UNIVERSITY OF CALIFORNIA

Los Angeles

Development and Benchmarking of Imputation Methods for Microbiome and Single-cell Sequencing Data

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Statistics

by

Ruochen Jiang

2021

ABSTRACT OF THE DISSERTATION

Development and Benchmarking of Imputation Methods for Microbiome and Single-cell Sequencing Data

by

Ruochen Jiang
Doctor of Philosophy in Statistics
University of California, Los Angeles, 2021
Professor Jingyi Jessica Li, Chair

Next generation sequencing (NGS) has revolutionized biomedical research and has a broad impact and applications. Since its advent around 15 years ago, this high scalable DNA sequencing technology has generated numerous biological data with new features and brought new challenges to data analysis. For example, researchers utilize RNA sequencing (RNA-seq) technology to more accurately quantify the gene expression levels. However, the NGS technology involves many processing steps and technical variations when measuring the expression values in the biological samples. In other words, the NGS data researchers observed could be biased due to the randomness and constraints in the NGS technology. This dissertation will mainly focus on microbiome sequencing data and single-cell RNA-seq (scRNA-seq) data. Both of them are highly sparse matrix-form count data. The zeros could either be biological or non-biological, and the high sparsity in the data have brought challenges to data analysis.

Missing data imputation problem has been studied in statistics and social science as the survey data often experience non-response to some of the survey questions and those unresponded questions will be marked as "NA" or missing values in the data. Imputation methods are used to provide a sophisticated guess for the missing values, and the purpose is to avoid discarding the collected samples and for the ease of using the state-of-the-art statistical methods. In machine learning, the famous Netflix data challenge regarding film

recommendation system also falls into the missing data imputation problem category. Netflix wants to find a way to predict users' fondness of the movies they have not watched. The potential scores these users would give to the unwatched films are regarded as missing values in the data. NGS data imputation problem is different from the previous two cases in that the missing values in the NGS data are not so well-defined. The zeros in the NGS data could either come from the biological origin (should not be regarded as missing values) or non-biological origin (due to the limitation of the sequencing technology and should be regarded as missing values). The size (number of samples and features) of the NGS matrix data is usually larger than the size of survey data but smaller than the size of the recommendation system data. In addition, in most cases, the percentage of missing values in the survey data is less than the percentage of zeros in the NGS data, and the missing values in the film recommendation system data have the highest percentage ($> 99.9\%$). As a result, the commonly used missing data imputation methods in statistics and machine learning are not directly applicable to NGS data. In recent years, numerous imputation methods have been proposed to deal with the highly sparse scRNA-seq data. In light of this, this dissertation aims to address two questions. First, the microbiome sequencing data, having additional information comparing to the scRNA-seq data, lacks an imputation method. Secondly, whether to use imputation or not in scRNA-seq data analysis is still a controversial problem.

The first part of this dissertation focuses on the first imputation method developed for the microbiome sequencing data: mbImpute. Microbiome studies have gained increased attention since many discoveries revealed connections between human microbiome compositions and diseases. A critical challenge in microbiome data analysis is the existence of many non-biological zeros, which distort taxon abundance distributions, complicate data analysis, and jeopardize the reliability of scientific discoveries. To address this issue, we propose the first imputation method for microbiome data—mbImpute—to identify and recover likely non-biological zeros by borrowing information jointly from similar samples, similar taxa, and optional metadata including sample covariates and taxon phylogeny. Comprehensive simulations verify that mbImpute achieves better imputation accuracy under multiple metrics, compared with five state-of-the-art imputation methods designed for non-microbiome data.

In real data applications, we demonstrate that mbImpute improves the power of identifying disease-related taxa from microbiome data of type 2 diabetes and colorectal cancer, and mbImpute preserves non-zero distributions of taxa abundances.

The second part of this dissertation focuses on how to deal with high sparsity in the scRNA-seq data. ScRNA-seq technologies have revolutionized biomedical sciences by enabling genome-wide profiling of gene expression levels at an unprecedented single-cell resolution. A distinct characteristic of scRNA-seq data is the vast proportion of zeros unseen in bulk RNA-seq data. Researchers view these zeros differently: some regard zeros as biological signals representing no or low gene expression, while others regard zeros as false signals or missing data to be corrected. As a result, the scRNA-seq field faces much controversy regarding how to handle zeros in data analysis. We first discuss the sources of biological and non-biological zeros in scRNA-seq data. Second, we evaluate the impacts of non-biological zeros on cell clustering and differential gene expression analysis. Third, we summarize the advantages, disadvantages, and suitable users of three input data types: observed counts, imputed counts, and binarized counts and evaluate the performance of downstream analysis on these three input data types. Finally, we discuss the open questions regarding non-biological zeros, the need for benchmarking, and the importance of transparent analysis.

The dissertation of Ruochen Jiang is approved.

Mark Stephen Handcock

Guido Francisco Montufar

Frederic Paik Schoenberg

Arash Ali Amini

Jingyi Jessica Li, Committee Chair

University of California, Los Angeles

2021

To my parents and wife, thank you my beloved ones for your continuous support in my life and study.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

Throughout my doctoral studies, I have received a great amount of support from my mentors, my collaborators, my family and my friends.

I want to express my deepest gratitude for my advisor Dr. Jessica Jingyi Li, without whom, I would not be able to start my doctoral study at UCLA. She is a great mentor to me, not only for my research, but also for my life. She is a dedicated, passionate and meticulous researcher. Her attitude towards science has always inspired me. Besides being my mentor, she is also my good friend. When I meet with difficulties and important choices, she is always there to help and offer her valuable advice. I want to thank her for her mentoring and the opportunities and supports she provides.

I would like to thank Dr. Arash A. Amini, Dr. Frederic Schoenberg, Dr. Guido F. Montufar, and Dr. Mark Stephen Handcock for serving on my doctoral committee and for providing valuable feedback on my research. I would like to thank my long term collaborator Dr. Mark D. Biggin for his support and advice.

I thank all current members of Jessica's group for their helpful discussions and suggestions about my research. Among them, I want to especially thank Dr. Vivian Wei Li, Dongyuan Song, Elvis Han Cui and Tianyi Sun, who are all great collaborators and labmates.

At last, I want to thank my families and friends for their love and support. I want to thank my wife Junjun Wang and my parents, Ying Li and Dongyu Jiang, for their unconditional support, love and encouragement. At this time, I am looking forward to our baby, Roy Jiang, who is not yet born, but already brings happiness to our family. I also want to thank my friend Zhanhao Peng for his accompany during my doctoral study.

VITA

2012-2016	B.S. in Applied Mathematics, UCLA
2016-2018	M.S. in Statistics, UCLA
2018-2021	P.h.D in Statistics, UCLA
2018-2021	Graduate Research Assistant, Department of Statistics, UCLA
2018	Outstanding master student Award, UCLA

PUBLICATIONS

Jiang, R., Li, W. V., & Li, J. J. (2021). mbImpute: an accurate and robust imputation method for microbiome data. Genome biology, 22(1), 1-27.

Jiang, R., Sun, T., Song, D., & Li, J. J. (2020). Zeros in scRNA-seq data: good or bad? How to embrace or tackle zeros in scRNA-seq data analysis?. bioRxiv.

Jiang, R., Mark B. Biggin & Li, J. J. (2021). Distinct translational control by RNA secondary structure, uORFs, CDS length, codon usage and poly-A tail length during early Drosophila development. (manuscript)

CHAPTER 1

Introduction

Next generation sequencing (NGS) technology has revolutionized genomics research and provides new opportunities for biologists and health care providers [1, 2]. On the other hand, newly generated sequencing data has also brought new challenges to data analysis. Particularly, we are interested in the challenge of high sparsity in microbiome sequencing data and single-cell RNA-seq (scRNA-seq) data. In other words, both involve a lot of zero values. Some of these zeros truly represent that the microbiome species does not exist in a certain environment (for microbiome data) or a gene is not expressed in the biological system when measured (for single-cell RNA-seq data). However, some of the zeros arise due to the limitations in the sequencing technology [3]. The prevalence of zeros has threatened the data analysis and disturbs the biological findings [4]. Some of the algorithm developers have built zero-inflated models to better describe the observed distributions with high sparsity [5]. In contrast, others developed imputation methods to make the data less sparse and they illustrated that the imputed data could lead to more biologically meaningful discoveries [6].

Missing data imputation finds its root in statistics, and social science [7]. In the survey data, researchers commonly meet with the problem of non-response to some of the questions they asked [8]. The non-responses will lead to missing values in the collected data. Multiple imputation is the most popular paradigm developed for missing data imputation in the survey data [9]. The core of multiple imputation is to build a Bayesian model that describes the joint distribution of the complete data (including both observed data and missing data) and missing mechanism [9]. Other commonly used statistical approaches include mean or median imputation and hot deck [10]. On the other hand, there are also some machine learning methods proposed for imputation in the survey data, for example, multi-layer perception

(MLP), KNN, and random forest [11, 12]. However, these methods, except for the mean / median imputation, are usually not applicable to NGS data imputation as the feature number in the NGS data could go from $\sim 10^2$ to $\sim 10^5$, which is much larger than the feature numbers of the survey data, usually below $\sim 10^2$. The much larger scale in the NGS data has made the computational time for those methods unaffordable.

In 2007, the famous Netflix data challenge aimed to develop a good film recommendation system for the company [13]. This challenge is also a missing data imputation problem as we can treat the existing user ratings as known values and the values to be imputed as the missing rating that we want to predict. One of the most commonly used approaches is low-rank matrix factorization (MF) [14, 15]. Comparing to the NGS data, the number of features (films in the Netflix case) is comparable, while the number of samples (users in the Netflix case) is even larger comparing to the samples (cells in single-cell RNA-seq data or human samples in the human microbiome data). In addition, the percentage of missing data in the Netflix case could be $\sim 99.9\%$, while in NGS sequencing data, the percentage of zero values is relatively lower ($\sim 90\%$ in single-cell RNA-seq data and $\sim 70\%$ in microbiome sequencing data).

Besides the aforementioned distinctions between NGS sequencing data and survey data or recommendation system data, another important issue for dealing with the zeros in NGS data is that the "missing values" are not clearly defined. For the survey data, researchers know that some people do not respond to certain questions and the "NA"'s emerge. For recommendation system data, the data collector can recognize which of the films are rated and the films that are not rated incur missing values. However, in the NGS data, the term "missing value" is hard to be defined. Although some empirical evidence shows that using imputation in some datasets could enable better clustering or differential abundance analysis [4, 16, 17], the definition of missing values, or values that need to be imputed, is quite vague and controversial. In Chapter 3, I specifically discussed about the missing values in scRNA-seq data.

Due to the mentioned unique characteristics of NGS data, new imputation methods have been developed. Hou *et al.* benchmarked the newly developed 18 imputation methods for scRNA-seq data [6]. Zhang *et al.* provided technical details regarding these newly proposed imputation methods [18]. On the other hand, although microbiome sequencing data also has high sparsity and endures similar problems in sequencing data generation, this type of data has additional information including sample metadata and microbiome evolutionary relationship. No imputation method has been proposed for microbiome sequencing data. This dissertation focused on dealing with the high sparsity in microbiome sequencing data and scRNA-seq data. In Chapter 2, I introduced the first imputation method specifically designed for microbiome sequencing data. In Chapter 3, I discussed the origin of zeros in scRNA-seq data, and how different perspectives towards data pre-processing would change the downstream analysis results.

1.1 mbImpute: an accurate and robust imputation method for microbiome data

The first part of this dissertation introduced a new imputation method specially designed for microbiome sequencing data. It was inspired by the previous work scImpute by Vivian on the scRNA-seq data [4]. To clarify, I want to point out that the microbiome data, especially human microbiome data, treats collected patients' biological samples as samples and microbial species as features. On the other hand, single-cell RNA-seq data treat cells as samples and genes as features. During our inspection on the microbiome data structure and features, we found that it is distinctive from scRNA-seq data in several ways. First, microbiome sequencing data involves fewer samples than scRNA-seq data. Second, the percentage of zeros in scRNA-seq data is larger than the microbiome sequencing data if we omit the features with zero values across all samples. Third, the microbiome sequencing data usually has side information including sample metadata, and microbe phylogenetic distances. The sample metadata includes features such as age, gender and BMI of the human samples, while the phylogenetic distances measure the evolutionary similarities among microbial species. We developed the first imputation method, mbImpute, to deal with the high sparsity in the microbiome data, and it can leverage sample metadata and microbe phylogenetic distances if available.

We examine the performance of mbImpute using extensive simulation and real microbiome data study. Using simulation, we first show that mbImpute can recover false zeros more precisely compared with other already developed imputation methods. Second, we show that mbImpute is able to empower the downstream differential abundance (DA) analysis. The purpose of DA analysis is to find taxa that exhibit different abundance (represented by the values in the microbiome sequencing data) among two study groups (usually control vs. diseased patients). In real data analysis, we confirm that mbImpute empowers the DA analysis and the newly discovered DA taxa are biologically meaningful. Besides, mbImpute empowers the taxon-taxon correlation analysis.

1.2 Sources of zeros in single-cell RNA-seq data and how they affect data analysis

The second part of this dissertation focused on the controversy regarding dealing with the high sparsity in scRNA-seq data. We first illustrated the origin of zeros in the scRNA-seq data and then clarified some of the ambiguous concepts that have been widely used in the scRNA-seq field. We showed the distributional differences between Unique molecular identifier (UMI) implemented scRNA-seq data and non-UMI data. In summary, we found that using Negative binomial distribution is good enough to capture the variation in the UMI data, while around half of the genes' expressions are better fitted by zero-inflated distributions due to the high sparsity. We further examined how the downstream analysis is affected by the increased proportion of introduced zero values. We focused on two types of analysis: the cell-level clustering analysis and gene-level differential expression (DE) analysis. The clustering analysis is at the cell level as the researchers want to examine how the cells are clustered together using the expression profile and then they can assign cell types to these clusters. The DE analysis is at the gene level as the researchers aim to find genes that exhibit different expression patterns among different conditions. In addition, there are currently three different perspectives towards pre-processing scRNA-seq data before the downstream analysis. Some researchers prefer direct modeling on the original data, some prefer to run imputation before the downstream analysis, and there are recommendations to binarize the scRNA-seq data before the downstream analysis (i.e., set the non-zero values to zero). We presented our perspective on this matter, and the pre-processing method researchers shall use would depend on their knowledge or training (whether they are tool users or method developers) and the data type (UMI data vs. non-UMI data).

Our novelty in the methodology is that we developed five mechanisms to introduce zeros to the scRNA-seq data. There are three types of missing mechanisms in the literature [7]: missing completely at random (MCAR), missing at random (MAR) and missing not at random (MNAR). Let $X \in \mathbb{R}^{n\times p}$ be the matrix containing both observed and missing values. This can be the gene expression matrix and missing values refers to false zeros. Also we denote X_o as observed values and X_m as missing values. Let $R \in \mathbb{R}^{n\times p}$ be the associated binary matrix indicating whether the value is missing. That is, for $i = 1, \cdots, n$ and $j = 1, \cdots, p$,

$$R_{ij} = \begin{cases} 1 & \text{if } X_{ij} \text{ is observed.} \\ 0 & \text{if } X_{ij} \text{ is a missing value.} \end{cases}$$

Then MCAR, MAR, MNAR corresponds to

$$\mathbb{P}(R|X) = \begin{cases} \mathbb{P}(R) & \text{MCAR} \\ \mathbb{P}(R|X_o) & \text{MAR} \\ \mathbb{P}(R|X_m) & \text{MNAR} \end{cases} \tag{1.1}$$

However, many of the currently developed imputation methods only consider the missing mechanism, MCAR, the easiest to be implemented. Based on our discussion of the origins of the zero expression values in the scRNA-seq data, we proposed five missing mechanisms that correspond to one of MCAR, MAR or MNAR. We found that different ways to introducing zeros would affect the results of the downstream analysis.

1.3 Summary

During my doctoral study, I have developed one novel imputation method as a leading author, and the details of this project will be described in Chapter 2 of this dissertation [19]. I have also written a perspective paper regarding the high sparsity in the scRNA-seq data as a leading author [20], and the details are in Chapter 3. In addition, I have worked on the modeling of RNA translation control. This collaborative project is under the process of manuscript writing and omitted from this dissertation.

CHAPTER 2

mbImpute: an accurate and robust imputationmethod for microbiome data

2.1 Introduction

Microbiome studies explore the collective genomes of microorganisms living in a certain environment such as soil, sea water, animal skin, and human gut. Numerous studies have confirmed the importance of microbiomes in natural environments and human bodies [21]. For example, new discoveries have revealed the important roles microbiomes play in complex diseases such as obesity [22], diabetes [23], pulmonary disease [24, 25], and cancers [26]. These studies have shown the potential of human microbes as biomarkers for disease diagnosis or as therapeutic targets for disease treatment [5].

The development of high-throughput sequencing technologies has advanced microbiome studies in the last decade [27]. Two sequencing technologies are primarily used: the 16S ribosomal RNA (rRNA) amplicon sequencing and the shotgun metagenomic sequencing. The 16S rRNA amplicon sequencing measures 16S rRNAs, which can be used to identify and distinguish microbes [28]. The 16S sequencing reads are either clustered into operational taxonomic units (OTUs) [29] or mapped to amplicon sequence variants (ASVs) [30, 31]. The shotgun metagenomic sequencing, also known as the whole-genome sequencing (WGS), sequences all DNAs in a microbiome sample, including whole genomes of microbial species and host DNAs [29, 32–38]. The WGS sequencing reads are mapped to known microbial genome databases to quantify the abundances of microbial species. Despite the vast differences between the two technologies, 16S and WGS data can both be processed into the same data structure containing abundances of microbes in microbiome samples: a taxon count matrix with rows as microbiome samples (which often correspond to subjects or individuals) and columns as taxa (i.e., OTUs or ASVs for 16S rRNA data and species for WGS data), and each entry corresponds to the number of reads mapped to a taxon in a microbiome sample. It is worth noting that the total read count per microbiome sample, i.e., the sum of entries in a row of the count matrix, differs by five orders of magnitude between the two technologies: $\sim 10^3$ per sample for 16S rRNA data and $\sim 10^8$ for WGS data [39].

A critical challenge in microbiome data analysis is the existence of many zeros in taxon counts, an ubiquitous phenomenon for both 16S rRNA and WGS data [39]. The large proportion of zeros belongs to three categories by origin: biological, technical, and sampling zeros [40]. Biological zeros represent true zero abundances of non-existent taxa in microbiome samples. In contrast, technical and sampling zeros are non-biological zeros with different origins: technical zeros arise from pre-sequencing experimental artifacts (e.g., DNA degradation during library preparation and inefficient sequence amplification due to factors such as GC content bias) [41], while sampling zeros are due to limited sequencing depths. Although WGS data have much larger per-sample total read counts than 16S data have, they still suffer from sampling zeros because they sequence more nucleic acid sequences (microbial genomes instead of 16S rRNAs) and their effective sequencing depths are reduced by widespread host DNA contaminations [42–44].

This data sparsity issue challenges microbiome data analysis, as most state-of-the-art methods have poor performance on data containing too many zeros. Adding a pseudo-count of one to zeros is a common, simple approach [45, 46], but it is ad-hoc and suboptimal because it cannot distinguish biological zeros from technical and sampling zeros [47, 48]. Kaul et al. [49] developed an approach to distinguish these three types of zeros and to correct only the sampling zeros; however, their correction is still a simple addition of a pseudo-count of one, ignoring the fact that the (unobserved) actual counts of sampling zeros may not be exactly one.

In particular, this data sparsity issue hinders the differentially abundant (DA) taxon analysis, which aims to identify the taxa that exhibit significantly different abundances between two groups of samples [32]. Microbiome researchers employ two major types of statistical methods to identify DA taxa. Methods of the first type use parametric models [5, 45, 50–57]. For example, the zero-inflated negative binomial generalized linear model (ZINB-GLM) is used in [5, 50, 51], the negative binomial regression is used in the DESeq2-phyloseq method [52, 53], and the zero-inflated Gaussian model is used in the metagenomeSeq method [54]. However, these parametric model assumptions may not hold for a particular dataset [58]. Methods of the second type perform non-parametric statistical tests that do not assume specific data distributions. Widely-used methods include the Wilcoxon rank-sum test [33–38] and ANCOM [46]. A major drawback of these non-parametric methods is that a taxon would be called DA if its zero proportions differ significantly between two groups of samples, but this difference is unlikely biologically meaningful due to the prevalence of technical and sampling zeros. Note that both types of DA methods require the input taxon abundances to be in one of three units: counts [5, 50, 51, 53], log-transformed counts [54], and proportions (i.e., each taxon's count is divided by the sum of all taxa's counts in a sample) [45, 46, 55–57]; regardless of the unit, DA taxon analysis is always biased by the prevalence of technical and sampling zeros.

In addition to DA taxon analysis, other microbiome data analyses, such as the construction of taxon interaction networks [59–62], are also impeded by the data sparsity challenge. Although zero-inflated modeling is commonly used for sparse data, it requires a specific model formulation for each analysis task, which is often complicated or unrealistic for most microbiome researchers. Hence, a flexible and robust approach is needed to address the sparsity issue of microbiome data.

Imputation is a widely-used technique to recover missing data and facilitate data analysis. It has successful applications in many fields, e.g., recommender systems (e.g., the Netflix challenge [63]), image and speech reconstruction [64–66], imputation of unmeasured epigenomics datasets [67], missing genotype prediction in genome-wide association studies [68], and the more recent gene expression recovery in single-cell RNA-sequencing (scRNA-seq) data [4, 16, 17, 69, 70]. Microbiome and scRNA-seq data have the same count matrix structure if one considers microbiome samples and taxa as analogs to cells and genes, respectively; both data have large proportions of non-biological zeros. Given the successes of scRNA-seq imputation methods, we hypothesize that imputation can also relieve the data sparsity issue in microbiome data. Although there are methods utilizing matrix completion in the microbiome field, their main purpose is to perform community detection or dimension reduction instead of imputation [71, 72]. Two distinct features of microbiome data make it suboptimal to directly apply existing imputation methods. First, microbiome data are often accompanied by metadata including sample covariates and taxon phylogeny, which, however, cannot be used by existing imputation methods. In particular, phylogenetic information is known to be valuable for microbiome data analysis [73–80], as closely-related taxa in a phylogeny are likely to have similar functions and abundances in samples [81–84]. Second, microbiome data have a much smaller number of samples (often in hundreds) than the number of cells (often in tens of thousands) in scRNA-seq data, making those deep-learning based imputation methods inapplicable [70, 85]. On the other hand, the smaller sample size allows microbiome data to afford an imputation method that focuses more on imputation accuracy than computational time.

Here we propose mbImpute, the first imputation method designed for microbiome data including both 16S and WGS data. The mbImpute method identifies and corrects the zeros and low counts that are unlikely biological (for ease of terminology, we will refer to them as non-biological zeros in the following text) in microbiome taxon count data. The goal of mbImpute is to provide a principled data-driven approach to relieve the microbiome data sparsity issue due to prevalent non-biological zeros. To achieve this, mbImpute leverages three sources of information: a taxon count matrix, sample covariates (e.g., sample library size and subjects' age, gender, and body mass index), and taxon phylogeny, with the latter two sources being optional. There are two main steps in mbImpute (Fig. 2.1): first, mbImpute identifies likely non-biological zeros; second, it imputes these zeros by borrowing information from similar taxa (determined by both phylogeny and counts), similar microbiome samples (in terms of taxon counts), and sample covariates if available (see an illustration of the imputation step in Additional File 1: Fig. S1). The imputed data are expected to contain recovered taxon counts and would thus facilitate various downstream analyses, such as the identification of DA taxa and the construction of taxon interaction networks. Microbiome researchers can use mbImpute to avoid the hassle of dealing with sparse data in individual analysis tasks and to enjoy the flexibility of building up data analysis pipelines.

2.2 Results

2.2.1 mbImpute outperforms non-microbiome imputation methods in recovering missing taxon abundances and empowering DA taxon identification

As there are no imputation methods for microbiome data, we benchmark mbImpute against five state-of-the-art imputation methods designed for non-microbiome data: four popular scRNA-seq imputation methods (scImpute [4], SAVER [16], MAGIC [17], and ALRA [69]) and a widely-used general imputation method softImpute [86]. We design two simulation studies, and the common goal is to obtain a "complete" microbiome dataset without non-biological zeros, so that we can evaluate imputation accuracy by comparing the imputed data with the complete data. In the first study, we simulate complete data from a generative model fitted to a WGS dataset of type 2 diabetes (T2D) samples [37]; In the second, more realistic simulation study, we extract a sub-dataset with fewer than 15% zeros as the complete data from another WGS dataset of T2D samples [38]. In both simulation studies (see Additional File 1: Simulation 1 and Simulation 2 [4, 5, 16, 21–23, 26, 27, 29, 32–38, 45, 46, 49–51, 70, 86–117]), we introduce non-biological zeros into the complete data by mimicking the observed zero patterns in real datasets, obtaining what we call the zero-inflated data. After applying the six imputation methods to the zero-inflated data in both studies, we compare these methods' imputation accuracy in three aspects: (1) the mean squared error (MSE) between the imputed data and the complete data, (2) each taxon's Pearson correlation between its imputed abundances and complete abundances, and (3) the Wasserstein distance between the distributions of taxa's abundance mean/(standard deviation) ratios in the imputed data and the complete data. Fig. 2.2a–d illustrate the comparison results, indicating that mbImpute achieves the best overall performance in all three aspects. In particular, Fig. 2.2c–d and Additional File 1: Fig. S2 show that the imputed data by mbImpute best resemble the complete data, verifying the advantage of mbImpute in recovering missing taxon abundances in microbiome data.

We next demonstrate that mbImpute is a robust method. The core of mbImpute is to borrow three-way information from similar samples, similar taxa, and sample covariates to impute non-biological zeros in microbiome data (see Methods). In the aforementioned second simulation study (Additional File 1: Simulation 2), we scramble samples in the real T2D WGS data when we select the complete data, a situation not optimal for mbImpute; however, mbImpute still outperforms existing imputation methods (Fig. 2.2a–b). To further test for the robustness of mbImpute, we design a third simulation study including four simulation schemes, where the information useful for imputation is encoded in sample covariates only, samples only, taxa only, or three sources together (see Additional File 1: Simulation 3). Additional File 1: Fig. S3 shows that mbImpute effectively recovers non-biological zeros and reduces the MSE under every scheme. These results verify the robustness of mbImpute in selectively leveraging the information useful for imputation.

To further evaluate the performance of mbImpute on 16S rRNA sequencing data, we use a 16S simulator `sparseDOSSA` [105] to generate the abundances of 150 taxa in 100 samples under two conditions (see Additional File 1: Simulation 4). Among these 150 taxa, 45 are predefined as truly DA taxa. We apply five state-of-the-art DA methods: the Wilcoxon rank-sum test, ANCOM [46], metagenomeSeq [54], DESeq2-phyloseq [52, 53], and Omnibus test [118]. To evaluate the accuracy of DA taxon identification, we calculate the precision, recall, and F_1 score (i.e., the harmonic mean of precision and recall) of each method, with or without using mbImpute as a preceding step, by comparing each method's detected DA taxa to the truly DA taxa. Note that metagenomeSeq uses the zero-inflated Gaussian linear model for log-transformed microbiome data, but this model does not fit well to imputed data, which have many zeros removed; hence, we use the Gaussian linear model without zero-inflation to evaluate metagenomeSeq on imputed data. Under the false discovery rate (FDR) thresholds of 0.05 (Fig. 2.2e) and 0.1 (Additional File 1: Fig. S4), the mbImpute-empowered DA methods consistently have better recall rates and F_1 scores than those of the same DA methods without imputation. Notably, mbImpute improves both precision and recall rates of metagenomeSeq.

To evaluate the robustness of mbImpute to sequencing depth, we simulate 16S rRNA sequencing data based on real data for 300 taxa in 54 samples with four sequencing depths: 1000, 2000, 5000, and 10,000 reads per sample (see Additional File 1: Simulation 5). Additional File 1: Fig. S5a shows that mbImpute has better imputation accuracy as sequencing depth increases. This is an expected result because a larger sequencing depth leads to fewer missing data so that mbImpute can be better trained with more non-missing data. We further evaluate the performance of the five non-microbiome imputation methods along with mbImpute. Additional File 1: Fig. S6 shows that softImpute and ALRA, the two low-rank matrix factorization methods, also have better imputation accuracy as sequencing depth increases, yet their accuracies are worse than those of mbImpute at all sequencing depths. Unexpectedly, the four other imputation methods developed for scRNA-seq data—SAVER, scImpute, MAGIC, and ALRA—show no improvement over the baseline, "no imputation". One possible reason is that the sequencing depths used in this simulation ($\sim 10^3$) are much lower than those of typical scRNA-seq data ($\sim 10^6$). These results again suggest that scRNA-seq imputation methods are unsuitable for microbiome 16S rRNA sequencing data. We also check the robustness of mbImpute to outlier samples. Taking the sample with the 2000-read per-sample sequencing depth, we generate one or two outlier samples by assigning large abundance values to 62 lowly abundant taxa in the existing 54 samples and setting other taxa's abundance to zero (see Additional File 1). Additional File 1: Fig. S5b shows that the imputation accuracy of mbImpute is robust to the introduction of outlier samples. Additional File 1: Fig. S7 shows the abundance distributions of four example taxa with outlier values before and after imputation. We observe that the existence of outliers does not distort the post-imputation distribution of non-outlier samples.

2.2.2 mbImpute empowers DESeq2-phyloseq in DA taxon analysis

We find that mbImpute works well with DESeq2-phyloseq [52, 53], a widely used DA method for microbiome data, on real WGS datasets. We perform DA analysis on two T2D WGS datasets (Qin et al. and Karlsson et al.) and four CRC WGS datasets (Zeller et al., Feng et al., Vogtmann et al., and Yu et al.), with or without using mbImpute as a preceding step. The goal of DA analysis is to identify the DA taxa between the diseased and control samples. These DA taxa may serve as potential targets for early detection or treatment of disease [33]. Note that mbImpute does not utilize the samples' group information (whether each sample belongs to the diseased or control group) for its imputation, so that mbImpute will not falsely increase sample similarity within groups.

We start with the five DA methods—Wilcoxon rank-sum test, ANCOM, metagenomeSeq, DESeq2-phyloseq, and Omnibus test—for identifying disease-related DA taxa in the two T2D and four CRC datasets. Under the FDR threshold 0.05, only DESeq2-phyloseq and Omnibus test identify DA taxa in all datasets (Additional File 1: Table S1). Hence, we focus on evaluating the accuracy of DESeq2-phyloseq and Omnibus test on the original and imputed data (for DESeq2-phyloseq applied to the imputed data, we refer to it as mbImpute-empowered DESeq2-phyloseq). For a sanity check on the DA taxon identification results in each dataset, we plot the distribution of taxa's p-values calculated by DESeq2-phyloseq or Omnibus test before and after mbImpute is applied (Additional File 1: Figs. S8–9). We find that all the p-value distributions for DESeq2-phyloseq match our expectation (i.e., the expected p-value distribution should have a mode near zero and be uniform elsewhere). However, the p-value distributions for Omnibus test exhibit abnormality for the Karlsson et al. T2D and Vogtmann et al. CRC datasets. Specifically, the distributions have an unexpected mode near one for the Karlsson et al. dataset after imputation and for the Vogtmann et al. dataset before and after imputation. This phenomenon suggests that the distributional assumption of Omnibus test does not hold for these data. Hence, we focus on the comparison between DESeq2-phyloseq and mbImpute-empowered DESeq2-phyloseq in the following analysis.

To investigate whether the DA taxa identified by DESeq2-phyloseq or mbImpute-empowered DESeq2-phyloseq are meaningful disease markers, we evaluate the predictive power of the identified DA taxa for sample disease conditions (control or diseased). For each microbiome dataset, we use the DA taxa, identified by DESeq2-phyloseq or mbImpute-empowered DESeq2-phyloseq, as features and apply the random forest algorithm to predict sample disease conditions. We use the 5-fold cross-validated precision-recall area under the curve (PR-AUC) to evaluate the prediction accuracy (Fig. 2.3a). We observe that mbImpute-empowered DESeq2-phyloseq leads to overall better prediction accuracy than DESeq2-phyloseq does across the six datasets.

Then we focus on the Karlsson et al. T2D dataset and the Vogtmann et al. CRC dataset, which exhibit the largest improvement in prediction accuracy when the DA taxa identified by mbImpute-empowered DESeq2-phyloseq are used. For the Karlsson et al. T2D dataset, we observe that mbImpute-empowered DESeq2-phyloseq outputs a greater number of small p-values than DESeq2-phyloseq does (Additional File 1: Fig. S7), suggesting that more taxa are identified as DA after imputation (in fact, all the DA taxa identified before imputation are still found as DA after imputation). Hence, the improvement in prediction accuracy implies that the DA taxa identified only after imputation contribute to the distinction between control and T2D samples. In particular, we examine three example taxa (*Ruminococcus* species) identified as DA only after imputation. Fig. 2.3b shows the distributions of these three taxa's abundances (on the log-scale) before and after imputation. For each taxon, we observe that the imputed abundances and the original non-zero abundances have similar ranges and both suggest that the taxon is more abundant in T2D samples than in control samples. However, this abundance difference is obscured by the prevalent zeros before imputation and thus cannot be captured by DESeq2-phyloseq. Literature evidence is consistent with the post-imputation result of the first two taxa. Specifically, the first taxon has decreased abundances in T2D patients after the Acarbose treatment [119]. The second taxon, *Ruminococcus callidus*, is shown to be enriched in T2D mouse models [120].

For the Vogtmann et al. CRC dataset, the 5-fold cross-validated PR-AUC increases by almost 10% when the DA taxa identified after imputation, instead of those identified before imputation, are used as features. In fact, fewer taxa are identified as DA after imputation (Additional File 1: Fig. S8). At the q-value threshold 0.05, DESeq2-phyloseq identifies 53 DA taxa, while mbImpute-empowered DESeq2-phyloseq identifies 40 DA taxa, with only 17 taxa in overlap. This result suggests that the 23 DA taxa identified only after imputation contribute much to the distinction between control and CRC samples. We examine three of these 23 taxa: *Ruminococcus gnavus*, *Lachnospiraceae bacterium_2_1_58FAA*, and *Granulicatella adiacens*. Fig. 2.3c shows that each taxon has its imputed abundances and its original non-zero abundances in similar ranges; its imputed and original non-zero abundances both suggest it to be more abundant in CRC samples than in control samples. However, this abundance difference is obscured by the prevalent zero abundances before imputation and thus cannot be captured by DESeq2-phyloseq. To confirm the post-imputation result, we find literature evidence for the three taxa. First, several studies have reported that *Ruminococcus gnavus* is associated with a higher risk of CRC [115, 121–123]. Second, two studies have shown that *Lachnospiraceae bacterium_2_1_58FAA* is positively associated with colorectal neoplasms, from which CRC arises [115]. Third, *Granulicatella adiacens* is reported to be associated with CRC progression in both human [99] and mouse studies [124]. We also examine the taxa identified as DA before imputation but not as DA after imputation, and we find that these taxa only differ in zero proportions and have similar non-zero abundance distributions between control and CRC samples (Additional File 1: Fig. S10). We argue that such taxa are unlikely to be truly DA because it is questionable whether zero proportion differences are biologically meaningful given the prevalence of technical and sampling zeros. Together, our analysis results on the Karlsson et al. T2D dataset and the Vogtmann et al. CRC dataset suggest that compared to DESeq2-phyloseq, mbImpute-empowered DESeq2-phyloseq can detect DA taxa that are more predictive of sample conditions, and we verify that some DA taxa only detected by mbImpute-empowered DESeq2-phyloseq are functionally relevant by literature evidence.

For all the DA taxa identified by DESeq2-phyloseq and mbImpute-empowered DESeq2-phyloseq in the two T2D and four CRC data datasets, we query the GMrepo database [115] and find two T2D- and one CRC-related functional terms. For each term, we perform the Fisher's exact test to check its enrichment in the DA taxa identified from the corresponding disease-related datasets. Our results show that all three terms are more enriched in the DA taxa identified after mbImpute is applied (Table 2.1; Additional Files 1–7), providing functional support to the efficacy of mbImpute in empowering DESeq2-phyloseq.

Furthermore, we analyze the overlap of the DA taxa identified in the two T2D datasets,

DA method	T2D term 1*	T2D term 2**	CRC term***
DESeq2-phyloseq	0.54	0.76	0.0027
mbImpute-empowered DESeq2-phyloseq	0.03	0.17	0.0010

Table 2.1: Fisher's exact test p-values about the enrichement of T2D- and CRC-related functional terms in the DA taxa found by DESeq2-phyloseq or mbImpute-empowered DESeq2-phyloseq. For each term, the DA taxa identified by each method from the corresponding datasets are pooled to do the test.
*T2D term 1: "The time period before the development of symptomatic diabetes. For example, certain risk factors can be observed in subjects who subsequently develop INSULIN RESISTANCE as in type 2 diabetes (DIABETES MELLITUS, TYPE 2)."
**T2D term 2: "A cluster of symptoms that are risk factors for CARDIOVASCULAR DISEASES and TYPE 2 DIABETES MELLITUS. The major components of metabolic syndrome include ABDOMINAL OBESITY; atherogenic DYSLIPIDEMIA; HYPERTENSION; HYPERGLYCEMIA; INSULIN RESISTANCE; a proinflammatory state; and a prothrombotic (THROMBOSIS) state."
***CRC term: "Tumors or cancer of the COLON or the RECTUM or both. Risk factors for colorectal cancer include chronic ULCERATIVE COLITIS; FAMILIAL POLYPOSIS COLI; exposure to ASBESTOS; and irradiation of the CERVIX UTERI."

Qin et al. and Karlsson et al. There is no overlap in the two sets of DA taxa identifed by DESeq2-phyloseq, but *Clostridium bolteae* is identified by mbImpute-empowered DESeq2-phyloseq in both datasets. In fact, *Clostridium bolteae* has been reported as eriched in CRC samples in Qin et al. but not in Karlsson et al. In our analysis on the Karlsson et al. dataset, *Clostridium bolteae* has FDR-adjusted p-values 0.347 and 0.036 before and after imputation, respectively (abundance distributions in Additional File 1: Fig. S11). Literature evidence suggests that *Clostridium bolteae* is positively associated with T2D in both human [125] and mouse studies [126].

For the four CRC datasets (Feng et al., Yu et al., Vogtmann et al., and Zeller et al.), we analyze the DA taxa identified in at least two datasets before and after imputation. Specifically, DESeq2-phyloseq and mbImpute-empowered DESeq2-phyloseq respectively identify four and 18 taxa (with three taxa in overlap) that have significantly lower abundances in CRC samples than in normal samples. Among these taxa, DESeq2-phyloseq only identifies *Bifidobacterium animalis*, while mbImpute-empowered DESeq2-phyloseq additionally identifies three other *Bifidobacterium* species: *Bifidobacterium bifidum*, *Bifidobacterium catenulatum*, and *Bifidobacterium longum*. Additional File 1: Figs. S12–14 show the distributions of these three taxa's abundances (on the log-scale) before and after imputation. Literature evidence indicates that *Bifidobacterium* is beneficial to the immune system against CRC [127–129] and has been used as probiotics [130]; all the four *Bifidobacterium* species detected by mbImpute-empowered DESeq2-phyloseq have been reported to have significantly lower abundances in CRC samples [131, 132]. Together, our overlap analysis on T2D and CRC datasets suggests that mbImpute helps recover the DA taxa that are detected in one dataset but missed in another due to prevalent zeros.

2.2.3 mbImpute preserves distributional characteristics of taxa's non-zero abundances and recovers downsampling zeros

In the DA analysis described in the last section, we observe that mbImpute can well maintain the distributions of taxa's non-zero abundances, see Fig. 2.3b–c. To further verify the property of mbImpute in preserving characteristics of non-zero abundances, we examine pairwise taxon-taxon relationships in the two T2D WGS datasets: Qin et al. and Karlsson et al. For a pair of taxa, we calculate two Pearson correlations based on the raw data on the log-scale: one using all the samples ("raw all-sample correlation") and the other only using the samples where both taxa have non-zero abundances ("raw non-zero-sample correlation"). In this section, we perform our analysis on the log-scale of the taxa count matrix since one of the assumptions for Pearson correlation is the normality of both variables, and microbiome count data on the log-scale better resemble a continuous normal distribution. For the same pair of taxa, we also calculate a Pearson correlation based on the imputed data by mbImpute on the log-scale, using all the samples ("imputed all-sample correlation"). As shown in Fig. 2.4a–b, there are vast differences between the raw all-sample correlations and the corresponding raw non-zero-sample correlations. However, the imputed all-sample correlations better resemble the corresponding raw non-zero-sample correlations, suggesting that mbImpute well preserves pairwise taxon-taxon correlations encoded in taxa's non-zero abundances.

