [4].

***De novo* comparative metagenomics offers a path forward.** In order to examine the complete genomic content, metagenomic samples can be compared using their sequence signature (or frequency of k-mers). This approach relies on three core tenets of k-mer-based analytics: (i) closely related organisms share k-mer profiles and cluster together, making taxonomic assignment unnecessary [5,6], (ii) k-mer frequency is correlated with the abundance of an organism [7], and (iii) k-mers of sufficient length can be used to distinguish specific organisms [8]. In 2012, Compareads [9] method was proposed, followed by Commet [10]. Both of these tools compute the number of shared reads between metagenomes using a k-mer-based read similarity measure. The number of shared reads between datasets is then used to compute a Jaccard distance between samples.

Given the computational intensity of all-vs-all sequence analysis, several other methods have been employed to reduce the dimensionality of metagenomes and speed up analyses by creating unique k-mer sets and computing the genetic distance between pairs of metagenomes, such as MetaFast [11] and Mash [12]. The fastest of these methods, Mash [13], indexes samples by unique k-mers to create size-reduced sketches, and compares these sketches using the MinHash algorithm [14] for computing a genetic distance using Jaccard similarity. Yet, the tradeoff for speed is that samples are reduced to a subset of unique k-mers (1k by default) that may lead an unrepresentative k-mer profile of the samples. Further, given that Mash uses Jaccard similarity only the genetic distance between samples is accounted for (or genetic content in microbial communities) without considering abundance (dominant vs rare organisms in the sample) which is central to microbial ecology and ecosystem processes [15].

Recently, Simka[13] was developed to compute a distance matrix between metagenomes by dividing the input datasets into abundance vectors from subsets of k-mers, then rejoining the resulting abundances in a cumulative distance matrix. The methodology can be parallelized to execute the analyses on a high-performance compute cluster (HPC). Simka also provides various ecological distance metrics to let the user choose the metric most relevant to their analysis. However, the computational time varies based on the distance metric, where some distances scale linearly and other distances metrics, like Jensen-Shannon, scale quadratically as additional samples are added [13]. Moreover, Simka normalizes datasets in an all-vs-all comparison by reducing the depth of sequencing for all samples to the least common denominator, therefore decreasing the resolution of the datasets. Lastly, computing k-mer analytics using HPC is subject to reduced fault tolerance for massive datasets.

Scaling sequence analysis using big data analytics via Hadoop. Hadoop is an attractive platform for performing large-scale sequence analysis because it provides a distributed file system and distributed computation for analyzing massive amounts of data. Hadoop clusters are comprised of commodity servers so that the processing power increases as more computing resources are added. Hadoop also offers a high-level programming abstraction, called MapReduce [16] that greatly simplifies the implementation of new analytical tools, and a high-performance distributed file system (HDFS) for storing data sets. Programmers do not need specialized training in distributed systems and networking to implement distributed programs using MapReduce. Hadoop also provides fault-tolerance by default. When a Hadoop node fails, Hadoop reassigns the failed node's tasks to another node containing a redundant copy of the data those tasks were processing. This differs from HPC where schedulers track failed nodes and either restart the failed computation from the most recent checkpoint, or from the beginning if checkpointing wasn't used. Thus, using a Hadoop infrastructure ensures that computations and data are protected even in the event of hardware failures. These benefits have led to new analytic tools based on Hadoop, making Hadoop a de facto standard in large-scale data analysis. In metagenomics, the development of efficient and inexpensive high-throughput sequencing technologies has lead to a rapid increase of the amount of sequence data for studying microbes in diverse environments. However, to date only Hadoop-enabled genomic or k-mer counting tools exist, and no comparative metagenomics tools are available (Supplemental Table 1).

Existing big data algorithms compare reads to limited genomic reference data. Recent progress has been made in translating bioinformatics algorithms to big data architectures to overcome scalability issues. Thus far, these algorithms compare large-scale NGS datasets to reference genomic datasets and replace computationally intensive algorithms such as sequence alignment [19], genetic variant detection [20,21], or short read mapping [22–25] (Supplemental Table 1). For example, BlastReduce and CloudBurst are parallel sequence mapping tools based on Hadoop MapReduce [23,24]. These tools, however, implement a query-to-a-reference approach that is inefficient for all-vs-all analyses of reads from metagenomes. Other algorithms such as BioPig [26] and Bloomfish [27] generate an index of sequence data for later partial sequence search and k-mer counting using Hadoop [28] (Supplemental Table 1). Also, some of these tools adopt traditional sequence indexing techniques such as a suffix array that are inefficient in reading and indexing data in HDFS, thus reducing performance. Moreover, neither tool offers an end-to-end solution for comparing metagenomes consisting of: data distribution on a Hadoop cluster, k-mer indexing and counting, distance matrix computation, and visualization. Finally none of these tools are enabled in an advanced cyberinfrastructure where users can compute analyses in a simple web-based platform (Supplemental Table 1).

Libra: a tool for scalable all-vs-all sequence analysis in an advanced cyberinfrastructure

Here, we describe a scalable algorithm called Libra that is capable of performing all-vs-all sequence analysis using Hadoop MapReduce (SciCrunch.org tool reference ID SCR_016608). We demonstrate for the first time that Hadoop MapReduce can be applied to all-vs-all sequence comparisons of large-scale metagenomic datasets comprised of mixed microbial communities. We demonstrate that Cosine Similarity, which is widely used in document clustering and information retrieval, is a good distance metric for comparing datasets to consider genetic distance and microbial abundance simultaneously, along with widely accepted distance metrics in biology such as Bray-Curtis [30] and Jensen-Shannon [31]. We validate this distance metric using simulated metagenomes (from both short and long read technologies) to show that Libra has exceptional sensitivity in distinguishing complex mixed microbiomes. Next, we show Libra's ability to distinguish metagenomes by both community composition and abundance using 48 samples (16S rRNA and WGS) from the human microbiome project (HMP) and the simulated CAMI "toy" PacBio dataset across diverse body sites, and compare the results to Mash and Simka. Finally, we show that Libra can scale to massive global-scale datasets by examining viral diversity in 43 Tara Ocean Viromes (TOV) from the 2009-2011 Expedition [32] that represent 26 sites containing about 4.2 billion reads. We show for the first time that viral communities in the ocean are similar across temperature gradients, irrespective of their location in the ocean. The resulting data demonstrate that Libra provides accurate, efficient, and scalable computation for comparative metagenomics that can be used to discern global patterns in microbial ecology.

To promote the broad use of the Libra algorithm we developed a web-based tool in iMicrobe (http://imicrobe.us), where users can run Libra using data in their free CyVerse [33,34] account or use datasets that are integrated into the iMicrobe Data Commons. These analyses are fundamental for determining relationships among diverse metagenomes to inform follow-up analyses on microbial-driven biological processes.

DATA DESCRIPTION

Staggered mock community. We performed metagenomic shotgun sequencing on a staggered mock community obtained from the Human Microbiome Consortium (HM-277D). The staggered mock community is comprised of genomic DNA from genera commonly found on or within the human body, consisting of 1,000 to 1,000,000,000 16S rRNA gene copies per organism per aliquot. The resulting DNA was subjected to whole genome sequencing as follows. Mixtures were diluted to a final concentration of 1 nanogram/microliter and used to generate whole genome sequencing libraries with the Ion Xpress Plug Fragment Library Kit and manual #MAN0009847, revC (Thermo Fisher Scientific, Waltham, MA, USA). Briefly, 10 nanograms of bacterial DNA was sheared using the Ion Shear enzymatic reaction for 12 min and Ion Xpress barcode adapters ligated following end repair. Following barcode ligation, libraries were amplified using the manufacturer’s supplied Library Amplification primers and recommended conditions. Amplified libraries were size selected to ~ 200 base pairs using the Invitrogen E-gel Size Select Agarose cassettes as outlined in the Ion Xpress manual and quantitated with the Ion Universal Library quantitation kit. Equimolar amounts of the library were added to an Ion PI Template OT2 200 kit V3. The resulting templated beads were enriched with the Ion OneTouch ES system and quantitated with the Qubit Ion Sphere Quality Control kit (Life Technologies) on a Qubit 3.0 fluorometer (Qubit, NY, NY, USA). Enriched templated beads were loaded onto an Ion PI V2 chip and sequenced according to the manufacturer's protocol using the Ion PI Sequencing 200 kit V3 on a Ion Torrent Proton sequencer. The sequence data comprised of ~80 million reads have been deposited to the NCBI Sequence Read Archive under accession SRP115095 under project accession PRJNA397434.

Simulated data derived from the staggered mock community. The resulting sequence data from the staggered mock community (~80 million reads) were used to develop simulated metagenomes to test the effects of varying read depth, and composition and abundance of organisms in mixed metagenomes. To examine read depth (in terms of raw read counts and file size), we used the known staggered mock community abundance profile to generate a simulated metagenome using GemSim [35] of 2 million reads (454 sequencing) and duplicated the dataset 2x, 5x and 10x. We also simulated the effects of sequencing a metagenome more deeply using GemSim [35] to generate simulated metagenomes with 0.5, 1, 5, and 10 million reads based on the relative abundance of organisms in the staggered mock community. Next, we developed four simulated metagenomes to test the effect of changing the dominant organism abundance and genetic composition including: 10 million reads from the staggered mock community (mock 1), the mock community with alterations in a few abundant species (mock 2), the mock community with many alterations in abundant species (mock 3), and mock 3 with additional sequences from archaea to further alter the genetic composition (mock 4) as described in Supplemental Table 2. The same community profiles were used to generate paired-end illumina dataset (100 million reads), using GemSim (illumina v4 error model). Finally, using SimLord [36], the community profiles were used to generate simulated third generation sequencing datasets (Pacific Bioscience SMRT sequencing - 1 million reads). SimLord default parameters were used to generate those simulated datasets.

All simulated datasets are available in iMicrobe (http://imicrobe.us) under project 265.

Human microbiome 16S rRNA gene amplicons and WGS reads. Human microbiome datasets were downloaded from the NIH Human microbiome project [37] including 48 samples from 5 body sites including: urogenital (posterior fomix), gastrointestinal (stool), oral (buccal mucosa, supragingival plaque, tongue dorsum), airways (anterior nares), and skin (retroauricular crease left and right) ([See Supplemental Table 3]). Matched datasets consisting of 16S rRNA reads, WGS reads, and WGS assembled contigs were downloaded from the 16S trimmed dataset and the HMIWGS/HMASM dataset respectively. For the WGS reads dataset, the analysis was run on the paired 1 read file.