We also explore the linear relationship of each taxon pair using the standard major axis (SMA) regression, which, unlike the least-squares regression, treats two taxa symmetrically. For a pair of taxa, we perform two SMA regressions on the raw data: one using all the samples ("raw all-sample regression") and the other using only the samples where both taxa have non-zero abundances ("raw non-zero-sample regression"). We also perform the

Removal rate	40%	70%
% of downsampling zeros identified	95.83% ± 0.46%	92.83% ± 0.92%
Pearson correlation before imputation	0.7565 ± 0.0023	0.5261 ± 0.0016
Pearson correlation after imputation	0.8747 ± 0.0100	0.7582 ± 0.0235

Table 2.2: Effectiveness of mbImpute in indentifying zeros due to downsampling of Qin et al.'s T2D WGS dataset. For each of two removal rates 40% and 70%, we repeat independent downsampling for ten times. For each removal rate (column), the first row lists the average percentage of downsampling zeros identified by mbImpute; the second row lists the average Pearson correlation between a downsampled matrix and the original matrix (on the log-scale) before imputation; the third row lists the average Pearson correlation (on the log-scale) after mbImpute is used. Each average calculated across the ten downsampling and is accompanied with an error margin, i.e., 1.96 times the standard error over the ten downsampling.

SMA regression on the imputed data by mbImpute, using all the samples ("imputed all-sample regression"). Fig. 2.4a–b show that the raw all-sample regressions and the raw non-zero-sample regressions return vastly different lines. Especially, the two lines between *Eubacterium sirasum* and *Ruminococcus obeum* in the Karlsson et al. data (Fig. 2.4b bottom left) have slopes with opposite signs. In contrast, the imputed all-sample regressions output lines with slopes similar to those of the raw non-zero-sample regressions. This result again confirms mbImpute's capacity for preserving characteristics of taxa's non-zero abundances in microbiome data.

Furthermore, we systematically evaluate the performance of mbImpute in preserving raw non-zero-sample correlations on the two T2D WGS datasets and the four CRC WGS datasets, with each dataset containing samples in two groups: diseased and control. Fig. 2.4c show that the imputed all-sample correlations resemble the raw non-zero-sample correlations much better than the raw all-sample correlations do, on every dataset including all samples ("whole" in Fig. 2.4c). Moreover, within each sample group in each dataset ("diseased" and "control" in Fig. 2.4c), the imputed all-sample correlations still better resemble the raw non-zero-sample correlations than the raw all-sample correlations do. Note that the resemblance is defined based on the Pearson correlation of two sets of correlations. Additional File 1: Fig. S15 shows that the same conclusion holds when the resemblance is defined based on the Spearman correlation. Note that mbImpute does not use the group information of each sample in its imputation process.

Our results echo existing concerns about spurious taxon-taxon correlations in microbiome data due to excess non-biological zeros [133, 134]. In other words, taxon-taxon correlations cannot be accurately estimated from raw data using all samples. Without imputation, an intuitive approach is to use taxa's non-zero abundances to estimate taxon-taxon correlations; however, this approach reduces the sample size for estimating each taxon pair's correlation because it does not use the samples containing zero abundances for either taxon, and it also makes different taxon pairs' correlation estimates rely on different samples. To address these issues, mbImpute provides another approach: its imputed data allow taxon-taxon correlations to be estimated from all samples. Moreover, we observe that mbImpute makes log-transformed taxon abundances closer to be normally distributed (Additional File 1: Fig. S16); thus, the Pearson correlation is a more meaningful measure for taxon-taxon associations on the imputed data than on the raw data.

In addition, based on the T2D WGS dataset generated by Qin et al., we verify mbImpute's capacity to identify non-biological zeros generated by downsampling. In each sample (i.e., each row in the sample-by-taxon count matrix), we assign every taxon a sampling probability proportional to its count, i.e., the larger the count, the more likely the taxon is to be sampled; based on these probabilities, we sample 60% or 30% of the non-zero taxon counts, and we set the unsampled counts to zeros (corresponding to a removal rate of 40% or 70%); we repeat the downsampling independently for ten times. After applying mbImpute to the downsampled count matrices, we find that mbImpute correctly identifies 95.83% and 92.83% (on average) of the newly introduced non-biological zeros under the two removal rates. Before imputation, the average Pearson correlations between the downsampled matrices and the original matrix (on the log-scale) are 0.76 and 0.53 under the two removal rates. After applying mbImpute to all the three matrices, the correlations are increased to 0.87 and 0.76 (Table 2.2). This result confirms the effectiveness of mbImpute in recovering zeros due to downsampling.

2.2.4 mbImpute increases the similarity of microbial community structure between 16S rRNA and WGS data

We further show that mbImpute can enhance the similarity of taxon-taxon correlations inferred from micrbiome data measured by two technologies—16S rRNA sequencing and WGS.

We use two microbiome datasets of healthy human stool samples: a 16S rRNA dataset from the Human Microbiome Project [135] and a WGS dataset from the control samples in Qin et al. We compare the genus-level taxon-taxon correlations between these two datasets, and we perform the comparison again after applying mbImpute. Fig. 2.5 shows that mbImpute increases the similarity between the taxon correlation structures in the two datasets. Before imputation, the Pearson correlation between the two correlation matrices (one computed from 16S rRNA taxon abundances and the other from WGS taxon abundances) is 0.59; mbImpute increases the correlation to 0.64. In particular, we observe three taxon groups (highlighted by magenta, green, and purple squares in Fig. 2.5) supported by both 16S rRNA and WGS data after imputation. Notably, in the magenta squares, *Acidaminococcus* has correlations with *Dialister* and *Blautia* only after imputation, and this result is consistent with the literature: *Acidaminococcus* and *Dialister* are both reported to have low abundances in healthy human stool samples [136]; *Acidaminococcus* and *Blautia* are both associated with risks of T2D and obesity, lipid profiles, and homeostatic model assessment of insulin resistance [137]. The green squares contain three bile-tolerant genera: *Alistipes*, *Bilophila*, and *Bacteroides* [138]. The raw 16S and WGS data only reveal the correlation between *Bacteroides* and *Alistipes*, but mbImpute recovers the correlations *Bilophila* has with *Alistipes* and *Bacteroides*. The purple squares indicate a strong correlation between *Sutterella* and *Prevotella* after imputation, yet this correlation is not observed in raw WGS data. We verify this correlation in the MACADAM database [139], which contains metabolic pathways associated with microbes. Out of 1260 pathways, *Sutterella* and *Prevotella* are associated with 154 and 278 pathways, respectively, and 122 pathways are in common; Fisher's exact test finds that the overlap is statistically significant (p-value $< 2.2 \times 10^{-16}$), suggesting that *Sutterella* and *Prevotella* may be functionally related. Overall, our results indicate that mbImpute can facilitate meta-analysis of 16S and WGS data by alleviating the hurdle of prevalent non-biological zeros.

We perform a negative control study to confirm that the increased similarity between 16S rRNA and WGS data is not an artifact introduced by mbImpute. We use a 16S rRNA dataset of human oral samples and a WGS dataset of human stool samples, which are expected to have different genus-level taxon-taxon correlations. Same as in the previous study, we compare the genus-level taxon-taxon correlations between the two datasets before and after applying mbImpute. Additional File 1: Fig. S17 shows that mbImpute decreases the similarity between the taxon correlation matrices of the two datasets. Before imputation, the Pearson correlation between the two correlation matrices is 0.21; mbImpute decreases the correlation to 0.09.

2.3 Discussion

A critical challenge in microbiome data analysis is statistical inference of taxon abundance from highly sparse and noisy data. Our proposed method, mbImpute, will address this challenge and facilitate analysis of both 16S and WGS data; mbImpute works by correcting non-biological zeros and retaining taxa's non-zero abundance distributions after imputation. As the first imputation method designed for microbiome data, mbImpute is shown to outperform multiple state-of-the-art imputation methods developed for other data types. In the DA analysis, we show that mbImpute-empowered DESeq2-phyloseq has better performance in selecting predictive taxa for disease conditions comparing to DESeq2-phyloseq. The reason is that mbImpute-empowered DESeq2-phyloseq is able to identify the taxa missed by the DESeq2-phyloseq (due to excess zeros) but should be called DA (i.e., having non-zero abundances that exhibit significantly different means between two sample groups). We then demonstrate that mbImpute preserves taxa's non-zero abundance distributions. As a result, taxon-taxon correlations calculated from all samples after imputation better resemble the taxon-taxon correlations calculated from non-zero counts only. Hence, mbImpute can facilitate taxon network analysis by allowing all taxon pairs to have meaningful correlations computed from all samples. Moreover, mbImpute improves the reproducibility of DA taxon identification across studies and the consistency of microbial community detection between 16S and WGS data.

In the application of mbImpute, two practical concerns are what normalization method and phylogenetic distance metric work the best with mbImpute. First, the goal of normalization is to make taxon counts comparable across samples, a necessary assumption of mbImpute. In our results, we think our way of normalization is sufficient to meet this assumption. However, the appropriate normalization method for mbImpute is case by case in future applications, depending on whether confounders such as batch effects are observable; hence, users' judgement is indispensable. We recognize that benchmarking normalization methods for microbiome data is a separate project. Hence, we refer users to benchmark papers [39, 140] to guide their choice of benchmark methods. Second, users may specify the phylogenetic distances between taxa based on their domain knowledge. In our results, we define the phylogenetic distance between two taxa as the number of branches connecting them in a phylogenetic tree, but alternative choices exist, such as the total lengths of the branches. If users want to choose a distance metric, we recommend that they supply the phylogenetic distances defined by candidate metrics into mbImpute and choose the metric that leads to the smallest cross-validated MSE, i.e., the cross-validated imputation error of mbImpute on non-missing data.

Regarding the mbImpute-empowered DA analysis, we note that it offers a new perspective of identifying DA taxa from microbiome 16S and WGS data after imputation. We have summarized three statistical definitions of DA taxa in microbiome data in Additional File 1: Statistical definitions of DA taxa. Note that mbImpute-empowered DA analysis is advantageous in that it alleviates the existence of non-biological zeros, and it uses all available samples for DA testing. A controversial question is, if a taxon has few zeros in condition 1 but few non-zeros in condition 2, and the non-zero values have similar magnitudes in the two conditions, whether or not should this taxon be called DA. We note that mbImpute is unlikely to impute the predominant zeros in condition 2 because it would treat these zeros as biological zeros. Hence, mbImpute-empowered DA analysis is likely to call such a taxon as DA.

There has been a long-standing concern about sample contamination in microbiome sequencing data, e.g. contamination from DNA extraction kits and laboratory reagents [141, 142]. Existing studies have attempted to address this issue via calibrated sequencing operations [142–144] and computational methods [145, 146]. We recommend researchers to perform contamination removal before applying mbImpute. Moreover, by its design, mbImpute is robust to certain types of sample contamination that result in outlier taxa and samples. For each outlier taxon, mbImpute would borrow little information from other taxa to impute this outlier taxon's abundances. Similarly, mbImpute is robust to the existence of outlier samples that do not resemble any other sample.

In statistical inference, a popular and powerful technique is the use of indirect evidence by borrowing information from other observations, as seen in regression, shrinkage estimation, empirical Bayes, among many others [147]. Imputation follows the indirect evidence principle, where the most critical issue is to decide what observations to borrow information from so as to improve data quality instead of introducing excess biases. To achieve this, mbImpute employs penalized regression to selectively leverage similar samples, similar taxa, and sample covariates to impute likely non-biological zeros, whose identification also follows the indirect evidence principle by incorporating sample covariates into consideration. Also, mbImpute provides a flexible framework to make use of microbiome metadata: it selectively borrows metadata information when available, but it does not rely on the existence of metadata (see Methods).

In the comparison of mbImpute with softImpute, a general matrix imputation method widely used in other fields, we observe that softImpute's imputed taxon abundances exhibit artificial spikes and smaller variances than those of the original non-zero abundances, possibly due to its low-rank assumption. In contrast, mbImpute is a regression-based method that focuses more on local matrix structures, and we find that it retains well the original non-zero abundance distributions. We will investigate the methodological differences between mbImpute and softImpute in a future study.

Moreover, we observe that, similar to each taxon's non-zero abundances, the imputed abundances exhibit a bell-shaped distribution across samples on the log-scale. This suggests that statistical methods utilizing normal distributional assumptions become suitable and applicable to imputed taxon abundances. A possible use of imputed microbiome data is to construct a taxon-taxon interaction network, to which network analysis methods may be applied to find taxon modules and hub taxa [148]. As a preliminary exploration, we construct Bayesian networks of taxa based on the two T2D datasets Qin et al. and Karlsson et al. after applying mbImpute. Interesting changes are observed in taxon interactions from control samples to T2D samples (Additional File 1: Figs. S18–19). For example, two genera, *Ruminococcus* and *Eubacterium*, have interactive species in control samples but not in T2D samples. In future research, differential network analysis methods can be applied to find taxon communities that differ between two sample groups.

2.4 Methods

2.4.1 mbImpute methodology

Here we describe mbImpute, a statistical method that corrects prevalent non-biological zeros in microbiome data. As an overview, mbImpute takes a taxon count matrix as input; pre-processes the data; identifies the likely non-biological zeros and imputes them based on the input count matrix, sample covariates, and taxon phylogeny; and outputs an imputed count matrix.

Notations

We denote the sample-by-taxon taxa count matrix as $\mathbf{M} = (M_{ij}) \in \mathbb{Z}_{\geqslant 0}^{n \times m}$, where n is the number of microbiome samples and m is the number of taxa. We denote the sample covariate matrix (i.e., metadata) as $\mathbf{X} \in \mathbb{R}^{n \times q}$, where q equals the number of covariates plus one (for the intercept). (By default, mbImpute includes sample library size as a covariate.) In addition, we define a phylogenetic distance matrix of taxa as $\mathbf{D} = (D_{jj'}) \in \mathbb{Z}_{\geqslant 0}^{m \times m}$, where $D_{jj'}$ represents the number of branches connecting taxa j and j' in the phylogenetic tree.

Data pre-processing

mbImpute requires every taxon's counts across samples to be on the same scale before imputation. If this condition is unmet, normalization is needed. However, how to properly normalize microbiome data is challenging, and multiple normalization methods have been developed in recent years [48, 149, 150]. Regarding the choice of an appropriate normalization method, users may refer to benchmark papers [39, 140]. To give users the flexibility of choosing an appropriate normalization method, mbImpute allows users to input a normalized count matrix by specifying that the input matrix does not need normalization. Otherwise, mbImpute normalizes samples by library size.

Default normalization (optional) To account for the varying library sizes (i.e., total counts) of samples, mbImpute first normalizes the count matrix $\mathbf{M}$ by row. The normalized count matrix is denoted as $\mathbf{M}^{(\mathcal{N})} = (M_{ij}^{(\mathcal{N})}) \in \mathbb{R}_{\geqslant 0}^{n \times m}$, where

$$M_{ij}^{(\mathcal{N})} = 10^6 \cdot \frac{M_{ij}}{\sum_{j'=1}^{m} M_{ij'}} .$$

After this normalization, every sample has a total count of 10^6.

mbImpute applies the logarithmic transformation to the normalized counts so as to reduce the effects of extremely large counts [98]. The resulted log-transformed normalized matrix is denoted as $\mathbf{Y} = (Y_{ij}) \in \mathbb{R}_{>0}^{n \times m}$, with

$$Y_{ij} = \log_{10}\left(M_{ij}^{\mathcal{N}} + 1.01\right),$$

where the value 1.01 is added to make $Y_{ij} > 0$ to avoid the occurrence of infinite values in a later parameter estimation step, following Li and Li [4, 97]. This logarithmic transformation allows us to fit a continuous probability distribution to the transformed data, thus simplifying the statistical modeling. In the following text, we refer to $\mathbf{Y}$ as the sample-by-taxon abundance matrix.

mbImpute step 1: identification of taxon abundances that need imputation

mbImpute assumes that each taxon's abundances, i.e., a column in $\mathbf{Y}$, follow a mixture model. The model consists of two components: a Gamma distribution for the taxon's likely non-biological zeros and low abundances and a normal distribution for the taxon's actual abundances, with the normal mean incorporating sample covariate information (including sample library size as a covariate). Specifically, mbImpute assumes that the abundance of taxon j in sample i, Y_{ij}, follows the following mixture distribution:

$$Y_{ij} \sim p_j \cdot \Gamma\left(\alpha_j, \beta_j\right) + (1 - p_j) \cdot \mathcal{N}\left(X_{i\cdot}^{\mathsf{T}} \gamma_j, \sigma_j^2\right) ,$$

where $p_j \in (0, 1)$ is the missing rate of taxon j, i.e., the probability that taxon j is falsely undetected, $\Gamma\left(\alpha_j, \beta_j\right)$ denotes the Gamma distribution with shape parameter $\alpha_j > 0$ and rate parameter $\beta_j > 0$, and $\mathcal{N}\left(X_{i\cdot}^{\mathsf{T}} \gamma_j, \sigma_j^2\right)$ denotes the normal distribution with mean $X_{i\cdot}^{\mathsf{T}} \gamma_j$ and standard deviation $\sigma_j > 0$. In other words, with probability p_j, Y_{ij} is a missing value that needs imputation; with probability $1 - p_j$, Y_{ij} is sampled from the non-missing abundance distribution of taxon j and does not need imputation. mbImpute models the normal mean parameter as a linear function of sample covariates: $X_{i\cdot}^{\mathsf{T}} \gamma_j$, where $X_{i\cdot} \in \mathbb{R}^q$ denotes the i-th row in the covariate matrix $\mathbf{X}$, i.e., the covariates of sample i, and $\gamma_j \in \mathbb{R}^q$ represents the q covariates' effects (including the intercept) on taxon j's abundance. This formulation allows a taxon to have similar expected abundances (when not missing) in samples with similar covariates.

The intuition behind this model is that taxon j's non-missing abundance in a sample is drawn from a normal distribution, whose mean depicts the expected abundance given the sample covariates. However, due to library preparation and under-sampling issues in sequencing, false zero or low counts could have been introduced into the data, creating another mode near zero in taxon j's abundance distribution. mbImpute models that mode using a Gamma distribution with mean α_j/β_j, which is close to zero.

mbImpute fits this mixture model to taxon j's abundances using the expectation-maximization (EM) algorithm to obtain the maximum likelihood estimates $\hat{p}_j$, $\hat{\alpha}_j$, $\hat{\beta}_j$, $\hat{\gamma}_j$, and $\hat{\sigma}_j^2$. Additional File 1: Fig. S20 shows four examples where the fitted mixture model well captures the bimodality of an individual taxon's abundance distribution. However, some taxa are observed to have an abundance distribution containing a single mode that can be well modelled by a normal distribution. When that occurs, the EM algorithm would encounter a convergence issue. To fix this, mbImpute uses a likelihood ratio test (LRT) to first decide if the Gamma-normal mixture model fits to taxon j's abundances significantly better than a normal distribution $Y_{ij} \sim \mathcal{N}\left(X_{i\cdot}^{\mathsf{T}}\eta_j, \omega_j^2\right)$ does. Given the maximum likelihood estimates $\hat{\eta}_j$ and $\hat{\omega}_j^2$ and under the assumption that Y_{ij}'s are all independent, the LRT statistic of taxon j is:

$$\Lambda_j = -2\ln \frac{\prod_{i=1}^{n} f_{\mathcal{N}}\left(Y_{ij}; X_{i\cdot}^{\mathsf{T}}\hat{\eta}_j, \hat{\omega}_j^2\right)}{\prod_{i=1}^{n}\left[\hat{p}_j \cdot f_{\Gamma}\left(Y_{ij}; \hat{\alpha}_j, \hat{\beta}_j\right) + (1-\hat{p}_j)\cdot f_{\mathcal{N}}\left(Y_{ij}; X_{i\cdot}^{\mathsf{T}}\hat{\gamma}_j, \hat{\sigma}_j^2\right)\right]} \,,$$

which asymptotically follows a Chi-square distribution with 3 degrees of freedom (because the mixture model has three more parameters than in the normal model) under the null hypothesis that the normal model is the correct model. We summarize the LRT p-values calculated on six real WGS datasets and observe that few taxa have p-values greater than 0.05 (see Additional File 1: Fig. S21a). Additional File 1: Fig. S21b shows the distribution of one randomly picked taxon with LRT p-value greater than 0.05 in each dataset; these taxa's log-transformed counts do not have a mode close to zero. If the LRT p-value $\leqslant 0.05$, mbImpute uses the mixture model to decide which abundances of taxon j need imputation. Specifically, mbImpute decides if Y_{ij} needs imputation based on the estimated posterior probability that Y_{ij} comes from the Gamma component:

$$d_{ij} = \frac{\hat{p}_j \cdot f_{\Gamma}\left(Y_{ij}; \hat{\alpha}_j, \hat{\beta}_j\right)}{\hat{p}_j \cdot f_{\Gamma}\left(Y_{ij}; \hat{\alpha}_j, \hat{\beta}_j\right) + (1-\hat{p}_j)\cdot f_{\mathcal{N}}\left(Y_{ij}; X_{i\cdot}^{\mathsf{T}}\hat{\gamma}_j, \hat{\sigma}_j^2\right)} \,,$$

where $f_{\Gamma}(\cdot; \hat{\alpha}_j, \hat{\beta}_j)$ and $f_{\mathcal{N}}(\cdot; X_{i\cdot}^{\mathsf{T}}\hat{\gamma}_j, \hat{\sigma}_j^2)$ represent the probability density functions of the estimated Gamma and normal components in the mixture model. Otherwise, if the LRT p-value > 0.05, mbImpute concludes that none of taxon j's abundances need imputation and sets $d_{1j} = \cdots = d_{nj} = 0$.

Based on the d_{ij}'s, mbImpute defines a set Ω of (sample, taxon) pairs whose abundances are unlikely missing and thus do not need imputation:

$$\Omega = \{(i,j) : d_{ij} < d_{\text{thre}}, i = 1, \ldots, n; j = 1, \ldots, m\} \,,$$

and a complement set Ω^c containing other (sample, taxon) pairs whose abundances need imputation:

$$\Omega^c = \{(i, j) : d_{ij} \geqslant d_{\text{thre}}, i = 1, \ldots, n; j = 1, \ldots, m\} \ .$$

Although $d_{\text{thre}} = 0.5$ is used as the default threshold on d_{ij}'s to decide the abundances that need imputation, mbImpute is fairly robust to this threshold choice because most d_{ij}'s are concentrated around 0 or 1. We show this phenomenon in Additional File 1: Fig. S22, which displays the distribution of all the d_{ij}'s in the data from Zeller et al. [33], Feng et al. [34], Yu et al. [35], Vogtmann et al. [36], Qin et al. [38], and Karlsson et al. [37].

To summarize, mbImpute does not impute all zeros in the taxon count matrix; instead, it first identifies the abundances that are likely missing using a mixture-modelling approach, and it then only imputes these values in the next step.

mbImpute step 2: imputation of the missing taxon abundances

In step 1, mbImpute identifies a set Ω of the (sample, taxon) pairs whose abundances do not need imputation. To impute the abundances in Ω^c, mbImpute first learns inter-sample and inter-taxon relationships from Ω by training a predictive model for Y_{ij}, the abundance of taxon j in sample i. The rationale is that taxon j should have similar abundances in similar samples, and that in every sample, the taxa similar to taxon j should have abundances similar to taxon j's abundance. In addition, sample covariates are assumed to carry predictive information of taxon abundances. Hence, for interpretability and stability reasons, mbImpute uses a linear model to combine the predictive power of similar taxa, similar samples, and sample covariates:

$$Y_{ij} = Y_{i\cdot}^{\mathsf{T}} \kappa_j + Y_{\cdot j}^{\mathsf{T}} \tau_i + X_{i\cdot}^{\mathsf{T}} \zeta_j + \epsilon_{ij} \ ,$$

where $Y_{i\cdot} \in \mathbb{R}^m_{>0}$ denotes the m taxa's abundances in sample i, $Y_{\cdot j} \in \mathbb{R}^n_{>0}$ denotes taxon j's abundances in the n samples, $X_{i\cdot} \in \mathbb{R}^q$ denotes sample i's covariates (including the intercept), and ϵ_{ij} is the error term. The parameters to be estimated include $\kappa_j \in \mathbb{R}^m$, $\tau_i \in \mathbb{R}^n$ and $\zeta_j \in \mathbb{R}^q$, $i = 1, \ldots, n$; $j = 1, \ldots, m$. Note that κ_j represents the m taxa's coefficients (i.e., weights) for predicting taxon j, with the j-th entry set to zero, so that taxon j would not predict itself; τ_i represents the n samples' coefficients (i.e., weights) for predicting sample i, with the i-th entry set to zero, so that sample i would not predict itself; ζ_j represents the coefficients of sample covariates for predicting taxon j. In the model, the first term $Y_{i\cdot}^{\mathsf{T}}\kappa_j$ borrows information across taxa, the second term $Y_{\cdot j}^{\mathsf{T}}\tau_i$ borrows information across samples, and the third term $X_{i\cdot}^{\mathsf{T}}\zeta_j$ borrows information from sample covariates. The total number of unknown parameters is $m(m-1)+n(n-1)+mq$, while our data $\mathbf{Y}$ and $\mathbf{X}$ together have $nm+nq$ values only. Given that often $m \gg n$, the parameter estimation problem is high dimensional, as the number of parameters far exceeds the number of data points. mbImpute performs regularized parameter estimation by using the Lasso-type ℓ_1 penalty, which leads to good prediction and simultaneously selects predictors (i.e., similar samples and similar taxa) to ease interpretation. That is, mbImpute estimates the above parameters by minimizing the following loss function:

$$L\left(\{\kappa_j,\zeta_j\}_{j=1}^m,\{\tau_i\}_{i=1}^n\right) := \sum_{(i,j)\in\Omega}\left[Y_{ij}-\left(Y_{i\cdot}^{\mathsf{T}}\kappa_j+Y_{\cdot j}^{\mathsf{T}}\tau_i+X_{i\cdot}^{\mathsf{T}}\zeta_j\right)\right]^2+\lambda\left(\sum_{j=1}^m\sum_{j'\neq j}^m D_{jj'}^{\psi}|\kappa_{jj'}|+\sum_{i=1}^n\sum_{i'\neq i}^n|\tau_{ii'}|\right),$$

where $\lambda,\psi \geqslant 0$ are tuning parameters chosen by cross-validation, $D_{jj'}$ represents the phylogenetic distance between taxa j and j', $\kappa_{jj'}$ represents the j'-th element of κ_j, and $\tau_{ii'}$ represents the i'-th element of τ_i. Here $D_{jj'}^{\psi}$, i.e., $D_{jj'}$ to the power of ψ, represents the penalty weight of $|\kappa_{jj'}|$ (in our R package implementation, the mbImpute function can take any distance matrix D as input that reflects the relationship among taxa specified by the user.) The intuition is that if two taxa are closer in the phylogenetic tree, they are more closely related in evolution and tend to have more similar DNA sequences and biological functions [111, 116], and thus we want to borrow more information between them. For example, if $D_{j_1j_2} > D_{j_1j_3}$, i.e., taxa j_1 and j_2 are farther away than taxa j_1 and j_3 in the phylogenetic tree, then the estimate of $\kappa_{j_1j_2}$ is more likely to be shrunk to zero than the estimate of $\kappa_{j_1j_3}$, and mbImpute would use taxon j_3's abundance more than taxon j_2's to predict taxon j_1's abundance. The tuning parameter ψ is introduced because the distance $D_{jj'}$, the number of branches connecting taxa j and j', may not be the best penalty weight for the prediction purpose. Choosing ψ by cross-validation is expected to enhance the predication accuracy.

mbImpute performs the estimation using the R package `glmnet` [90] and obtains the parameter estimates: $\hat{\kappa}_j \in \mathbb{R}^m$, $\hat{\tau}_i \in \mathbb{R}^n$, and $\hat{\zeta}_j \in \mathbb{R}^q$, $i = 1, \dots, n$; $j = 1, \dots, m$. Finally, for $(i,j) \in \Omega^c$, the abundance of taxon j in sample i is imputed as:

$$\hat{Y}_{ij} = Y_{i\cdot}^{\mathsf{T}} \hat{\kappa}_j + Y_{\cdot j}^{\mathsf{T}} \hat{\tau}_i + X_{i\cdot}^{\mathsf{T}} \hat{\zeta}_j \,,$$

and mbImpute does not alter Y_{ij} if $(i,j) \in \Omega$.

Note that mbImpute does not require the availability of the sample covariate matrix $\mathbf{X}$ or the phylogenetic tree. In the absence of sample covariates, the loss function becomes

$$L\left(\{\kappa_j\}_{j=1}^m, \{\tau_i\}_{i=1}^n\right) := \sum_{(i,j)\in\Omega} \left(Y_{ij} - \left(Y_{i\cdot}^{\mathsf{T}} \kappa_j + Y_{\cdot j}^{\mathsf{T}} \tau_i\right)\right)^2 + \lambda \left(\sum_{j=1}^m \sum_{j' \neq j}^m D_{jj'}^{\psi} |\kappa_{jj'}| + \sum_{i=1}^n \sum_{i' \neq i}^n |\tau_{ii'}| \right) ,$$

minimizing which returns the parameter estimates: $\hat{\kappa}_j \in \mathbb{R}^m$ and $\hat{\tau}_i \in \mathbb{R}^n$, $i = 1, \dots, n$; $j = 1, \dots, m$. Finally, for $(i,j) \in \Omega^c$, the abundance of taxon j in sample i is imputed as:

$$\hat{Y}_{ij} = Y_{i\cdot}^{\mathsf{T}} \hat{\kappa}_j + Y_{\cdot j}^{\mathsf{T}} \hat{\tau}_i \,,$$

and mbImpute does not alter Y_{ij} if $(i,j) \in \Omega$. In the absence of the phylogenetic tree, mbImpute sets $D_{jj'} = 1$ for all $j \neq j' \in \{1, \dots, m\}$.

When m is large, mbImpute does not estimate $m(m-1) + n(n-1) + mq$ parameters but uses the following strategy to increase its computational efficiency. For each taxon j, mbImpute selects the k taxa closest to it (excluding itself) in phylogenetic distance and sets the other $(m-k)$ taxa's coefficients in κ_j to zero. This strategy reduces the number of parameters to $mk + n(n-1) + mq$ and decreases the computational complexity from $O(m^2)$ to $O(m)$.

In summary, mbImpute step 2 includes two phases: training on Ω and prediction (imputation) on Ω^c, as illustrated in Additional File 1: Fig. S1.

2.5 Availability of Data and Materials

2.5.1 Imputation methods

We compare mbImpute with five existing imputation methods designed for non-microbiome data: softImpute and four scRNA-seq imputation methods (scImpute, SAVER, MAGIC, and ALRA). All these imputation methods take a count matrix as input and ouput an imputed count matrix with the same dimensions.

1. softImpute

We use `R` package `softImpute` (version 1.4) and the following command to impute a taxon count matrix (a sample-by-taxon matrix):

```
complete(taxa_count_matrix, softImpute(taxa_count_matrix, rank.max = cv.rankmax))
```

where `rank.max` is chosen by 10-fold cross-validation.

2. scImpute

We use `R` package `scImpute` (version 0.0.9) with the input as a taxon-by-sample count matrix (transpose of the matrix in Fig. 1):

```
scimpute(count_path = "taxa_count_matrix_trans.csv", Kcluster = 1, out_dir = "sim_imp")
```

where `taxa_count_matrix_trans.csv` is the input file containing the transposed taxon count matrix.

3. SAVER

We use `R` package `SAVER` (version 1.1.2) with the input as a taxon-by-sample count matrix (transpose of the matrix in Fig. 1):

```
saver(t(taxa_count_matrix), ncores = 1, estimates.only = TRUE)
```

4. MAGIC

We use Python package MAGIC (version 2.0.3) and the following commands to impute a taxon count matrix:

```
magic_op = magic.MAGIC()
magic_op.set_params(n_pca = 40)
magic_op.fit_transform(taxa_count_matrix)
```

5. ALRA

We apply R functions normalize_data, choose_k, and alra, which were released on Aug 10, 2019 at https://github.com/KlugerLab/ALRA, and the following commands to impute a taxon count matrix:

```
normalized_mat = normalize_data(taxa_count_matrix)
k_chosen = choose_k(normalized_mat, K = 49, noise_start = 44)$k
alra(normalized_mat, k = k_chosen)$A_norm_rank_k_cor_sc
```

2.5.2 DA analysis methods

In simulation studies, we compare five existing DA methods: the Wilcoxon rank-sum test, ANCOM, metagenomeSeq, DESeq2-phyloseq, and Omnibus test. We apply each method to taxon counts, with or without using mbImpute as a preceding step. When mbImpute is used as a preceding step, we call the resulting method a mbImpute-empowered DA method. In real data studies, we compare mbImpute-empowered DESeq2-phyloseq and mbImpute-empowered Omnibus test with DESeq2-phyloseq and Omnibus test, respectively. Each method calculates a p-value for each taxon and identifies the DA taxa by setting a p-value threshold to control the FDR. See Additional File 1 for the statistical definitions of DA taxa.

1. Wilcoxon rank-sum test

We implement the Wilcoxon rank-sum test using the `R` function `pairwise.wilcox.test` in the package `stats` (version 3.5.1). For each taxon, we perform the test on its counts in two sample groups to obtain a p-value, which suggests if this taxon is DA between the two groups. In simulations, we use the following command to implement a two-sided test for each taxon:

```
pairwise.wilcox.test(x = taxon_counts, g = condition, p.adjust.method = "none")
```

2. ANCOM

We use the `ANCOM.main` function released on Sep 27, 2019 at https://github.com/FrederickHuangLin ANCOM [46]. Since this function does not provide an option for a one-sided test, we use its default settings and report its identified DA taxa based on a two-sided test with a significance level 0.05 (`sig = 0.05`), in both simulations and real data analysis. We note that no external FDR control is implemented. Specifically, we use the following command to obtain the result of ANCOM:

```
ANCOM.main(taxa_count_matrix, covariate_matrix, adjusted = F, repeated = F, main.var
= "condition", adj.formula = NULL, repeat.var = NULL, multcorr = 2, sig = 0.05,
prev.cut = 0.90, longitudinal = F)
```

where `taxa_count_matrix` is a sample-by-taxon count matrix and `covariate_matrix` is a sample-by-covariate matrix, same as the input of mbImpute.

3. metagenomeSeq

We use two `R` packages, `metagenomeSeq` (version 1.28.2) and `phyloseq` (version 1.30.0). Specifically, we use the following command to obtain the result:

```
mseq_obj <- phyloseq_to_metagenomeSeq(physeq2)
pd <- pData(mseq_obj)
mod <- model.matrix(∼ 1 + condition, data = pd)
```

```
ran_seq <- fitFeatureModel(mseq_obj, mod)
```

where `physeq2` is an object created from a count matrix and sample covariates using the `phyloseq` package.

4. DESeq2-phyloseq

We use the `DESeq2` (version 1.26.0) package combined with `phyloseq` (version 1.30.0). Specifically, we use the following command to obtain the result of DESeq2:

```
Deseq2_obj <- phyloseq_to_deseq2(physeq2, ∼ condition)
results <- DESeq(Deseq2_obj, test="Wald", fitType="parametric")
```

where `physeq2` is an object created from a count matrix and sample covariates using the `phyloseq` package.

5. Omnibus test

We use the R package `mbzinb` (version 0.2). Specifically, we use the following command to obtain the result of Omnibus test:

```
mbzinb_data <- mbzinb.dataset(taxa_count_matrix, covariate_matrix)
mbzinb_test_result <- mbzinb.test(mbzinb_data, group = "condition")
```

For the Wilcoxon rank-sum test, MetagenomeSeq, DESeq2-phyloseq, and Omnibus test, after obtaining the p-values of all taxa and collecting them into a vector `p_values`, we adjust them for FDR control using the `R` function `p.adjust` in the package `stats` (version 3.5.1):

```
p.adjust(p_values, method = "fdr")
```

Then we set the FDR threshold to 0.05 in both simulation and real data analysis. The taxa whose adjusted p-values do not exceed this threshold are called DA. ANCOM directly outputs the DA taxa.

2.5.3 Classification

We use a 5-fold cross-validated precision-recall area under the curve (PR-AUC) to evaluate the classification results using identified DA taxa as features and diseased / control group as classification labels. We use the R package randomForest (version 4.6-14) to perform the random forest classification and the R package PRROC (version 1.3.1) to calculate the PR-AUC.

2.5.4 T2D and CRC datasets

We apply mbImpute to six real microbiome datasets, each corresponding to an independent study on the relationship between microbiomes and the occurrence of a human disease. All the six datasets were generated by the whole genome shotgun sequencing and are available in the R package curatedMetagenomicData [103]. We compare the disease-enriched DA taxa identified by DESeq2-phyloseq and mbImpute-empowered DESeq2-phyloseq. Below is the description of the six datasets and our analysis.

Two T2D datasets [37, 38]. The Karlsson *et al.* dataset contains 145 fecal samples from 70-year-old European women to study the relationship between human gut microbiome compositions and T2D status. The samples/subjects are in three groups: 53 women with T2D, 49 women with impaired glucose tolerance (IGT), and 43 women as the normal control (CON). The eleven sample covariates include the subject's age, the number of reads in each sample, the triglycerides level, the hba1c level, the ldl (low-density lipoprotein cholesterol) level, the c peptide level, the cholesterol level, the glucose level, the adiponectin level, the hscrp level, and the leptin level. In our analysis, we consider the 147 species-level taxa (having at least 10% non-zero counts in both T2D and CON groups) with phylogenetic information available in the R package curatedMetagenomicData. Qin et al. [38] performed deep shotgun metagenome sequencing on 369 Chinese T2D patients and non-diabetic controls (CON). The two sample covariates include the body mass index, and the number of reads in each sample. We analyze 156 species-level taxa (having at least 10% non-zero counts in both T2D and CON groups) with phylogenetic information. From both datasets, we identify DA taxa by comparing the T2D and CON groups.

Four CRC datasets [33–36]. Zeller et al. [33] and Feng et al. [34] studied CRC-related microbiomes in three conditions: CRC, small adenoma (ADE; diameter $<$ 10 mm), and control (CON). Zeller et al. [33] sequenced the fecal samples of patients across two countries (France and Germany) in these three groups: 191 patients with CRC, 66 patients with ADE, and 42 patients in CON. The sample covariates include the subject's age category, gender, body mass index and country, and the number of reads in each sample. We include 188 species-level taxa (having at least 10% non-zero counts in both CRC and CON groups) with phylogenetic information. Feng et al. [34] sequenced samples from 154 human subjects aged between 45–86 years old in Australia, including 46 patients with CRC, 47 patients with ADE, and 61 in CON. The sample covariates include the subject's age category, gender, body mass index, and number of reads in each sample. We include 182 species-level taxa that have at least 10% non-zero counts in both CRC and CON groups. Yu et al. [35] and Vogtmann et al. [36] studied CRC-related microbiomes in two conditions: CRC vs. CON. In detail, Yu et al. [35] sequenced 128 Chinese samples, including 75 patients with CRC and 53 patients in CON. The only sample covariate is the number of reads in each sample. We study 173 species-level taxa that have at least 10% non-zero counts in both CRC and CON groups. Vogtmann et al. [36] included 104 samples from Washington DC and sequenced their fecal samples, including 52 with CRC and 52 in CON. The sample covariates include the subject's age category, gender, body mass index, and the number of reads in each sample. We include 167 species-level taxa that have at least 10% non-zero counts in both CRC and CON groups. From all the four datasets, we identify DA taxa by comparing the CRC and CON groups.

2.5.5 16S rRNA sequencing datasets

We include two 16S rRNA sequencing datasets from the `R` package `HMP16SData` [151] (version 1.6.0). The two datasets correspond to the healthy human stool samples and healthy human oral samples. The healthy stool 16S dataset includes 187 samples and 43140 OTUs, and the healthy oral 16S data includes 190 samples and 43140 OTUs.

2.5.6 Software and code

The mbImpute R package is available at https://github.com/ruochenj/mbImpute [152]. The source code and data for reproducing the results are available at https://doi.org/10.5281/zenodo.4840266 [153]. Both the R package and the source code are under the MIT license.

Acknowledgements

The authors would like to thank Dr. Hongzhe Li at University of Pennsylvania for pointing us to this research direction. The authors also appreciate the comments and feedback from the members of the Junction of Statistics and Biology at UCLA (http://jsb.ucla.edu).

Figures

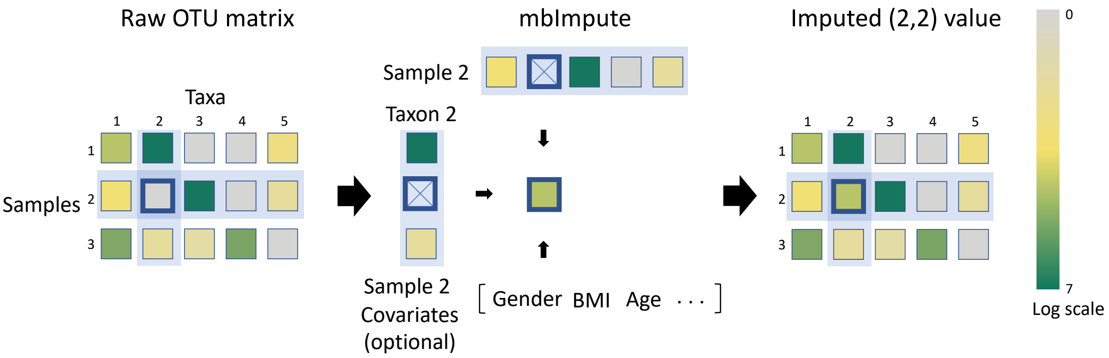


Figure 2.1: An illustration of mbImpute. After mbImpute identifies likely non-biological zeros, it imputes them (e.g. the abundance of taxon 2 in sample 2) by jointly borrowing information from similar samples, similar taxa, and sample covariates if available (details in Methods).