Tara ocean viromes. Tara oceans viromes were downloaded from European Nucleotide Archive (ENA) at EMBL and consisted of 43 viromes from 43 samples at 26 locations across the world's oceans collected during the Tara Oceans (2009-2012) scientific expedition (Supplemental Table 4) [32]. Metadata for the samples was downloaded from PANGAEA [38]. These samples were derived from multiple depths including: 16 surface samples (5-6 meters), 18 deep chlorophyll maximum samples (DCM; 17-148 meters), and one mesopelagic sample (791 meters). Quality control procedures were applied according to methods described by Brum and colleagues [32].

CAMI Human microbiome project toy dataset

The human microbiome project toy dataset from the Critical Assessment of Metagenome Interpretation 2nd Challenge was downloaded from https://data.cami-challenge.org/participate. This dataset is composed of 49 simulated PacBio reads from five different body sites of the human host, namely gastrointestinal tract, oral cavity, airways, skin and urogenital tract.

RESULTS AND DISCUSSION

Libra computational strategy. Libra uses Hadoop MapReduce to perform massive all-vs-all sequence comparisons between next-generation sequence (NGS) datasets. Libra uses a scalable algorithm and efficient resource usage to make all-vs-all comparisons feasible on large datasets. Hadoop allows parallel computation over distributed computing resources via its simple programming interface called *MapReduce*, while hiding much of the complexity of distributed computing (e.g. node failures) for robust fault-tolerant computation. Taking advantage of Hadoop, Libra can scale to larger input datasets and more computing resources. Furthermore, many cloud providers such as Amazon and Google offer Hadoop clusters on a pay-as-you-go basis, allowing scientists to scale their Libra computations to match their datasets and budgets.

Libra is implemented using three different MapReduce jobs — 1) k-mer histogram construction, 2) inverted index construction, and 3) distance matrix computation. Fig 1 shows a workflow of the Libra algorithm.

Figure 1. The Libra Workflow.

Libra consists of three MapReduce jobs (yellow boxes) — 1) Libra constructs a k-mer histogram of the input samples for load-balancing. The k-mer histogram of the input samples is computed in parallel by running multiple Map tasks and a Reduce task that combines their results; 2) Libra constructs the inverted index in parallel. In the Map phase, a separate Map task is spawned for every data block in the input sample files. Each Map task generates k-mers from the sequences stored in a data block then passes them to the Reduce tasks. Each Reduce task then counts k-mers it receives and produces an index chunk; 3) In the distance matrix computation, the work is split by partitioning the k-mer space in the beginning of a MapReduce job. The k-mer histogram files for input samples are loaded and the k-mer space is partitioned according to the k-mer distributions. A separate Map task is spawned for each partition to perform the computation in parallel and merged to produce the complete distance matrix.

Libra distance computation. Jaccard and Bray-Curtis distance have been extensively used to compare metagenomes based on their sequence signature [11–13]. While Mash only computes the Jaccard distance between samples, Simka and Libra implement several classical ecology distances, allowing the user to choose the best-suited distance for the considered dataset [13]. Libra provides three distance metrics — Cosine Similarity, Bray-Curtis and Jensen-Shannon. In this paper, we demonstrate Cosine Similarity as the default distance metric. This distance uses a vector space model to compute the distance between two NGS samples based on their k-mer composition and abundance, while simultaneously normalizing for sequencing depth. Cosine Similarity is widely used in document clustering and information retrieval. This distance metric was previously used to evaluate the accuracy of methods to reconstruct genomes from "virtual metagenomes" derived from 16S rRNA data based on shared KEGG orthologous gene counts [39], but has not been applied in analyzing sequence signatures between metagenomes. Libra users can also weight k-mers based on their abundance (using boolean weighting, natural weighting and logarithmic weighting) to account for differences in microbial community composition and sequencing effort as detailed below.

Cosine Similarity allows for an accurate and normalized comparison of metagenomes.

We explored the effects of varying: (1) the size of the datasets, (2) depth of sequencing, (3) the abundance of dominant microbes in the community, and (4) genetic composition of the community by adding in an entirely new organism (in our case we added archaea). We constructed simulated metagenomes and compared Libra's distance based on the Cosine Similarity against those from Mash and Simka. Simulated datasets were derived from genomic

DNA from a staggered mock community of bacteria obtained from the human microbiome consortium and sequenced deeply using the Ion Torrent sequencing platform (80 million reads, see methods).

First, we examined the effect of the size of the dataset by using GemSim [35] to obtain a simulated metagenome composed of 1 million reads (454 sequencing) from the mock community and duplicating that dataset 2x and 10x. Overall, we found that altering the size of the metagenome (by duplicating the data) had no effect on the distance between metagenomes for Mash, Simka, or Libra. In each case the distance of the duplicated datasets to the 1x mock community was less than 0.0001 (data not shown).

Because metagenomes don't scale exactly with size and instead have an increasing representation of low-abundance organisms, we created a second simulated dataset from the mock community using GemSim [35] 0.5, 1, 5, and 10 million reads (454 sequencing) to mimic the effect of reducing the sequencing. Given the abundance of organisms in the mock community, the 0.5 M read dataset is mainly comprised of dominant species. Because Simka normalizes all samples to the lowest read count, no changes between samples were measurable when using Jaccard and Bray-Curtis distances (Fig 2A). In contrast, Mash and Libra (natural weighting) take into account all of the reads in the metagenomes, therefore they measure a larger difference when you compare the smallest (0.5M read sample) and largest (10 million read sample). These results suggest that Libra (natural weighting) and Mash are appropriate for comparing datasets at different sequencing depths, whereas using Simka could lead to undesired effects.

Figure 2. Analysis of simulated metagenomes using Mash, Simka and Libra.

A. Distance to staggered mock community simulated metagenome composed of 10 million reads (mock1 10M), for simulated metagenomes of same community sequenced at various depth. Simulated metagenomes (454 sequencing) were obtained using GemSim and the known abundance profile of the staggered mock community (see Supplemental Table 2). In order to mimic various sequencing depths, the simulated metagenomes were generated at 0.5, 1, 5 or 10 million reads (noted mock1 0.5M; mock1 1M; mock1 5M; mock1V2 10M). The distances between the 4 simulated metagenomes and a 10 million read simulated metagenome (mock1 10M) were computing using Mash, Simka (Jaccard and Bray-curtis distance) and Libra (natural weighting).

B. Distance to staggered mock community simulated metagenome (mock 1), for simulated metagenomes from increasingly distant communities. The mock 1 relies on the known abundance profile from the staggered mock community. The mock 2 community profile was obtained by randomly inverting 3 species abundance from mock 1 profile. The mock 3 profile was obtained by randomly inverting 2 species abundances from mock 2 profile. Finally mock 4 profile was obtained by adding high abundance archeal genomes not present in any the other mock communities. Simulated metagenomes (454 sequencing) were generated using GemSim at 10 million reads. The distance between the mock 1 community to mock 2, mock 3, mock 4 and a replicate community (mock1 V2) was computed using Mash, Simka (Jaccard and Bray-curtis distance) and Libra (cosine distance, natural and logarithmic weighting).

In addition to natural variation in population-level abundances, artifacts from sequencing can result in high-abundance k-mers. Libra allows users to select the optimal methodology for weighting high abundance k-mers in their datasets including boolean, natural, and logarithmic. These options for weighting k-mers are important for different biological scenarios as described below and shown in simulated datasets. To examine the effect of weighting, we compared and contrasted the natural and logarithmic weight in Libra, with other distances obtained from Mash and Simka (Jaccard and Bray-Curtis). We also examined the effect of adding an entirely new species by spiking a simulated dataset with sequences derived from archaea (that were not present in the mock community). The simulated datasets (454 technology) were comprised of the staggered mock community (mock 1), the mock community with alterations in a few abundant species (mock 2), the mock community with many alterations in abundant species (mock 3), and mock 3 with additional sequences from archaea to alter the genetic composition of the community (mock 4) (see Supplemental Table 2). The resulting data showed that Libra (logarithmic weighting) shows a stepwise increase in distance among the mock communities (Fig 2B). This suggests that logarithmic weighting in Libra allows for a comparison of distantly related microbial communities. Mash also shows a stepwise distance between communities, but is compressed relative to Libra, making differences less distinct. Simka (Bray-Curtis and Jaccard) and Libra (cosine distance, natural weighting) reach the maximum difference between mock communities 3 and 4 (Fig 2B). This indicates that these distances are more appropriate when comparing metagenomes with small fluctuations in the community (e.g., data from a time-series analysis), whereas Libra (cosine distance, logarithmic weighting) can be used to distinguish metagenomes that vary in both genetic composition and abundance over a wide-range of species diversity by dampening the effect of high-abundance k-mers. Because of this important difference, we used the cosine distance with the logarithmic weighting in all subsequent analyses. Further, we also found that cosine distance provides the fastest computation among all distance metrics (see Methods). We confirmed these findings using Illumina simulated datasets (Supplemental Figure 1A), to show that these results are consistent across short read technologies.

Given the availability of long read (~10K) sequencing technologies like Oxford Nanopore and PacBio sequencing, we repeated the analyses above on simulated long read data (Supplemental Figure 1B). We show that simulated PacBio long read data for the mock community derived from SimLoRD [36] shows a similar stepwise distance pattern between each of the mock communities (Supplemental Figure 1B), but has a higher overall distance between mock 1 and each of the mock communities (mock 2 - 4) likely due to the high simulated random error rate compared to simulated short read data.

Libra accurately profiles differences in bacterial diversity and abundance in amplicon and WGS datasets from the human microbiome.

Microbial diversity is traditionally assessed using two methods: the 16S rRNA gene to classify bacterial and archaeal groups at the genus to species level, or whole genome shotgun sequencing (WGS) for finer taxonomic classification at the species or subspecies level. Further, WGS datasets provide additional information on functional differences between metagenomes. Here we compare and contrast the effect of different algorithmic approaches (Mash vs Libra vs Simka), distance metric (Libra vs Simka), data type (16S rRNA vs WGS), and sequence type (WGS reads vs assembled contigs) in analyzing data from 48 samples across 8 body sites from the Human Microbiome Project. Specifically, we examine matched datasets (16S rRNA reads, WGS reads, and WGS assembled contigs) classified as urogenital (posterior fomix), gastrointestinal (stool), oral (buccal mucosa, supragingival plaque, tongue dorsum), airways (anterior nares), and skin (retroauricular crease left and right) ([See Supplemental Table 2]).

Because the HMP datasets represent microbial communities, abundant bacteria will have more total read counts than rare bacteria in the samples. Thus, each sample can vary by both taxonomic composition (the genetic content of taxa in a sample) and abundance (the relative proportion of those taxa in the samples). Importantly, the 16S rRNA amplicon dataset is useful in showing how well each algorithm performs in detecting and quantifying small-scale variation for single a gene at the genus-level, whereas the WGS dataset demonstrates the effect of including the complete genetic content and abundance of organisms at the species-level in a community [40]. Also, we examine differences in each algorithm when read abundance is excluded using assembled contigs that only represent the genetic composition of the community.

Using the 16S rRNA reads, both Mash and Libra clustered samples by broad categories but not individual body-sites (Fig 3A and B). Similar to what is described in previous work [13], samples from the airways and skin co-cluster, whereas other categories including urogenital, gastrointestinal, and oral are distinct [13]. These results indicate that limited variation in the 16S rRNA gene may only allow for clustering for broad categories. Further, the Mash algorithm shows lower overall resolution (Fig 3A) as compared to Libra (Fig 3B). Indeed, amplicon sequencing analysis is not an original intended use of Mash, given that it reduces the dimensionality of the data by looking at presence/absence of unique k-mers, whereas Libra examines the complete dataset accounting for both the genetic composition of organisms and their abundance. In contrast, Simka (Jaccard-ab and Bray-Curtis) fails to cluster samples by broad categories: some skin samples are found associated with stool and formix samples (Fig 3C and D). Moreover, Simka Jaccard-ab fails to cluster the mouth samples together (Fig 3C). This result suggests that applying Simka and these well-used distance metrics are not appropriate for these datasets.

Figure 3. Clustering of HMP 16S rRNA datasets using Mash, Libra and Simka.

48 Human metagenomic samples from the HMP projects clustered by Mash (A), Libra (B) or Simka using Jaccard-ab (C) and Bray-Curtis distances (D) from 16S rRNA sequencing runs. The samples were clustered using Ward's method on their distance scores. Mash, Simka, and Libra report distance in the same range (0-1). Heat maps showing the pairwise dissimilarity between samples were therefore scaled between 0 (green) and 1 (red). A key below the heatmap colors the samples by body sites.

When using WGS reads, both Mash and Libra show enhanced clustering by body-site (Fig 4A and B), however Mash shows decreased resolution (Fig 4A) as compared to Libra (Fig 4B). Again, these differences reflect the effect of using all of the read data (Libra) rather than a subset (Mash). The effect of using all of the read data compared to a subset (when sketching in Mash) has been previously described in Benoit *et al*. [13]. Importantly, the Libra algorithm depends on read abundance that provides increased resolution for interpersonal variation as seen in skin samples (Fig 4B). Similar to the 16S rRNA datasets, Simka (Jaccard-ab and Bray-Curtis) failed to cluster the samples by body site, where some skin and stool samples cluster with formix samples (Fig 4C and D). Similarly, Simka Jaccard-ab also fails to cluster the mouth samples together (Fig 4C). Overall Simka shows an enhanced clustering by body-site using WGS data compared to the 16S rRNA data using these distance metrics, however the clustering is still not accurate. In order to confirm the independence of these result toward the sequencing technology, we performed the same experiment on the *CAMI HMP* “toy dataset” (simulated PacBio long reads) [Supplemental Figure 2]. This analysis shows that each of the tools is able to cluster the samples broadly by body site. However there are small misclassifications shared across all tools, suggesting that the increased error rate for this technology could have a limited impact on k-mer based analytics.

Figure 4. Clustering of WGS samples using Mash, and Libra and Simka.

48 Human metagenomic samples from the HMP projects clustered by Mash (A), Libra (B) or Simka using Jaccard-ab (C) and Bray-Curtis distances (D) from whole genome shotgun sequencing runs. The samples were clustered using Ward’s method on their distance scores. Heat maps illustrate the pairwise dissimilarity between samples, scaled between 0 (green) and 1 (red). A key below the heatmap colors the samples by body sites.

When abundance is taken out of the equation by using assembled contigs ([See Supplemental Figure 3]) Mash performs well in clustering distinct body sites whereas Libra shows discrepancies and less overall resolution. Thus, as designed Libra requires reads rather than contigs to perform accurately and obtain high-resolution clustering (Fig 4). Simka (Jaccard-ab and Bray-Curtis) was not able to distinguish any assembled datasets and scored all sample-to-sample distances to the maximum, even considering presence-absence distance metric proposed by Simka (data not shown). This phenomenon may be explained by the normalization method used by Simka, which does not provide enough data to compare the samples when normalized by the smallest number of contigs (in our dataset 69 contigs).

Libra allows for ecosystem-scale analysis: clustering the Tara ocean viromes to unravel global patterns.

To demonstrate the scale and performance of the Libra algorithm, we analyzed 43 Tara Ocean Viromes (TOV) from the 2009-2011 Expedition [32] representing 26 sites, 43 samples, and 4.2 billion reads from the global ocean (see Methods). Phages (viruses that infect bacteria) are abundant in the ocean [41] and can significantly impact environmental processes through host mortality, horizontal gene transfer, and host-gene expression. Yet, how phages change over space and time in the global ocean and with environmental fluxes is just beginning to be explored. The primary challenge is the majority of reads in viromes (often > 90%) do not match known proteins or viral genomes [3] and no conserved genes like the bacterial 16S rRNA gene exist to differentiate populations. To examine known and unknown viruses simultaneously, viromes are best compared using sequence signatures to identify common viral populations. Two approaches exist to cluster viromes based on sequence composition. The first approach uses protein clustering to examine functional diversity in viromes between sites [3,32,42]. Protein clustering, however, depends on accurate assembly and gene finding that can be problematic in fragmented and genetically diverse viromes [43]. Further, assemblies from viromes often include only a fraction of the total reads (e.g., only ⅓ in TOV [32]). To examine global viral diversity in the ocean using all of the reads we examined TOV using Libra. The complete pairwise analysis of ~4.2 billion reads in the TOV dataset [32] finished in 18 hours using a 10-node Hadoop cluster (see Methods and Table 2). Importantly, Libra exhibits remarkable performance in computing the distance matrix, wherein k-mer matches for all TOV completed within 1.5 hours (see Table 1). This step usually represents the largest computational bottleneck for bioinformatics tools that compute pairwise distances between sequence pairs for applications such as hierarchical sequence clustering [44–47]. A direct comparison of the runtime of the Simka, Mash and Libra is not possible given that each tool is tuned to a different computational architecture with a different number of servers and total CPU/memory (Mash runs on a single server; Simka runs on an HPC; and Libra on Hadoop).

Table 1. Execution times for the Libra based on the Tara Ocean Virome (TOV) dataset.

Stage	Execution Time
Preprocessing (k-mer histogram construction / Inverted index construction)	16:32:55
Distance matrix computation	1:24:27
Total	17:57:22

Overall, we found that viral populations in the ocean are largely structured by temperature in four gradients (Fig 5) similar to their bacterial hosts [2]. Interestingly, samples from different Longhurst Provinces but the same temperature gradient cluster together. Also, water samples from the surface (SUR) and deep chlorophyll maximum (DCM) at the same station, cluster more closely together than samples from the same depth at nearby sites (Fig 5). Also noteworthy, samples that were derived from extremely cold environments (noted as C0 in Fig 5) lacked similarity to all other samples (at a 30% similarity score), indicating distinctly different viral populations. These samples include a mesotrophic sample that have previously been shown to have distinctly different viral populations than surface ocean samples [48]. Taken together, these data indicate that viral populations are structured globally by temperature, and at finer resolution by station (for surface and DCM samples) indicating that micronutrients and local conditions play an important role in defining viral populations.

Figure 5. Visualizing the genetic distance among marine viral communities using Libra. Similarities between samples from 43 TOV from the 2009-2012 Tara Oceans Expedition. Lines (edges) between samples represent the similarity and are colored and thickened accordingly. Lines with insignificant similarity (less than 30%) are removed. Each of the sample names are color coded by Longhurst Province. Inner circles show temperature ranges. Sample names show the temperature range, station, and depth as indicated on the legend. The analysis is performed using Libra (k=20, Logarithmic weighting and Cosine Similarity).

INNOVATIONS

Scientific collaboration is increasingly data driven given large-scale next generation sequencing datasets. It is now possible to generate, aggregate, archive, and share datasets that are terabytes and even petabytes in size. Scalability of a system is becoming a vital feature that decides feasibility of massive ‘omic’s analyses. In particular, this is important for metagenomics where patterns in global ecology can only be discerned by comparing the sequence signatures of microbial communities from massive ‘omics datasets, given that most microbial genomes have not been defined. Current algorithms to perform these tasks run on local workstations or high-performance computing architectures. Libra presents three main innovations: the use of a scalable Hadoop framework enabling massive dataset comparison, linear calculations for complex distance metrics allowing for high accuracy and clustering of the metagenomes based on their k-mer content, and a web-based tool imbedded in the CyVerse advanced cyberinfrastructure through iMicrobe (http://imicrobe.us) for broader use of the tool in the scientific community. The work described here is the first step in implementing a cloud-based resource for comparative metagenomics that can be broadly used by scientists to analyze large-scale shared data resources. Moreover, the code can be ported to any Hadoop cluster (e.g., Wrangler at TACC, Amazon EMR, or private Hadoop clusters). This computing paradigm is consistent with recent efforts to increase the accessibility of big datasets in the cloud, such as the Pan Cancer Analyses of Whole Genomes Project [49].

METHODS

Libra Algorithm Detailed Description:

k-mer size. Libra calculates the distances between samples based on their k-mer composition. Canonical representation of the k-mer is used to reduce the number of stored k-mers**.** Several considerations should be taken into account for choosing the k-mer size *k*. Larger values of *k* result in fewer matches due to sequencing errors and fragmentary metagenomic data. However, smaller values of *k* give less information about the sequence similarities. In Libra, *k* is a configurable parameter chosen by the user, and is set by default to *k* equal to 21. This value was reported to be at the inflection point where the k-mer matches move from random to representative of the read content, and is generally resilient to sequencing error and variation [50,51].

Distance Matrix Computation. Libra provides three distance metrics — Cosine Similarity, Bray-Curtis and Jensen-Shannon. Cosine Similarity is the default.