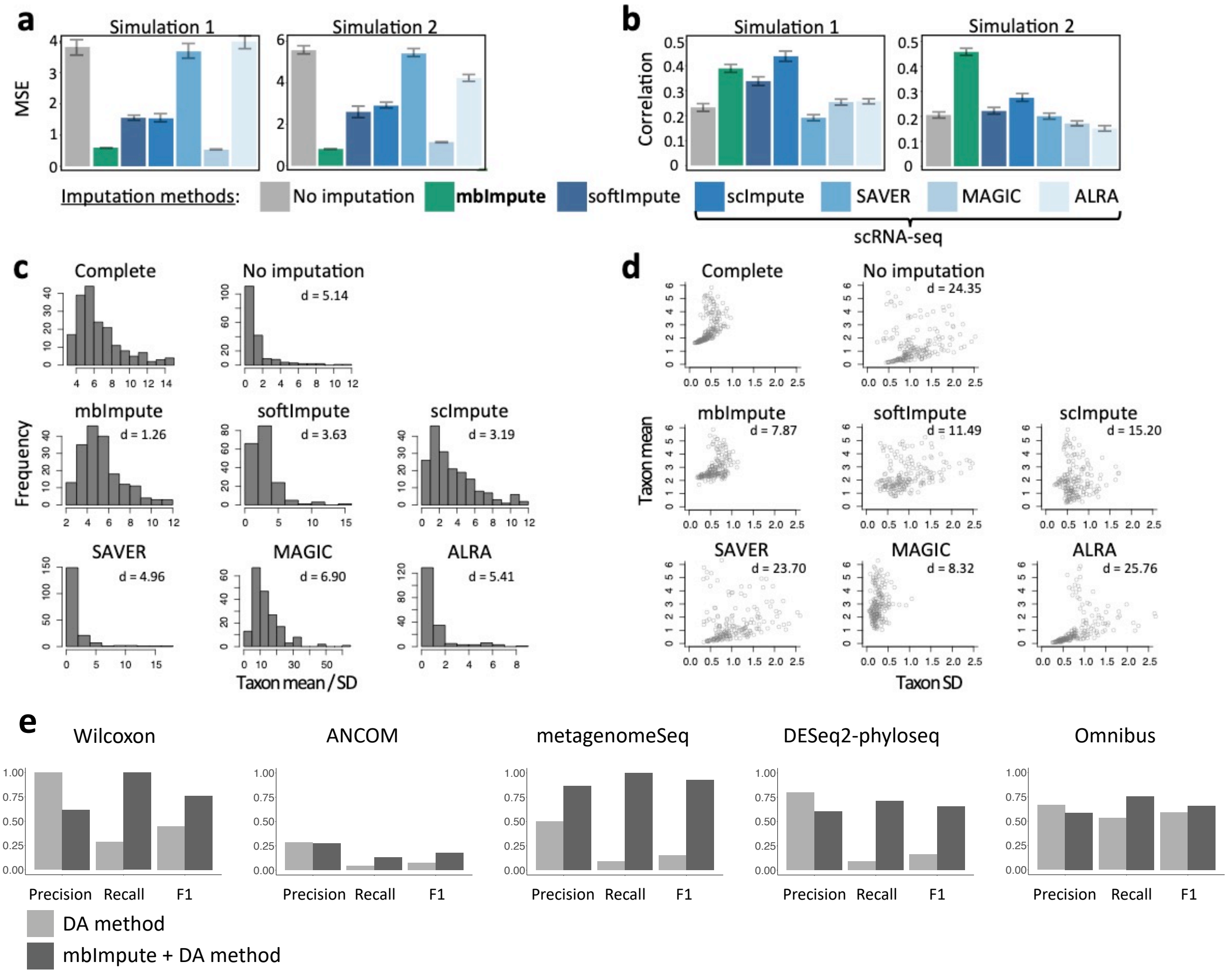


Figure 2.2: mbImpute outperforms state-of-the-art imputation methods designed for non-microbiome data and enhances the identification of DA taxa. **(a)** Mean squared error (MSE) and **(b)** mean Pearson correlation of taxon abundances between the complete data and the zero-inflated data ("No imputation," the baseline) or the imputed data by each imputation method (mbImpute, softImpute, scImpute, SAVER, MAGIC, and ALRA) in Simulations 1 and 2 (see Additional File 1). **(c)-(d)** For each taxon, the mean and standard deviation (SD) of its abundances are calculated for the complete data, the zero-inflated data, and the imputed data by each imputation method in Simulation 1; (c) shows the distributions of the taxon mean / SD and the Wasserstein distance between every distribution and the complete distribution; (d) shows the taxa in two coordinates, mean vs. SD, and the average Euclidean distance between the taxa in every (zero-inflated or imputed) dataset and the complete data in these two coordinates. **(e)** Accuracy (Precision, recall, and F_1 scores) of five DA methods (Wilcoxon rank-sum test, ANCOM, metagenomeSeq, DESeq2-phyloseq, and Omnibus test) with the FDR threshold 0.05 on raw data (light color) and imputed data by mbImpute (dark color) in the 16S data simulation.

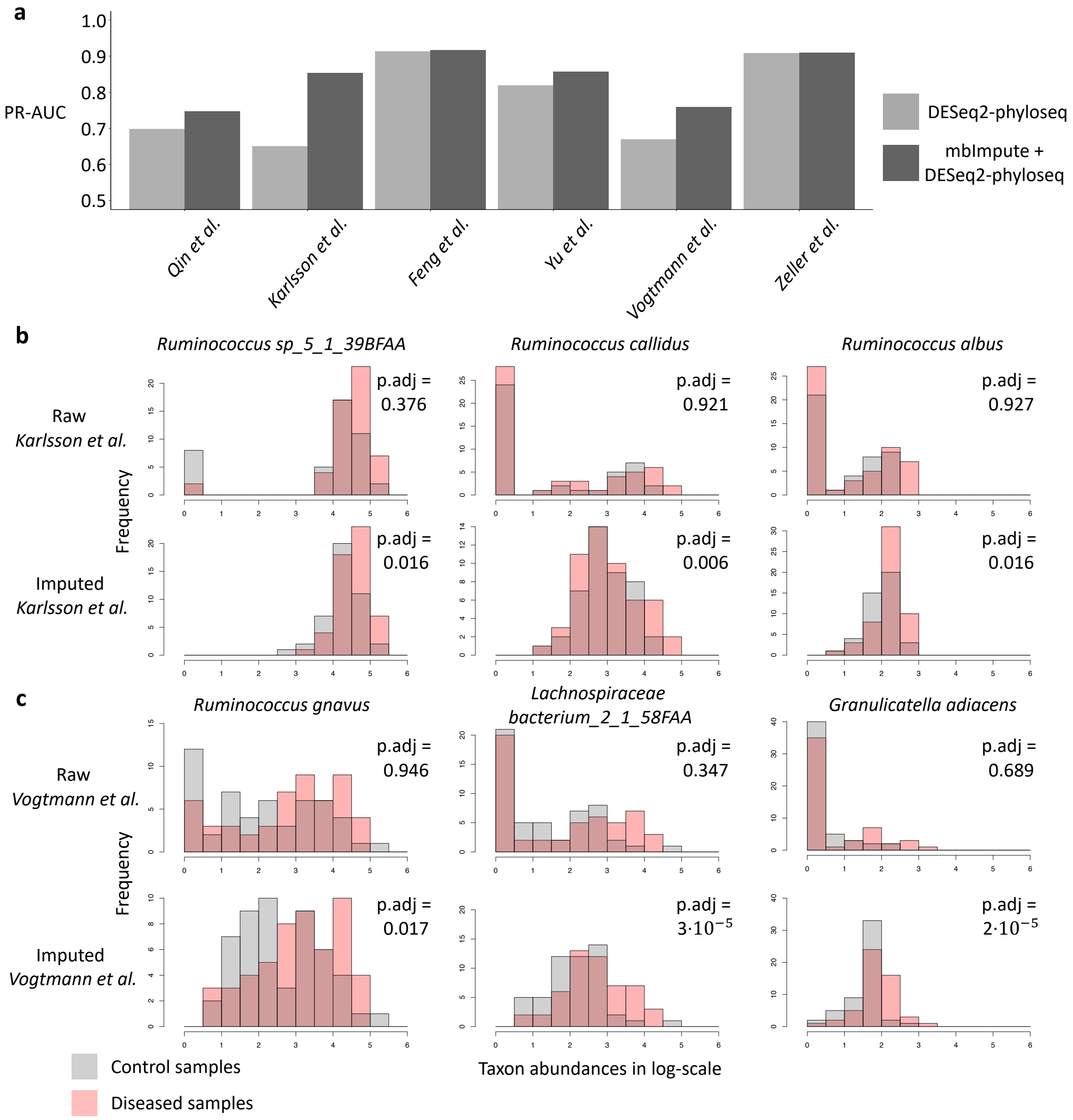


Figure 2.3: mbImpute empowers DESeq2-phyloseq in identifying DA taxa. **(a)** The barplots show classification accuracy, measured by 5-fold cross-validated precision-recall area under the curve (PR-AUC), by the random forest algorithm for predicting samples' disease conditions in two T2D datasets (Qin et al. and Karlsson et al.) and four CRC datasets (Feng et al., Vogtmann et al., Yu et al., and Zeller et al.). The features are the DA taxa detected by DESeq2-phyloseq (light color) or mbImpute-empowered DESeq2-phyloseq (dark color; labeled as mbImpute + DESeq2-phyloseq). **(b)** The histograms show the distributions of three taxa in control and T2D samples in Karlsson et al. before and after mbImpute is applied. The three taxa, *Ruminococcus sp_5_1_39BFAA*, *Ruminococcus callidus*, and *Ruminococcus albus*, are identified as enriched in T2D samples only after imputation. **(c)** The histograms show the distributions of three taxa in control and CRC samples in Vogtmann et al. before and after mbImpute is applied. The three taxa, *Ruminococcus gnavus*, *Lachnospiraceae bacterium_2_1_58FAA*, and *Granulicatella adiacens*, are identified as enriched in CRC samples only after imputation. In (b) and (c), adjusted p-values calculated by DESeq2-phyloseq are listed.

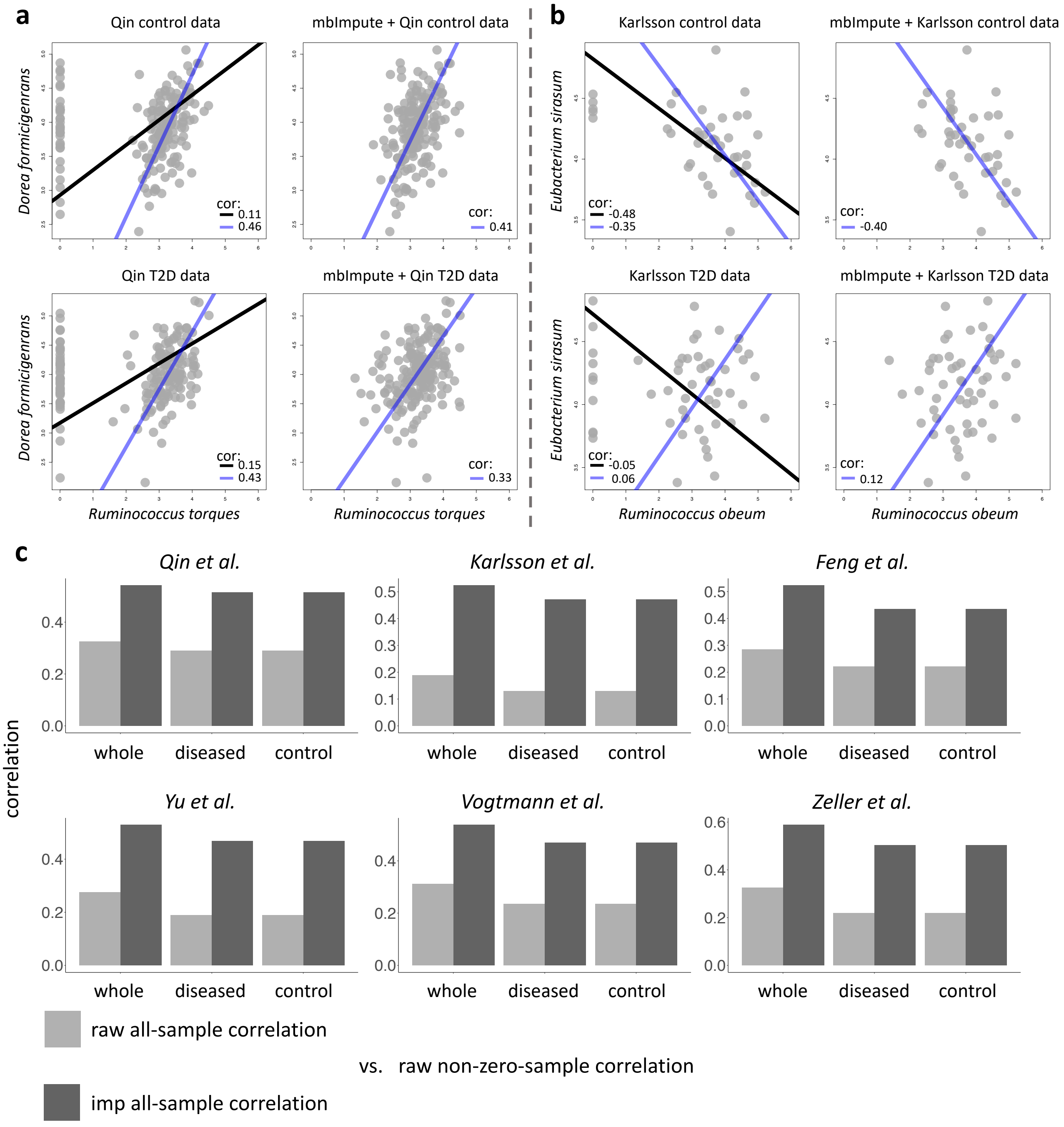
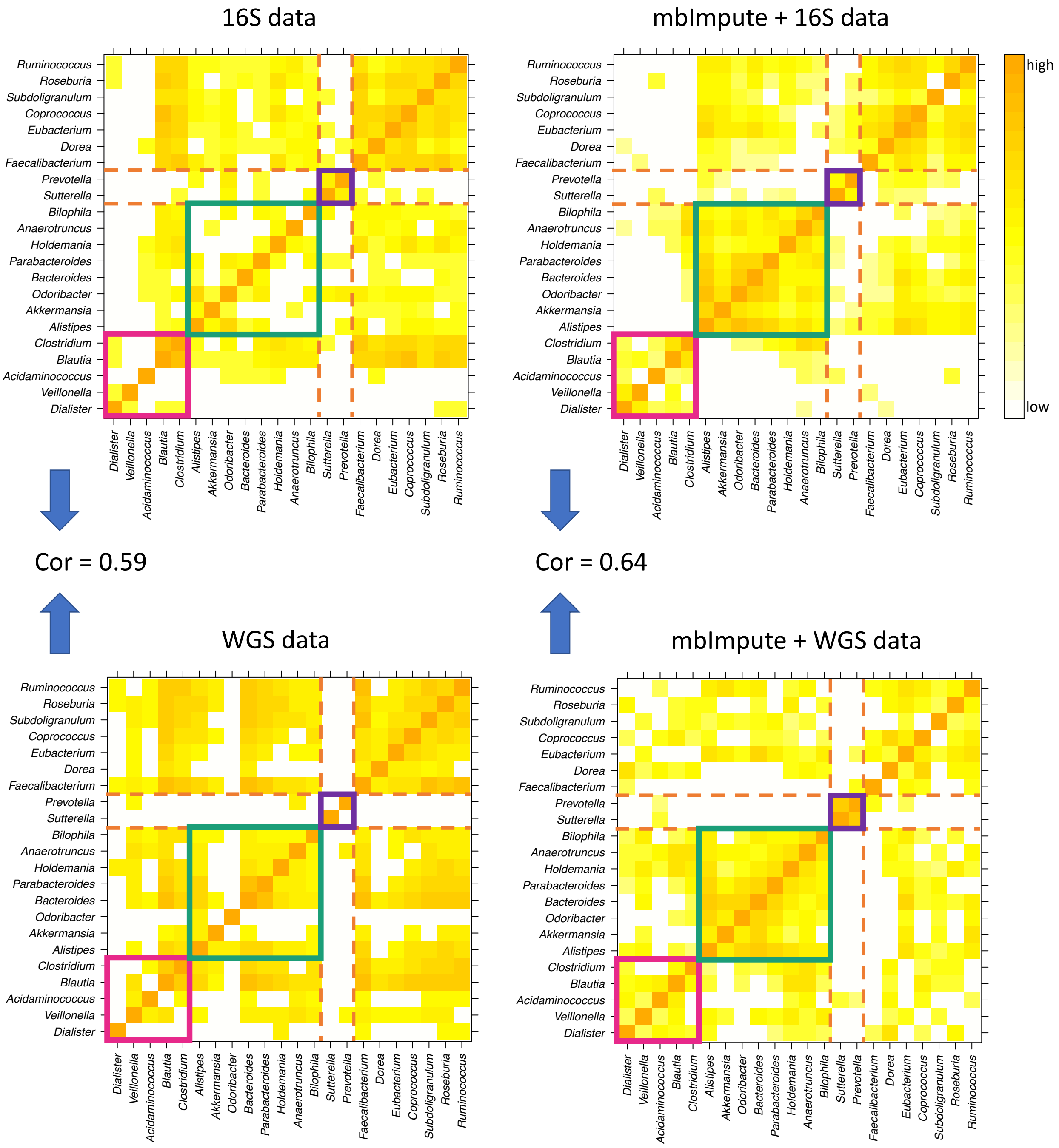


Figure 2.4: mbImpute preserves distributional characteristics of taxa's non-zero abundances. **(a)** Top: two scatter plots show the relationship between the abundances of *Dorea formicigenerans* and *Ruminococcus torques* in Qin et al.'s control samples, with or without using mbImpute as a preceding step. The left plot shows two standard major axis (SMA) regression lines and two corresponding Pearson correlations based on the raw data (black: based on all the samples; blue: based on only the samples where both taxa have non-zero abundances). The right plot shows the SMA regression line (blue) and the Pearson correlation using all the samples in the imputed data. Bottom: two scatter plots for the same two taxa in Qin et al.'s T2D samples, with lines and legends defined the same as in the top panel. **(b)** Four scatter plots show the SMA regression lines and correlations between *Eubacterium sirasum* and *Ruminococcus obeum* in Karlsson et al.'s control and T2D samples, with lines and legends defined the same as in (a). **(c)** Each bar shows the Pearson correlation between taxon-taxon correlations in raw data (light gray) or imputed data (dark gray) using all samples and taxon-taxon correlations in raw data using non-zero samples only. The two correlations are calculated for two T2D datasets and four CRC datasets using diseased samples, control samples, and whole data.



16S data

mbImpute + 16S data

WGS data

mbImpute + WGS data

Figure 2.5: mbImpute improves the similarity of taxon-taxon correlations between 16S and WGS data of microbiomes in healthy human stool samples. Four Pearson correlation matrices are calculated based on a common set of genus-level taxa's abundances in 16S and WGS data, with or without using mbImpute as a preceding step. Before imputation, the Pearson correlation between the two correlation matrices is 0.59, and this correlation increases to 0.64 after imputation. For illustration purposes, each heatmap shows square roots of Pearson correlations, with the bottom 40% of values truncated to 0. The magenta, green, and purple squares highlight three taxon groups, each of which contains strongly correlated taxa and is consistent between the 16S and WGS data after imputation.

2.6 Supplementary materials

Simulation 1 for benchmarking imputation methods

To compare mbImpute with existing imputation methods developed for non-microbiome data, we generated microbiome abundances from a generative model fitted to the T2D data with 53 subjects and 344 taxa [37]. Below we describe the data generation process step by step.

Complete data generation

1. We followed the data pre-processing steps of mbImpute (see Methods in the main text) to convert the OTU count matrix to log-transformed normalized abundances. Then we removed the taxa with greater than 95% of zero counts (equivalently, $\log_{10}(1.01)$ abundances) across subjects and kept 193 taxa. We denote the abundance matrix after this filtering step by $\mathbf{Y} = (Y_{ij}) \in \mathbb{R}^{n \times m}$, where $n = 53$ and $m = 193$. We also collected the sample covariate matrix $\mathbf{X} \in \mathbb{R}^{n \times q}$, where $q = 12$.
2. Following step 1 of mbImpute (see Methods in the main text), we identified a set Ω of (sample, taxon) pairs whose abundances are unlikely missing and thus do not need imputation.
3. Following step 2 of mbImpute (see Methods in the main text), we fit the following model to the (sample, taxon) pairs in Ω:

$$Y_{ij} = Y_{i\cdot}^{\mathsf{T}} \kappa_j + Y_{\cdot j}^{\mathsf{T}} \tau_i + X_{i\cdot}^{\mathsf{T}} \zeta_j + \epsilon_{ij} \,,$$

   by minimizing the loss function

$$\sum_{(i,j)\in\Omega} \left(Y_{ij} - \left(Y_{i\cdot}^{\mathsf{T}} \kappa_j + Y_{\cdot j}^{\mathsf{T}} \tau_i + X_{i\cdot}^{\mathsf{T}} \zeta_j\right)\right)^2 + \lambda \left(\sum_{j=1}^{m} \sum_{j' \neq j}^{m} D_{jj'}^{\psi} |\kappa_{jj'}| + \sum_{i=1}^{n} \sum_{i' \neq i}^{n} |\tau_{ii'}| \right)$$

   and obtaining the parameter estimates: $\hat{\kappa}_j \in \mathbb{R}^m$, $\hat{\tau}_i \in \mathbb{R}^n$, and $\hat{\zeta}_j \in \mathbb{R}^q$, $i = 1, \ldots, n$;

$j = 1, \ldots, m$. The tuning parameters $\lambda, \psi \geqslant 0$ were chosen by cross-validation. Note that we constrained κ_j to have only 5 non-zero entries corresponding to the 5 taxa closest to taxon j in phylogenetic distance, i.e., $\{j' : D_{jj'}$ is among the five smallest of all $D_{jr}, r \neq j\}$.

4. We generated the complete, log-transformed abundance of taxon j in sample i, as

$$Y_{ij}^{\text{comp}} = Y_{i\cdot}^{\mathsf{T}} \hat{\kappa}_j + Y_{\cdot j}^{\mathsf{T}} \hat{\tau}_i + X_{i\cdot}^{\mathsf{T}} \hat{\zeta}_j \, .$$

   We denote the resulting matrix $\mathbf{Y}^{\text{comp}} = (Y_{ij}^{\text{comp}})$ as the complete data that contain log-transformed abundances without missing values.

Zero-inflated data generation

Next, we introduced zero inflation to $\mathbf{Y}^{\text{comp}}$ and generated $\mathbf{Y}^{\text{zi}}$ by mimicking real data as follows. With the identified Ω, we calculated z_k^{real} (taxon k's proportion of likely false zeros across samples) and μ_k^{real} (taxon k's average abundance after we excluded likely false zeros) for each taxon k in Karlsson *et al.*'s data $\mathbf{Y}$, $k = 1, \ldots, m$. Next we introduced zeros into $\mathbf{Y}^{\text{comp}}$ in the following non-parametric way for each taxon j, $j = 1, \ldots, m$:

1. We calculated taxon j's average abundance in $\mathbf{Y}^{\text{comp}}$ as the mean of column j, denoted by μ_j^{comp};
2. We randomly sampled a value from $\{z_k^{\text{real}} : \mu_k^{\text{real}} \in (\mu_j^{\text{comp}} - 0.5, \mu_j^{\text{comp}} + 0.5)\}$ and denoted it as z_j^{zi}, i.e., the proportion of false zeros to be introduced into taxon j's abundances in $\mathbf{Y}^{\text{comp}}$;
3. We randomly drew false zero indicators $I_{ij} \sim \text{Bernoulli}(z_j^{\text{zi}})$ independently for sample $i = 1, \ldots, n$;
4. We set $Y_{ij}^{\text{zi}} = \max\left(Y_{ij}^{\text{comp}} \cdot I_{ij}, \, \log_{10}(1.01)\right)$.

The reason why we set the minimum value of $\mathbf{Y}^{\text{zi}}$ to $\log_{10}(1.01)$ is that mbImpute step 1 sets the minimum log-transformed abundance to $\log_{10}(1.01)$ to facilitate the fitting of the Gamma-normal mixture model.

Evaluation criteria for imputation accuracy

After applying an imputation method to $\mathbf{Y}^{\text{zi}}$ (mbImpute also used $\mathbf{X}$), we obtained $\mathbf{Y}^{\text{imp}}$ and evaluated the imputation performance by the following three criteria. Note that mbImpute identified 8 taxa with too many zeros in $\mathbf{Y}^{\text{zi}}$ and excluded them from imputation. For a fair comparison, we excluded these 8 taxa from the calculation of the evaluation criteria for every imputation method, so m was reduced to $193 - 8 = 185$ in the following.

1. Mean squared error (MSE) between $\mathbf{Y}^{\text{imp}}$ and $\mathbf{Y}^{\text{comp}}$:

$$\text{MSE} = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}(Y_{ij}^{\text{comp}} - Y_{ij}^{\text{imp}})^2\,,$$

   which is shown in Fig. 2a.

2. Pearson correlation between $Y_{\cdot j}^{\text{imp}}$ and $Y_{\cdot j}^{\text{comp}}$ for $j = 1, \ldots, m$. The mean of these m correlations is shown in Fig. 2b.

3. Mean and standard deviation (SD) of taxon j in the imputed data vs. those in the complete data:

$$\bar{Y}_{\cdot j}^{\text{imp}} = \frac{1}{n}\sum_{i=1}^{n} Y_{ij}^{\text{imp}} \quad \text{vs.} \quad \bar{Y}_{\cdot j}^{\text{comp}} = \frac{1}{n}\sum_{i=1}^{n} Y_{ij}^{\text{comp}}\,,$$

$$\text{sd}_{\cdot j}^{\text{imp}} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(Y_{ij}^{\text{imp}} - \bar{Y}_{\cdot j}^{\text{imp}}\right)^2} \quad \text{vs.} \quad \text{sd}_{\cdot j}^{\text{comp}} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(Y_{ij}^{\text{comp}} - \bar{Y}_{\cdot j}^{\text{comp}}\right)^2}\,,$$

   $j = 1, \ldots, m$. In Fig. 2c, we computed the Wasserstein distance between the distribution of $\left\{\bar{Y}_{\cdot 1}^{\text{imp}}/\text{sd}_{\cdot 1}^{\text{imp}}, \ldots, \bar{Y}_{\cdot m}^{\text{imp}}/\text{sd}_{\cdot m}^{\text{imp}}\right\}$ vs. that of $\left\{\bar{Y}_{\cdot 1}^{\text{comp}}/\text{sd}_{\cdot 1}^{\text{comp}}, \ldots, \bar{Y}_{\cdot m}^{\text{comp}}/\text{sd}_{\cdot m}^{\text{comp}}\right\}$. In Fig. 2d, we computed the Euclidean distance between the imputed data and the complete data as

$$\sqrt{\sum_{j=1}^{m}\left(\bar{Y}_{\cdot j}^{\text{imp}} - \bar{Y}_{\cdot j}^{\text{comp}}\right)^2 + \left(\text{sd}_{\cdot j}^{\text{imp}} - \text{sd}_{\cdot j}^{\text{comp}}\right)^2}\,.$$

Simulation 2 for benchmarking imputation methods based on real WGS data

To further benchmark mbImpute against existing imputation methods developed for non-microbiome data, we used a semi-simulation approach by obtaining a subset of microbiome WGS data with at least 86% non-zeros from a T2D dataset composed of 344 subjects and 469 taxa [38]. Below we describe the data generation process step by step.

Complete data generation

1. We followed the data pre-processing steps of mbImpute (see Methods in the main text) to convert the OTU count matrix to log-transformed normalized abundances, denoted by $\mathbf{Y} = (Y_{ij}) \in \mathbb{R}^{344\times 469}$. We also collected the sample covariate matrix $\mathbf{X} \in \mathbb{R}^{344\times 3}$.
2. Following step 1 of mbImpute (see Methods in the main text), we identified a set Ω of (sample, taxon) pairs whose abundances are unlikely missing and thus do not need imputation.
3. For each taxon, we checked if it has at least 43 non-zero counts, or equivalently, at least 43 abundances greater than $\log_{10}(1.01)$, across the 344 subjects. If yes, we kept the taxon; otherwise, we filtered it out. This step left us with 145 taxa.
4. Denote the index set of samples where taxon j has non-zero counts as

   $$\mathcal{I}_j = \{i : Y_{ij} > \log_{10}(1.01),\, i = 1, \ldots, 344\}$$

   Note that all $|\mathcal{I}_j| \geqslant 43$ due the the filtering step. Then we constructed a new index set of samples $\mathcal{I}'_j$, with $|\mathcal{I}'_j| = 50$, in the following way:

   (a) if $|\mathcal{I}_j| < 50$, $\mathcal{I}'_j = \mathcal{I}_j \cup$ a random subset of $\mathcal{I}_j^c$ with size $50 - |\mathcal{I}_j|$;

   (b) otherwise, $\mathcal{I}'_j =$ a random subset of $\mathcal{I}_j$ with size 50.
5. We constructed the complete data by column-stacking taxon j's abundances in the samples in $\mathcal{I}'_j$, with the 50 samples randomly ordered, $j = 1, \ldots, 145$, resulting in

$\mathbf{Y}^{\text{comp}} \in \mathbb{R}^{50 \times 145}$, which we assumed to have no missing values. By our construction, $\mathbf{Y}^{\text{comp}}$ has at least 86% values greater than $\log_{10}(1.01)$, corresponding to non-zero values on the count scale.

6. We constructed the sample covariate matrix $\mathbf{X}^{\text{comp}} \in \mathbb{R}^{50 \times 3}$ as follows: for sample i and covariate r ($i = 1, \ldots, 50$; $r = 1, 2, 3$),

   (a) if covariate r is categorical (e.g., gender), X_{ij}^{comp} was decided by the majority vote of the j-th covariate values of the original samples rearranged into the i-th row of $\mathbf{Y}^{\text{comp}}$: majority($\{X_{i'j} :$ sample i' is in the i-th row of $\mathbf{Y}^{\text{comp}}\}$);

   (b) if covariate r is numerical (e.g., BMI), X_{ij}^{comp} was set to the average of the j-th covariate values of the original samples rearranged into the i-th row of $\mathbf{Y}^{\text{comp}}$: average($\{X_{i'j} :$ sample i' is in the i-th row of $\mathbf{Y}^{\text{comp}}\}$).

Zero-inflated data generation for mimicking WGS data

Next, we introduced zero inflation to $\mathbf{Y}^{\text{comp}}$ and generated $\mathbf{Y}^{\text{zi}}$ by mimicking real WGS data as follows. With the identified Ω, we calculated z_k^{real} (taxon k's proportion of likely false zeros across samples) and μ_k^{real} (taxon k's average abundance after we excluded likely false zeros) for each taxon k in Qin *et al.*'s data, $k = 1, \ldots, 469$. Next we introduced zeros into $\mathbf{Y}^{\text{comp}}$ in the following non-parametric way for each taxon j, $j = 1, \ldots, 145$:

1. We calculated taxon j's average abundance in $\mathbf{Y}^{\text{comp}}$ as the mean of column j, denoted by μ_j^{comp};
2. We randomly sampled a value from $\{z_k^{\text{real}} : \mu_k^{\text{real}} \in (\mu_j^{\text{comp}} - 0.5, \mu_j^{\text{comp}} + 0.5)\}$ and denoted it as z_j^{zi}, i.e., the proportion of false zeros to be introduced into taxon j's abundances in $\mathbf{Y}^{\text{comp}}$;
3. We randomly drew false zero indicators $I_{ij} \sim \text{Bernoulli}(z_j^{\text{zi}})$ independently for sample $i = 1, \ldots, 50$;
4. We set $Y_{ij}^{\text{zi}} = \max\left(Y_{ij}^{\text{comp}} \cdot I_{ij}, \log_{10}(1.01)\right)$.

The reason why we set the minimum value of $\mathbf{Y}^{\text{zi}}$ to $\log_{10}(1.01)$ is that mbImpute step 1 sets the minimum log-transformed abundance to $\log_{10}(1.01)$ to facilitate the fitting of the Gamma-normal mixture model.

Evaluation criteria for imputation accuracy

After applying an imputation method to $\mathbf{Y}^{\text{zi}}$ (mbImpute also used $\mathbf{X}^{\text{comp}}$), we obtained $\mathbf{Y}^{\text{imp}}$ and evaluated the imputation performance by the following three criteria, where $n = 50$ and $m = 145$.

1. Mean squared error (MSE) between $\mathbf{Y}^{\text{imp}}$ and $\mathbf{Y}^{\text{comp}}$:

$$\text{MSE} = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}(Y_{ij}^{\text{comp}} - Y_{ij}^{\text{imp}})^2 \,.$$

2. Pearson correlation between $Y_{\cdot j}^{\text{imp}}$ and $Y_{\cdot j}^{\text{comp}}$ for $j = 1, \ldots, m$. The mean of these m correlations is shown in Fig. 2b.

Simulation 3 for evaluating the accuracy and robustness of mbImpute

To evaluate the accuracy and robustness of mbImpute, we simulated microbiome abundances based on real data [33] using four schemes. We set the number of samples to $n = 50$ and the number of taxa to $m = 200$. Under each scheme, we first generated the complete data, as described below.

Scheme 1: Covariate

1. We randomly sampled (without replacement) 50 samples' six covariates (gender, age, disease subtype, BMI, country, and number of reads) from Zeller *et al.*'s data. Then we added a 50-length vector of ones to the sampled covariates to form a sample covariate matrix $\mathbf{X} \in \mathbb{R}^{n \times q}$, with $q = 7$.
2. We randomly sampled (without replacement) 200 taxa' normalized and log-transformed

abundances (see Methods in the main text) from Zeller *et al.*'s data. We fit the Gamma-normal mixture model (step 1 of mbImpute) to each sampled taxon j, and we denote the estimated coefficient vector in the normal mean as $\hat{\gamma}_j \in \mathbb{R}^q$, $j = 1, \ldots, m$.

3. We simulated the log-transformed abundance of taxon j in sample i, $i = 1, \ldots, n$, from the following model:

$$Y_{ij}^{\text{comp}} = X_{i\cdot}^{\mathsf{T}} \hat{\gamma}_j + \epsilon_{ij} \,,$$

where $\epsilon_{ij} \sim \mathcal{N}(0, 1)$ independently. We denote the resulting matrix $\mathbf{Y}^{\text{comp}} = (Y_{ij}^{\text{comp}})$ as the complete data that contain log-transformed abundances without missing values.

Scheme 2: Sample

1. We randomly divided $n = 50$ samples into 5 groups. For each sample i, $i = 1, \ldots, n$, we denote its group index by $k(i) \in \{1, \ldots, 5\}$.
2. We generated a sample-to-sample distance matrix $\mathbf{D}^{\text{samp}} = (D_{ii'}^{\text{samp}})_{n \times n}$:

$$D_{ii'}^{\text{samp}} = \begin{cases} 2(|k(i) - k(i')| + 1) & \text{if } i \neq i' \\ 0 & \text{otherwise} \end{cases}$$

so that two samples from closer groups (in terms of group index) had a smaller distance.

3. We converted the distance matrix $\mathbf{D}^{\text{samp}}$ into a sample-to-sample covariance matrix $\mathbf{\Sigma}^{\text{samp}}$ such that two samples having a smaller distance would have a larger covariance

$$\Sigma_{ii'}^{\text{samp}} = \begin{cases} \exp\left\{(-D_{ii'}^{\text{samp}}/6)^{1.3}\right\} & \text{if } i \neq i' \\ \exp\left\{(-D_{ii'}^{\text{samp}}/6)^{1.3}\right\} + 1 & \text{otherwise} \end{cases},$$

where the constants 6 and 1.3 were chosen to make $\mathbf{\Sigma}^{\text{samp}}$ positive definite.

4. We simulated the log-transformed abundances of taxon j in all the n samples, $j = 1, \ldots, m$, from the following model:

$$Y_{\cdot j}^{\text{comp}} \overset{\text{ind}}{\sim} \mathcal{N}(3 \cdot \mathbf{1}_n, \Sigma^{\text{samp}}) \,,$$

which we collected as columns to form $\mathbf{Y}^{\text{comp}} = (Y_{ij}^{\text{comp}})$, i.e., the complete data containing log-transformed abundances without missing values.

Scheme 3: Taxon

1. We randomly divided $m = 200$ taxa into 10 groups. For each taxon j, $j = 1, \ldots, m$, we denote its group index by $r(j) \in \{1, \ldots, 10\}$.
2. We generated a taxon-to-taxon distance matrix $\mathbf{D}^{\text{taxon}} = (D_{jj'}^{\text{taxon}})_{m \times m}$:

$$D_{jj'}^{\text{taxon}} = \begin{cases} 2(|r(j) - r(j')| + 1) & \text{if } j \neq j' \\ 0 & \text{otherwise} \end{cases}$$

   so that two taxa from closer groups (in terms of group index) had a smaller distance.

3. We converted the distance matrix $\mathbf{D}^{\text{taxon}}$ into a taxa-to-taxa covariance matrix $\mathbf{\Sigma}^{\text{taxon}}$ such that two samples having a smaller distance would have a larger covariance

$$\Sigma_{jj'}^{\text{taxon}} = \begin{cases} \exp\left\{(-D_{jj'}^{\text{taxon}}/12)^{1.3}\right\} & \text{if } j \neq j' \\ \exp\left\{(-D_{jj'}^{\text{taxon}}/12)^{1.3}\right\} + 1 & \text{otherwise} \end{cases},$$

   where the constants 12 and 1.3 were chosen to make $\mathbf{\Sigma}^{\text{taxon}}$ positive definite.

4. We simulated the log-transformed abundances of sample i in all the m taxa, $i = 1, \ldots, n$, from the following model:

$$Y_{i\cdot}^{\text{comp}} \overset{\text{ind}}{\sim} \mathcal{N}(3 \cdot \mathbf{1}_m, \mathbf{\Sigma}^{\text{taxon}}),$$

   which we collected as rows to form $\mathbf{Y}^{\text{comp}} = (Y_{ij}^{\text{comp}})$, i.e., the complete data containing log-transformed abundances without missing values.

Scheme 4: Taxon + Sample + Covariate

We combined the data generation procedures from the above three schemes. Following the same notations as above, we simulated the log-transformed abundance of taxon j in sample

i, $i = 1, \ldots, n$, $j = 1, \ldots, m$, as follows:

1. We generated baseline values, following scheme 1:

$$Y_{ij}^{\text{comp1}} = X_{i\cdot}^{\mathsf{T}} \hat{\gamma}_j + \epsilon_{ij}$$

   to form $\mathbf{Y}^{\text{comp1}} = (Y_{ij}^{\text{comp1}})$.

2. We introduced a sample correlation structure as in scheme 2:

$$Y_{\cdot j}^{\text{comp2}} \overset{\text{ind}}{\sim} \mathcal{N}(Y_{\cdot j}^{\text{comp1}}, \mathbf{\Sigma}^{\text{samp}}) \,,$$

   which we collected as columns to form $\mathbf{Y}^{\text{comp2}} = (Y_{ij}^{\text{comp2}})$

3. We introduced a taxon correlation structure as in scheme 3:

$$Y_{i\cdot}^{\text{comp}} \overset{\text{ind}}{\sim} \mathcal{N}(Y_{i\cdot}^{\text{comp2}}, \mathbf{\Sigma}^{\text{taxon}}) \,,$$

   which we collected as rows to form $\mathbf{Y}^{\text{comp}} = (Y_{ij}^{\text{comp}})$, i.e., the complete data containing log-transformed abundances without missing values.

Next, we introduced zero inflation to $\mathbf{Y}^{\text{comp}}$ and generated $\mathbf{Y}^{\text{zi}}$ by mimicking real data as follows. We applied the step 1 of mbImpute to Zeller *et al.*'s data, which contain 486 taxa [33]; that is, we identified likely false zeros. Then we calculated z_k^{real} (taxon k's proportion of likely false zeros across samples) and μ_k^{real} (taxon k's average abundance after we excluded likely false zeros) for each taxon k in Zeller *et al.*'s data, $k = 1, \ldots, 486$. Next we introduced zeros into $\mathbf{Y}^{\text{comp}}$ in the following non-parametric way for each taxon j, $j = 1, \ldots, m$:

1. We calculated taxon j's average abundance in $\mathbf{Y}^{\text{comp}}$ as the mean of column j, denoted by μ_j^{comp};
2. We randomly sampled a value from $\{z_k^{\text{real}} : \mu_k^{\text{real}} \in (\mu_j^{\text{comp}} - 0.5, \mu_j^{\text{comp}} + 0.5)\}$ and denoted it as z_j^{zi}, i.e., the proportion of false zeros to be introduced into taxon j's abundances in $\mathbf{Y}^{\text{comp}}$;

3. We randomly drew false zero indicators $I_{ij} \sim \text{Bernoulli}(z_j^{\text{zi}})$ independently for sample $i = 1, \ldots, n$;
4. We set $Y_{ij}^{\text{zi}} = \max\left(Y_{ij}^{\text{comp}} \cdot I_{ij}, \log_{10}(1.01)\right)$.

The reason why we set the minimum value of $\mathbf{Y}^{\text{zi}}$ to $\log_{10}(1.01)$ is that mbImpute step 1 sets the minimum log-transformed abundance to $\log_{10}(1.01)$ to facilitate the fitting of the Gamma-normal mixture model.

After applying mbImpute to $\mathbf{Y}^{\text{zi}}$ (mbImpute also used $\mathbf{X}$), we obtained $\mathbf{Y}^{\text{imp}}$ and evaluated the imputation performance by calculating the mean squared error (MSE) between $\mathbf{Y}^{\text{imp}}$ and $\mathbf{Y}^{\text{comp}}$:

$$\text{MSE}^{\text{mbImpute}} = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}(Y_{ij}^{\text{comp}} - Y_{ij}^{\text{imp}})^2$$

and the MSE between the zero-inflated matrix $\mathbf{Y}^{\text{zi}}$ and $\mathbf{Y}^{\text{comp}}$:

$$\text{MSE}^{\text{no imputation}} = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}(Y_{ij}^{\text{comp}} - Y_{ij}^{\text{zi}})^2 .$$

The results are summarized in Supplementary Fig. S3.

Simulation 4 for DA analysis on 16S rRNA data

We used the `R` package `sparseDOSSA` by [154] to simulate the 16S rRNA sequencing data with known DA taxa. Specifically, we used the following command to obtain simulated data:

```
metadata <- matrix(rbinom(n = 100, size = 1, prob = 0.5), nrow = 1, ncol = 100)
simulated_data <- sparseDOSSA(number_features = 150, number_samples = 100,
percent_spiked = 0.3, UserMetadata = metadata)
```

In order to evaluate the performance of four DA methods (the Wilcoxon rank-sum test, ANCOM, metagenomeSeq, and DESeq2-phyloseq) before and after mbImpute is applied, we use three evaluation metrics: precision, recall, and F1 score. The results are summarized in Fig. 2e in the main text.

Simulation 5 for robustness to sequencing depth and outlier samples

To evaluate the robustness of mbImpute to sequencing depth and the existence of outlier samples, we performed simulation based on the 16S rRNA sequencing data of 54 healthy human stool samples in the R package `HMP16SData` (version 1.6.0). Below we describe the data generation process step by step.

Complete data generation

1. We randomly sampled 300 taxa in the 16S rRNA sequencing data that each has no more than 70% of zeros across samples, obtaining a 54×300 OTU count matrix.
2. We followed the data pre-processing steps of mbImpute (see Methods in the main text) to convert the OTU count matrix to a log-transformed normalized abundance matrix, denoted by $\mathbf{Y} = (Y_{ij}) \in \mathbb{R}^{54\times 300}$. We also collected the sample covariate matrix $\mathbf{X} \in \mathbb{R}^{54\times 2}$ and the phylogenetic distance matrix $\mathbf{D} \in \mathbb{Z}_{\geqslant 0}^{300\times 300}$.
3. We create a complete abundance matrix denoted by $\mathbf{Y}^{\text{comp}}$ with the same dimensions as $\mathbf{Y}$. Specifically, for each taxon j, we identified a set Ω_j of samples in which taxon j's original counts are not 0, and we denote the rest of the samples as Ω_j^c. From the j-th column of $\mathbf{Y}$, we copied the abundances in samples in Ω_j to the corresponding entries in the j-th column of $\mathbf{Y}^{\text{comp}}$, and we calculated the mean $\hat{\mu}_j$ and the standard deviation $\hat{\sigma}_j$ of the abundances in the samples in Ω_j. Then we sampled abundances for the samples in Ω_j^c independently from $\mathcal{N}(\hat{\mu}_j, \hat{\sigma}_j)$ to fill in the corresponding entries in the j-th column $\mathbf{Y}^{\text{comp}}$.
4. We transform $\mathbf{Y}^{\text{comp}}$ to a complete count matrix $\mathbf{M}^{\text{comp}}$ by setting the (i, j)-th entry as $M_{ij}^{\text{comp}} = \lfloor (10^{Y_{ij}^{\text{comp}}} - 1.01) \rfloor$.

Complete data adjusted by sequencing depth

1. Samples (rows) in $\mathbf{M}^{\text{comp}}$ have total counts around 2,000. We varied the sequencing depth as s = 1,000, 2,000, 5,000, or 10,000.

2. For each sequencing depth, we sampled a 300-dimensional count vector for sample i, $i = 1, \ldots, 54$, from a multinomial distribution, whose total is the sequencing depth and whose probability vector is $M_{i\cdot}^{\text{comp}}/(\sum_{j=1}^{300} M_{ij}^{\text{comp}})$, where $M_{i\cdot}^{\text{comp}}$ denotes the i-th row of $\mathbf{M}^{\text{comp}}$. We stacked the sampled vectors by row into a count matrix $\mathbf{M}_s^{\text{comp}}$, whose log-transformed abundance matrix is denoted by $\mathbf{Y}_s^{\text{comp}}$.

3. We transform $\mathbf{Y}_s^{\text{comp}}$ to a complete count matrix $\mathbf{M}_s^{\text{comp}}$ by setting the (i, j)-th entry as $(M_s)_{ij}^{\text{comp}} = \lfloor (10^{(Y_s)_{ij}^{\text{comp}}} - 1.01) \rfloor$.

Zero-inflated data generation

We introduced zero inflation to $\mathbf{Y}_s^{\text{comp}}$, following the same non-parametric procedure as in simulation 2. We denote the resulting zero-inflated matrix by $\mathbf{Y}_s^{\text{zi}}$ and the corresponding count matrix by $\mathbf{M}_s^{\text{zi}}$, $s =$ 1,000, 2,000, 5,000, and 10,000.

Zero-inflated data with outlier samples

1. Given $s =$ 2,000, we use $\mathbf{Y}_s^{\text{zi}}$ to define the lowly abundant taxa as those that have mean abundances below the median and at least 10 non-zero abundances.

2. To generate one outlier sample, for the 62 lowly abundant taxa based on our definition, we set their abundances in the outlier sample to be 62 values randomly sampled from the top 100 maximum abundances (for each taxon, we calculated the maximum abundance). For the other taxa, we set their abundances in the outlier sample to zero.

3. To generate the second outlier sample, we repeated step 2.

4. We denote the resulting abundance matrices and count matrices with one or two outlier samples by $\mathbf{Y}_{1\text{o}}^{\text{zi}}$ and $\mathbf{M}_{1\text{o}}^{\text{zi}}$ or $\mathbf{Y}_{2\text{o}}^{\text{zi}}$ and $\mathbf{M}_{2\text{o}}^{\text{zi}}$, respectively.

Evaluation criteria for imputation accuracy

After applying mbImpute to the zero-inflated abundance matrices, each denoted by $\mathbf{Y}^{\text{zi}}$, we obtained the corresponding imputed abundances matrices, each denoted by $\mathbf{Y}^{\text{imp}}$, and we evaluated the imputation performance by calculating the mean squared error (MSE) between $\mathbf{Y}^{\text{imp}}$ and the corresponding complete abundance matrix $\mathbf{Y}^{\text{comp}}$:

$$\text{MSE}^{\text{mbImpute}} = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}(Y_{ij}^{\text{comp}} - Y_{ij}^{\text{imp}})^2$$

and the MSE between the zero-inflated matrix $\mathbf{Y}^{\text{zi}}$ and $\mathbf{Y}^{\text{comp}}$:

$$\text{MSE}^{\text{no imputation}} = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}(Y_{ij}^{\text{comp}} - Y_{ij}^{\text{zi}})^2 \,.$$

Note that the MSE is only calculated for non-outlier samples when outlier samples are introduced. The results are summarized in Supplementary Figs. S5 and S6.

Statistical definitions of DA taxa

Here we list three possible statistical definitions of DA taxa. For a given taxon j, we denote M_j^1 and M_j^2 as its (random) counts in two samples from subject groups 1 and 2. The first and most straightforward definition of DA is whether the null hypothesis

$$H_0 : \mathbb{E}[M_j^1] = \mathbb{E}[M_j^2]$$

is rejected. The Wilcoxon rank-sum test is based on this definition. However, this definition has an obvious drawback: it ignores the existence of excess false zeros, i.e., many observed zero values of M_j^1 and M_j^2 are not reliable. This drawback motivates the second definition, which introduces latent variables Z_j^1 and Z_j^2 indicating whether taxon j is detected in the two samples. The second definition relies on zero-inflated models: $M_j^r|(Z_j^r = 0) = 0$, $r = 1, 2$, and it defines taxon j as DA if the null hypothesis

$$H_0 : \mathbb{E}[M_j^1 | (Z_j^1 = 1)] = \mathbb{E}[M_j^2 | (Z_j^2 = 1)]$$

is rejected. The metagenomSeq is based on this definition. A drawback of this definition is that many observations of M_j^1 and M_j^2 would not be used for testing this hypothesis, if their corresponding Z_j^1 and Z_j^2 are inferred as zeros, resulting in a power loss. To relieve this issue, imputation can be used to rescue the likely false zeros and infer their actual values, and mbImpute achieves this by borrowing information from similar samples, similar taxon, and sample covariates, leading to the third definition of DA taxa. Assuming that the imputation is successful, we denote M_j^{imp1} and M_j^{imp2} as taxon j's imputed counts, and Y_j^{imp1} and Y_j^{imp2} as the imputed abundances on the logarithmic scale, in the two samples. Then the third definition calls taxon j DA if the null hypothesis

$$H_0 : \mathbb{E}[M_j^{\text{imp1}}] = \mathbb{E}[M_j^{\text{imp2}}] \text{ or } H_0 : \mathbb{E}[Y_j^{\text{imp1}}] = \mathbb{E}[Y_j^{\text{imp2}}]$$

is rejected. This third definition is advantageous in that (1) compared with the first definition, it is less affected by the existence of false zeros, whose different proportions in the two subject groups may lead to false positive DA taxa (i.e., the taxa whose non-zero counts do not exhibit a clear difference between the two groups), and (2) compared with the second definition, it uses all the observations of taxon j for testing, leading to an increase in statistical power.

Fisher's exact test for detecting the enrichment of T2D- and CRC-related terms in DA taxa

We use functional terms in the GMrepo [115] database to understand the DA taxa identified by DESeq2-phyloseq or mbImpute-empowered DESeq2-phyloseq. Among all functional terms, we identified two T2D-related terms:

- “The time period before the development of symptomatic diabetes. For example, certain risk factors can be observed in subjects who subsequently develop INSULIN RESISTANCE as in type 2 diabetes (DIABETES MELLITUS, TYPE 2).”
- “A cluster of symptoms that are risk factors for CARDIOVASCULAR DISEASES and TYPE 2 DIABETES MELLITUS. The major components of metabolic syndrome include ABDOMINAL OBESITY; atherogenic DYSLIPIDEMIA; HYPERTENSION; HYPERGLYCEMIA; INSULIN RESISTANCE; a proinflammatory state; and a prothrombotic (THROMBOSIS) state.”

and one CRC-related term:

- “Tumors or cancer of the COLON or the RECTUM or both. Risk factors for colorectal cancer include chronic ULCERATIVE COLITIS; FAMILIAL POLYPOSIS COLI; exposure to ASBESTOS; and irradiation of the CERVIX UTERI.”

Given a T2D- or CRC-related functional term, we performed the Fisher’s exact test to check its enrichment in the DA taxa identified by DESeq2-phyloseq or mbImpute-empowered DESeq2-phyloseq from the corresponding T2D or CRC datasets. Specifically, we constructed a two-by-two contingency table (see below), whose rows indicate whether taxa are annotated by the functional term or not, and whose columns indicate whether taxa are identified as DA or not; each entry in the contingency table is the number of taxa satisfying the row and column conditions.

	identified as DA	not identified as DA
annotated by the term	a	b
not annotated by the term	c	d

Based on the above contingency table, the p-value of the Fisher’s exact test is calculated as

$$p = \frac{\binom{a+b}{a}\binom{c+d}{c}}{\binom{a+b+c+d}{a+c}} .$$

A smaller p-value shows stronger evidence against the null hypothesis that there is no dependence between whether taxa are related to the term and whether taxa are identified as DA. In other words, a smaller p-value indicates a stronger enrichment of the term in the DA taxa.

To test the enrichment of each T2D-related term, we combined the DA taxa identified from the two T2D datasets by DESeq2-phyloseq or mbImpute-empowered DESeq2-phyloseq. To test the enrichment of each CRC-related term, we combined the DA taxa identified from the four CRC datasets by DESeq2-phyloseq or mbImpute-empowered DESeq2-phyloseq. The results are summarized in Table 1 in the main text.

Supplementary Table

	Wilcoxon	ANCOM	MetagenomeSeq	DESeq2-phyloseq	Omnibust test
Qin et al.	25	20	9	11	20
Karlsson et al.	5	6	0	1	4
Feng et al.	42	18	0	30	28
Yu et al.	19	22	4	54	23
Vogtmann et al.	0	0	0	53	2
Zeller et al.	25	34	10	36	20

Table 2.3: The number of DA taxa identified by each DA method in each dataset at the FDR threshold 5%. There are two T2D datasets (*Qin et al.* and *Karlsson et al.*) and four CRC datasets (*Feng et al.*, *Vogtmann et al.*, *Yu et al.*, and *Zeller et al.*)

Supplementary figures

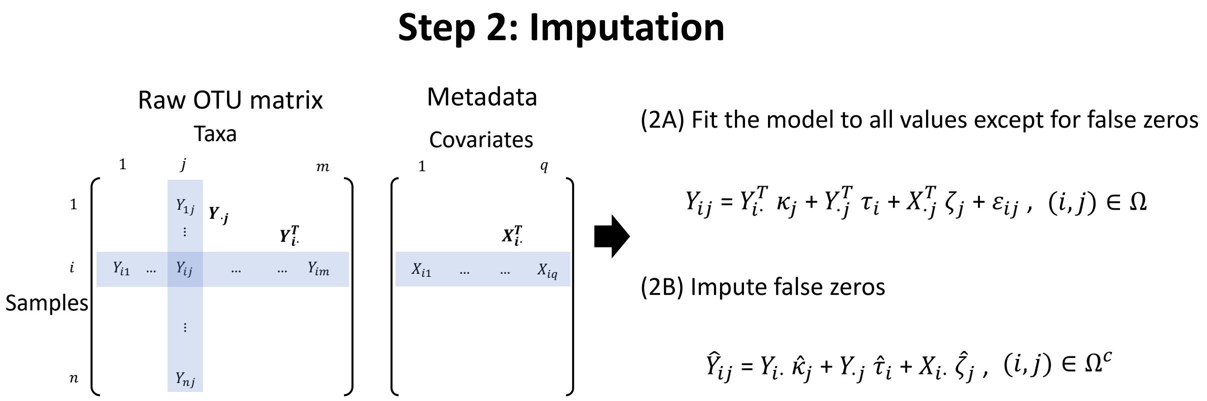
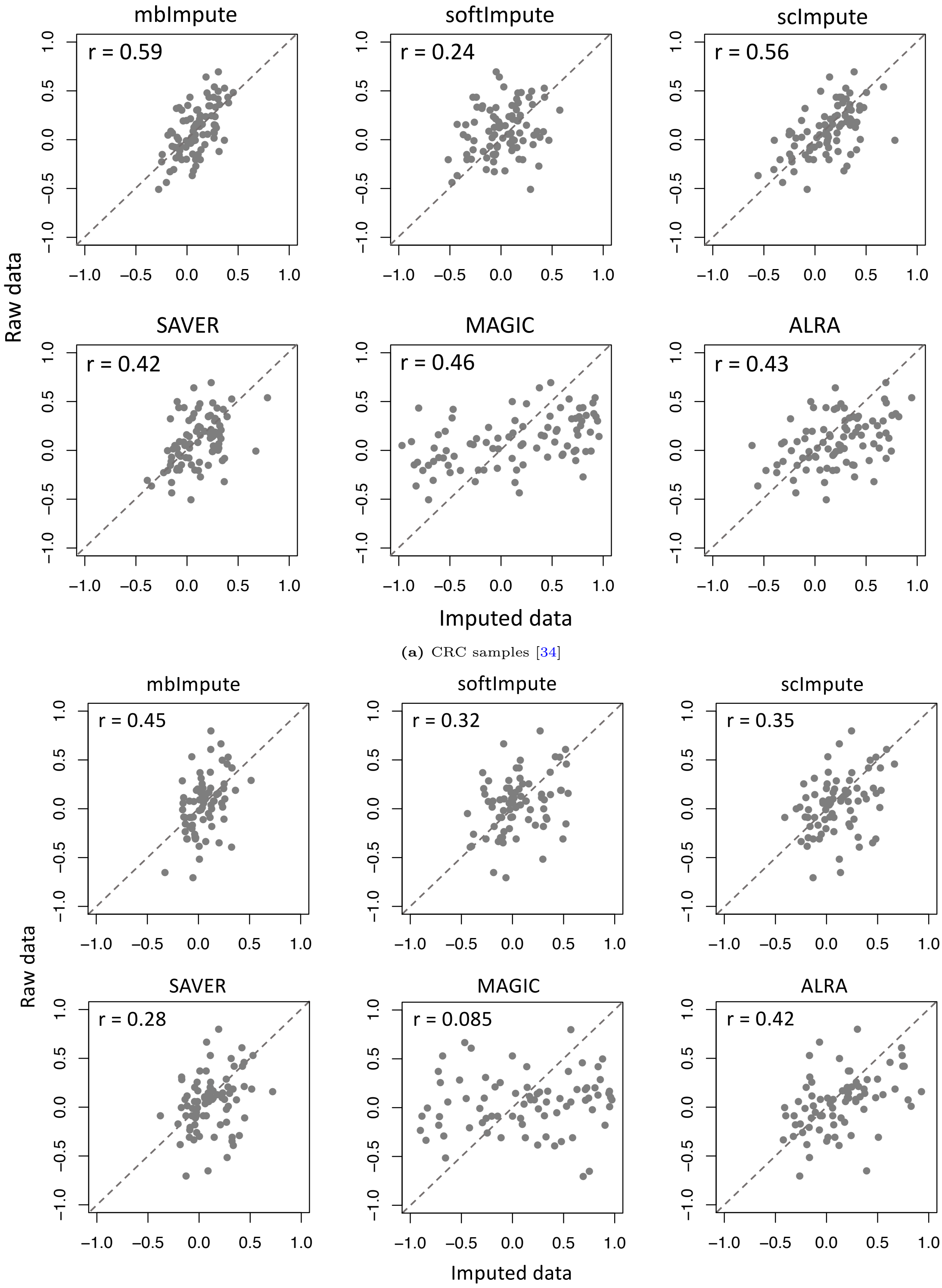

Step 2: Imputation
Raw OTU matrix
Taxa
Metadata
Covariates
Samples
1
j
m
1
q
1
i
n
Y_{1j}
$\boldsymbol{Y}_{\cdot j}$
$\boldsymbol{Y}_{i\cdot}^T$
Y_{i1} ... Y_{ij} ... ... Y_{im}
Y_{nj}
$\boldsymbol{X}_{i\cdot}^T$
X_{i1} ... ... X_{iq}
(2A) Fit the model to all values except for false zeros
$Y_{ij} = Y_{i\cdot}^T\,\kappa_j + Y_{\cdot j}^T\,\tau_i + X_{\cdot j}^T\,\zeta_j + \varepsilon_{ij}\ ,\ \ (i,j) \in \Omega$
(2B) Impute false zeros
$\hat{Y}_{ij} = Y_{i\cdot}\,\hat{\kappa}_j + Y_{\cdot j}\,\hat{\tau}_i + X_{i\cdot}\,\hat{\zeta}_j\ ,\ \ (i,j) \in \Omega^c$

Figure 2.6: A diagram illustrating the step 2 of mbImpute. After step 1 that identifies likely false zeros, mbImpute borrows information across taxa, samples, and sample covariates to impute these likely false zeros. For details, see Methods in the main text.



(a) CRC samples [34]

(b) Control samples [34]

Figure 2.7: Correlations between phylogenetically closely related taxa in the raw data vs. those in the imputed data. For each pair of taxa connected by a path shorter than 4 branches in the phylogenetic tree, we computed their correlation in the raw data using their mutual non-zero abundances (vertical axis) and their correlation in the imputed data using all the abundances (horizontal axis). The Pearson correlation between these two sets of correlations is reported for each imputed dataset, showing that mbImpute achieves the highest correlation in both (a) the CRC samples and (b) the control samples.

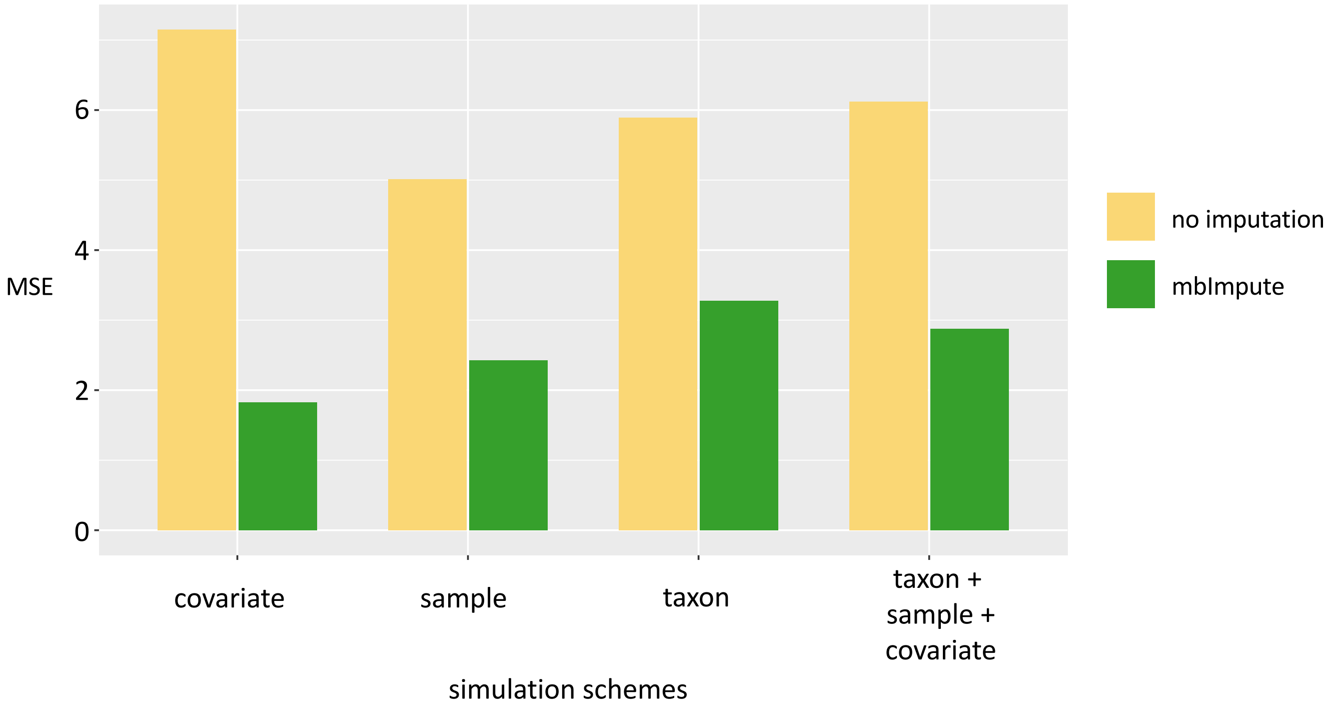


Figure 2.8: mbImpute reduces MSE under the four different simulation schemes in Simulation 3.

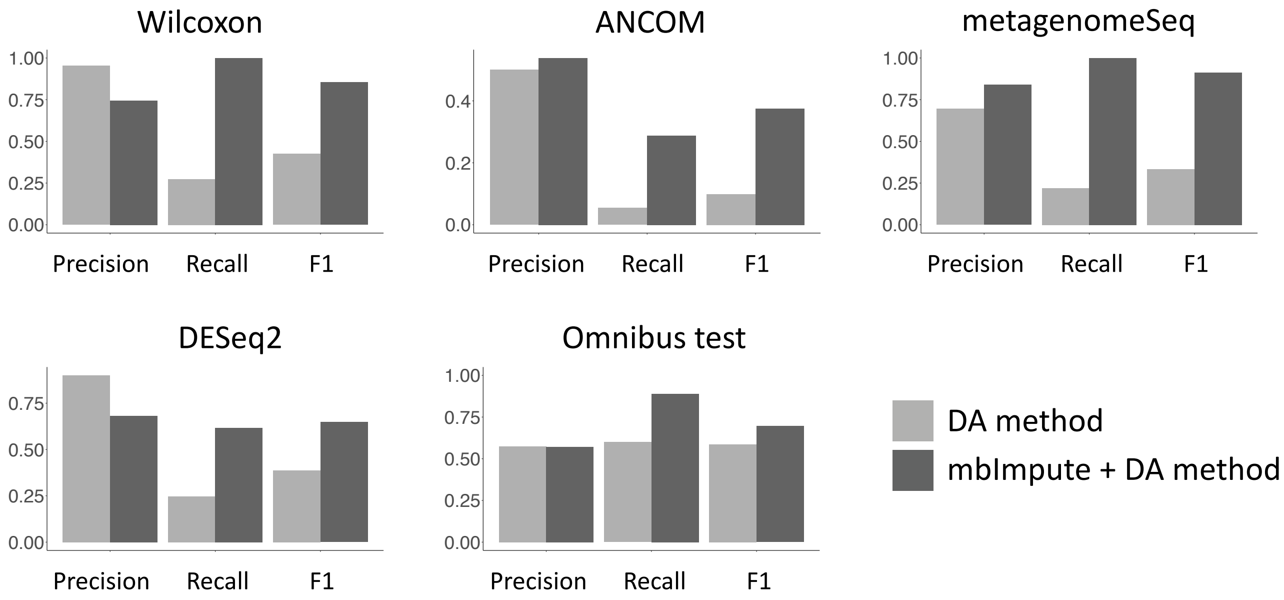


Figure 2.9: mbImpute enhances DA taxa identification in 16S simulation. Accuracy (Precision, recall, and F_1 scores) of four DA methods (Wilcoxon rank-sum test, ANCOM, metagenomeSeq, DESeq2-phyloseq, and Omnibus test) with an FDR threshold 0.1 on raw data (light color) and imputed data by mbImpute (dark color) in 16S data simulation.

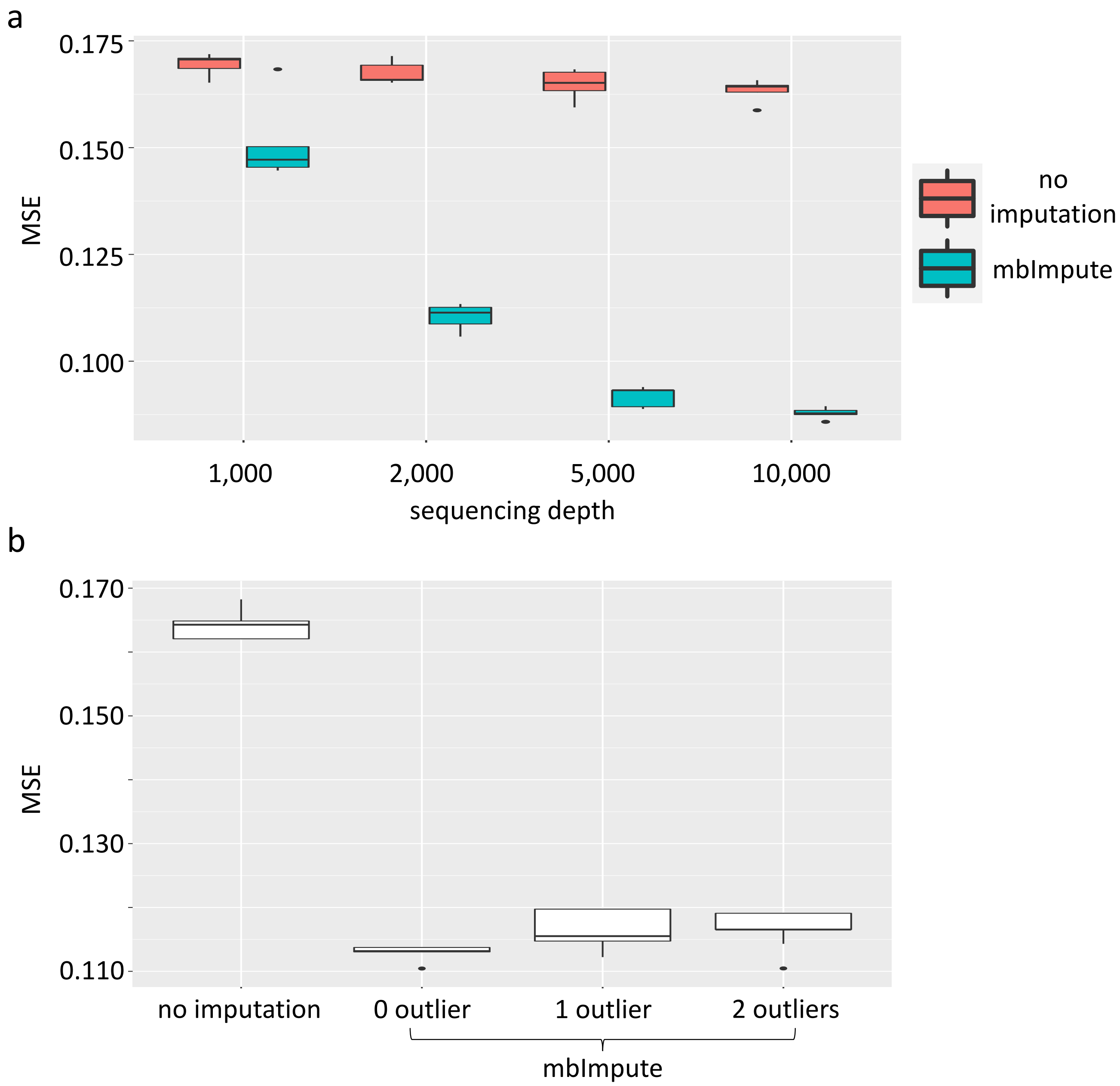


Figure 2.10: Robustness of mbImpute to sequencing depth and outlier samples (Simulation 5; independently repeated for five times). (a) Distributions of MSE before and after mbImpute is applied to simulated 16S rRNA sequencing data with four sequencing depths (number of reads per sample). **(b)** Distributions of MSE before and after mbImpute is applied to the data in (a) with the 2,000-read per-sample sequencing depth and 0, 1 or 2 outlier samples added.

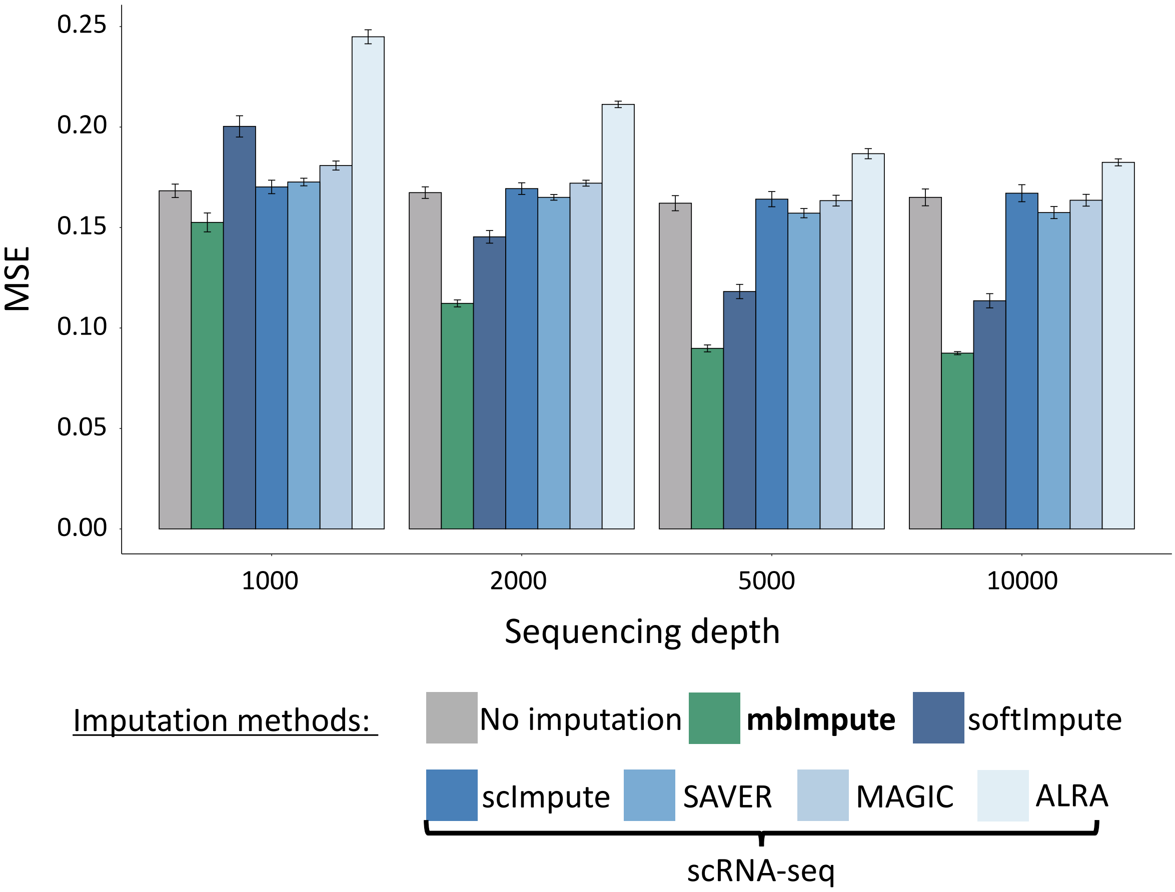


Figure 2.11: Effects of sequencing depth on the imputation accuracy of six imputation methods (Simulation 5; independently repeated for five times). Mean squared error (MSE) between the complete data and the zero-inflated data ("No imputation," the baseline) or the imputed data by each imputation method (mbImpute, softImpute, scImpute, SAVER, MAGIC, and ALRA) under each of four sequencing depths. The results of the baseline and mbImpute are the same as those in Supplementary Fig. S5a. Only mbImpute is consistently better than the baseline; its has the smallest MSE at every sequencing depth, and its MSE decreases as the sequencing depth increases.

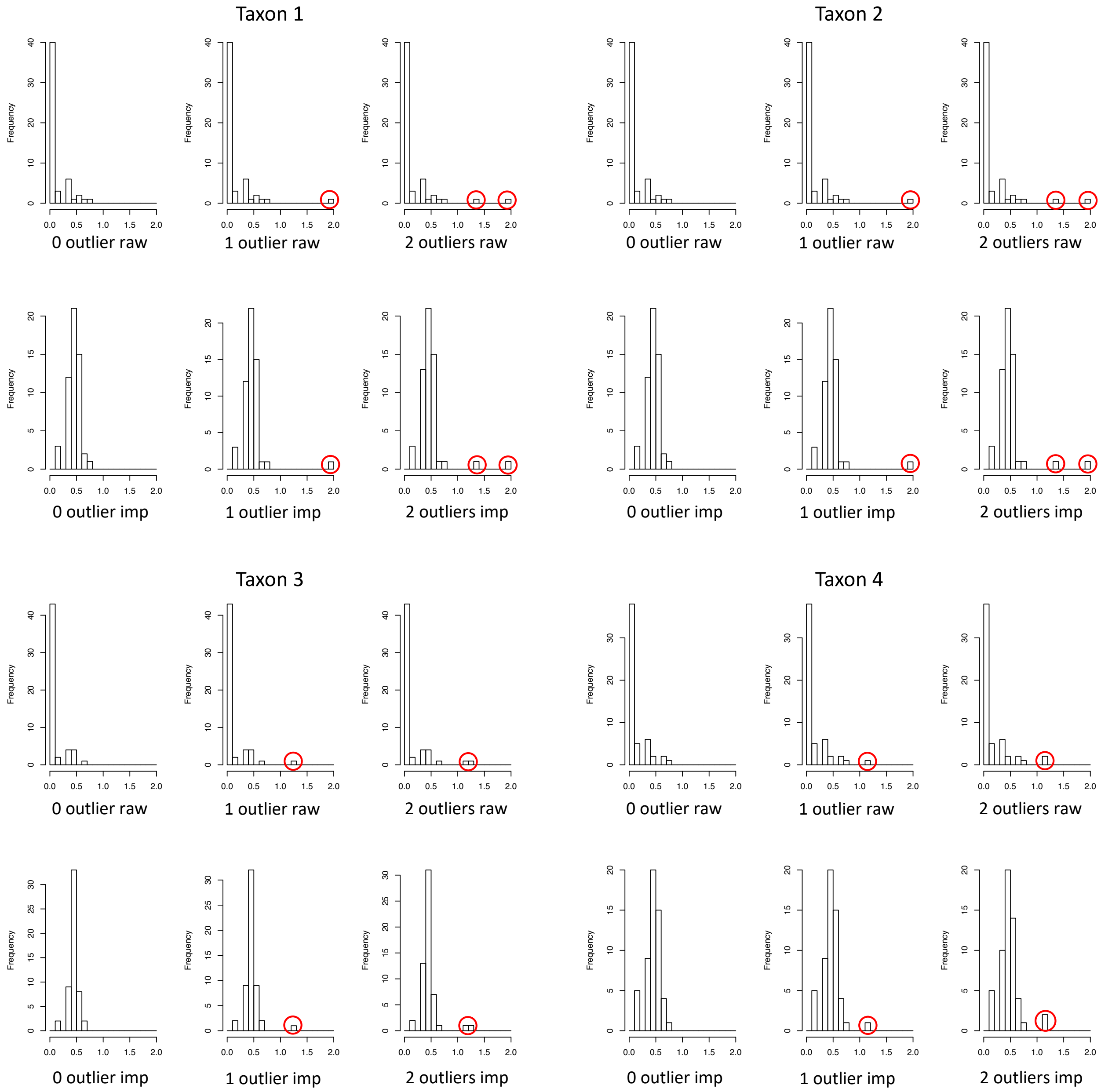


Figure 2.12: Abundance distributions of four taxa with one or two outlier samples before and after mbImpute is applied. For each taxon, the top row (”raw”) displays the abundance distributions with 0, 1, or 2 outlier samples before imputation, and the bottom row (”imp”) shows the distributions after mbImpute is applied. The abundance values in the outlier samples are marked by red circles.