Cosine Similarity metric. Libra constructs a vector V_s for each sample *s* from the weight of each k-mer *k* in the sample ($w_{k.s}$). Each dimension in the vector corresponds to the weight of the corresponding k-mer:

$$V_s = (w_{1,s}, w_{2,s}, w_{3,s}, \ldots, w_{n,s})$$

The weight of a k-mer in a sample (w_{ks}) can be derived from frequency of the k-mer (f_{ks}) in several ways. The simplest uses the raw frequency of the k-mer ($w_{ks} = f_{ks}$), called *Natural Weighting*. Another uses *Logarithmic Weighting* ($w_{ks} = 1 + \log(f_{ks})$) to not give too much weight to highly abundant k-mers. In this weighting w_{ks} grows logarithmically with the frequency

$\square_{\square\square}$, reducing the effect on the distance of highly abundant k-mers caused by sequencing artifacts.

Once their vectors have been constructed, the distance between two samples ($\square_1$ and $\square_2$) is derived using distance metrics. For example, the distance between the two samples using Cosine Similarity is determined as following:

$$\square\square\square\square\square\square\square\square(\square_1, \square_2) = 1 - Cosine\square\square\square\square\square\square\square\square\square\square(\square_1, \square_2)$$

$$= 1 - \square\square\square(\square_{\square 1}, \square_{\square 2}) = 1 - \frac{\square_{\square 1} \cdot \square_{\square 2}}{||\square_{\square 1}|| \times ||\square_{\square 2}||} = 1 - \frac{\square_{\square 1,\square 2}}{\square_{\square 1} \times \square_{\square 2}}$$

$$\square\square\square\square\square, \square_{\square 1,\square 2} = \square_{\square 1} \cdot \square_{\square 2} = \sum_{\square \in \square 1 \cap \square 2} \square_{\square\square,\square 1} \times \square_{\square\square,\square 2},$$

$$\square_{\square} = ||\square_{\square}|| = \sqrt{\sum_{\square \in \square} (\square_{\square\square,\square})^2}$$

In other words, $\square_{\square 1,\square 2}$ is the dot product of the vectors $\square_{\square 1}$and $\square_{\square 2}$, and $\square_{\square}$ is the magnitude (length) of the vector $\square_{\square}$. The distance between two NGS samples is the cosine of the angle between their vectors $\square_{\square}$; the magnitude of the vector $\square_{\square}$ is not taken into account in the metric thereby normalizing samples with different numbers of total base pairs.

Inverted Index Construction. A naïve implementation would require the storage of one vector with 4^k dimensions per sample, where *k* is he k-mer length. For a k of 21, each vector would have more than one million dimensions. To reduce the overhead, Libra stores and computes the distance on a single *inverted index* with the k-mer frequencies from multiple samples and performs the distance computation on the index directly. The inverted index is indexed by k-mer, and each entry is an index record containing a list of pairs, each of which contains a sample identifier and the frequency of the k-mer in the sample.

$$\square\square\square\square\square\ \square\square\square\square\square\square = \square\_\square\square\square : \{< \square\square\square\square\square\square\_\square\square, \square\square\square\square\square\square\square\square\square >, < \square\square\square\square\square\square\_\square\square, \square\square\square\square\square\square\square\square\square > \ldots\}$$

The records in the index are stored in an alphabetical order by k-mer, allowing the record for a particular k-mer to be found via binary search. The k-mer record contains the k-mer frequency in each sample, not the weight, to allow for different weighting functions to be applied during distance matrix computation.

Sweep line algorithm. To compute the distance between two samples $\square_1$ and $\square_2$, Libra must compute the three values $\square_{\square 1,\square 2}$, $\square_{\square 1}$, *and* $\square_{\square 2}$*.* The values are calculated by scanning through the vectors $\square_{\square 1}$and $\square_{\square 2}$ and computing the values. The time for the distance matrix computation is proportional to the number of dimensions (the number of k-mers) in the two vectors. In general, computing all-vs-all comparisons on n samples would require $\square \times (\square - 1)/2$ vector scans, which becomes prohibitively expensive as *n* gets large. Libra uses a sweep line algorithm [38] to greatly reduce the computational time. The sweep line algorithm only requires a single scan of all vectors to compute the distance of all pairs of samples ([See Supplemental Figure 4]). Briefly, Libra sweeps a line through all the vectors simultaneously starting with the first component. Libra outputs a record of the non-zero values of the following format:

$$\square\square\square\square\square\square = \square\_\square\square\square : \{< \square\square\square\square\square\square\_\square\square, \square\square\square\square\square\square >, < \square\square\square\square\square\square\_\square\square, \square\square\square\square\square\square >, \ldots\}$$

Libra then moves the sweep line to the next component and performs the same operation. From the output records, contributions to $\square_{\square}$ for each sample in the record are computed and accumulated. Contributions to $\square$ are also computed from the record by extracting sample pairs. For example, the record $\{< \square_1, \square >, < \square_2, \square >, < \square_4, \square >\}$ has three sample pairs $(\square_1\square_2), (\square_1\square_4)$ $\square\square\square$ $(\square_2\square_4)$. Libra then computes contribution to $\square$ for each pair, e.g. $\square * \square$ is added to $\square_{\square 1,\square 2}$, $\square * \square$ is added to $\square_{\square 1,\square 4}$, and $\square * \square$ is added to $\square_{\square 2,\square 4}$. Using this method, Libra computes the distances of every sample pairs in an input dataset in linear time. Other distance metrics, such as Bray-Curtis and Jensen-Shannon, can also be computed in the same fashion.

The sweep algorithm is particularly easy to implement on an inverted index; it consists of simply stepping through the (sorted) k-mers. Furthermore, the sweep algorithm is easily parallelized. The k-mer space is partitioned and a separate sweep is performed on each partition computing the contributions of its k-mer frequencies to the □ and □ values. At the end of the computation the intermediate □ and □ values are combined together to produce the final □ and □ values and thereby the distance matrix. Each sweep uses binary search to find the first k-mer in the partition.

Terabyte Sort. Libra groups samples automatically based on the number and size (by default 4GB per group). Similar to Terabyte Sort [52], the index records are partitioned by k-mer ranges and the records in each partition is stored in a separate *chunk file*. All k-mers in partition □ appear before the k-mers in partition□ $+ 1$ in lexicographic order. This facilitates breaking computation and I/O down into smaller tasks, so that work of creating an index can be distributed across several machines.

k-mer space partitioning. Both the inverted index construction and the distance matrix computation require partitioning the k-mer space so that different partitions can be processed independently. For the partitioning to be effective, the workload should be balanced across the partitions. Simply partitioning into fixed-size partitions based on the k-mer space will not ensure balanced workloads, as the k-mers do not appear with uniform frequency. Some partitions may have more k-mer records than others, and thereby incur higher processing costs. Instead, the partitions should be created based on the k-mer distribution, so that each partition has roughly the same number of records ([See Supplemental Figure 5]).

Computing the exact k-mer distribution across all the samples is too expensive in both space and time, therefore Libra approximates the distribution instead. A histogram is constructed using the first 6 letters of the k-mers in each sample, which requires much less space and time to compute. In practice, partitioning based on this histogram adequately partitions the k-mer space so that the workloads are sufficiently balanced across the partitions.

Scalability benchmarking for Libra. We used synthetic datasets for a scalability benchmark. Each dataset contains 10 billion bytes (approximately 9.3 GB). We used four datasets consisting of 10 (93GB), 20 (186GB), 30 (279GB) and 40 (372GB) samples in the benchmark. Each experiment was run three times, and an average of the three runs reported ([See Supplemental Table 4 for details]). The runtime of Libra increased linearly with increased input volume (Figure 6). This shows that Libra efficiently handles increased volume of input and efficiently computes distances between all sample pairs while the number of sample pairs increases quadratically.

Figure 6 Scalability testing for Libra. Runtimes of Libra on four datasets consisting of 10, 20, 30 and 40 samples (total sizes of 93GB, 186GB, 279GB and 372GB, respectively). Libra was performed with default parameters (k=20, Logarithmic weighting and Cosine Similarity). Runtimes were averaged out over 3 runs. The total runtime of Libra increased linearly with increased input volume. Both index construction and distance matrix computation showed linearly increased runtimes for the increased input volume. This shows that Libra performs efficiently and scales to input although the number of distances between sample pairs to be computed increases quadratically.

Benchmarking runtimes of different distance metrics in Libra. We used the same synthetic dataset with 40 samples (372GB in total) in the scalability benchmarking (Figure 7). We measured the runtimes of Libra for the different distance metrics. Once the index is constructed all distance metrics are calculated using that index; thus, runtimes of the inverted index construction for the different metrics are the same. Each experiment was run three times and the average reported ([See Supplemental Table 4 for details]). Differences in runtimes are mainly due to different computational workload of distance metrics (Figure 7). For example, Jensen-Shannon requires more multiplications and divisions in nested loops than Cosine Similarity, incurring more computational workload. Yet, distance matrix computation with Jensen-Shannon took only 12.64% of total runtime.

Figure 7. **Runtime for different distance metrics**. Runtimes for three different distance metrics (Cosine Similarity, Bray-Curtis and Jensen-Shannon) in Libra with 40 samples of input (372GB in total). Libra was performed with default parameters (k=20 and Logarithmic weighting). Runtimes were averaged over 3 runs. An inverted index was reused for all three distance metrics because the inverted index Libra constructs is independent of the distance metrics. Cosine Similarity took the shortest runtime among the three metrics while Jensen-Shannon took the longest. Jensen-Shannon took almost twice as long as Cosine Similarity because it requires more mathematical computations. Because of its fastest runtime, Cosine Similarity is used by default in Libra.

Advanced cyberinfrastructure for Libra in iMicrobe. To improve access to Libra we made it available on the iMicrobe website (https://www.imicrobe.us). A researcher with a CyVerse account can run Libra on iMicrobe by filling-out a simple web form specifying the input files and parameters. Input files are selected from the CyVerse Data Store where they have either been uploaded by the user to their home directory or are part of the iMicrobe Data Commons. When a job is submitted, the user is presented with the status of the job, and on completion the output files and visualization of results. To deploy Libra on iMicrobe, we developed a job dispatch service to automate execution of Libra on a University of Arizona Hadoop cluster. The service is written in NodeJS and accepts a JSON description of the job inputs and parameters, stages the input files onto the UA Hadoop cluster, executes Libra with the given parameters, and transfers the resulting output files to the user's home directory in the CyVerse Data Store. The service provides a RESTful interface that mimics the Agave API Jobs service and is secured using an Agave OAuth2 token. Source code is located at https://github.com/hurwitzlab/occ-plan-b.