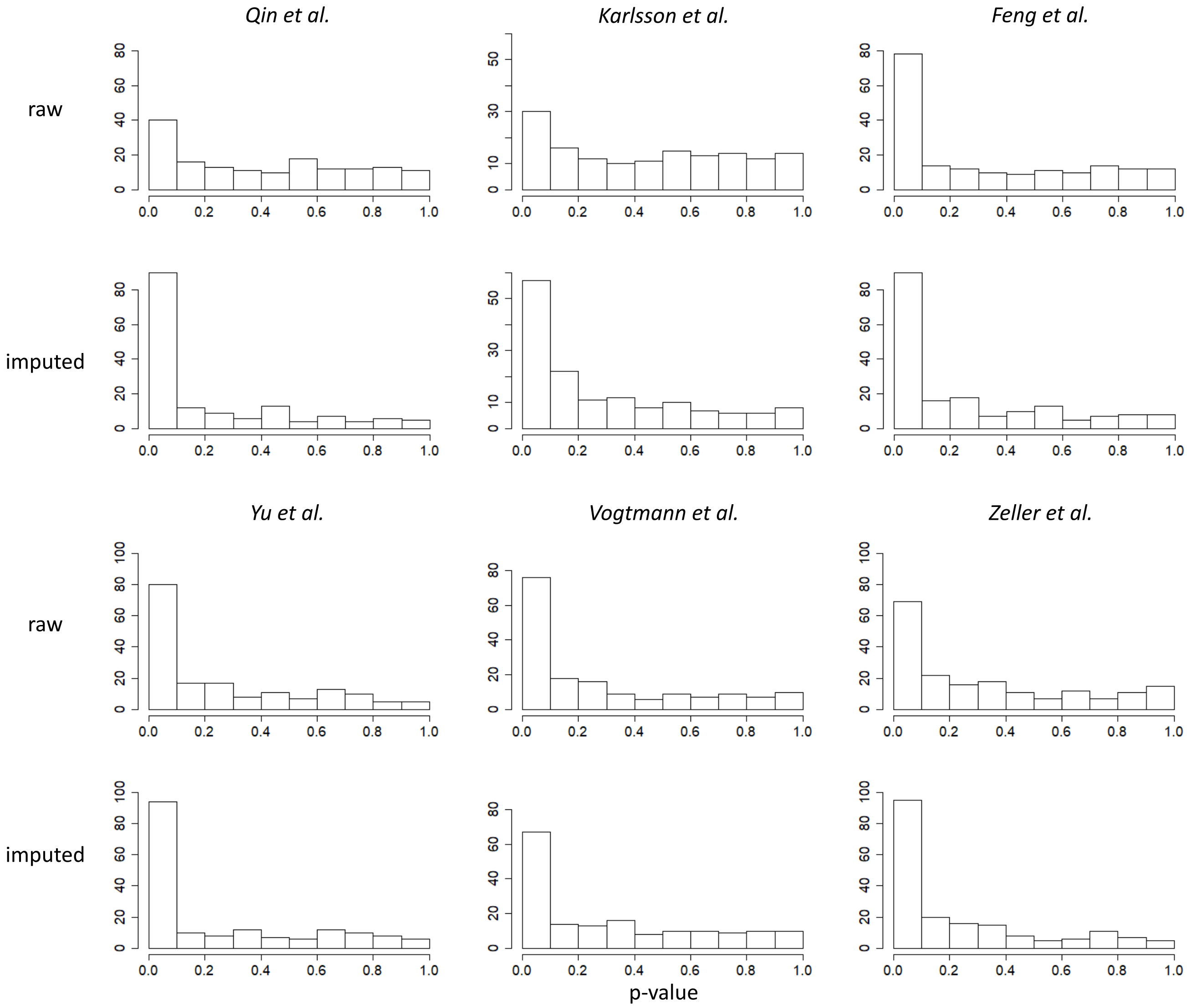


Figure 2.13: Distributions of DESeq2-phyloseq p-values before and after mbImpute is applied.

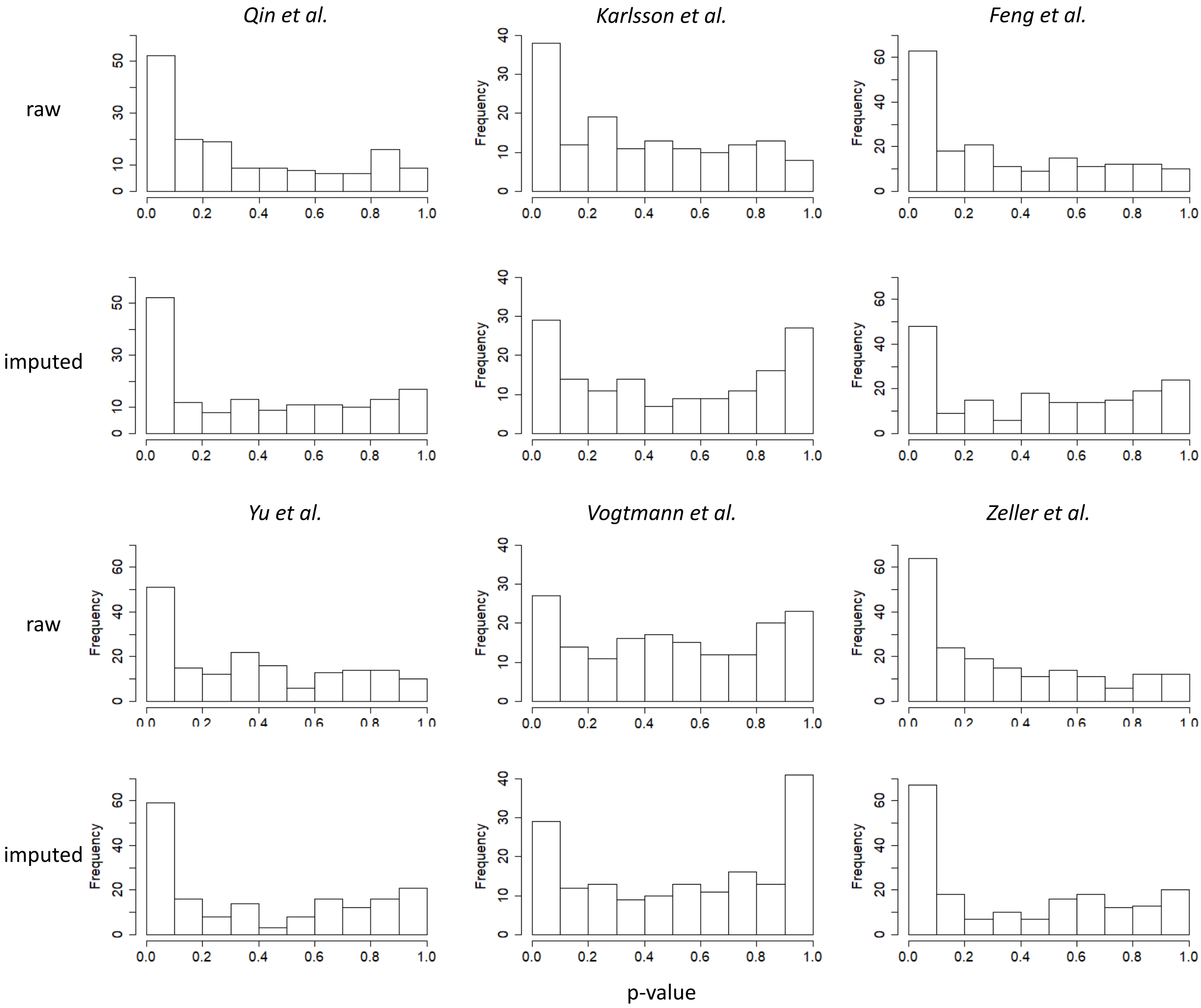


Figure 2.14: Distributions of Omnibus test p-values before and after mbImpute is applied.

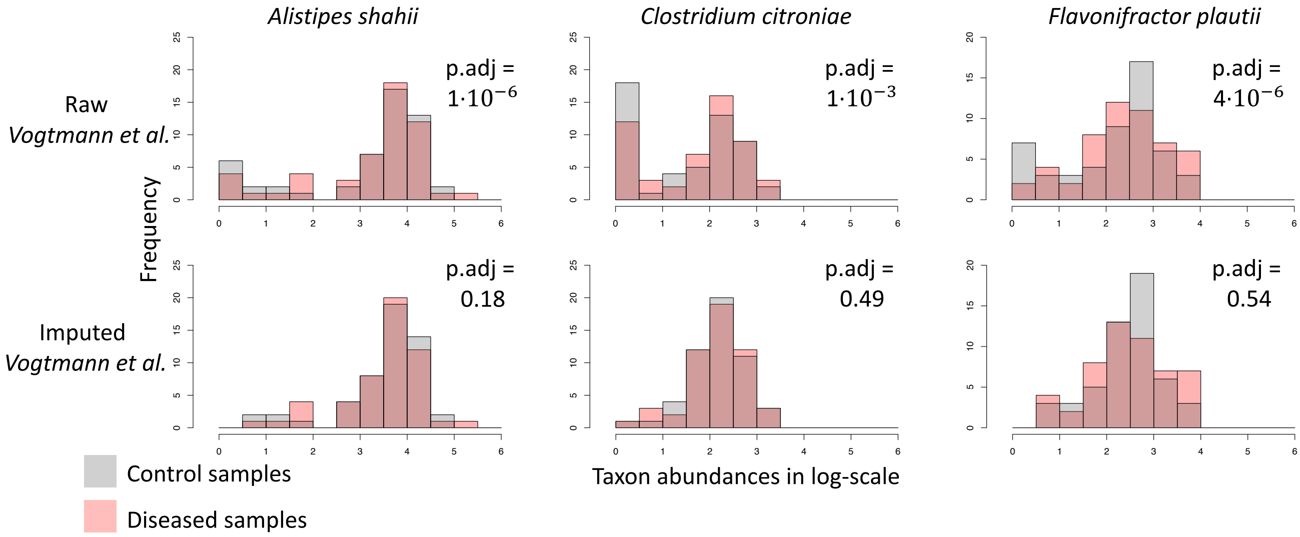


Figure 2.15: Abundance distributions of three taxa in control and CRC samples in *Vogtmann et al.* before and after mbImpute is applied. The three taxa, *Alistipes shahii*, *Clostridium citroniae*, and *Flavonifractor plautii*, are identified as DA by DESeq2-phyloseq before imputation but not as DA after imputation. Adjusted p-values are listed.

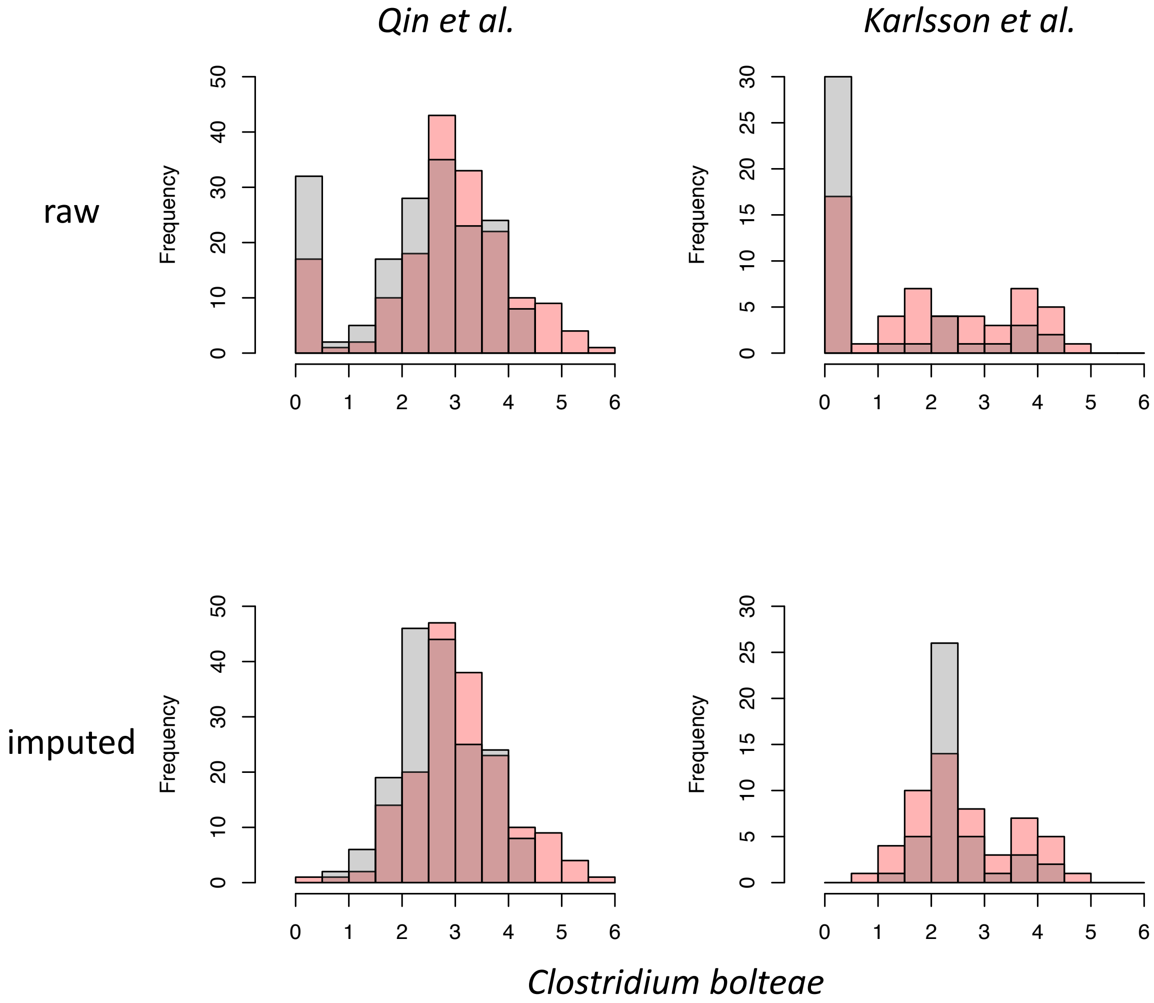


Figure 2.16: Distributions of *Clostridium bolteae* abundances before and after mbImpute is applied. Two T2D datasets, Qin *et al.* and Karlsson *et al.*, are included in this figure.

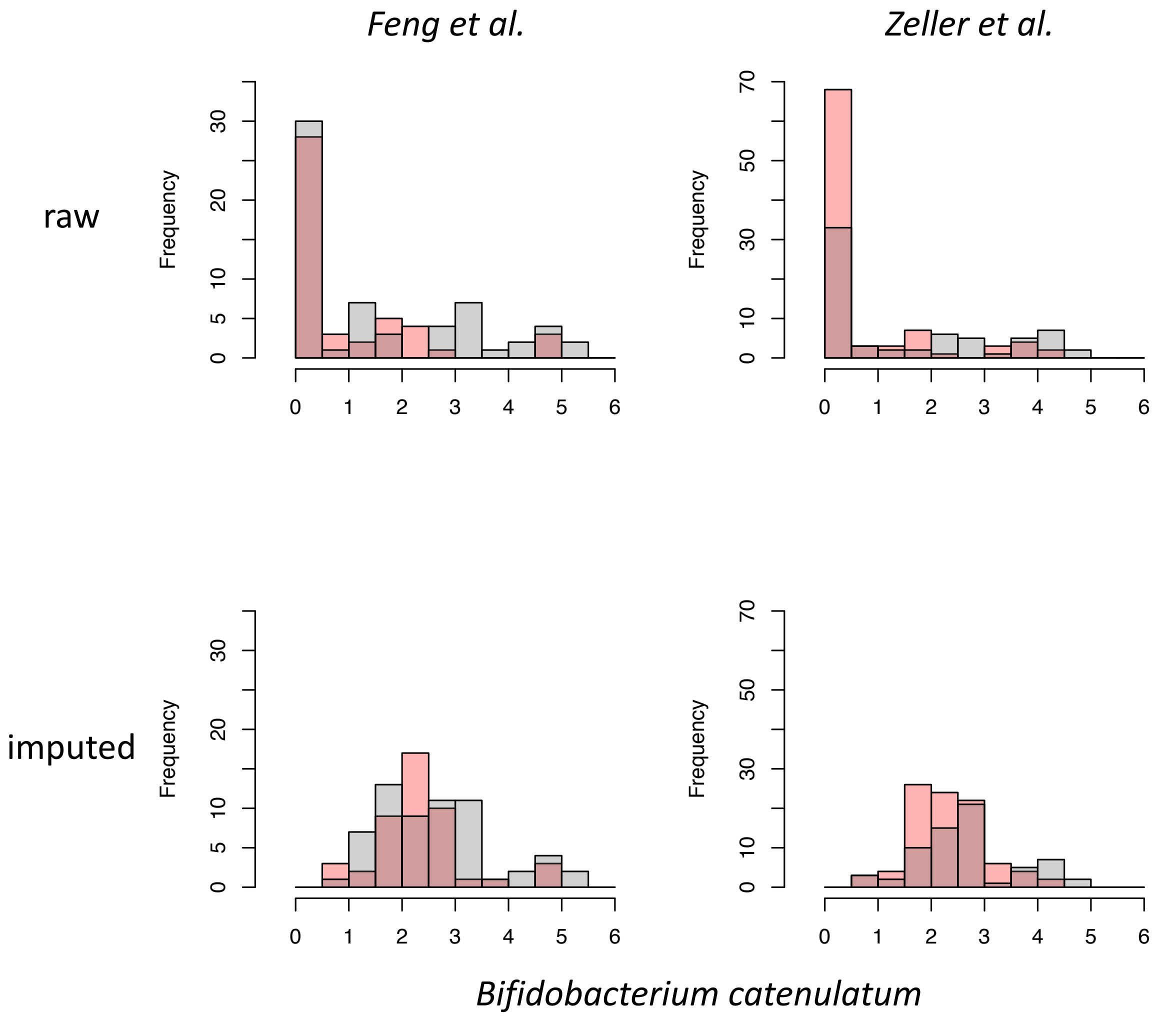


Figure 2.17: Distributions of *Bifidobacterium catenulatum* abundances before and after mbImpute is applied. Two CRC datasets, Feng *et al.* and Zeller *et al.*, are included in this figure.

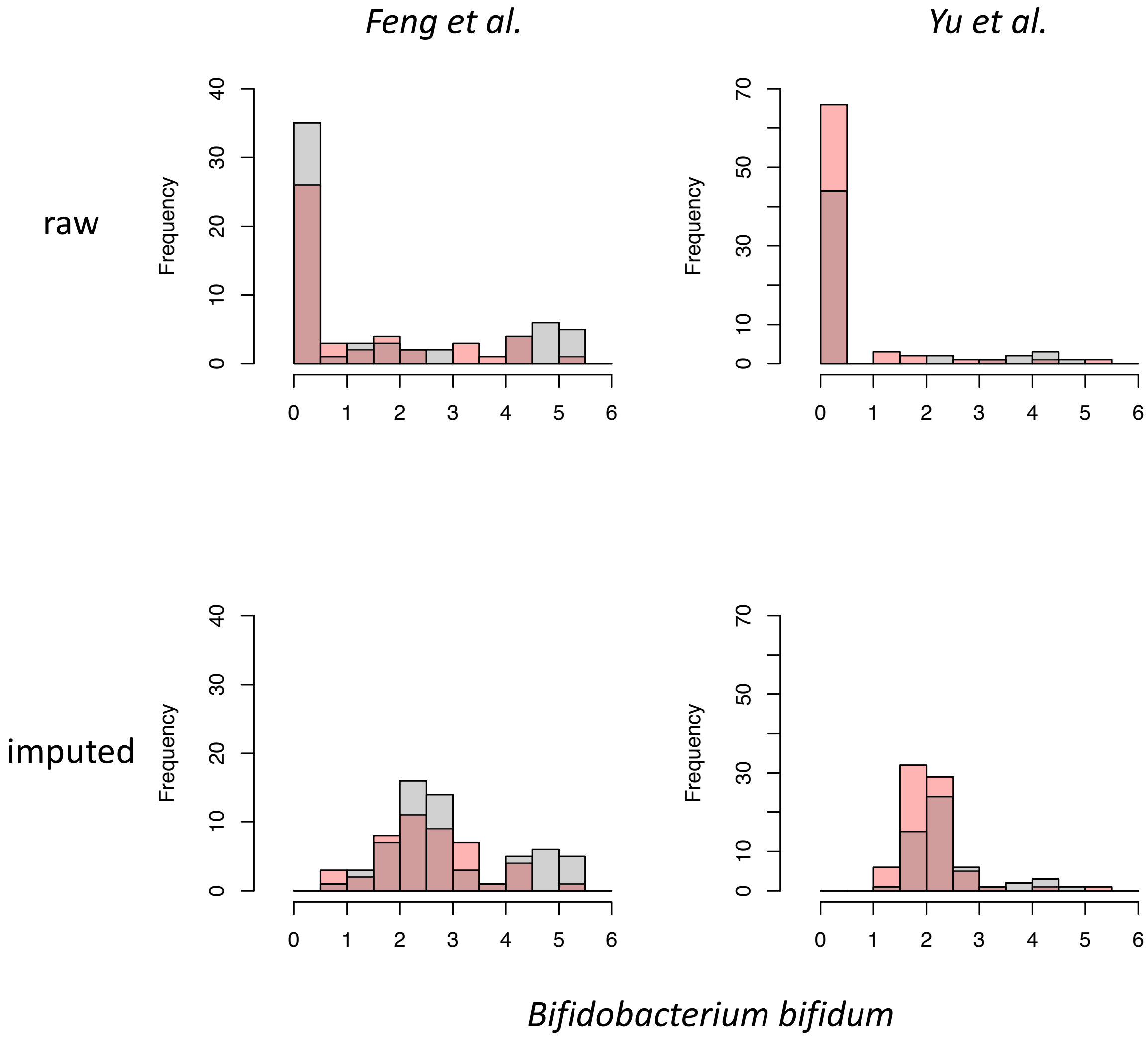


Figure 2.18: Distributions of *Bifidobacterium bifidum* abundances before and after mbImpute is applied. Two CRC datasets, Feng *et al.* and Yu *et al.*, are included in this figure.

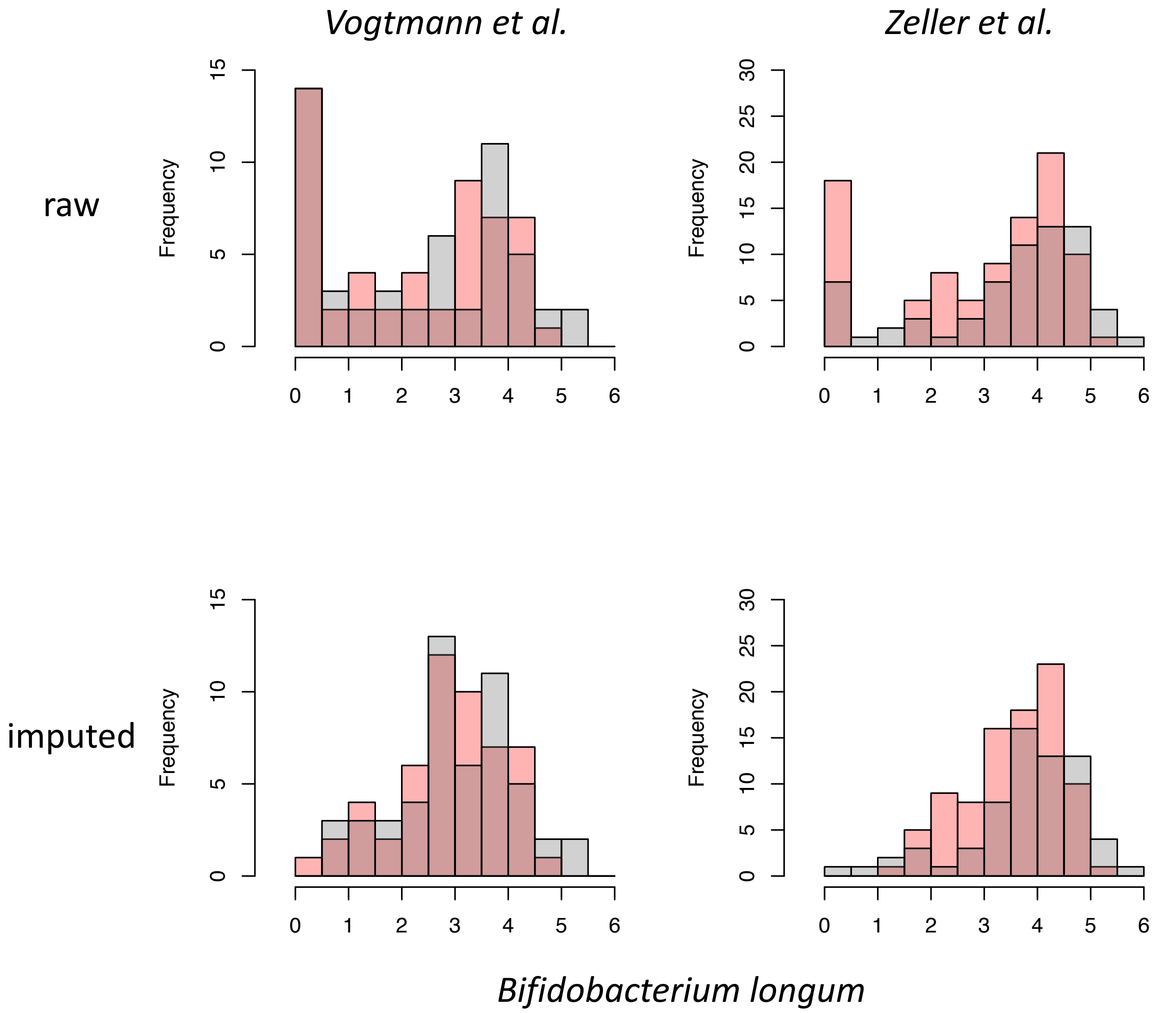


Figure 2.19: Distributions of *Bifidobacterium longum* abundances before and after mbImpute is applied. Two CRC datasets, Vogtmann *et al.* and Zeller *et al.*, are included in this figure.

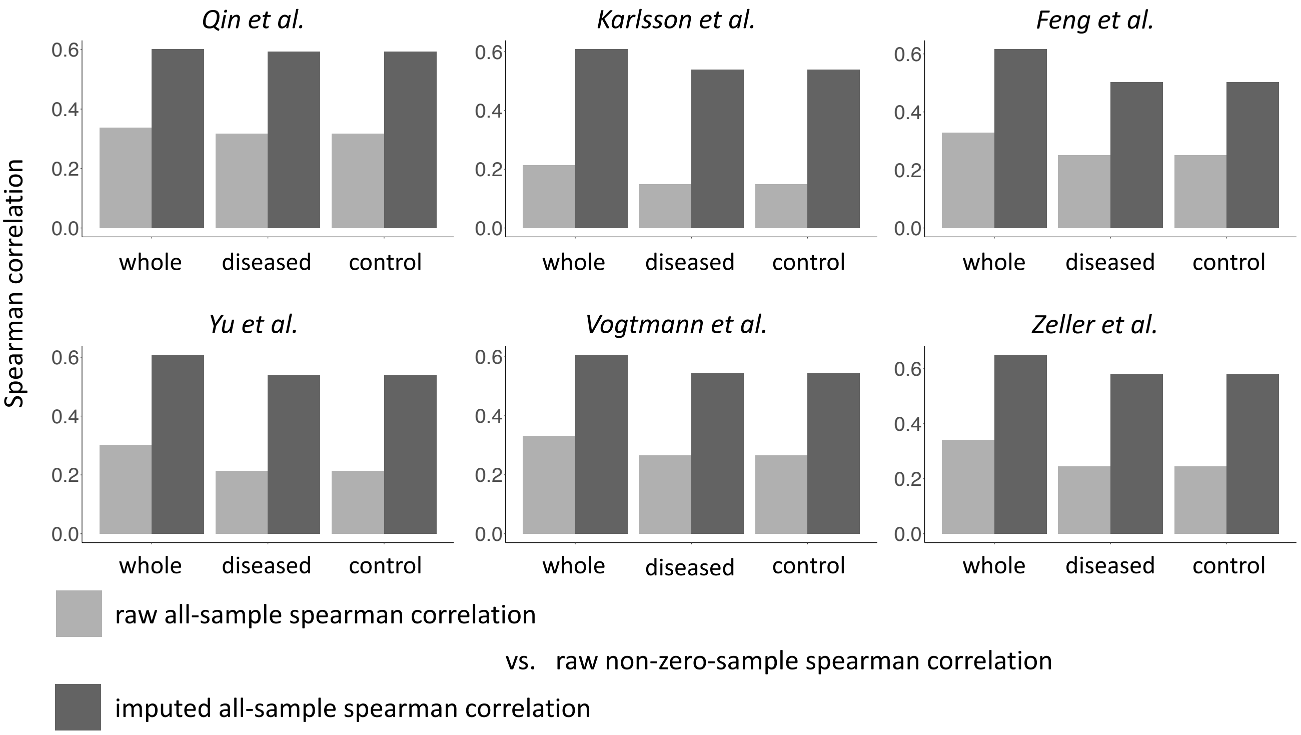


Figure 2.20: Each bar shows the Spearman correlation between taxon-taxon correlations in raw data (light gray) or imputed data (dark gray) using all samples and taxon-taxon correlations in raw data using non-zero samples only. The two correlations are calculated for two T2D datasets and four CRC datasets using diseased samples, control samples, and whole data.

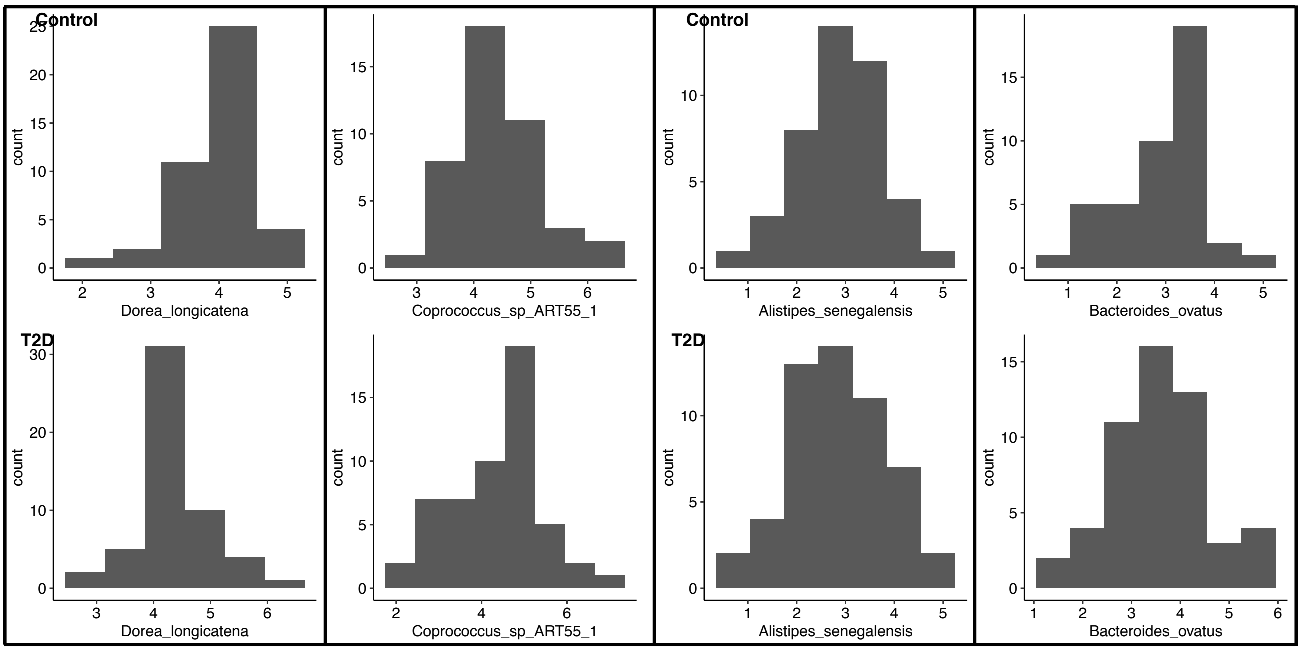


Figure 2.21: Distributions of four randomly chosen taxa from Karlsson *et al.* after mbImpute is applied. The imputed abundances are approximately normal.

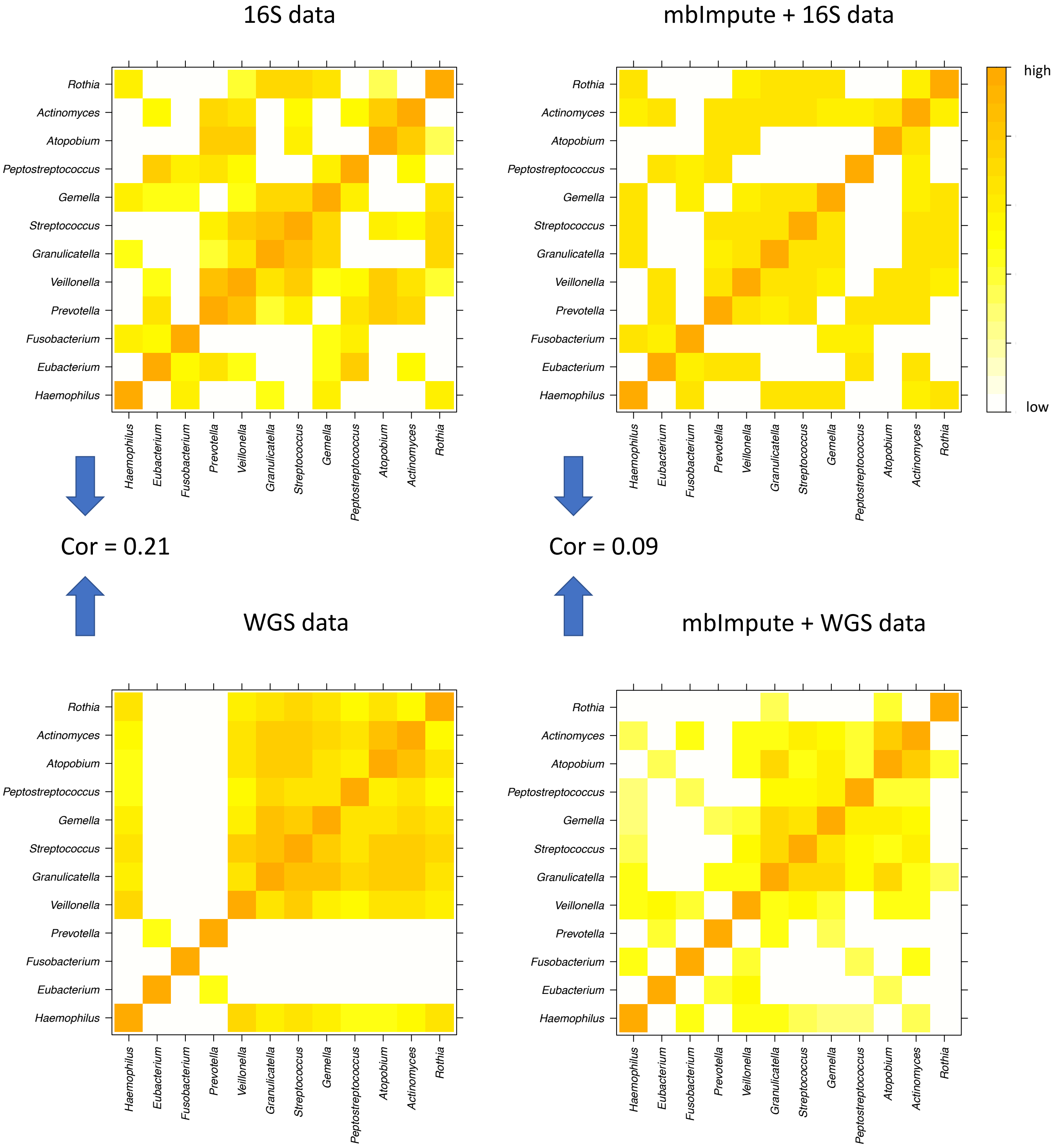


Figure 2.22: Four Pearson correlation matrices are calculated based on a common set of genus-level taxa's abundances in 16S healthy human oral samples and WGS healthy human stool samples, with or without using mbImpute as a preceding step. Before imputation, the Pearson correlation between the two correlation matrices is 0.21, and this correlation decreases to 0.09 after imputation. For illustration purposes, each heatmap shows square roots of Pearson correlations, with the bottom 40% of values truncated to 0.

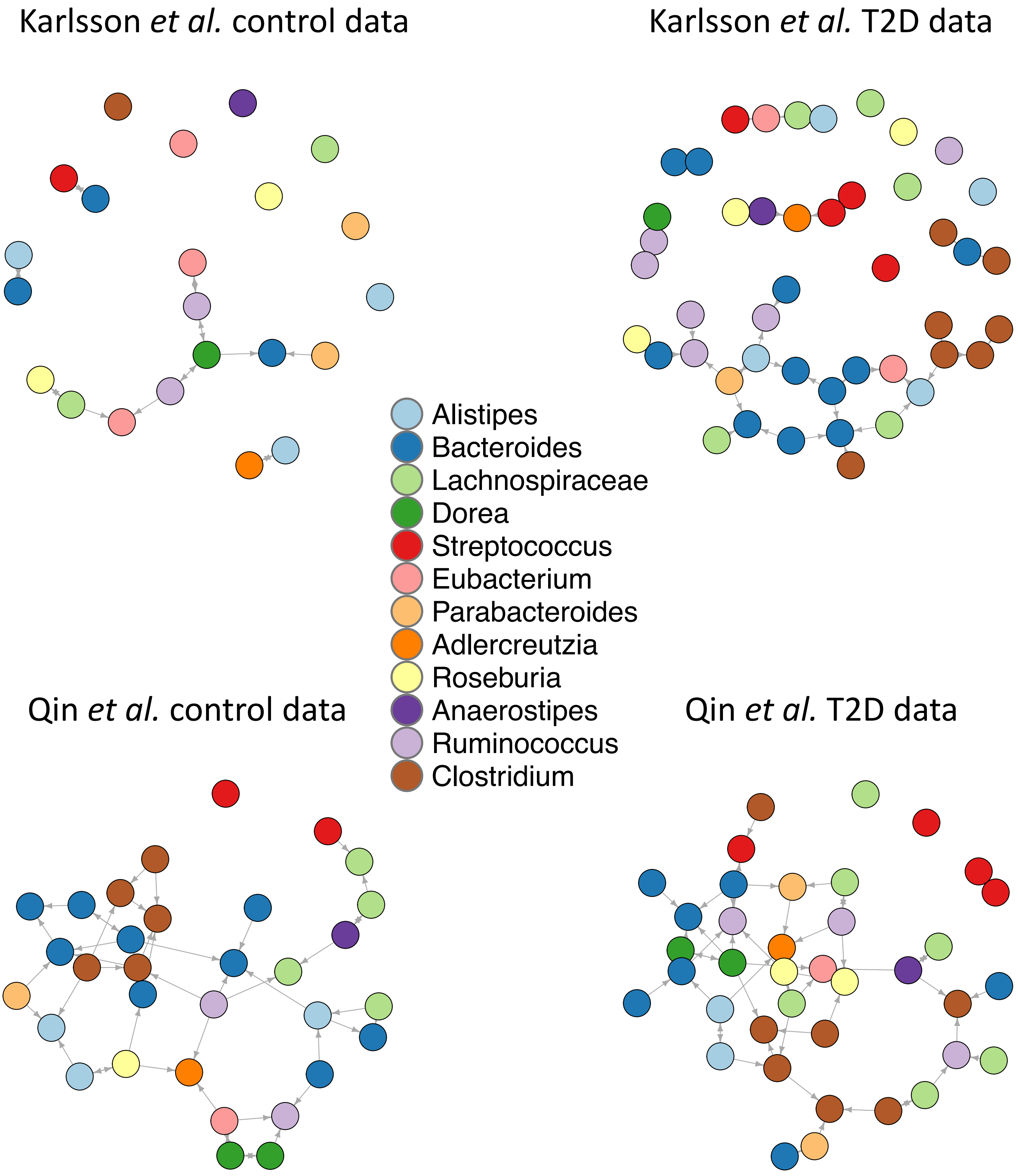


Figure 2.23: Visualization of genus-level taxon interaction networks constructed from two T2D datasets after mbImpute is applied. Four genus-level interaction networks constructed by the PC algorithm (Kalisch *et al.*, 2007) on the Karlsson *et al.* and Qin *et al.*'s control samples and T2D samples after mbImpute is applied.

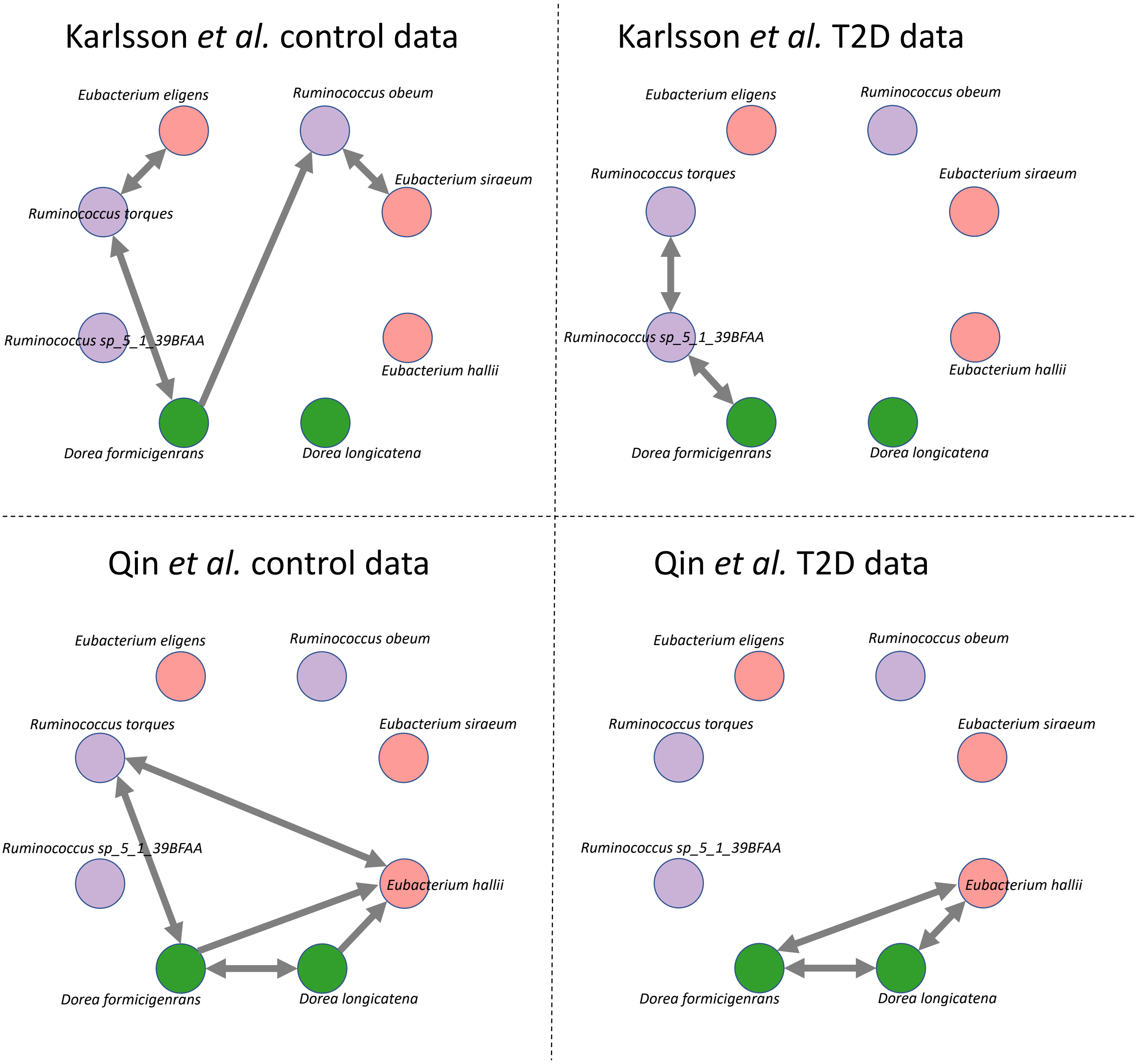


Figure 2.24: Visualization of species-level taxon interaction networks of three genera (*Eubacterium*, *Ruminococcus*, and *Dorea*) from two T2D datasets after mbImpute is applied. Four strain-level interaction networks constructed by the PC algorithm on the Karlsson *et al.* and Qin *et al.*'s control samples and T2D samples after mbImpute is applied.

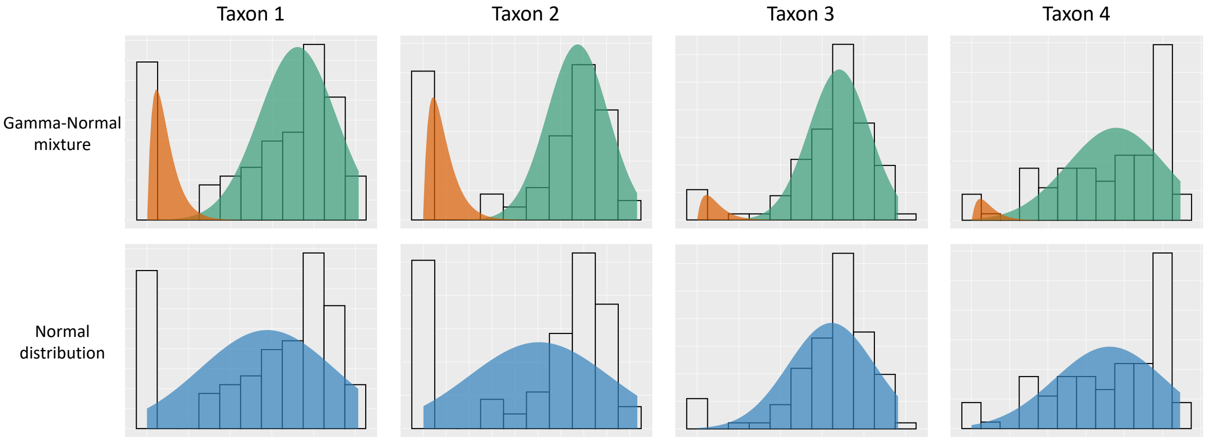


Figure 2.25: Abundance distributions of four taxa in the data from [33]. The same histogram is displayed in each column to show each taxon's abundance distribution. Top panel: fitted Gamma-normal mixture model, where the red and green areas represent the Gamma and normal components, respectively. Bottom panel: fitted normal distribution represented by the blue area. For details, see Methods in the main text.

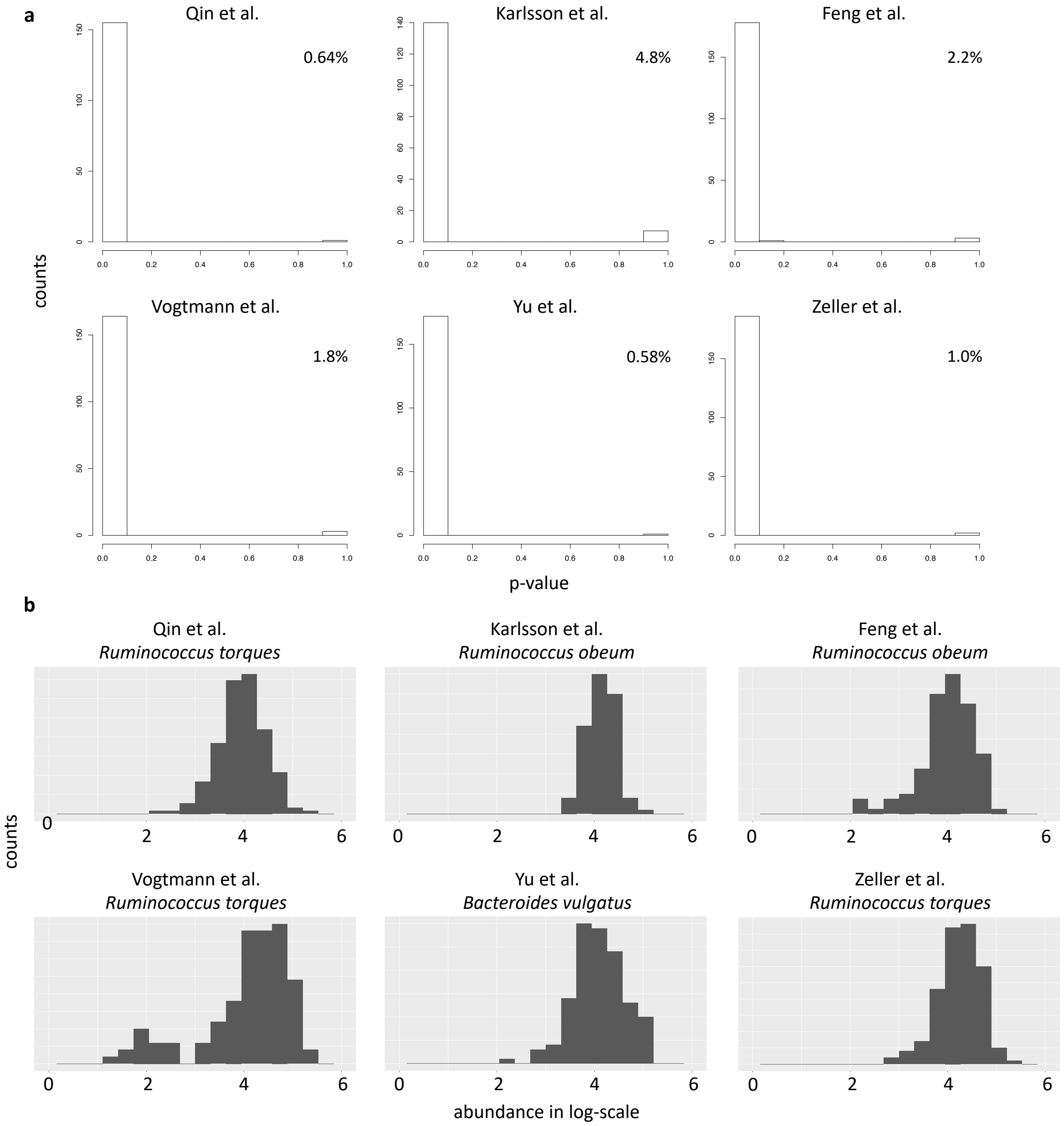


Figure 2.26: Outputs of step 1 of mbImpute on six real datasets. (a) Distributions of the LRT p-values in step 1 of mbImpute. Most p-values are close to 0. For each dataset, the percentage of taxa with > 0.05 p-values are shown at the top right. **(b)** Distributions of six example taxa, each of which has the LRT p-value > 0.05 in each dataset. For these taxa, mbImpute does not perform imputation.

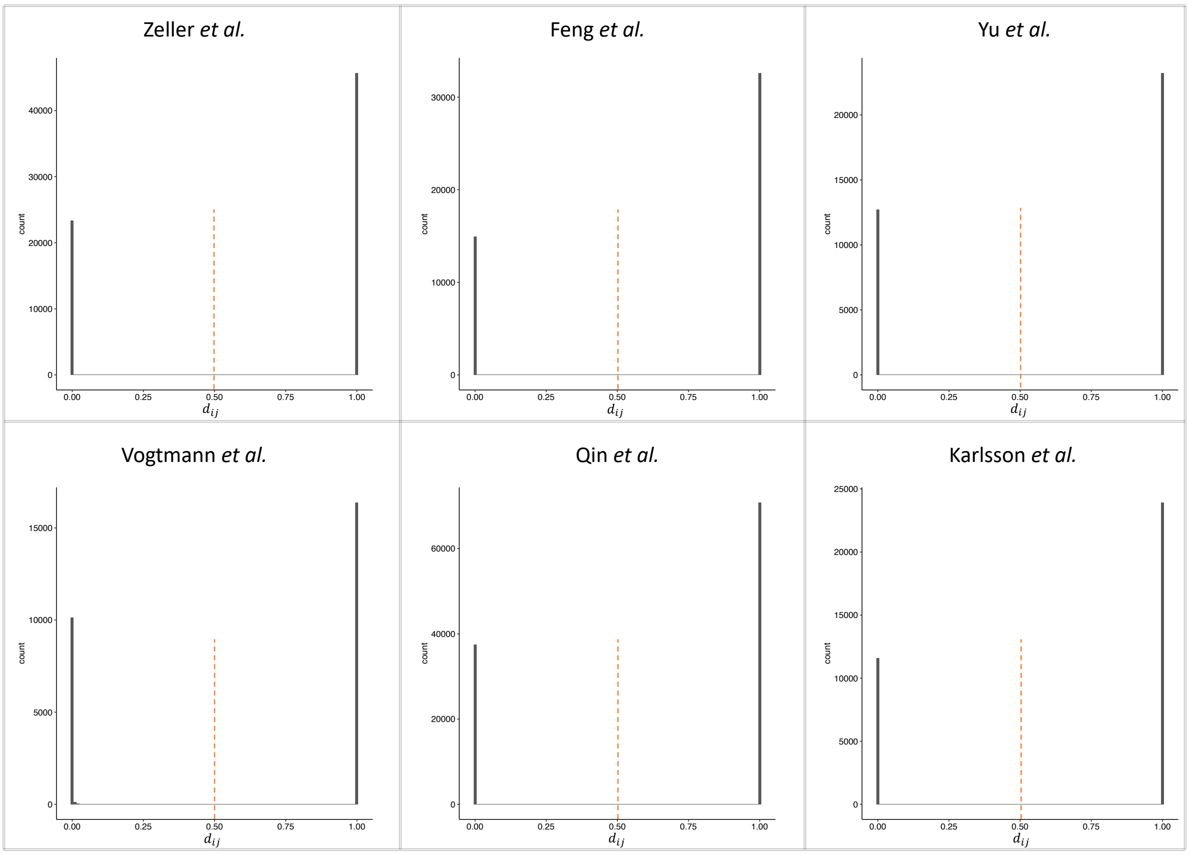


Figure 2.27: The distributions of d_{ij}'s calculated in step 1 of mbImpute applied to six real datasets. The 0.5 threshold is indicated by the red dotted line. For details, see Methods in the main text.

CHAPTER 3

How to deal with zeros in single-cell RNA-seq data and how they will affect downstream analysis

3.1 Introduction

The rapid development of single-cell technologies has brought unprecedented opportunities to quantify transcriptome heterogeneity among individual cells and transcriptome dynamics along cell developmental trajectories [155–158]. Several single-cell RNA sequencing (scRNA-seq) protocols have been developed. Two major types of protocols are (1) tag-based, unique molecular identifier (UMI)-based protocols such as Drop-seq [159] and 10x Genomics Chromium [160, 161] and (2) full-length, non-UMI-based protocols such as Smart-seq2 [162] and Fluidigm C1 [163]. Data generated by different protocols exhibit disparate accuracy and noise levels in quantifying gene expression in single cells, posing many computational and analytical challenges for researchers to extract biological knowledge from scRNA-seq data [3, 164]. Facing these challenges, computational researchers have developed hundreds of computational and statistical methods for various scRNA-seq data analytical tasks, including the selection of informative marker genes [165–169], the identification of cell types and states [167, 170–176], the reconstruction of cell developmental trajectories [177–182], and the identification of cell-type-specific genes [53, 166, 181, 183–190].

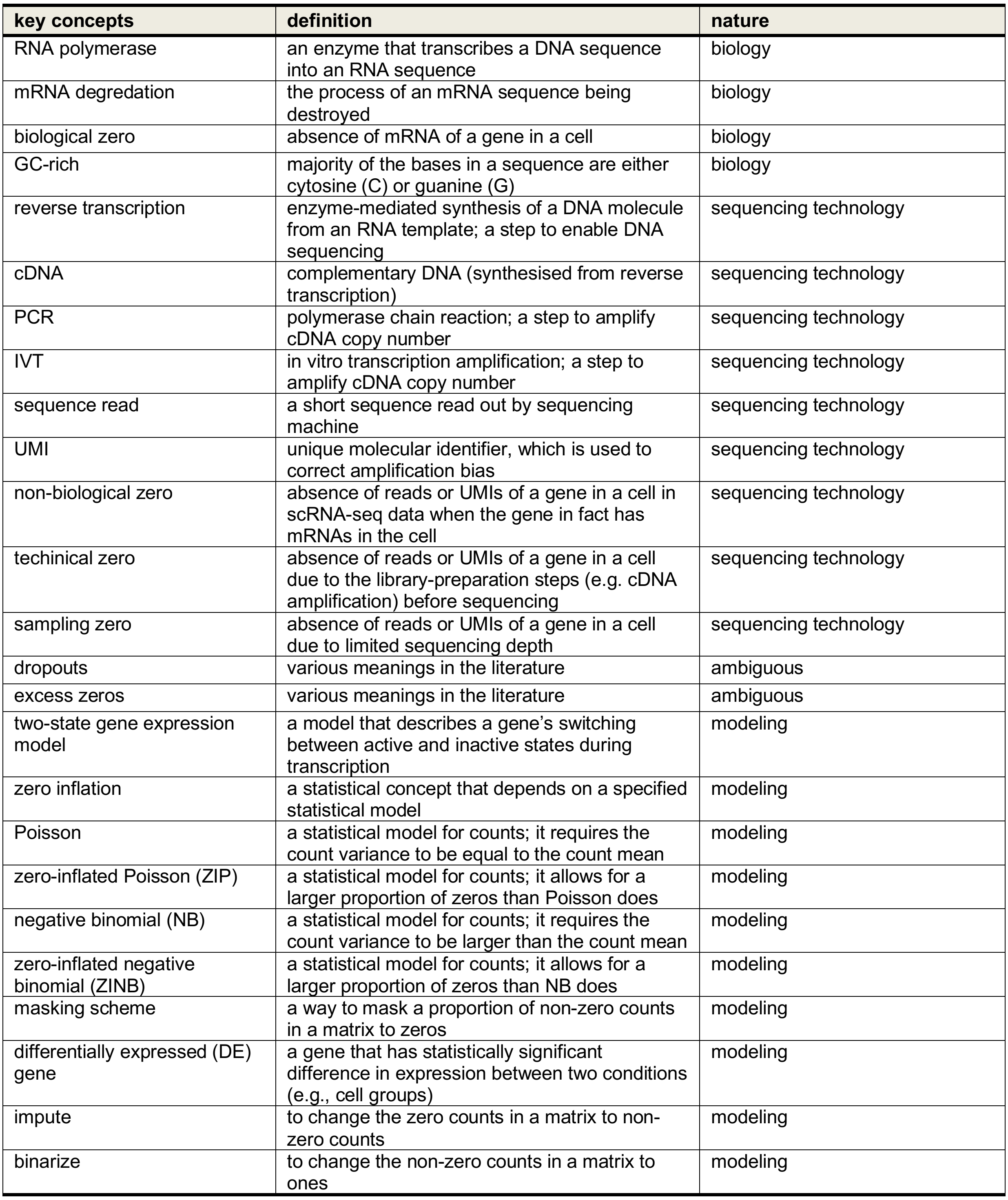
A universal analytical challenge for scRNA-seq data generated by any protocol is the vastly high proportion of genes with zero expression measurements in each cell. This data sparsity issue is apparent when scRNA-seq data are compared with bulk RNA-seq data [188, 191, 192], which contain aggregated gene expression measurements from many cells. The proportion of zeros in scRNA-seq data can be as high as 90% [193]. Such excess zeros would bias the estimation of gene expression correlations [194] and hinder the capture of gene expression dynamics [195] from scRNA-seq data. In early scRNA-seq data analyses, the high data sparsity provoked the use of zero-inflated models [188, 190, 196] and the development of imputation methods for reducing zeros [85, 173, 194, 195, 197–211]. More recently, however, there are voices against the use of zero-inflated models for scRNA-seq data generated by UMI protocols [212]. Besides, there is a proposal for treating zeros as useful information that researchers should embrace [213]. These mixed statements raised a fundamental question to the scRNA-seq field: should we use or remove these zeros in scRNA-seq data analysis?

Here we offer some perspectives on these two puzzling questions by discussing the sources of zeros in scRNA-seq data, the impacts of zeros on various data analyses, the existing approaches for handling zeros, and the pros and cons of these approaches. In detail, we first define biological and non-biological zeros arising from scRNA-seq data generation, and we clarify several ambiguous terms about zeros in the scRNA-seq literature. Second, we summarize three commonly used approaches for handling zeros—direct statistical modeling, imputation, and binarization—and discuss their respective pros and cons. Third, we use scRNA-seq data generated by Drop-seq, 10x Genomics, and Smart-seq2 to demonstrate the relation between zero patterns and protocols. In this case, we use simulation studies to evaluate the effects of zeros and zero-generation mechanisms on cell clustering and differentially expressed (DE) gene identification. Last, we provide some practical advice and future directions for bioinformatics tool developers and users in dealing with the highly sparse scRNA-seq data.

We summarize the key concepts used in this paper, including their definitions and nature (biology, technology, or modeling), in Table3.1.

3.2 Sources of zeros in scRNA-seq data

Zero measurements in scRNA-seq data have two sources: biological and non-biological (Fig. 3.1). While biological zeros carry meaningful information about cell states, non-



key concepts	definition	nature
RNA polymerase	an enzyme that transcribes a DNA sequence into an RNA sequence	biology
mRNA degredation	the process of an mRNA sequence being destroyed	biology
biological zero	absence of mRNA of a gene in a cell	biology
GC-rich	majority of the bases in a sequence are either cytosine (C) or guanine (G)	biology
reverse transcription	enzyme-mediated synthesis of a DNA molecule from an RNA template; a step to enable DNA sequencing	sequencing technology
cDNA	complementary DNA (synthesised from reverse transcription)	sequencing technology
PCR	polymerase chain reaction; a step to amplify cDNA copy number	sequencing technology
IVT	in vitro transcription amplification; a step to amplify cDNA copy number	sequencing technology
sequence read	a short sequence read out by sequencing machine	sequencing technology
UMI	unique molecular identifier, which is used to correct amplification bias	sequencing technology
non-biological zero	absence of reads or UMIs of a gene in a cell in scRNA-seq data when the gene in fact has mRNAs in the cell	sequencing technology
techinical zero	absence of reads or UMIs of a gene in a cell due to the library-preparation steps (e.g. cDNA amplification) before sequencing	sequencing technology
sampling zero	absence of reads or UMIs of a gene in a cell due to limited sequencing depth	sequencing technology
dropouts	various meanings in the literature	ambiguous
excess zeros	various meanings in the literature	ambiguous
two-state gene expression model	a model that describes a gene's switching between active and inactive states during transcription	modeling
zero inflation	a statistical concept that depends on a specified statistical model	modeling
Poisson	a statistical model for counts; it requires the count variance to be equal to the count mean	modeling
zero-inflated Poisson (ZIP)	a statistical model for counts; it allows for a larger proportion of zeros than Poisson does	modeling
negative binomial (NB)	a statistical model for counts; it requires the count variance to be larger than the count mean	modeling
zero-inflated negative binomial (ZINB)	a statistical model for counts; it allows for a larger proportion of zeros than NB does	modeling
masking scheme	a way to mask a proportion of non-zero counts in a matrix to zeros	modeling
differentially expressed (DE) gene	a gene that has statistically significant difference in expression between two conditions (e.g., cell groups)	modeling
impute	to change the zero counts in a matrix to non-zero counts	modeling
binarize	to change the non-zero counts in a matrix to ones	modeling

Table 3.1: A summary of the key concepts used in this paper, including their definitions and nature.

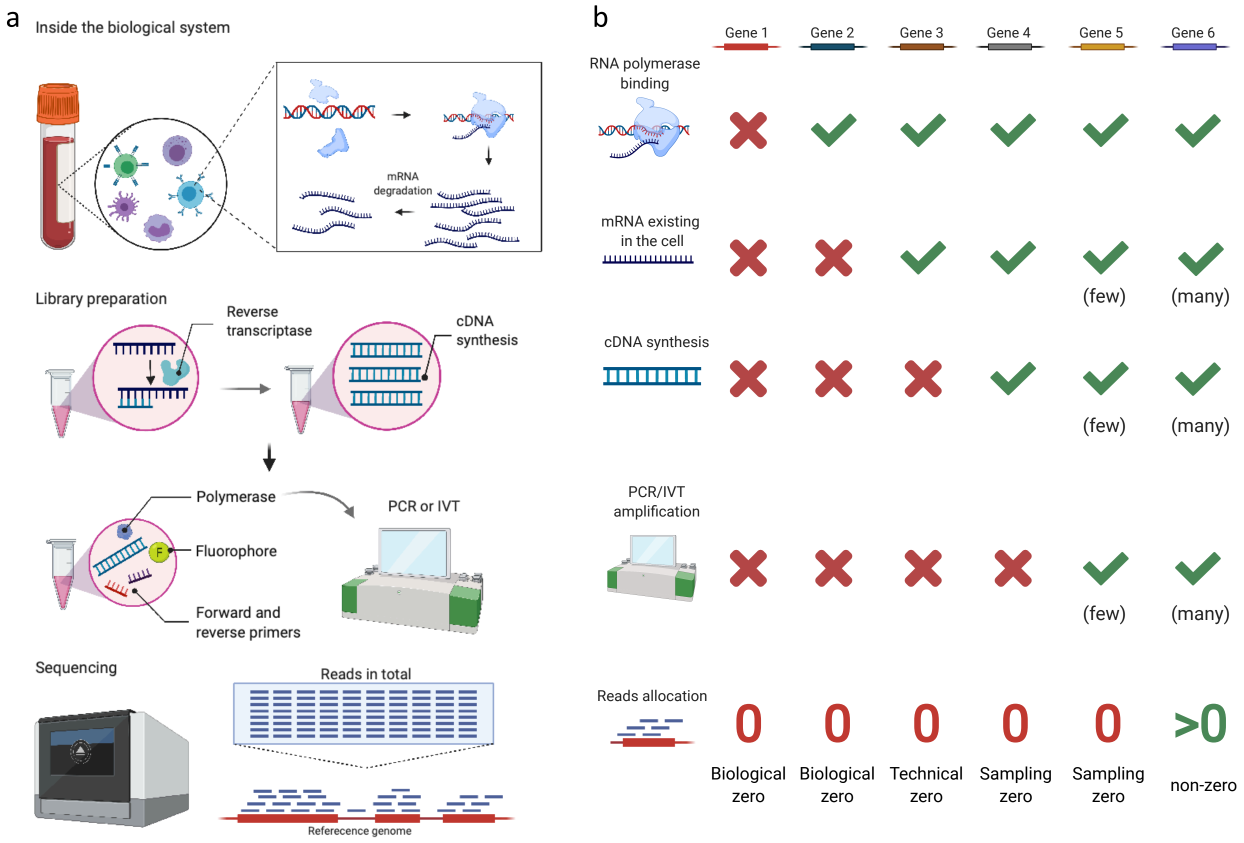


Figure 3.1: Sources of zeros in scRNA-seq data. (a) An overview of a scRNA-seq experiment. Biological factors that determine true gene expression levels include transcription and mRNA degradation (top panel). Technical procedures that affect gene expression measurements include cDNA synthesis, PCR or IVT amplification, and sequencing depth (bottom three panels). Finally, every gene's expression measurement in each cell is defined as the number of reads mapped to that gene in that cell. **(b)** How the biological factors and technical procedures in (a) lead to biological, technical, and sampling zeros in scRNA-seq data. Red crosses indicate occurrences of zeros, while green checkmarks indicate otherwise. Biological zeros arise from two scenarios: no transcription (gene 1) or no mRNA due to faster mRNA degradation than transcription (gene 2). If a gene has mRNAs in a cell, but its mRNAs are not captured by cDNA synthesis, the gene's zero expression measurement is called a technical zero (gene 3). If a gene has cDNAs in the sequencing library, but its cDNAs are too few to be captured by sequencing, the gene's zero expression measurement is called a sampling zero. Sampling zeros occur for two reasons: a gene's cDNAs have few copies because they are not amplified by PCR or IVT (gene 4), or a gene's mRNA copy number is too small so that its cDNAs still have few copies after amplification (gene 5). If the factors and procedures above do not result in few cDNAs of a gene in the sequencing library, the gene would have a non-zero measurement (gene 6). The figure is created with BioRender.com.

biological zeros represent missing values artificially introduced during the generation of scRNA-seq data. In our paper, non-biological zeros include technical zeros, which occur during the preparation of biological samples for sequencing, and sampling zeros, which arise due to limited sequencing depths. Our classification of zeros in sequencing data into biological, technical, and sampling zeros is aligned with the classification in Silverman et al. [41] except a slight difference (we refer to the zeros due to inefficient amplificaiton, e.g., PCR, as sampling zeros, while Silverman et al called them technical zeros). The non-biological zeros have typically been viewed as impediments to the full and accurate interpretation of cell states and the differences between them. Fig. 3.1a provides an overview of a scRNA-seq experiment, and it highlights the biological factors and technical procedures that may lead to zeros in scRNA-seq data. Fig. 3.1b summarizes how biological factors result in biological zeros and how technical procedures cause non-biological zeros, including technical zeros and sampling zeros. It is worth noting that biological and non-biological zeros are hardly distinguishable in scRNA-seq data without biological knowledge or spike-in control (see Future directions).

Biological zeros in scRNA-seq data

A biological zero is defined as the true absence of a gene's transcripts or messenger RNAs (mRNAs) in a cell [41]. Biological zeros occur for two reasons (Fig. 3.1b). First, many genes are unexpressed in a cell (e.g., gene 1 in Fig. 3.1b), and cells of distinct types have different genes expressed—a fact that results in the diversity of cell types [214, 215]. Second, many genes undergo a bursty process of transcription (i.e., mRNA synthesis); that is, these genes are not transcribed constantly but intermittently, a well-known phenomenon in gene regulation [190, 191, 196, 216–218]. Specifically, in eukaryotic cells, transcription is initiated by the binding of specific transcription factors (TFs) and RNA polymerase to the promoter of a gene [219–221]. Due to the stochasticity of TF binding, a gene switches between active and inactive states, and its transcription only occurs during the active state [222]. Hence, systems biologists have used a two-state gene expression model to describe how the rates of three processes—active/inactive state switching, transcription, and mRNA degradation—

jointly determine the distribution of a gene's mRNA copy numbers, i.e., expression levels, in cells of the same type [222–224]. Fig. 3.2 illustrates the model and provides three example settings of model parameters along with their corresponding gene expression distributions. Depending on the gene's switching rates between the active and inactive states, transcription rate, and degradation rate, the resulting distribution may exhibit a mode near zero, which makes it appear that the gene expresses no mRNA at a particular time, in a large number of cells (e.g., gene 2 in Fig. 3.1b).

3.2.1 Non-biological zeros in scRNA-seq data

Non-biological zeros reflect the loss of information about truly expressed genes due to the inefficiencies of the technologies employed from sample collection to sequencing. Unlike biological zeros, non-biological zeros refer to the zero expression measurements of genes with transcripts in a cell. There are two types of non-biological zeros [41]: (1) technical zeros, which arise from library-preparation steps before sequencing, and (2) sampling zeros, which result from a limited sequencing depth.

One cause of technical zeros is the imperfect mRNA capture efficiency in the reverse transcription (RT) step from mRNA to cDNA. The efficiency has a considerable variation across protocols and may be as low as 20% [225], depending on multiple experimental parameters [226]. The efficiency may even differ between mRNA transcripts. For example, if an mRNA transcript has an intricate secondary structure or is bound to proteins, it would not be reversely transcribed to cDNA efficiently [3, 186, 190]. In summary, if a gene's mRNA transcripts in a cell are not converted into cDNA molecules (cDNAs), the gene would falsely appear as non-expressed in that cell in the sequencing library, resulting in a technical zero in scRNA-seq data (e.g., gene 3 in Fig. 3.1b).

The other type of non-biological zeros, sampling zeros, occurs due to a constraint on the total number of reads sequenced, i.e., the sequencing depth [212, 227], which is determined by the experimental budget and sequencing machine. During sequencing, cDNAs are randomly captured ("sampled") and sequenced into reads. Hence, a gene with fewer cDNAs is more

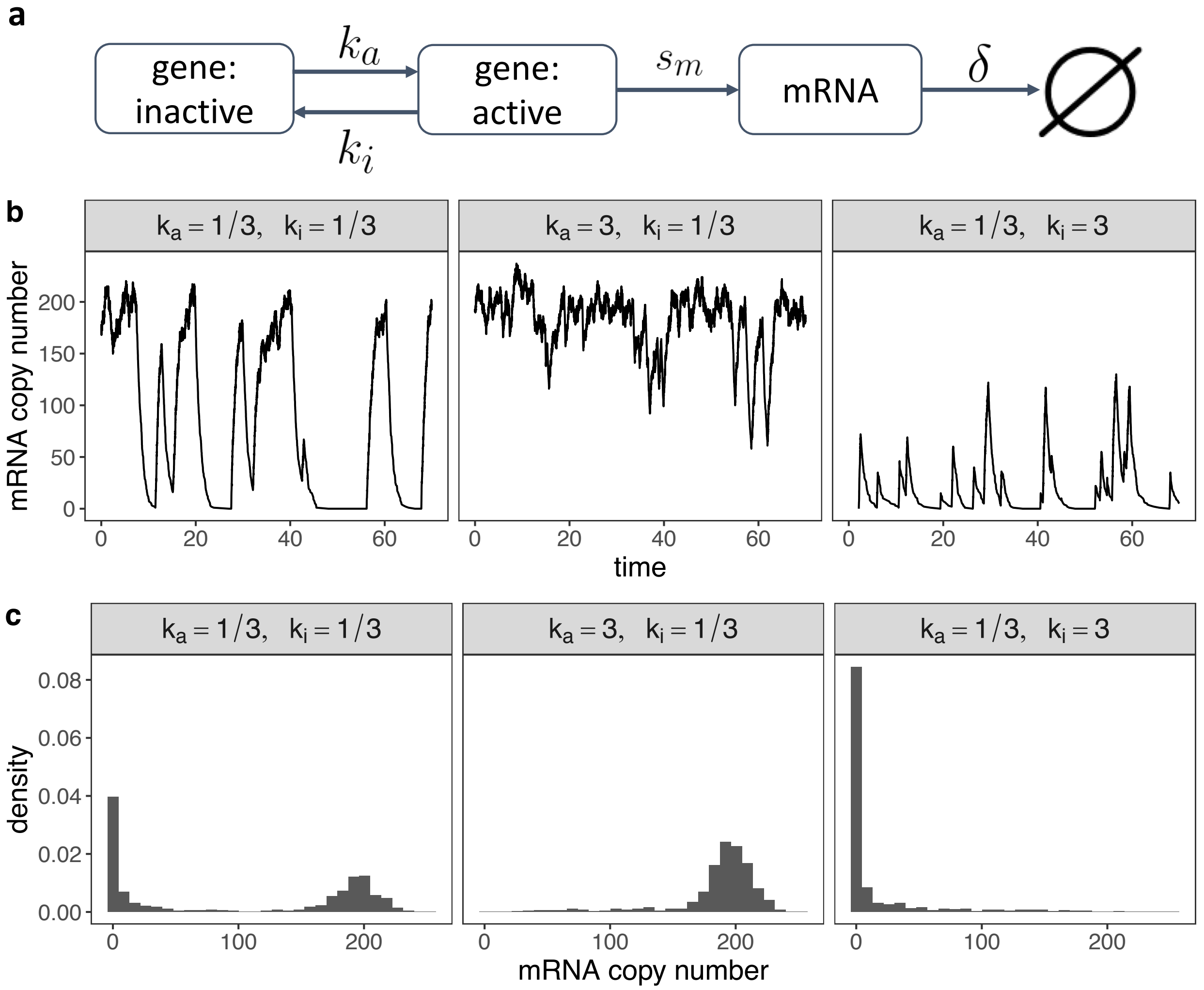
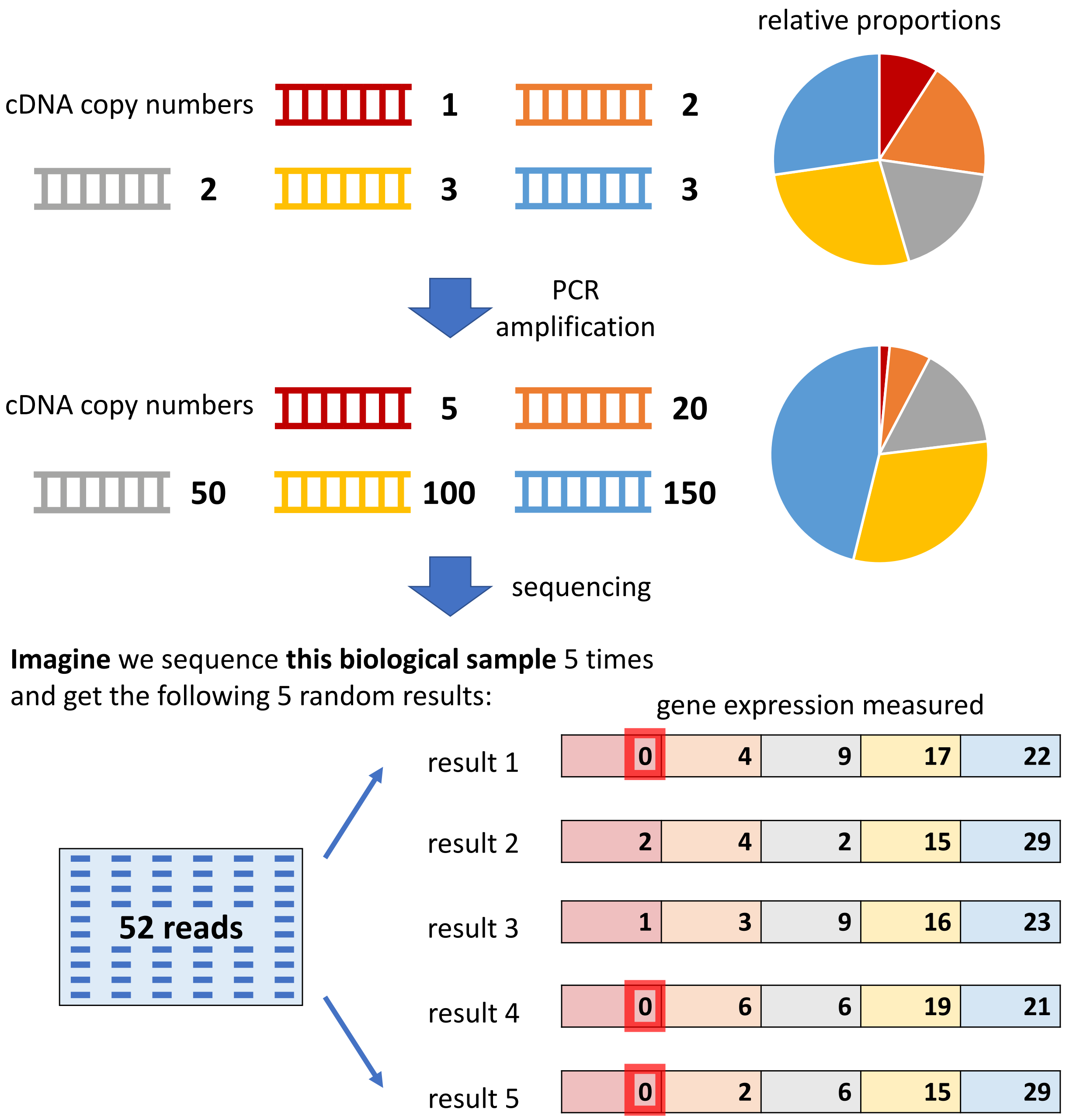


Figure 3.2: A two-state stochastic model of the expression levels of one gene. **(a)** A diagram of the two-state gene expression model [222–224], where a gene stochastically switches from an inactive state to an active state at rate k_a and from an active state to an inactive state at rate k_i. The gene transcribes mRNA at rate s_m only when it is in the active state. The transcribed mRNA then degrades at rate δ. **(b)** Given $s_m = 200$ and $\delta = 1$, the effects of k_a and k_i on the temporal dynamics of the gene's mRNA copy number. Three example values of k_a and k_i are provided. Left: when both k_a and k_i are small, the mRNA copy number switches between small and large values. Middle: when k_a is much larger than k_i, the mRNA copy number remains large most of the time. Right: when k_a is much smaller than k_i, the mRNA copy number remains small most of the time. **(c)** Distributions of the gene's mRNA copy number (across cells) corresponding to the three example settings in (b). Left: when the gene's mRNA copy number switches between small and large values, the resulting distribution is bimodal with two modes at zero and around s_m/δ. Middle: when the gene's mRNA copy number is large most of the time, the resulting distribution has a single mode around s_m/δ. Right: when the gene's mRNA copy number is small most of the time, the resulting distribution has a single mode at zero. In summary, when k_a is small, the gene is expected to have biological zeros in cells with non-negligible probability.

likely to be undetected due to this random sampling. If undetected, the gene's resulting zero read count is a "sampling zero." There are two reasons why a gene (in a cell) may have too few cDNAs in the sequencing library: too few cDNAs before amplification and inefficient cDNA amplification. Below we explain why cDNA amplification may cause some genes to have a disproportionally low cDNA copy number in the sequencing library (see Fig. 3.3).

The cDNA amplification step is essential for scRNA-seq, as it increases the number of cDNA copies of a gene so that the gene is more likely to be detected by sequencing. Polymerase chain reaction (PCR) [228] is the most widely-used amplification procedure. However, PCR amplification is non-linear; thus, the ratio between the copy numbers of two differentially expressed genes is artificially distorted by PCR, i.e., a ratio greater (or smaller) than one becomes even larger (or smaller) after PCR. As a remedy, *in vitro* transcription (IVT) has been developed for linear amplification [229]. However, compared with PCR, IVT requires more input cDNAs to ensure successful amplification; thus, PCR is still the dominant amplification procedure for scRNA-seq [230]. Though indispensable, cDNA amplification is known to introduce biases into cDNA copy numbers because the amplification efficiency depends on cDNA sequence and structure [231, 232]. For example, GC-rich cDNA sequences are harder to be amplified [41]. The amplification efficiency also depends on the design of cell barcodes, adapters, and primers; overlaps or complementarity of barcode, adapter, and primer sequences would induce cDNA secondary structures and thus reduce the amplification efficiency [233, 234]. Moreover, cDNA copy number biases would accumulate as the number of amplification cycles increases [231, 235]; that is, the more cycles, the larger the difference of two genes (with different amplification efficiency) in cDNA copy numbers [236, 237]. Since different scRNA-seq experiments may use different numbers of amplification cycles (e.g., 18 cycles in a Smart-seq2 experiment [238] and 14 cycles in a 10x Genomics experiment [239]), cDNA copy number biases differ among scRNA-seq datasets. In addition, the non-linear amplification nature of PCR would exaggerate the expression level differences between lowly-expressed and highly-expressed genes. Altogether, due to amplification biases, cDNA copy numbers in a sequencing library may not reflect cDNAs' actual proportions before amplification. As a result, the genes with small cDNA proportions in the sequencing library



result 1	0	4	9	17	22
result 2	2	4	2	15	29
result 3	1	3	9	16	23
result 4	0	6	6	19	21
result 5	0	2	6	15	29

Figure 3.3: A toy example showing how the PCR amplification may result in sampling zeros. Five genes have their cDNAs amplified by PCR. After the non-linear amplification, their relative proportions change. If the sequencing depth is limited to 52 reads, the first gene has sampling zeros in three out of five hypothetical sequencing experiments.

are likely to be missed by sequencing and thus result in sampling zeros (e.g., gene 4 suffering from inefficient amplification and gene 5 having too few cDNAs in Fig. 3.1b).