Experimental Environment Description:

Mash and Simka configurations. Mash v1.1 was run on the metagenomic datasets with the following parameters: -r –s 10000 –m 2 [19]. The analysis of assemblies was run without the parameter "-r", used for short sequences.

Simka v1.3.2 was run on the metagenomic datasets with the following parameters: -abundance-min 2 -max-reads [MINCOUNT] -simple-dist -complex-dist, where [MINCOUNT] is the smallest sequence count across the analysed samples.

Hadoop cluster configuration. The Libra experiments described in the paper were performed on a Hadoop cluster consisting of 10 physical nodes (9 MapReduce worker nodes). Each node contains 12 CPUs and 128 GB of RAM, and is configured to run a maximum of 7 YARN containers simultaneously with 10 GB of RAM per container. The remaining system resources are reserved for the operating system and other Hadoop services such as Hive or Hbase.

Rationale for not porting Libra to Spark. Spark [17] is increasingly popular for scientific data analysis [18] because of its outstanding performance provided by fast in-memory processing. Although Libra is currently implemented on Hadoop MapReduce, Libra can be easily ported to Spark because both Hadoop MapReduce and Spark have similar interfaces for data processing and partitioning. For example, Resilient Distributed Datasets (RDD) can be partitioned and distributed over a Spark cluster using Libra's k-mer range partitioning. RDDs are memory-resident, allowing Spark to significantly improve the performance of Libra's k-mer counting and distance matrix computation by avoiding slow disk I/O for intermediate data. We implemented Libra using Hadoop MapReduce because Spark requires much more RAM than Hadoop MapReduce, significantly increasing the cost of the cluster.

FUNDING

This work was supported by the National Science Foundation award #1640775 to BLH and JHH. System support and access for the Hadoop cluster was provided by University of Arizona Information Technology Services. The following reagent was obtained through BEI Resources, NIAID, NIH as part of the Human Microbiome Project: Genomic DNA from Microbial Mock Community B (Staggered, High Concentration), v5.2H, for Whole Genome Shotgun Sequencing, HM-277D.

Competing interests: The authors declare no competing interests.

Availability and Implementation:
Project home page: Program binary, source code and documentation for Libra are available in Github (https://www.github.com/iychoi/Libra); Libra web-based App is in iMicrobe under Apps (http://imicrobe.us); code to implement the Libra web-based App is in Github (https://github.com/hurwitzlab/occ-plan-b); **Operating system(s):** MapReduce 2.0 (Apache Hadoop 2.3.0 or above); **Programming language:** Java 7 (or above); **Other requirements:** none; **License:** Apache License Version 2.0; **Any restrictions to use by non-academics:** no license needed. Libra has been registered with the SciCrunch.org database under reference ID: SCR_016608.

REFERENCES

1. Yooseph S, Sutton G, Rusch DB, Halpern AL, Williamson SJ, Remington K, et al. The Sorcerer II Global Ocean Sampling expedition: expanding the universe of protein families. PLoS Biol. 2007;5:e16.

2. Sunagawa S, Coelho LP, Chaffron S, Kultima JR, Labadie K, Salazar G, et al. Structure and function of the global ocean microbiome. Science [Internet]. sciencemag.org; 2015;348.

Available from: http://www.sciencemag.org/content/348/6237/1261359.abstract

3. Hurwitz BL, Sullivan MB. The Pacific Ocean Virome (POV): a marine viral metagenomic dataset and associated protein clusters for quantitative viral ecology. PLoS One. 2013;8:e57355.

4. Morgan XC, Huttenhower C. Chapter 12: Human microbiome analysis. PLoS Comput Biol. journals.plos.org; 2012;8:e1002808.

5. Dubinkina VB, Ischenko DS, Ulyantsev VI, Tyakht AV, Alexeev DG. Assessment of k-mer spectrum applicability for metagenomic dissimilarity analysis. BMC Bioinformatics; 2016;17:38.

6. Teeling H, Waldmann J, Lombardot T, Bauer M, Glockner FO. TETRA: a web-service and a stand-alone program for the analysis and comparison of tetranucleotide usage patterns in DNA sequences. BMC Bioinformatics. 2004;5:163.

7. Wu Y-W, Ye Y. A novel abundance-based algorithm for binning metagenomic sequences using l-tuples. J Comput Biol. online.liebertpub.com; 2011;18:523–34.

8. Fofanov Y, Luo Y, Katili C, Wang J, Belosludtsev Y, Powdrill T, et al. How independent are the appearances of n-mers in different genomes? Bioinformatics. Oxford University Press; 2004;20:2421–8.

9. Maillet N, Lemaitre C, Chikhi R, Lavenier D, Peterlongo P. Compareads: comparing huge metagenomic experiments. BMC Bioinformatics. bmcbioinformatics.biomedcentral. …; 2012;13 Suppl 19:S10.

10. Maillet N, Collet G, Vannier T, Lavenier D, Peterlongo P. Commet: Comparing and combining multiple metagenomic datasets. 2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). ieeexplore.ieee.org; 2014. p. 94–8.

11. Ulyantsev VI, Kazakov SV, Dubinkina VB, Tyakht AV, Alexeev DG. MetaFast: fast reference-free graph-based comparison of shotgun metagenomic data. Bioinformatics. Oxford Univ Press; 2016;32:2760–7.

12. Ondov BD, Treangen TJ, Melsted P, Mallonee AB, Bergman NH, Koren S, et al. Mash: fast genome and metagenome distance estimation using MinHash. Genome Biol. biorxiv.org; 2016;17:132.

13. Benoit G, Peterlongo P, Mariadassou M, Drezen E, Schbath S, Lavenier D, et al. Multiple comparative metagenomics using multiset k-mer counting. PeerJ Comput Sci. PeerJ Inc.; 2016;2:e94.

14. Broder AZ. On the resemblance and containment of documents. Proceedings Compression and Complexity of SEQUENCES 1997 (Cat No97TB100171) [Internet]. Available from: http://dx.doi.org/10.1109/sequen.1997.666900

15. Koslicki D, Zabeti H. Improving Min Hash via the Containment Index with applications to Metagenomic Analysis [Internet]. bioRxiv. 2017 [cited 2018 Oct 19]. p. 184150. Available from: https://www.biorxiv.org/content/early/2017/09/04/184150.abstract

16. Dean J, Ghemawat S. MapReduce: Simplified Data Processing on Large Clusters. Commun ACM. New York, NY, USA: ACM; 2008;51:107–13.

17. Zaharia M, Chowdhury M, Franklin MJ, Shenker S, Stoica I. Spark: Cluster computing with working sets. HotCloud. static.usenix.org; 2010;10:95.

18. Guo R, Zhao Y, Zou Q, Fang X, Peng S. Bioinformatics applications on Apache Spark. Gigascience [Internet]. academic.oup.com; 2018; Available from: http://dx.doi.org/10.1093/gigascience/giy098

19. Kolker N, Higdon R, Broomall W, Stanberry L, Welch D, Lu W, et al. Classifying proteins into functional groups based on all-versus-all BLAST of 10 million proteins. OMICS. online.liebertpub.com; 2011;15:513–21.

20. McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, et al. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. Genome Res. genome.cshlp.org; 2010;20:1297–303.

21. Langmead B, Schatz MC, Lin J, Pop M, Salzberg SL. Searching for SNPs with cloud computing. Genome Biol. biomedcentral.com; 2009;10:R134.

22. Nguyen T, Shi W, Ruden D. CloudAligner: A fast and full-featured MapReduce based tool for sequence mapping. BMC Res Notes. biomedcentral.com; 2011;4:171.

23. Schatz MC. BlastReduce: high performance short read mapping with MapReduce. University of Maryland, http://cgis cs umd edu/Grad/scholarlypapers/papers/MichaelSchatz pd f [Internet]. cs.umd.edu; 2008; Available from: https://www.cs.umd.edu/sites/default/files/scholarly_papers/MichaelSchatz_1.pdf

24. Schatz MC. CloudBurst: highly sensitive read mapping with MapReduce. Bioinformatics. Oxford Univ Press; 2009;25:1363–9.

25. Pandey RV, Schlötterer C. DistMap: a toolkit for distributed short read mapping on a Hadoop cluster. PLoS One. dx.plos.org; 2013;8:e72614.

26. Nordberg H, Bhatia K, Wang K, Wang Z. BioPig: a Hadoop-based analytic toolkit for large-scale sequence data. Bioinformatics. Oxford Univ Press; 2013;29:3014–9.

27. Gao T, Guo Y, Wei Y, Wang B, Lu Y, Cicotti P, et al. Bloomfish: A Highly Scalable Distributed K-mer Counting Framework. 2017 IEEE 23rd International Conference on Parallel and Distributed Systems (ICPADS). ieeexplore.ieee.org; 2017. p. 170–9.

28. Menon RK, Bhat GP, Schatz MC. Rapid Parallel Genome Indexing with MapReduce. Proceedings of the Second International Workshop on MapReduce and Its Applications. New York, NY, USA: ACM; 2011. p. 51–8.

29. Huang A. Similarity measures for text document clustering. Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand. academia.edu; 2008. p. 49–56.

30. Michie MG. Use of the Bray-Curtis similarity measure in cluster analysis of foraminiferal data. Math Geol. Kluwer Academic Publishers-Plenum Publishers; 1982;14:661–7.

31. Lin J. Divergence measures based on the Shannon entropy. IEEE Trans Inf Theory. 1991;37:145–51.

32. Brum JR, Ignacio-Espinoza JC, Roux S, Doulcier G, Acinas SG, Alberti A, et al. Patterns and ecological drivers of ocean viral communities. Science [Internet]. sciencemag.org; 2015;348. Available from: http://www.sciencemag.org/content/348/6237/1261498.abstract

33. Goff SA, Vaughn M, McKay S, Lyons E, Stapleton AE, Gessler D, et al. The iPlant Collaborative: Cyberinfrastructure for Plant Biology. Front Plant Sci. ncbi.nlm.nih.gov; 2011;2:34.

34. Devisetty UK, Kennedy K, Sarando P, Merchant N, Lyons E. Bringing your tools to CyVerse Discovery Environment using Docker. F1000Res. 2016;5:1442.