3.2.2 Clarification of zero-related terminology

In the current scRNA-seq literature, much ambiguity exists in the use of terms including "dropouts," "excess zeros," and "zero-inflation" to describe the prevalence of zeros in scRNA-seq data [240]. We clarify the three terms by summarizing their various uses in the scRNA-seq field to facilitate our discussion.

Dropout or **dropouts** are widely used regarding the prevalence of zeros in scRNA-seq data. It was first introduced in the SCDE method paper: "dropout describes zero gene expression for the genes that show moderate or high expressions in only a proportion of cells [190]." Hence, dropouts, as a data-driven concept, are not equivalent to either biological or non-biological zeros. Nevertheless, the use of "dropouts" in later papers became inconsistent and confusing: most papers meant non-biological zeros [173, 188, 192, 202, 205, 241, 242]; some meant non-biological zeros and low expression measurements [195, 243]; some meant all zeros [196, 197, 244]. In addition, "dropout" was often used as an adjective to mean the existence of many zeros [245]. Such inconsistent uses of "dropouts" are emphasized in a recent work [240]. To avoid possible confusion, we will not use "dropout" or "dropouts" in the following text.

Excess zeros are used in various ways: some papers referred to the larger proportion of zeros in scRNA-seq data than in bulk RNA-seq data [192]; some meant non-biological zeros [195, 242]; some meant the additional zeros that cannot be explained by the negative binomial (NB) model [243]. To avoid confusion, we will not use "excess zeros" in the following text.

Zero inflation, unlike the first two terms, is a statistical concept that depends on a specified model, i.e., a count distribution such as the Poisson distribution and the NB distribution [241]. It means the proportion of zeros that exceeds what is expected under the specified model [192]. We will use "zero inflation" in the following discussion because its definition has no ambiguity.

3.3 Debate on zero-inflated modeling of scRNA-seq data

Since the advent of scRNA-seq, zero-inflated models have been widely used in bioinformatics tools on the observed scRNA-seq count data [188, 190, 196, 246]. Zero-inflated models are mixture probabilistic models with two components: a point mass at zero and a common distribution, including the Poisson and NB distributions for read or UMI counts and the normal distribution for log-transformed read or UMI counts. More recently, however, researchers have found that UMI counts are not zero-inflated when compared with the Poisson or NB distribution [41, 160, 239, 240].

The use of UMIs in scRNA-seq can correct the amplification biases in non-zero gene expression measurements [247]; that is, UMIs can be used to identify and remove reads from cDNA duplicates that are results of amplification, and thus some non-zero gene expression measurements would be reduced. However, UMIs cannot help recover sampling zeros, whose corresponding cDNA copy numbers stay unknown despite the use of UMIs [246]. In fact, UMIs cannot reduce any zeros, including biological and non-biological ones. The change of modeling choice—from zero-inflated models for non-UMI-based data to non-zero-inflated models for UMI-based data—indicates that whether or not to use zero-inflated models has nothing to do with the prevalence of zeros. In other words, the modeling choice is a statistical consideration and says nothing about the proportions of zeros or the distinction between biological and non-biological zeros.

Four count distributions—Poisson, zero-inflated Poisson (ZIP), NB, and zero-inflated NB (ZINB)—have been widely used to model a single gene's read or UMI counts across cells in scRNA-seq data. In fact, the former three models are special cases of the ZINB model (Fig. 3.4a). Poisson only has one parameter (λ) equal to both mean and variance

(Fig. 3.4b). Compared with Poisson, ZIP has one more zero-inflation parameter (p) to indicate the proportion of additional zeros that do not come from Poisson (Fig. 3.4c); when this zero-inflation parameter is zero, ZIP reduces to Poisson. Also, compared with Poisson, NB has one more dispersion parameter (ψ) that indicates the over-dispersion of variance relative to the mean (i.e., unlike Poisson, NB has variance greater than mean; Fig. 3.4d); when this dispersion parameter is positive infinity, NB reduces to Poisson. Compared with NB, ZINB has one more zero-inflation parameter (p) to indicate the proportion of additional zeros that do not come from NB (Fig. 3.4e); when this zero-inflation parameter is zero, ZINB reduces to NB.

For a fair comparison, we illustrate these four distributions, with example parameters such that they all have the same mean as one (Fig. 3.4b–e). With the same mean, ZIP and NB have more zeros than Poisson does, and ZINB has the most zeros. Between ZIP and NB, which one has more zeros depends on their parameter values, and when they have the same zero proportion, their non-zero distributions are still different. Moreover, when the four distributions have the same mean, compared with Poisson and ZIP, NB and ZINB have heavier right tails, i.e., greater probabilities of taking larger values.

Svensson shows that non-zero-inflated distributions can describe the variation in droplet scRNA-seq data using droplet-based ERCC spike-in data. To evaluate this claim on real scRNA-seq data with both droplet-based and full-length protocols, we perform the similar analysis on three real scRNA-seq PBMC datasets. More specifically, we fit the above four count distributions—two zero-inflated (ZIP and ZINB) and two non-zero-inflated (Poisson and NB)—to a non-UMI-based dataset generated by Smart-seq2 and two UMI-based datasets generated by 10x Genomics and Drop-seq. These three datasets are ideal for studying how the modeling choice depends on the experimental protocol, as they were generated by a benchmark study [193] that applied multiple scRNA-seq protocols to measure peripheral blood mononuclear cells (PBMCs) from the same batch, and the benchmark study labeled cells using the same cell types and curated genes to be the same across protocols. We first compare the three datasets in terms of their distributions of cell library size (i.e., the total number of reads or UMIs in each cell), numbers of cells, and distributions of the

a

distribution	mean	variance	zero proportion, i.e., P(X=0)
Poisson(λ)	λ	λ	$\exp(-\lambda)$
ZIP(λ, p)	$\lambda \cdot (1-p)$	$\lambda \cdot (1-p) + \lambda^2 \cdot (1-p) \cdot p$	$\exp(-\lambda) \cdot (1-p) + p$
NB(λ, ψ)	λ	$\lambda + \frac{\lambda^2}{\psi}$	$(\frac{\psi}{\psi+\lambda})^{\psi}$
ZINB(λ, ψ, p)	$\lambda \cdot (1-p)$	$\lambda \cdot (1-p) + \lambda^2 \cdot (1-p) \cdot (p + \frac{1}{\psi})$	$(\frac{\psi}{\psi+\lambda})^{\psi} \cdot (1-p) + p$

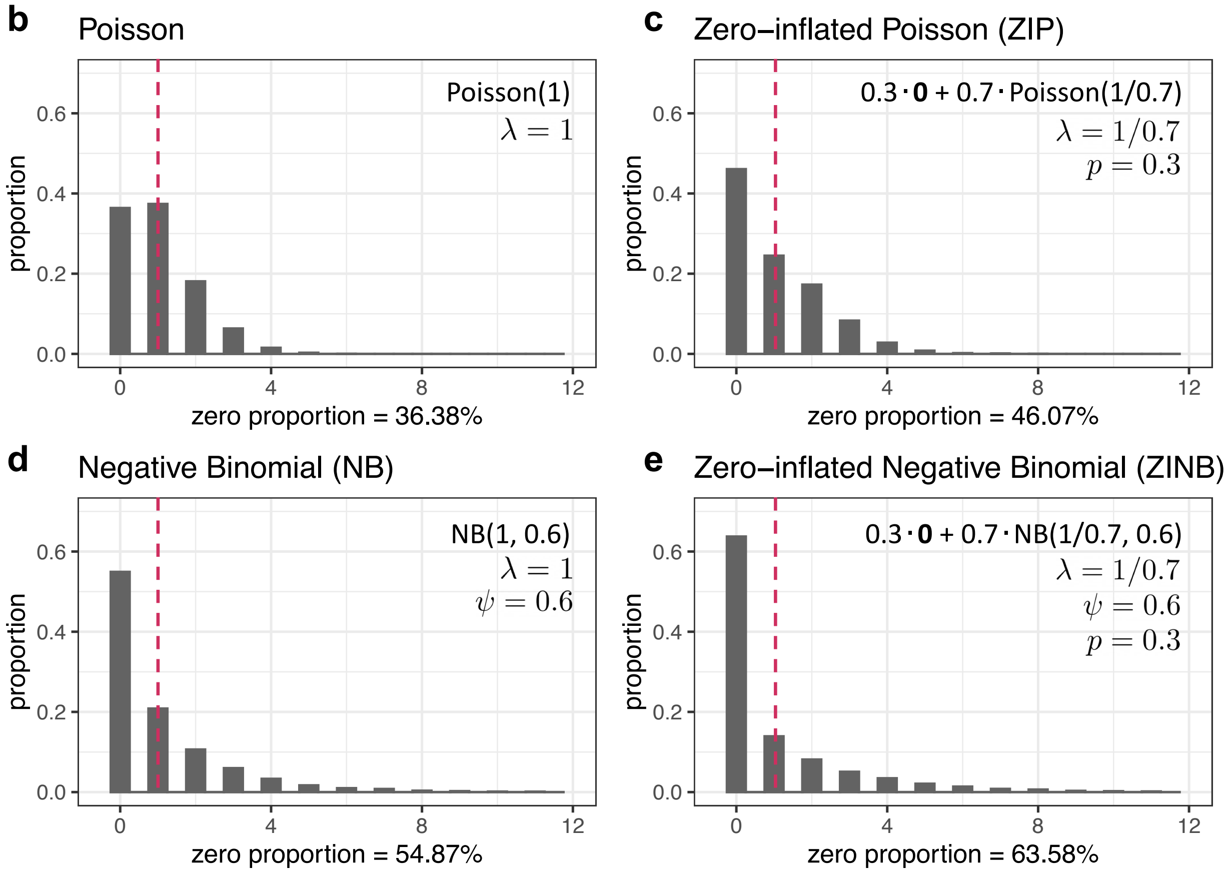


Figure 3.4: Four count distributions: Poisson, zero-inflated Poisson (ZIP), negative binomial (NB), and zero-inflated negative binomial (ZINB). (a) Parameterization, mean, variance, and zero proportion of each of the four distributions. **(b)**, **(c)**, **(d)**, and **(e)** Illustration of the probability mass functions of Poisson (b), ZIP (c), NB (d), and ZINB (e) distributions that all have mean equal to 1. The horizontal axis indicates each possible value, and the vertical axis indicates the probability of taking each possible value. For each distribution, the parameter values are listed on the top right, and the zero proportion is listed at the bottom.

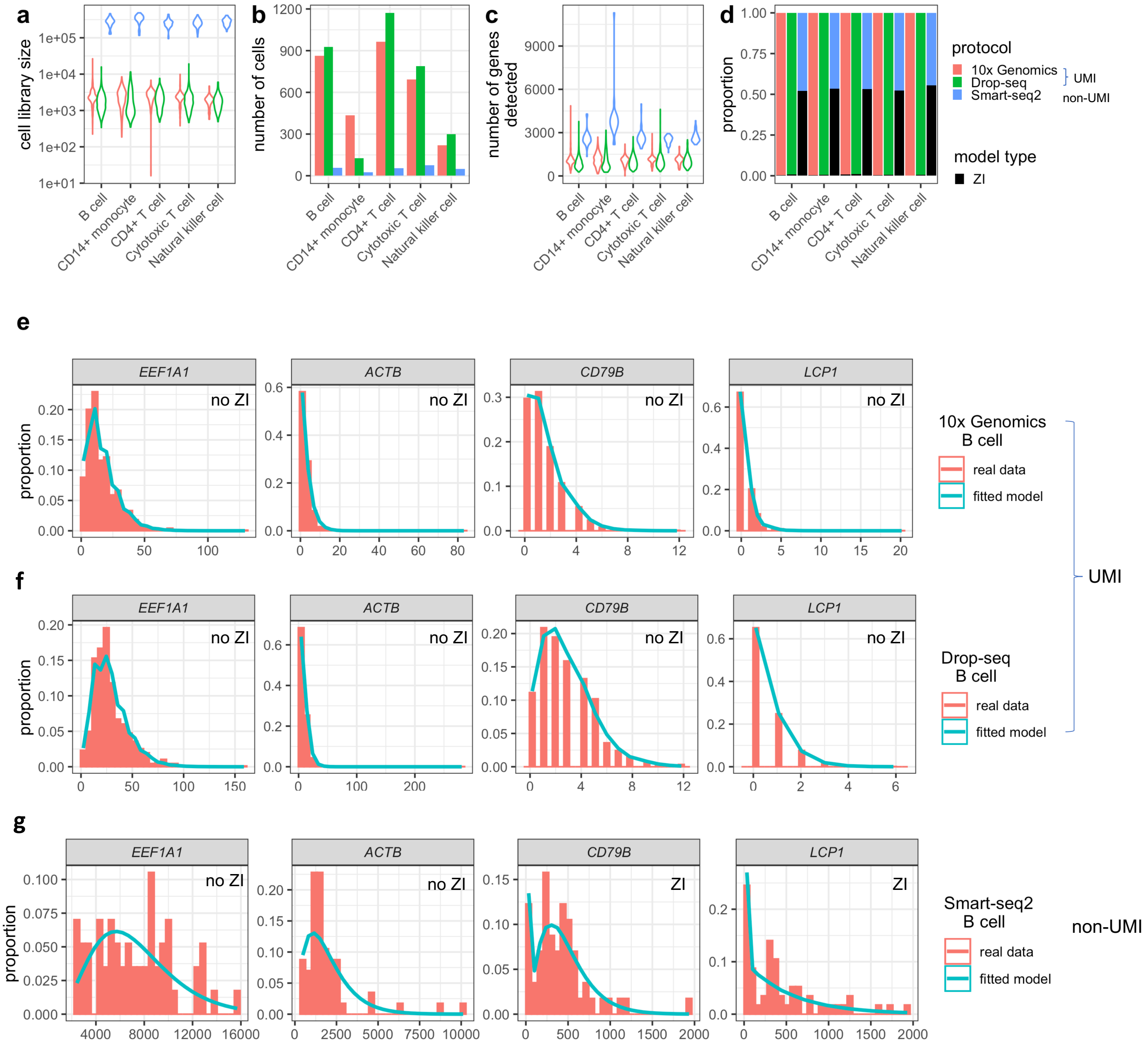


Figure 3.5: Statistical modeling of 10x Genomics, Drop-seq, and Smart-seq2 data for the same PMBC sample. 10x Genomics and Drop-seq data are UMI-based, while Smart-seq2 data are non-UMI-based. **(a)** Violin plots showing the distribution of cell library sizes for each of five PMBC cell types measured by each protocol [193]. For the UMI-based protocols (10x Genomics and Drop-seq) and Smart-seq2, a cell's library size is defined as the total number of UMIs and reads, respectively, in that cell. **(b)** Barplots showing the number of cells detected for each cell type by each protocol. **(c)** Violin plots showing the distribution of the number of genes detected per cell. **(d)** Barplots showing the proportions of genes for which zero-inflated (ZI) models are chosen (black) and genes for which non-zero-inflated models are chosen (non-black). The model selection is done by likelihood ratio tests. **(e)** Four example genes' distributions of UMI counts in B cells measured by 10x Genomics. The observed count distributions are shown in histograms. Non-zero-inflated (no ZI) models are chosen, and the fitted model distributions are shown in cyan curves. **(f)** The same four example genes' distributions of UMI counts in B cells measured by Drop-seq. **(g)** The same four example genes' distributions of read counts in B cells measured by Smart-seq2. ZI models are chosen for two genes.

number of genes detected per cell. Fig. 3.5a–c show that, compared with the two UMI-based datasets, the Smart-seq2 (non-UMI-based) dataset has larger cell library sizes, fewer cells, and more genes detected—a phenomenon consistent across the five cell types (B cells, CD14+ monocytes, CD4+ T cells, cytotoxic T cells, and natural killer cells).

Next, for each gene in each dataset, we fit the four distributions to its read or UMI counts in cells of each type, and we choose its distribution among the four distributions by likelihood ratio tests (see [248] for detail). The rationale is to choose the least complex distribution that fits the data well. Fig. 3.5d shows that non-zero-inflated distributions (Poisson and NB) are chosen for almost all genes in the 10x Genomics and Drop-seq datasets, while zero-inflated distributions (ZIP and ZINB) are chosen for about half of the genes in the Smart-seq2 dataset. This result is consistent with the recent advocate for not using zero-inflated models for UMI-based data [212, 240], and it suggests that zero-inflated modeling is still useful for Smart-seq2 data. For illustration purposes, in Fig. 3.5e–g, we plot the read or UMI count distributions for four genes (*EEF1A1*, *ACTB*, *CD79B*, and *LCP1*) in B cells in these three datasets, and we also plot the fitted chosen distribution for each gene. Specifically, non-zero-inflated distributions are chosen for all the four genes in the UMI-based datasets, while zero-inflated distributions are chosen for *CD79B* and *LCP1* in the Smart-seq2 dataset. Our results show that the same gene's expression distribution under the same biological condition may be described by different statistical models for data generated by different protocols, confirming that zero inflation provides no direct information on biological zeros, whose existence does not depend on protocols (Fig. 3.1b).

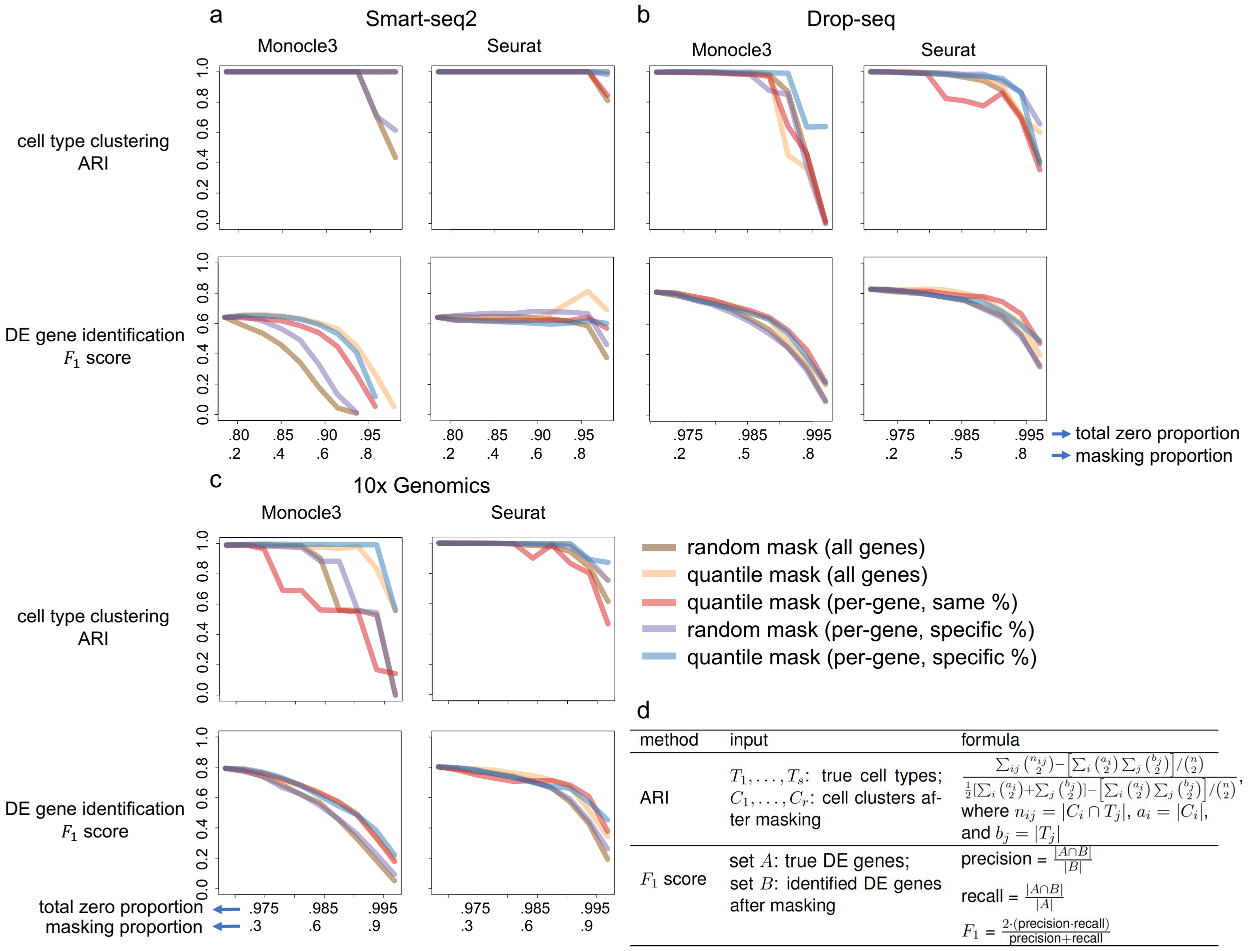
3.4 How non-biological zeros affect scRNA-seq data analysis

To evaluate the effects of non-biological zeros on scRNA-seq data analysis, such as cell clustering and DE gene identification, we need access to true cell types and true DE genes. Hence, we use scDesign2 [248], a probabilistic, flexible simulator we developed to generate realistic scRNA-seq count data from any protocol with gene correlations captured. First, we train scDesign2 on the three benchmark PBMC datasets (10x Genomics, Drop-seq, and

Smart-seq2) [193], which all contain the same five cell types (B cells, CD14+ monocytes, CD4+ T cells, cytotoxic T cells, and natural killer cells) and are used in Fig. 3.5. Second, we simulate the corresponding non-zero-inflated synthetic datasets, one for each protocol, in the form of gene-by-cell count matrices. In detail, after the first training step, every gene in each cell type has a fitted count distribution (Poisson, ZIP, NB, or ZINB) by scDesign2; in the second simulation step, we generate read or UMI counts for every gene in each cell type from the non-zero-inflation component (Poisson or NB). Note that we set the number of synthetic cells generated by scDesign2 equal to the number of real cells for each cell type. Hence, for each gene, this simulation procedure removes the statistical zero inflation, which we define in the last section, and provides the gene's expected expression level in each cell type (as the mean of its non-zero-inflation component).

Based on the three non-zero-inflated synthetic datasets (10x Genomics, Drop-seq, and Smart-seq2), we define the positive controls for two typical analyses: cell clustering and DE gene identification, which are ubiquitous in scRNA-seq data analysis pipelines. For cell clustering, the positive controls are provided by scDesign2 as the cell types from which it generates synthetic cells. For DE gene identification, the positive controls are provided by scDesign2 as the genes whose expected expression levels differ between cell types.

Using each of the five masking schemes (see Supplementary), we introduce a varying number of non-biological zeros, corresponding to masking proportions $p = 0.1, \dots, 0.9$, into the three synthetic datasets corresponding to 10x Genomics, Drop-seq, and Smart-seq2 protocols, creating three suites of zero-inflated datasets, one suite per protocol. Note that each suite contains one non-zero-inflated dataset and $45 = 9$ (# of masking proportions) $\times$ 5 (# of masking schemes) zero-inflated datasets. Then we apply Monocle3 (R package version `0.2.3.0`) [181] and Seurat (R package version `3.2.1`) [166], two popular multi-functional software packages, to the three suites of datasets. We use the two packages to perform cell clustering and DE gene identification, and we evaluate the analysis results based on our previously defined positive controls. Fig. 3.6a–c summarizes how the accuracy of the two analyses deteriorates as the masking proportion increases under each masking scheme and for each protocol.



method	input	formula
ARI	$T_1, \ldots, T_s$: true cell types; $C_1, \ldots, C_r$: cell clusters after masking	$\frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}$, where $n_{ij} = \lvert C_i \cap T_j \rvert$, $a_i = \lvert C_i \rvert$, and $b_j = \lvert T_j \rvert$

F_1 score	set A: true DE genes; set B: identified DE genes after masking	precision = $\frac{\lvert A \cap B \rvert}{\lvert B \rvert}$
		recall = $\frac{\lvert A \cap B \rvert}{\lvert A \rvert}$
		F_1 = $\frac{2 \cdot (\text{precision} \cdot \text{recall})}{\text{precision} + \text{recall}}$

Figure 3.6: Effects of non-biological zeros on cell clustering and DE gene identification. We introduce a varying number of non-biological zeros, which correspond to masking proportions 0.1–0.9, into the simulated **(a)** Smart-seq2, **(b)** Drop-seq, and **(c)** 10x Genomics datasets using five masking schemes. The horizontal axes show (top) the total zero proportion (including the zeros before masking and the non-biological zeros introduced by masking) and (bottom) the masking proportion (i.e., the proportion of non-zero counts masked by a masking scheme). After introducing non-biological zeros, we apply Monocle 3 and Seurat to each dataset to perform cell clustering and identify DE genes. For the two analyses, we evaluate their accuracy using the adjusted rand index (ARI) and F_1 score (given the false discovery rate 5%), respectively. **(d)** Technical definitions of the ARI and F_1 score.

The clustering results (top rows in Fig. 3.6a–c) show that the clustering accuracy (measured by the adjusted rand index; Fig. 3.6d) is robust to the introduction of non-biological zeros up to the masking proportion $p = 0.6$ (i.e., 60% non-zero counts are masked as zeros) for most masking schemes. Compared with Monocle3, Seurat is more robust to non-biological zeros under all the five masking schemes. Among all schemes, the two schemes that assume (1) dependence between masking and count values and (2) gene-specific masking proportions—`quantile mask (all genes)` and `quantile mask (per-gene, specific %)` (Fig. 3.8b)—have the least deteriorating effects on cell clustering. This result is reasonable as these two schemes tend to mask low counts to zeros so that the relative order of gene expression counts (from low to high) is better preserved than by the other three schemes. A recent article argues that zeros in scRNA-seq data carry biological meanings and should be embraced, and its argument is based on the assumption that most zeros correspond to low expression levels [213], an assumption aligned with these two masking schemes. Finally, among the three protocols, clustering on Smart-seq2 data is most robust to non-biological zeros, likely because Smart-seq2 data contain fewer zeros than the two UMI-based protocols' data do. It is worth noting that, between the two UMI-based protocols, clustering accuracy is better on 10x Genomics data than Drop-seq data.

The DE gene identification results (bottom rows in Fig. 3.6a–c) show that the F_1 scores (at 5% false discovery rate; Fig. 3.6d) are robust to non-biological zeros for Seurat, but not as much for Monocle3. The reason is that Seurat uses MAST [188], a method built upon a zero-inflated model, for DE gene identification, while Monocle3 uses non-zero-inflated models (including Poisson, quasi Poisson, and NB) that cannot account for additional non-biological zeros. Among the five masking schemes, the two random schemes that assume independence between masking and count values—`random mask (all genes)` and `random mask (per-gene, specific %)` (Fig. 3.8b)—have the most deteriorating effects on DE gene identification. This result is reasonable as these two schemes may mask high counts to zeros, so they disrupt every gene's count distribution more than the other three schemes do. Interestingly, although `quantile mask (per-gene, same %)` is unlikely a realistic generation mechanism of non-biological zeros as it masks the same proportion of non-zero counts for every gene, we observe that Seurat has robust F_1 scores as non-biological zeros are introduced by this scheme. This seemingly unexpected result reflects that zero-inflated models are robust for DE gene identification under quantile masking, even though the masking proportion may not be reasonable. Finally, regarding the three protocols, Seurat has better F_1 scores for Smart-seq2 data than Monocle3 does, a reasonable result given the observed zero-inflation in Smart-seq2 data (Fig. 3.5d). For the two UMI-based protocols, Monocle3 and Seurat have comparable performance in terms of F_1 scores. We have also observed that the DE analysis results for UMI data are better than for non-UMI-based data. One possible reason is the larger sample sizes (larger numbers of cells) in Drop-seq and 10x data that increase the power in statistical testing. To supplement the F_1 scores, we show the corresponding precision and recall rates in Supplementary Fig. 3.9. It is worth noting that although the false discovery rate is set to 5%, the precision rates of both Monocle3 and Seurat are far below the expected precision 95%, which is equal to one minus the false discovery rate. This phenomenon calls for better false discovery rate control in scRNA-seq DE analysis [249]. In addition, compared to Seurat, Monocle3 shows a greater fluctuation in both precision and recall as the masking proportion increases.

In summary, compared with DE gene identification, cell clustering is more robust to non-biological zeros. This result suggests that the sparsity in scRNA-seq data affects gene-level analyses more than cell-level analyses because the latter jointly uses all genes' expression levels. Overall, Seurat is more robust than Monocle3 is to non-biological zeros for both analyses. For cell clustering, Seurat has better accuracy regardless of protocols. For DE gene identification, Seurat is preferable for Smart-seq2 data, while Monocle3 has better accuracy for UMI-based data.

It is worth noting that many imputation methods evaluate their imputation accuracy based on only the `random mask (all genes)` scheme [85, 199, 250]. Our results indicate that non-biological zeros introduced by different masking schemes have different effects on cell clustering and DE gene identification, and quantile masking may be more realistic given previous reports that genes with lower expression values have more zeros than genes with higher expression [191, 196]. Hence, we urge that quantile masking schemes be considered in the future evaluation of computational methods that deal with non-biological zeros.

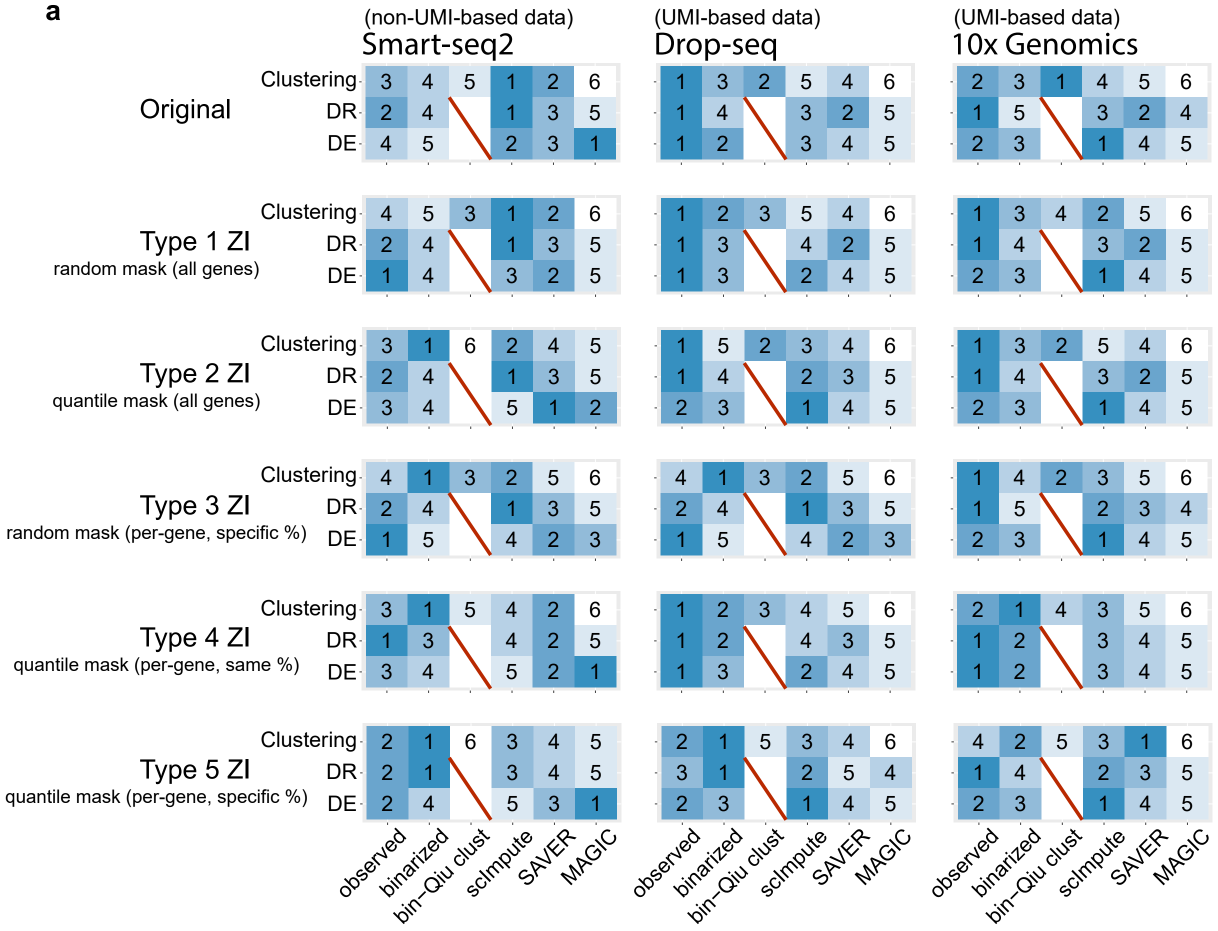
3.5 Input data: observed vs. imputed vs. binarized counts

Current scRNA-seq data analysis typically takes three types of input data: observed, imputed, and binarized counts. Researchers use imputed and binarized counts to deal with the vast proportion of zeros. Although log-transformed counts are often used as input data, this practice is under controversy [239, 251, 252] and is not the focus of our discussion. Here we summarize the advantages, disadvantages, and suitable users (bioinformatics tool developers vs. users) of each input data type.

Direct modeling of observed counts is the most common practice for bioinformatics tool developers [166, 181, 183–186, 188, 253]. An obvious advantage of direct modeling is that observed counts are not biased by any data pre-processing steps. Hence, observed counts are the preferred input data type for most tool developers. However, unlike tool developers, tool users need to apply existing bioinformatics tools to scRNA-seq data. If the observed counts do not work well with existing tools, for practical reasons, tool users may consider data pre-processing steps such as imputation and binarization so that existing tools can output reasonable analysis results.

Since the sparsity in scRNA-seq counts has posed a great hurdle for many existing tools, imputation has been proposed as a practical data pre-processing step, and many imputation methods have been developed [85, 173, 194, 195, 197–211]. Of course, imputation has the risk of biasing data, leading to false signals [245] or diminished biological variation [18, 195]. However, imputation has two practical advantages for single-cell biologists who are mostly tool users. First, many imputation methods have shown that their imputed counts, in which many zeros in the observed counts become non-zeros, agree better with biological knowledge and/or biologists' expectations. For example, the effectiveness of imputation has been supported by evidence that scRNA-seq data after imputation agree better with bulk RNA-seq data or single-cell RNA fluorescence in situ hybridization (FISH) data [85, 198, 254]. Second, imputation builds a bridge that connects sparse scRNA-seq data to many

a



b

Input data	Smart-seq2	Drop-seq	10x Genomics	Clustering	DR	DE
observed	3.0	**1.0**	**1.7**	**2.3**	**1.5**	2.8
binarized	4.3	3.0	3.7	3.5	4.3	3.8
bin-Qiu clust	NA	NA	NA	3.3	NA	NA
scImpute	**1.3**	3.7	2.7	2.8	2.0	**2.0**
SAVER	2.7	3.3	3.7	3.3	2.5	3.5
MAGIC	4.0	5.3	5.0	6.0	4.8	3.0

Figure 3.7: (Continued on the following page.)

Figure 3.7: Performance ranks of using observed, binarized, and imputed counts (from three experimental protocols ad under five masking schemes) in three downstream analyses. **(a)** We perform cell clustering (Clustering), cell dimension reduction (DR), and gene differential expression (DE) analysis on the observed, binarized, and imputed counts of Smart-seq2, Drop-seq, and 10x Genomics data. We consider three popular imputation methods: scImpute, SAVER, and MAGIC. In addition to the original data, we use five masking schemes (Type 1 ZI–Type 5 ZI) to introduce 50% non-biological zeros and evaluate the effects on the downstream analyses with different input data. The five masking schemes are `random mask (all genes)`, `quantile mask (all genes)`, `random mask (per-gene, specific %)`, `quantile mask (per-gene, same %)`, and `quantile mask (per-gene, specific %)`, corresponding to type 1 ZI–type 5 ZI, respectively. The six columns correspond to different input data types: observed counts, binarized counts, binarized counts analyzed by the Qiu's clustering algorithm (bin-Qiu clust), imputed counts by scImpute, imputed counts by SAVER, and imputed counts by MAGIC. For cell clustering, except bin-Qiu clust, clustering is conducted by the Louvain clustering algorithm (in Seurat); clustering performance is ranked by the ARI. For cell DR analysis, we apply UMAP (in Seurat) to perform DR and calculate the average Silhouette score (based on known cell types) for each input data type to evaluate the DR performance. For gene DE analysis, we apply the two-sample proportion test to binarized counts and MAST (in Seurat) to observed, binarized, and imputed data to perform DE analysis. To rank the DE performance by the F_1 score (at the 5% false discovery rate), since binarized counts have two DE methods, we compute the rank for the better-performing method in each comparison. In each row of each matrix, rank 1 indicates the best-performing input data type, while rank 6 indicates the worst. **(b)** On the original data, we compute the average ranks of the six input data types. Columns 4–6 show the average ranks for Smart-seq2 data, Drop-seq data, and 10x Genomics data across the three downstream analysis—cell clustering analysis, cell DR analysis, and gene DE analysis. Columns 7–9 show the weighted averages of the ranks for the three downstream analysis given the three protocols. Weights of 2, 1, 1 are used for Smart-seq2, Drop-seq, and 10x Genomics to ensure that the weights for non-UMI-based and UMI-based data are equal.

powerful tools designed for non-sparse data. For example, DESeq2 [53] and edgeR [184] are two popular DE gene identification methods for bulk RNA-seq data; however, they are not directly applicable to scRNA-seq data because their models do not account for data sparsity. Hence, if tool users cannot find a DE gene identification method that works well for their scRNA-seq data, they may consider reducing zeros by imputation methods to make DESeq2 or edgeR applicable [210, 211, 255], conditional on verified false discovery rate control [249, 256].

Moreover, a recent article provides a new perspective by proposing to use only binarized counts (with all non-zero counts truncated as ones) for cell clustering [213]. It argues that, by removing the magnitudes of non-zero counts, binarization alleviates the need for normalizing individual cells' sequencing depths. Further, its key message is that zeros are biologically meaningful because binarized counts can lead to reasonable cell clustering results. Other works also suggest that binarized counts can serve as useful data, in addition to observed counts, and be incorporated into scRNA-seq data modeling and analysis [176, 257]. Although binarized counts eliminate the expression differences between highly- and lowly-expressed genes, they highlight the co-expression patterns of genes, i.e., whether two genes are co-expressed in a cell, which have been used in marker gene selection [165] and gene network construction [258–260]. However, it remains unclear whether binarized counts can replace observed counts in scRNA-seq data analysis. Our intuition says that the answer is unlikely yes for all analyses because the magnitudes of non-zero counts reflect expression levels of genes in each cell. Qiu uses binarized counts to deal with cell clustering, a cell-level analysis [213]. For gene-level analyses such as DE gene identification, binarized counts are unlikely better than observed counts. For example, if a gene has similar percentages of zero counts in two cell types, but its non-zero counts are much larger in one cell type than the other, then this gene should be identified as DE using observed counts, but it would be missed as DE using binarized counts. In the previous section "How non-biological zeros affect scRNA-seq data analysis," we have compared the effects of non-biological zeros on clustering and DE gene analysis. For tool developers, it would be beneficial to consider using binarized counts in addition to observed counts for developing new analysis tools. For tool users, binarized counts can be used for exploratory data analysis because several efficient computational tools are applicable to binary counts only, e.g., scalable probabilistic principal component analysis [261].

We further evaluate the effects of the three input data types (observed, binarized, and imputed counts) on three popular downstream analyses: cell clustering, cell dimension reduction (two-dimensional visualization), and DE gene identification. To obtain the imputed counts, we use three popular imputation methods: scImpute, MAGIC, and SAVER, which demonstrate good performance in a recent benchmark study [6].

To benchmark cell clustering and dimension reduction results, we use the three real scRNA-seq PBMC datasets with labelled cell types [193]—one non-UMI-based dataset generated by Smart-seq2 and two UMI-based datasets generated by 10x Genomics and Drop-seq—which we have used in the previous section to evaluate the effects of non-biological zeros. To benchmark DE gene identification results, we generate synthetic datasets containing pre-defined true DE genes by scDesign2 [248] from two cell types (CD4+ T cells and cytotoxic T cells) in the three real datasets.

For cell clustering, we use two algorithms: Qiu's algorithm designed specifically for binarized counts [213] and the Louvain algorithm (implemented in Seurat). For all three input data types, we use the Louvain algorithm to cluster cells; for binarized counts only, we also use Qiu's algorithm. Based on the cell type labels provided in all three datasets, we calculate the ARI as a measure of clustering accuracy (Fig. 3.10; top row).

For cell dimension reduction, we perform UMAP (implemented in Seurat) on the observed, binarized and imputed counts (Figs. 3.12, 3.13, and 3.14 show the results of Smart-seq2, Drop-seq, and 10x Genomics, respectively). We use the average Silhouette score to evaluate how well the labeled cell types are separated in the two-dimensional UMAP space (Fig. 3.11; top row).

For DE gene analysis, we consider two DE methods. For all three inpute data types, we use MAST (implemented in Seurat) to perform DE gene identification; for binarized counts only, we also apply a two-sample proportion test to the binarized data. At a 5% false discovery rate, we use precision (Fig. 3.16), recall (Fig. 3.17) and F_1 score (Fig. 3.15) to evaluate the identification results.