35. McElroy KE, Luciani F, Thomas T. GemSIM: general, error-model based simulator of next-generation sequencing data. BMC Genomics. biomedcentral.com; 2012;13:74.

36. Stöcker BK, Köster J, Rahmann S. SimLoRD: Simulation of Long Read Data. Bioinformatics. academic.oup.com; 2016;32:2704–6.

37. Human Microbiome Project Consortium. Structure, function and diversity of the healthy human microbiome. Nature. nature.com; 2012;486:207–14.

38. Diepenbroek M, Grobe H, Reinke M, Schindler U, Schlitzer R, Sieger R, et al. PANGAEA—an information system for environmental sciences. Comput Geosci. Elsevier; 2002;28:1201–10.

39. Okuda S, Tsuchiya Y, Kiriyama C, Itoh M, Morisaki H. Virtual metagenome reconstruction from 16S rRNA gene sequences. Nat Commun. nature.com; 2012;3:1203.

40. Watts GS, Youens-Clark K, Slepian MJ, Wolk DM, Oshiro MM, Metzger GS, et al. 16S rRNA gene sequencing on a benchtop sequencer: accuracy for identification of clinically important bacteria. J Appl Microbiol. Wiley Online Library; 2017;123:1584–96.

41. Bergh O, Borsheim KY, Bratbak G, Heldal M. High abundance of viruses found in aquatic environments. Nature. 1989;340:467–8.

42. Hurwitz BL, Brum JR, Sullivan MB. Depth-stratified functional and taxonomic niche specialization in the “core”and “flexible”Pacific Ocean Virome. ISME J [Internet]. nature.com; 2014; Available from: http://www.nature.com/ismej/journal/vaop/ncurrent/full/ismej2014143a.html

43. Minot S, Wu GD, Lewis JD, Bushman FD. Conservation of gene cassettes among diverse viruses of the human gut. PLoS One. 2012;7:e42342.

44. Sun Y, Cai Y, Huse SM, Knight R, Farmerie WG, Wang X, et al. A large-scale benchmark study of existing algorithms for taxonomy-independent microbial community analysis. Brief Bioinform. Oxford Univ Press; 2012;13:107–21.

45. Edgar RC. Search and clustering orders of magnitude faster than BLAST. Bioinformatics. 2010;26:2460–1.

46. Niu BF, Fu LM, Sun SL, Li WZ. Artificial and natural duplicates in pyrosequencing reads of metagenomic data. BMC Bioinformatics. 2010;11:187.

47. Cai Y, Sun Y. ESPRIT-Tree: hierarchical clustering analysis of millions of 16S rRNA pyrosequences in quasilinear computational time. Nucleic Acids Res. Oxford Univ Press;

2011;39:e95.

48. Hurwitz BL, Brum JR, Sullivan MB. Depth-stratified functional and taxonomic niche specialization in the “core” and “flexible” Pacific Ocean Virome. ISME J. 2015;9:472–84.

49. Weinstein JN, Collisson EA, Mills GB, Shaw KRM, Ozenberger BA, Ellrott K, et al. The cancer genome atlas pan-cancer analysis project. Nat Genet. Nature Publishing Group; 2013;45:1113–20.

50. Hurwitz BL, Westveld AH, Brum JR, Sullivan MB. Modeling ecological drivers in marine viral communities using comparative metagenomics and network analyses. PNAS. 2014;111:10714–9.

51. Kurtz S, Narechania A, Stein JC, Ware D. A new method to compute K-mer frequencies and its application to annotate large repetitive plant genomes. BMC Genomics. 2008;9:517.

52. O’Malley O. Terabyte sort on apache hadoop. Yahoo, available online at: http://sortbenchmark org/Yahoo-Hadoop pdf,(May). Citeseer; 2008;1–3.

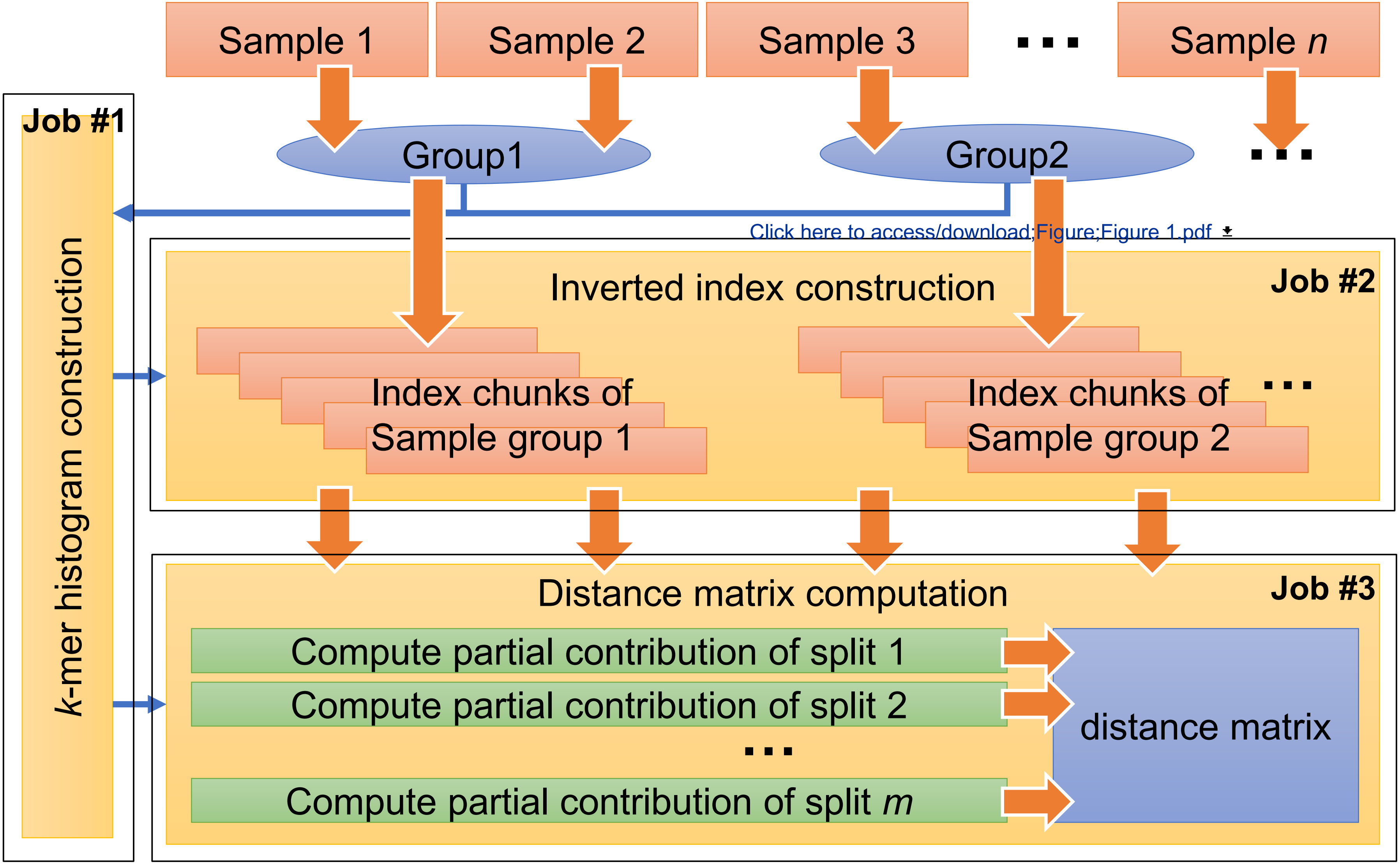

Sample 1
Sample 2
Sample 3
...
Sample *n*
Group1
Group2
...
Job #1
k-mer histogram construction
Inverted index construction
Job #2
Index chunks of Sample group 1
Index chunks of Sample group 2
...
Distance matrix computation
Job #3
Compute partial contribution of split 1
Compute partial contribution of split 2
...
Compute partial contribution of split *m*
distance matrix

Figure 2

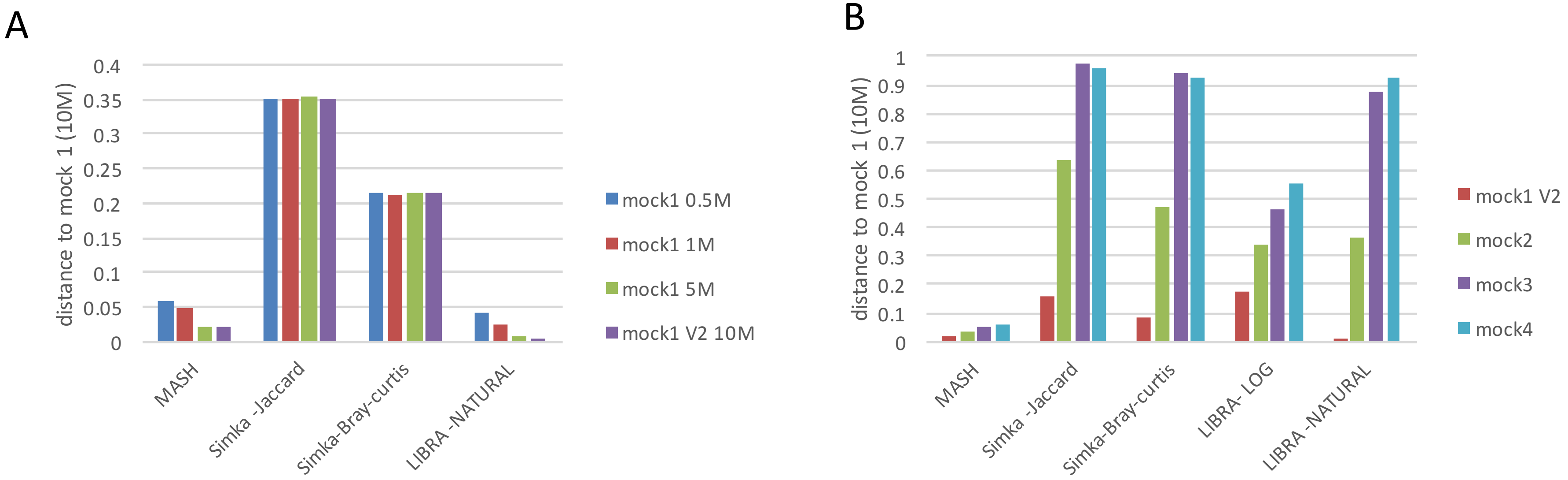
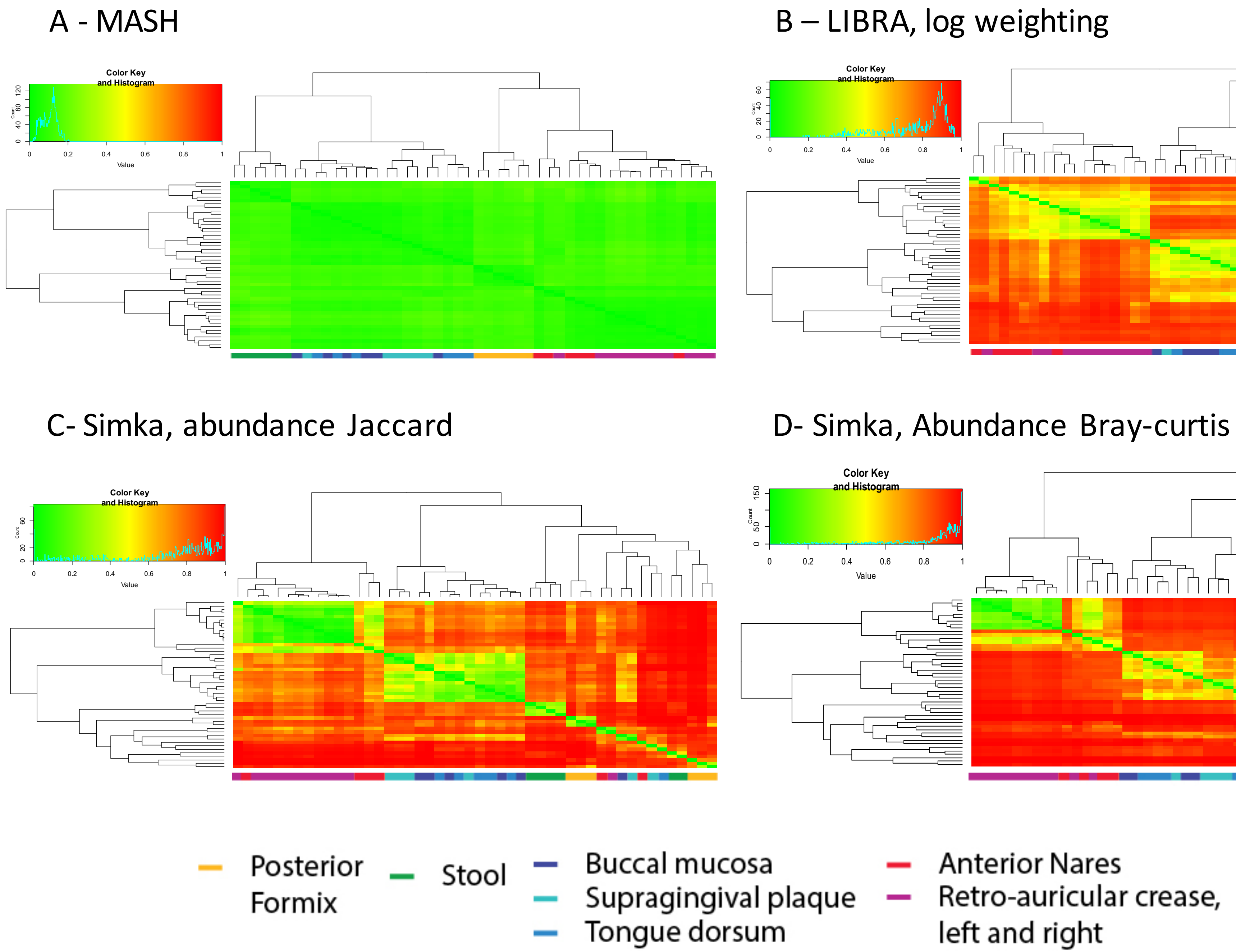

A
distance to mock 1 (10M)
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
MASH
Simka -Jaccard
Simka-Bray-curtis
LIBRA -NATURAL
🟦 mock1 0.5M
🟥 mock1 1M
🟩 mock1 5M
🟪 mock1 V2 10M
B
distance to mock 1 (10M)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
MASH
Simka -Jaccard
Simka-Bray-curtis
LIBRA- LOG
LIBRA -NATURAL
🟥 mock1 V2
🟩 mock2
🟪 mock3
🟦 mock4

Figure 3



A - MASH

B – LIBRA, log weighting

C- Simka, abundance Jaccard

D- Simka, Abundance Bray-curtis

Figure 4

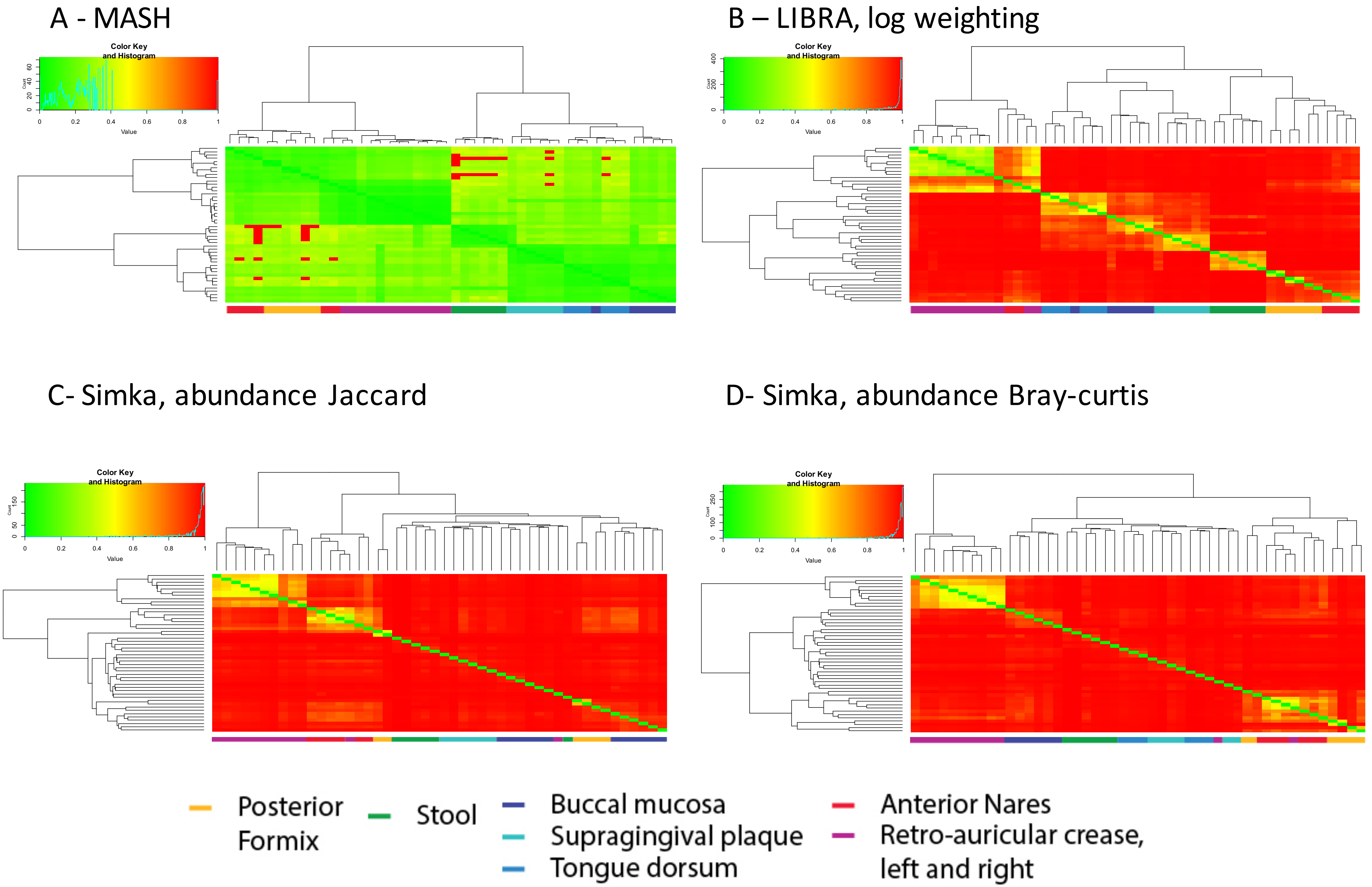
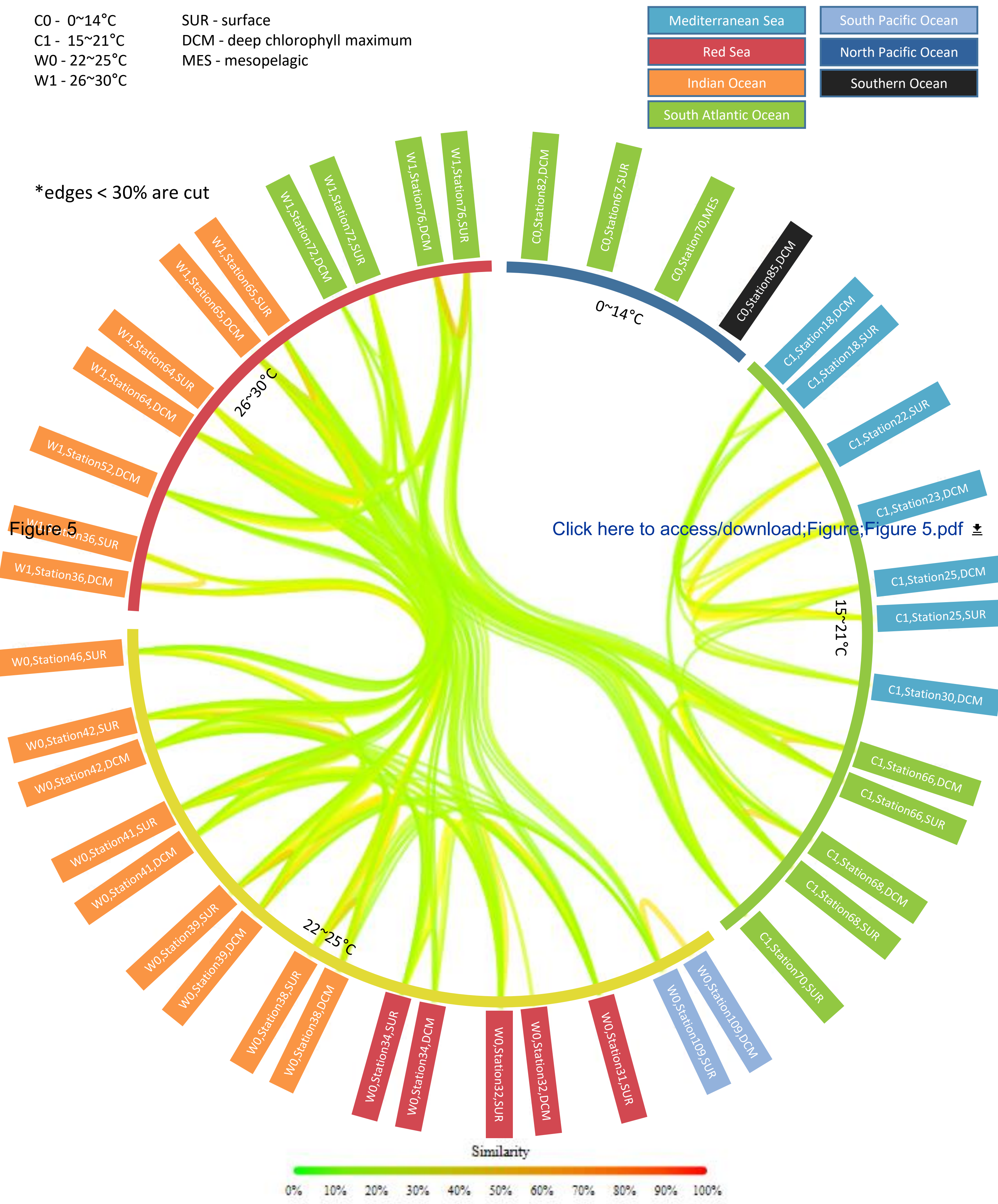
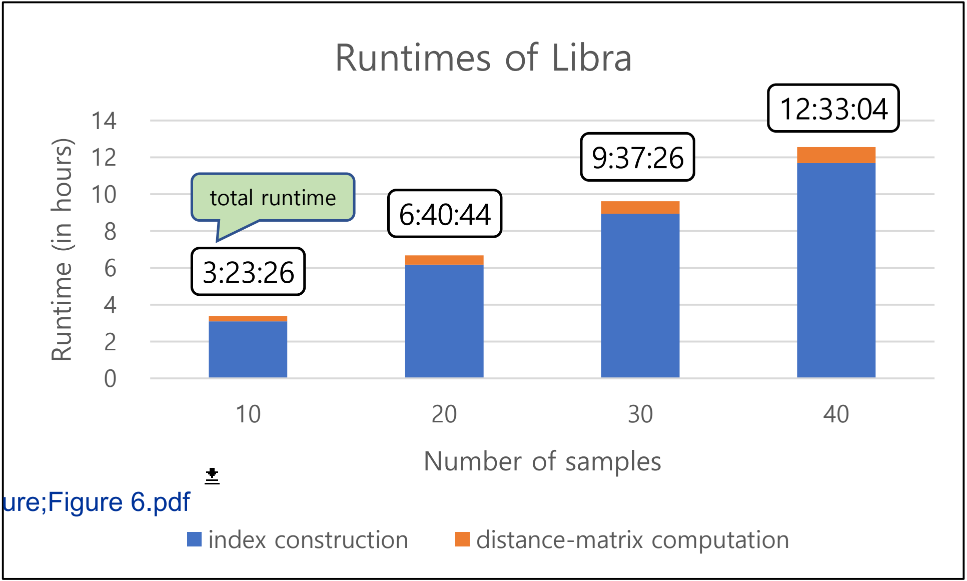
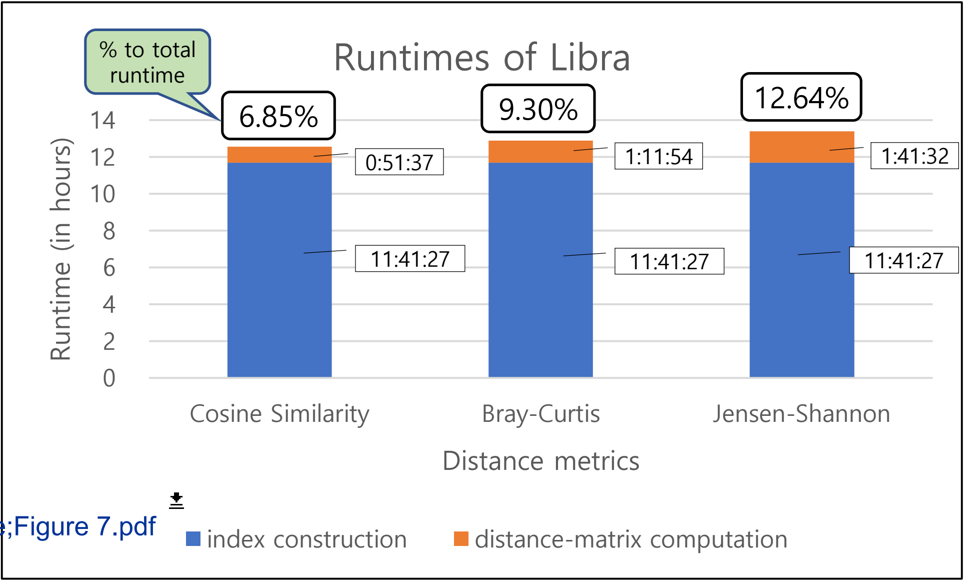

A - MASH
Color Key and Histogram
Count
0
20
40
60
0
0.2
0.4
0.6
0.8
1
Value
B – LIBRA, log weighting
Color Key and Histogram
Count
0
200
400
0
0.2
0.4
0.6
0.8
1
Value
C- Simka, abundance Jaccard
Color Key and Histogram
Count
0
50
150
0
0.2
0.4
0.6
0.8
1
Value
D- Simka, abundance Bray-curtis
Color Key and Histogram
Count
0
100
250
0
0.2
0.4
0.6
0.8
1
Value
Posterior Formix
Stool
Buccal mucosa
Supragingival plaque
Tongue dorsum
Anterior Nares
Retro-auricular crease, left and right

Figure 5


C0 - 0~14°C
C1 - 15~21°C
W0 - 22~25°C
W1 - 26~30°C
SUR - surface
DCM - deep chlorophyll maximum
MES - mesopelagic
Mediterranean Sea
Red Sea
Indian Ocean
South Atlantic Ocean
South Pacific Ocean
North Pacific Ocean
Southern Ocean
*edges < 30% are cut
0~14°C
15~21°C
22~25°C
26~30°C
C0,Station82,DCM
C0,Station67,SUR
C0,Station70,MES
C0,Station85,DCM
C1,Station18,DCM
C1,Station18,SUR
C1,Station22,SUR
C1,Station23,DCM
C1,Station25,DCM
C1,Station25,SUR
C1,Station30,DCM
C1,Station66,DCM
C1,Station66,SUR
C1,Station68,DCM
C1,Station68,SUR
C1,Station70,SUR
W0,Station109,DCM
W0,Station109,SUR
W0,Station31,SUR
W0,Station32,DCM
W0,Station32,SUR
W0,Station34,DCM
W0,Station34,SUR
W0,Station38,DCM
W0,Station38,SUR
W0,Station39,DCM
W0,Station39,SUR
W0,Station41,DCM
W0,Station41,SUR
W0,Station42,DCM
W0,Station42,SUR
W0,Station46,SUR
W1,Station36,DCM
W1,Station36,SUR
W1,Station52,DCM
W1,Station64,DCM
W1,Station64,SUR
W1,Station65,DCM
W1,Station65,SUR
W1,Station72,DCM
W1,Station72,SUR
W1,Station76,DCM
W1,Station76,SUR
Similarity
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%



gure;Figure 6.pdf



e;Figure 7.pdf

Supplementary Material Table 1

Supplementary Material Table 2

Supplementary Material Table 3

Supplementary Material Table 4

Supplementary Material Table 5

Supplementary Material Figure 1

Supplementary Material Figure 2

Supplementary Material Figure 3

Supplementary Material Figure 4

Supplementary Material Figure 5

THE UNIVERSITY OF ARIZONA®

College of Agriculture
and Life Sciences

Department of
Agricultural and
Biosystems

Shantz Bldg., B38, Room 403
1177 E. 4th Street
P.O. Box 210038
Tucson, AZ 85721-0038
Tel: (520) 621-1607
Fax: (520) 621-3963

August 24, 2018

Dear Editors,

Please find our paper for consideration at *Gigascience* as a research article titled "Libra: robust biological inferences of global datasets using scalable k-mer based all-vs-all metagenomic comparisons".

Microbiome research spans a broad array of disciplines from medicine, agriculture, bioenergy, and the environment, and is united in addressing core scientific questions relating microbial communities to biological and chemical processes in human, animal, or Earth systems. Given the preponderance of genomic data from diverse environments, there is a new desire to ask cross-cutting questions from the environment to human health. To move this work forward, microbiome datasets need to be holistically analyzed to examine how microbes move through living systems. Currently, only a subset of tools are available that make these analyses possible (through data reduction techniques and read count normalization), but none exploit big data architectures to scale compute and analyze complete datasets (100% of reads) in a linear and fault tolerant manner. This level of resolution is vital in metagenomic analyses where > 50% of the reads are unknown and the only way to understand functional changes in microbial communities is through all-vs-all analysis of diverse datasets to associate sequence patterns with environmental factors. To date, no tool offers a scalable and complete analysis of reads to explore global patterns in microbiome sciences.

Here we describe the first scalable algorithm for comparative metagenomics called Libra that is capable of performing an all-vs-all sequence analysis on hundreds of metagenomes in a Hadoop big data framework. Libra performs with unparalleled accuracy compared to equivalent tools using both simulated and real metagenomic datasets ranging from 80 million to 4.2 billion reads. In contrast to current methods, Libra's state-of-the-art algorithm and its implementation in a big data architecture does not require a reduction in dataset size or simplified distance metrics to achieve remarkable compute times and accuracy. As a result, Libra enables integration of massive datasets across disciplines to identify microbial and viral signatures linked to key biological processes. Moreover, Libra is available as an open-access web-based tool in iMicrobe (http://imicrobe.us) and in Github where the code is available for further optimization and reuse by the community. All authors declare no competing interests and have approved the manuscript for submission. The content of the manuscript has not been published, or submitted for publication elsewhere. Thank you for considering our paper for publication in *Gigascience*.

Sincerely,

Bonnie Hurwitz, PhD
Assistant Professor of Biosystems Engineering
University of Arizona, bhurwitz@email.arizona.edu