Fig. 3.7a–b summarizes the relative performance of the three input data types for the three protocols (Smart-seq2, Drop-seq, and 10x Genomics) in the three downstream analyses. In terms of cell clustering, for non-UMI-based Smart-seq2 data, the Louvain algorithm has better performance on scImpute and SAVER's imputed counts than on the observed or binarized counts; for UMI-based Drop-seq and 10x Genomics data, the Louvain algorithm on the observed counts and Qiu's algorithm on the binarized counts have comparable performance and outperform the Louvain algorithm applied to other input data types, suggesting that imputation does not improve the clustering of UMI-based data. Notably, Qiu's algorithm only works well for binarized counts of UMI-based data, likely due to its special design. In terms of cell dimension reduction, scImpute's imputed counts work the best for non-UMI-based data; the observed counts have the best performance for UMI-based data; binarized counts and MAGIC's imputed counts have poor performance for both non-UMI-based and UMI-based data. In terms of DE gene analysis, for non-UMI-based data, all three imputation methods' imputed counts outperform the observed and binarized counts, a result consistent with our previous discussion on the existence of zero-inflation in non-UMI-based data; for UMI-based data, the observed counts and scImpute's imputed counts lead to the best result for Drop-seq and 10x Genomics data, respectively.

Moreover, we evaluate the three input data types in the three downstream analyses after applying the five masking schemes (see Supplementary) to introducing additional non-biological zeros. Under each masking scheme, we mask 50% of the original non-zero counts as zeros in each of the three original datasets (Smart-seq2, Drop-seq, and 10x Genomics). In terms of cell clustering analysis, for non-UMI-based data, scImpute's imputed counts demonstrate robust performance and stay as a top-performing input data type under the first three masking schemes, including the two random masking schemes; interestingly, by the Louvain algorithm, the binarized counts do not perform well for the original data but become a top-performing input data type under the last four masking schemes, including the three quantile masking schemes. These two results suggest that scImpute's imputation and binarization ameliorate the effects of additional non-biological zeros in complementary ways. For UMI-based data, the observed counts lead to the overall best clustering results under all masking schemes (ranked the 1st in 6 out of 10 protocol-masking scheme combinations), while Qiu's algorithm is not robust to the introduction non-biological zeros by masking schemes. In terms of cell dimension reduction, scImpute's imputed counts are the best input data type for non-UMI-based data (ranked the 1st under 3 out of 5 masking schemes), while the observed counts are the best for UMI-based data (ranked the 1st in 8 out of 10 protocol-masking scheme combinations). In terms of DE gene analysis, for non-UMI-based data, there is no universal winner: the observed counts work the best under random masking schemes, while SAVER and MAGIC's imputed counts work the best under quantile masking schemes. For UMI-based data, scImpute's imputed counts have the best performance (ranked the 1st in 6 out of 10 protocol-masking scheme combinations), followed by the observed counts (ranked the 1st in 4 out of 10 protocol-masking scheme combinations).

In summary, the observed counts work well for UMI-based data and are robust to the introduction of non-biological zeros. As expected, the binarized counts work well under the quantile masking schemes, which largely preserve the ranks of gene expression levels. Qiu's clustering algorithm works well for the binarized counts of UMI-based data but is not robust to the introduction of non-biological zeros. Imputation methods show concrete improvement for non-UMI-based data, but not so much for UMI-based data. Among the imputation methods, scImpute shows the best performance, while MAGIC does not perform well; a likely reason is that the data we use contain discrete cell types instead of continuous cell trajectories. Notably, the performance of imputation methods depends heavily on the masking scheme, demonstrating the importance of considering multiple masking schemes for the development and benchmarking of imputation methods.

Future directions

ScRNA-seq technologies have advanced the revelation of genome-wide gene expression profiles at the cell level. Accordingly, many computational algorithms and statistical models have been developed for analyzing scRNA-seq data. A well-known challenge in scRNA-seq data analysis is the prevalence of zeros, and how to best tackle zeros remains a controversial topic. Modeling and analysis may be performed on observed, imputed, or binarized scRNA-seq counts. However, the relative advantages and disadvantages of these three strategies remain ambiguous. In this article, we attempt to address this controversy by discussing multiple intertwined topics: the biological and non-biological sources of zeros, the relationship between zero prevalence and scRNA-seq technologies, the extent to which zero prevalence affects various analytical tasks, and the three strategies' relative advantages, disadvantages, and suitable users. We benchmark the performance of analytical tasks on observed, binarized and imputed data with or without non-biological zeros introduced.

The prevalence of biological and non-biological zeros is a mixed result of intrinsic biological nature and complex scRNA-seq experiments. In particular, the generation mechanism of non-biological zeros is protocol dependent. Hence, it is infeasible to distinguish non-biological zeros from biological zeros purely based on observed counts. As a result, existing imputation methods have a glass ceiling if they use only observed counts as input. To better distinguish non-biological zeros from biological zeros, researchers need to utilize spike-in RNA molecules, whose true counts are known (e.g., External RNA Control Consortium spike-ins [262]), to investigate the generation mechanism of non-biological zeros. Such investigation requires consortium efforts such as the work by the Sequencing Quality Control (SEQC-2) consortium [263]. With a better understanding of how the generation of non-biological zeros depends on mRNA sequence features such as GC contents, statistical and mechanistic models may be developed to better distinguish non-biological zeros from biological zeros and thus to improve imputation accuracy.

The prevalence of biological and non-biological zeros is only one of the many obstacles in using scRNA-seq data for scientific discoveries. As scientific discovery is a trial-and-error process, scRNA-seq data analysis is unavoidably multi-step. Hence, bioinformatics tool developers must consider the pre-processing steps applied to input data and the downstream analyses users may perform on output data. Taking the popular Seurat package as an example, many data pre-processing steps are used before DE gene identification. These steps include filtering low-quality genes and cells, data normalization, gene selection, cell dimension reduction, and cell clustering. Hence, if tool developers are not aware of these pre-processing steps, their bioinformatics tools may not fit into the state-of-the-art scRNA-seq data analysis pipelines. Ultimately, the transparency and reproducibility of scRNA-seq data analysis call for a community collaboration between tool developers and users. Towards this goal, every research article, regardless of being tool development or data analysis, should contain a detailed description of each step and the underlying justifications [264].

Code availability

The R code for reproducing the results in Figs. 3.2–3.6 is available at `https://doi.org/10.5281/zenodo.4393041`.

3.6 Supplementary materials

Design of five masking schemes

Given the three synthetic datasets without zero inflation, we use five masking schemes to introduce a varying number of non-biological zeros into each dataset. Since there is no consensus on the generation mechanism of non-biological zeros, we design the five masking schemes to reflect two fundamental questions: whether the occurrence of non-biological zeros (1) depends on the actual gene expression levels and/or (2) is gene-specific. As the five masking schemes cover the extreme answers to both questions (Fig. 3.8a), we expect that they together cover the unknown generation mechanism of non-biological zeros and would thus reveal the realistic effects of non-biological zeros on cell clustering and DE gene identification.

We provide a toy example to demonstrate the five masking schemes in Fig. 3.8b and summarize their technical details in Fig. 3.8c. In short, for a dataset with n non-zero counts, given a masking proportion p, all schemes would mask approximately np non-zero counts. However, the five schemes differ in masking which np non-zero counts, and they can be categorized in two ways corresponding to the two aforementioned questions.

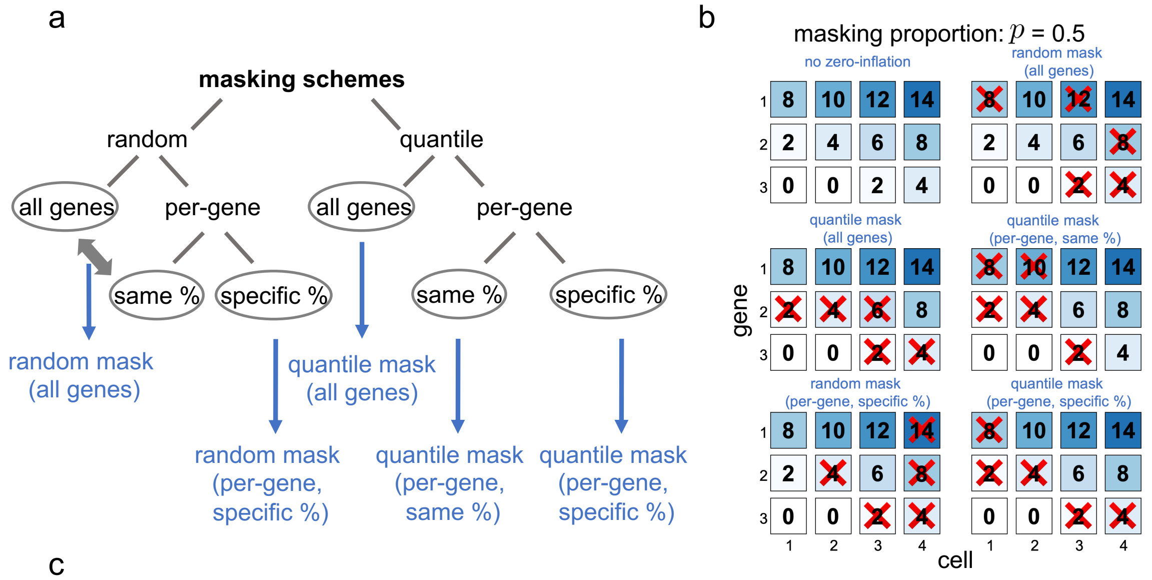
The first categorization is whether masking depends on the non-zero count values: random masking vs. quantile masking. While the two random masking schemes assume the independence between whether a non-zero count would be masked and the count value itself, the three quantile masking schemes assume a complete dependence by truncating non-zero values below a quantile (which corresponds to the masking proportion) to zero. Specifically, the two random masking schemes differ in the definition of independence: `random mask (all genes)` assumes the complete independence between masking and count values; `random mask (per-gene, specific %)` only assumes the conditional independence between masking and count values given each gene, and the masking proportion is gene-specific. Note that we define each gene's specific masking proportion as a function of the gene's non-zero counts based on an empirical formula in the literature [195, 196] (Fig. 3.8c); in short, the larger a gene's non-zero counts are, the smaller the gene's masking proportion is. Besides the two random masking schemes, the three quantile masking schemes differ in how they perform the truncation: `quantile mask (all genes)` truncates the lowest $100p\%$ non-zero counts of all genes; `quantile mask (per-gene, same %)` truncates the lowest $100p\%$ non-zero counts of each gene; `quantile mask (per-gene, specific %)` truncates the lowest non-zero counts of each gene based on the gene's specific masking proportion determined by the empirical formula.

The second categorization is regarding whether the masking proportion is gene-specific. Two schemes mask the same expected proportion $100p\%$ of non-zero counts for all genes: `random mask (all genes)` and `quantile mask (per-gene, same %)`. Three schemes use gene-specific masking proportions: `quantile mask (all genes)`, `random mask (per-gene, specific %)`, and `quantile mask (per-gene, specific %)`. Specifically, although `quantile mask (all genes)` does not use the empirical formula to determine gene-specific masking proportions as in `random mask (per-gene, specific %)` and `quantile mask (per-gene, specific %)`, it still truncates different proportions of non-zero counts for different genes. The reason is that its truncation threshold is set to the p-th quantile of all genes' non-zero counts, and different genes have different numbers of non-zero counts below that threshold. It is also worth noting that we do not include `random mask (per-gene, same %)` because it is theoretically equivalent to `random mask (all genes)`—both schemes are expected to randomly mask $100p\%$ of every gene's non-zero counts (Fig. 3.8a).

Note that random masking aims to reflect the random nature of sampling zeros. In a sequencing experiment, allocation of reads to genes is essentially random sampling from a multinomial distribution, whose probabilities are the proportions of genes in terms of cDNA copy numbers in the sequencing library. Due to the randomness of sampling, for two genes with moderately different non-zero proportions, it is possible that, in one experiment, the gene with the larger proportion receives a zero read count, i.e., a sampling zero, while the gene with the smaller proportion receives a non-zero read count. The magnitude of the randomness depends on the sequencing depth. For every gene, the standard deviation of its count over its expected count is equal to a large constant depending on its proportion (i.e.,

$\sqrt{(1-q_i)/q_i}$, where q_i is the proportion of gene i) multiplied by the inverse of the square root of the sequencing depth (i.e., $1/\sqrt{N}$, where N is the sequencing depth). Hence, the smaller the sequencing depth, the larger the standard deviation of every gene's count in relation to its expected count, the more likely that genes receive sampling zeros irrespective of their proportions. Moreover, the expected number of sampling zeros (i.e., $\sum_{i=1}^{I}(1-q_i)^N$, where I is the number of genes) decreases as the sequencing depth increases. In contrast, quantile masking aims to reflect gene proportions in the sequencing library and technical zeros, i.e., zero counts due to zero proportions without randomness. Quantile masking also reflects the fact that, despite of randomness, a gene with a small proportion is more likely to receive a sampling zero than a gene with a much larger proportion does.

Hence, for Drop-seq and 10x Genomics, since they sequence many cells, per-cell sequencing depth is low and thus randomness is influential, random masking better represents the occurrence of non-biological zeros, sampling zeros in particular, than quantile masking does. For Smart-seq2, since per-cell sequencing depth is high and thus randomness is negligible, quantile masking better resembles the generation mechanism of non-biological zeros, technical zeros in particular, than random masking does.


a
masking schemes
random
quantile
all genes
per-gene
all genes
per-gene
same %
specific %
same %
specific %
random mask (all genes)
quantile mask (all genes)
random mask (per-gene, specific %)
quantile mask (per-gene, same %)
quantile mask (per-gene, specific %)
b
masking proportion: p = 0.5
no zero-inflation
random mask (all genes)
quantile mask (all genes)
quantile mask (per-gene, same %)
random mask (per-gene, specific %)
quantile mask (per-gene, specific %)
gene
1
2
3
8 10 12 14
2 4 6 8
0 0 2 4
cell
1 2 3 4

c

masking scheme	parameter	method
random mask (all genes)	$p \in (0, 1)$	randomly mask $N \sim$ Binomial(n, p) of the n non-zero counts of all genes
quantile mask (all genes)	$p \in (0, 1)$	mask the smallest $\lfloor np \rfloor$ (floor) non-zero counts
qauntile mask (per-gene, same %)	$p \in (0, 1)$	mask the smallest $\lfloor n_i p \rfloor$ non-zero counts of gene i, which has n_i non-zero counts
random mask (per-gene, specific %)	$\lambda > 0, p \in (0, 1)$	randomly mask $N_i \sim$ Binomial(n_i, p_i) of the n_i non-zero counts of gene i, whose average (natrual) log-transformed non-zero count is μ_i, and $p_i = \exp(-\lambda \mu_i^2)$. λ is chosen to satisfy $\sum_{i=1}^{\text{\# of genes}} n_i p_i = np$, where n is the total number of non-zero counts of all genes
quantile mask (per-gene, specific %)	$\lambda > 0, p \in (0, 1)$	mask the smallest $\lfloor n_i p_i \rfloor$ non-zero counts of gene i. n_i and p_i are the same as in “random mask (per-gene, specific %)”

Figure 3.8: Five masking schemes for introducing non-biological zeros. (a) A tree diagram illustrating the design of the five masking schemes. From the top, the first division is about whether masking is independent of or completely dependent on count values, with the former as random masking and the latter as quantile masking. The second division is about whether masking is performed across all genes (with the same masking proportion) or within each gene (i.e., per-gene). If the latter, the third division is regarding whether the masking proportion is the same for all genes or specific to each gene depending on the gene's mean non-zero expression level. Note that random masking across all genes is equivalent to random masking per-gene with the same masking proportion (shown by the double arrow on the left). **(b)** A toy example illustration of the five masking schemes. The topleft plot shows the expression counts of three genes in four cells without zero-inflation; the other five plots show the expression counts after the five masking schemes are applied with the same masking proportion $p = 0.5$ (i.e., 50% of the non-zero gene expression counts are masked as zeros). **(c)** Technical explanation of each masking scheme. In the notations, p denotes the overall masking proportion across all genes, and p_i is the masking proportion of gene i.

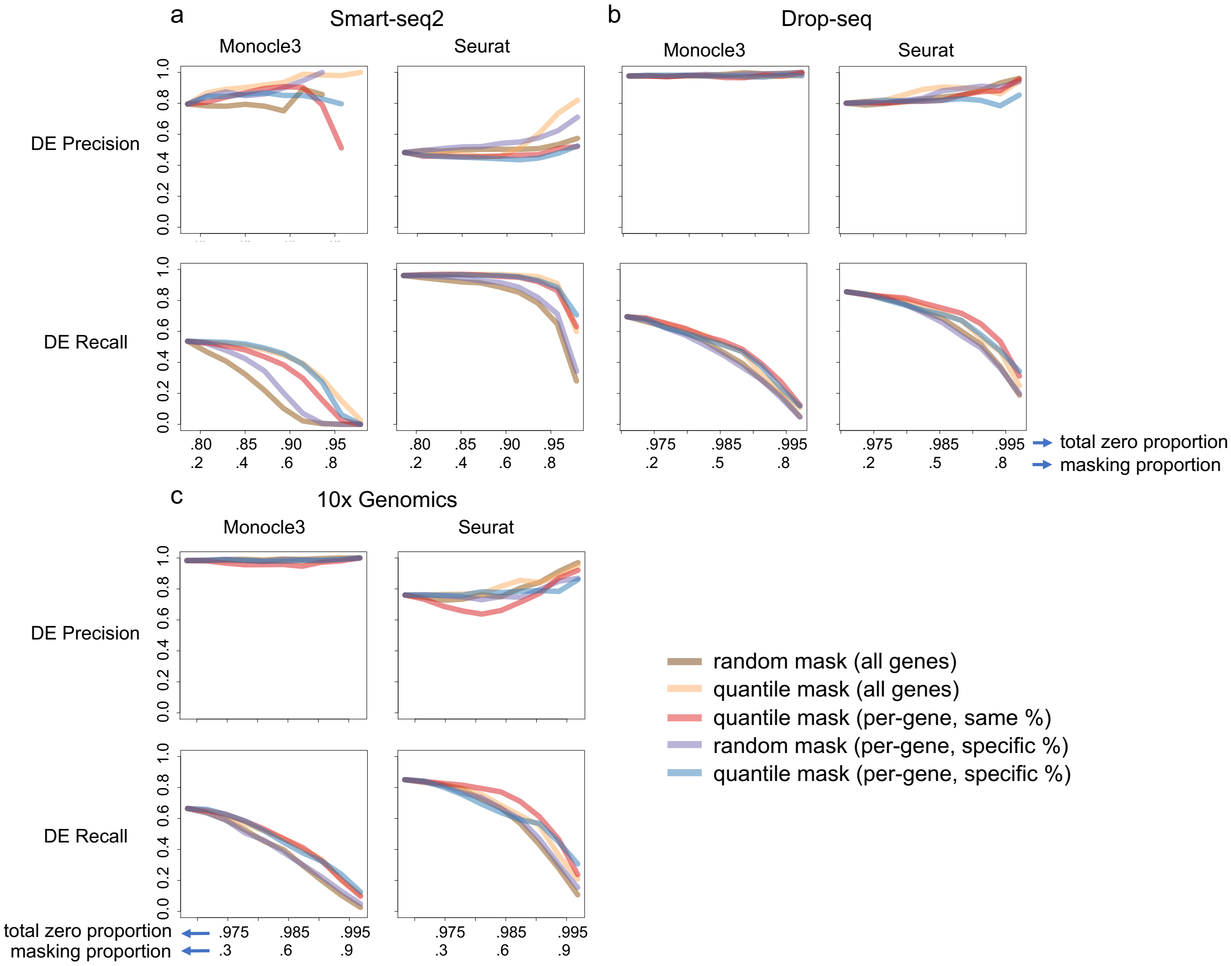


Figure 3.9: Effects of non-biological zeros on DE gene identification in terms of precision and recall. We introduce a varying number of non-biological zeros, which correspond to masking proportions 0.1–0.9, into the simulated **(a)** Smart-seq2, **(b)** Drop-seq, and **(c)** 10x Genomics datasets using five masking schemes. The horizontal axes show (top) the total zero proportion (including the zeros before masking and the non-biological zeros introduced by masking) and (bottom) the masking proportion (i.e., the proportion of non-zero counts masked by a masking schemes). After the introduction of non-biological zeros, we apply Monocle 3 and Seurat to each dataset to identify DE genes. We evaluate the accuracy using the precision and recall (given the false discovery rate 5%; defined in Fig. 3.6d), respectively.

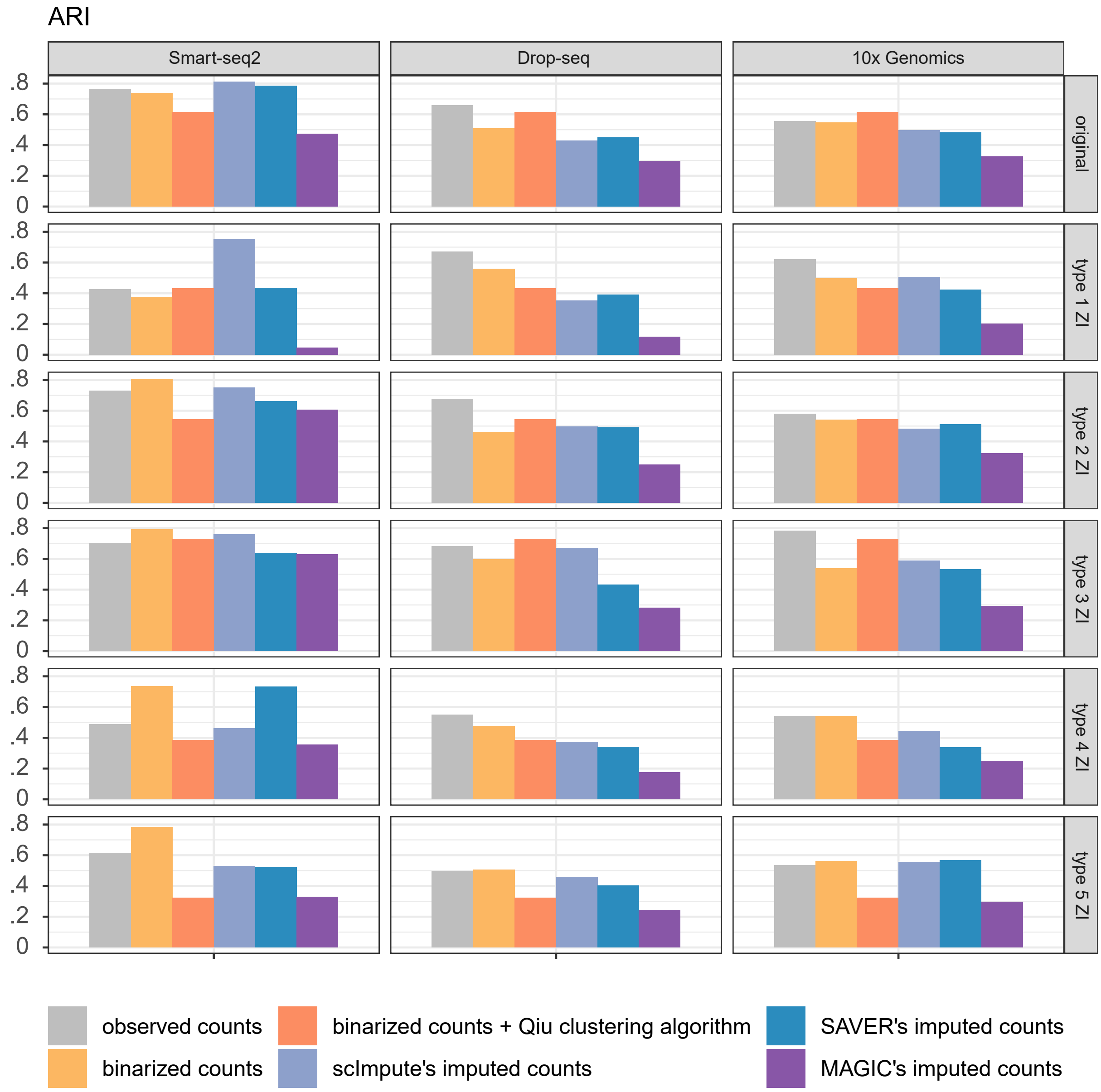


Figure 3.10: Evaluation of clustering analysis on observed, binarized and imputed data. We evaluate the clustering analysis on Smart-seq2, Drop-seq, and 10X Genomics data based on observed, binarized and imputed data. We perform this analysis before and after using the five masking schemes (type 1 ZI–type 5 ZI) to introduce non-biological zeros. Besides the bin-Qiu *et al.* which indicates the clustering algorithm developed specially for binarized data, we use Louvain clustering (in Seurat) on observed, binarized, and imputed data. We use ARI to evaluate the clustering results.

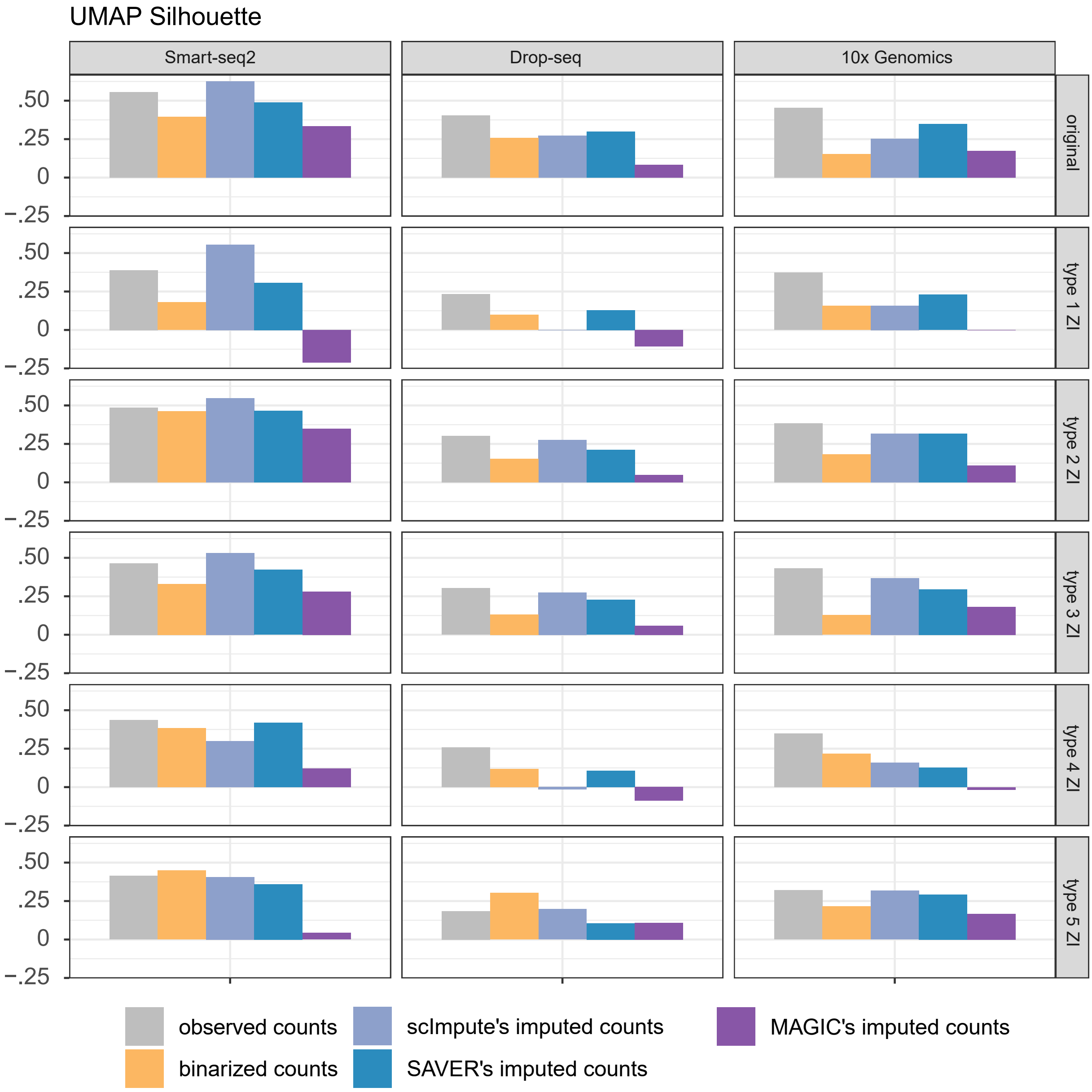


Figure 3.11: Evaluation of dimension reduction analysis on observed, binarized and imputed data. We evaluate the dimension reduction analysis on Smart-seq2, Drop-seq, and 10X Genomics data based on observed, binarized and imputed data. We perform this analysis before and after using the five masking schemes (type 1 ZI–type 5 ZI) to introduce non-biological zeros. We use UMAP (in Seurat) on observed, binarized, and imputed data to perform dimension reduction. We use Silhouette score to evaluate the dimension reduction results.

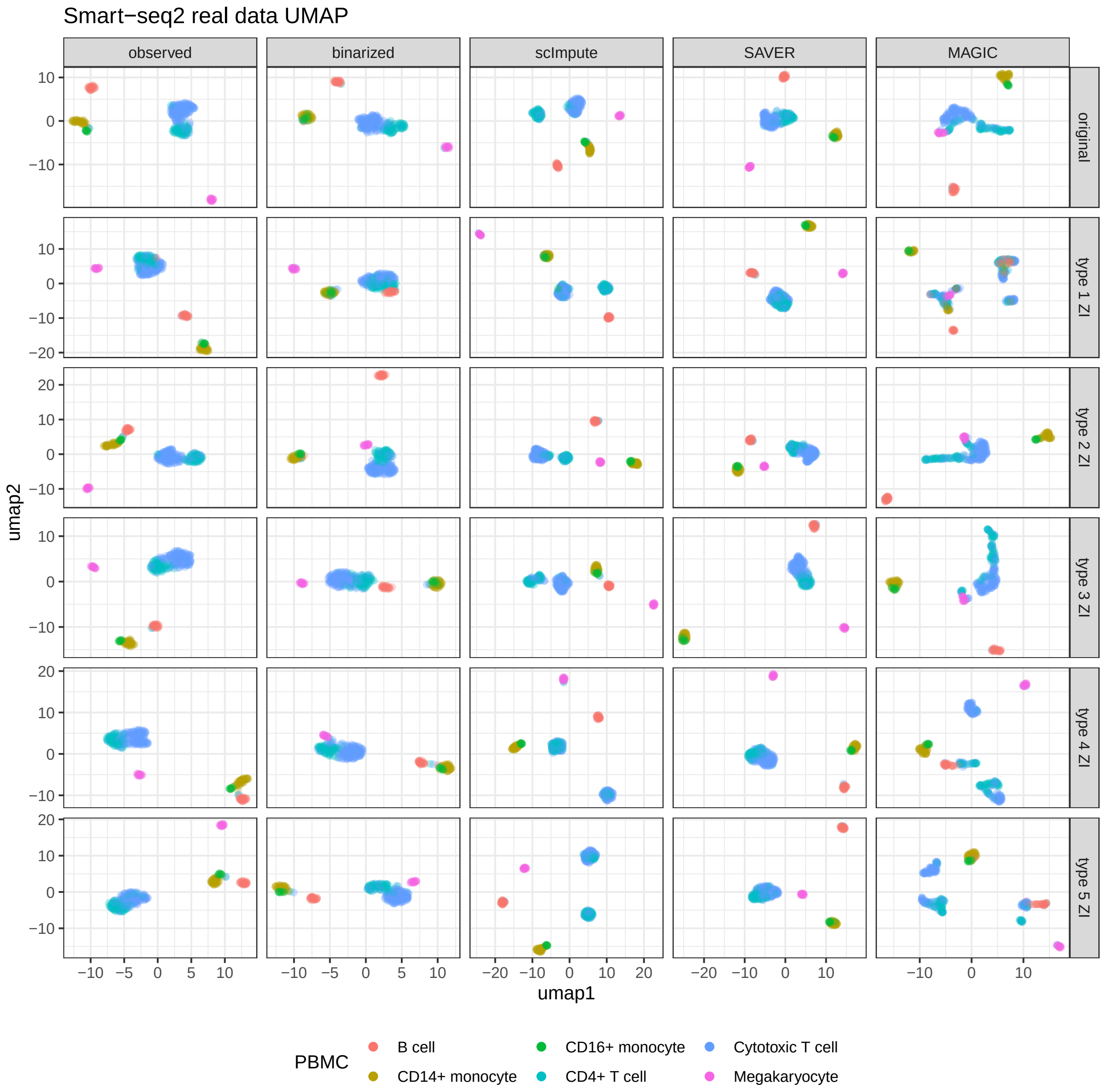


Figure 3.12: UMAP dimesion reduction visualization on observed, binarized and imputed Smart-seq2 data. We perform UMAP (in Seurat) on Smart-seq2's observed, binarized and imputed data. We perform this analysis before and after using the five masking schemes (type 1 ZI–type 5 ZI) to introduce non-biological zeros.

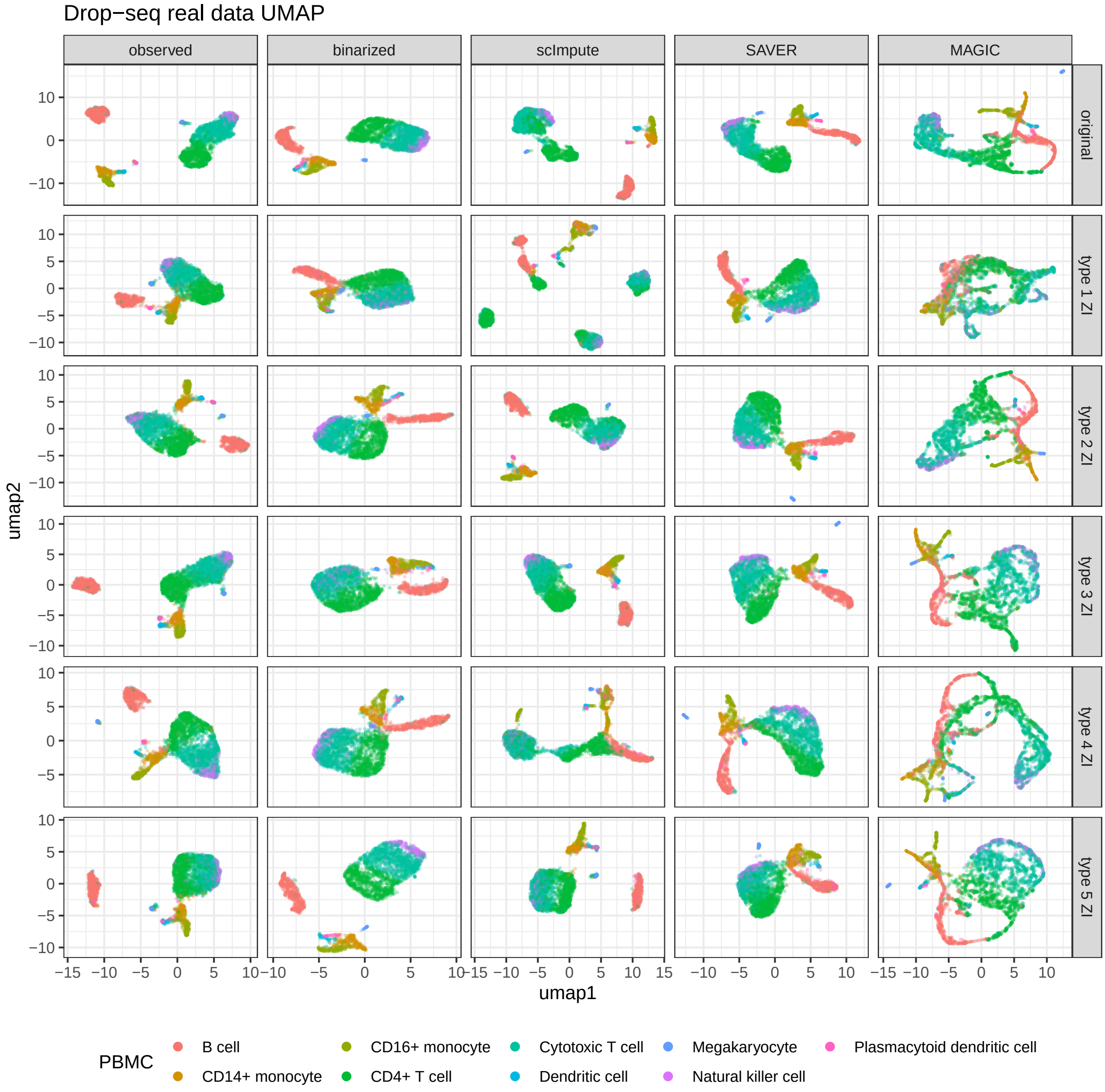


Figure 3.13: UMAP dimesion reduction visualization on observed, binarized and imputed Drop-seq data. We perform UMAP (in Seurat) on Drop-seq's observed, binarized and imputed data. We perform this analysis before and after using the five masking schemes (type 1 ZI–type 5 ZI) to introduce non-biological zeros.

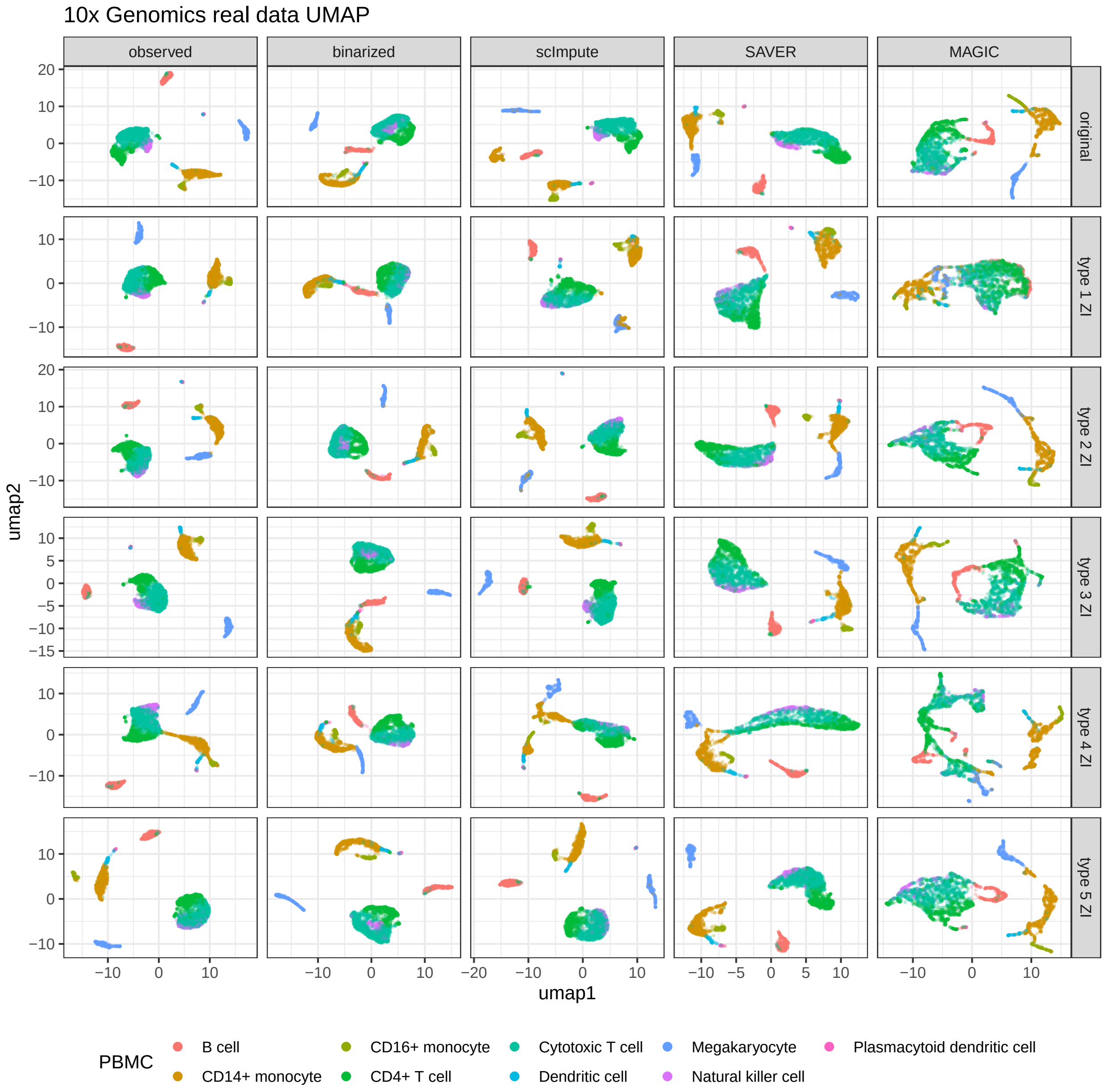


Figure 3.14: UMAP dimesion reduction visualization on observed, binarized and imputed 10X Genomics data. We perform UMAP (in Seurat) on 10X Genomics' observed, binarized and imputed data. We perform this analysis before and after using the five masking schemes (type 1 ZI–type 5 ZI) to introduce non-biological zeros.

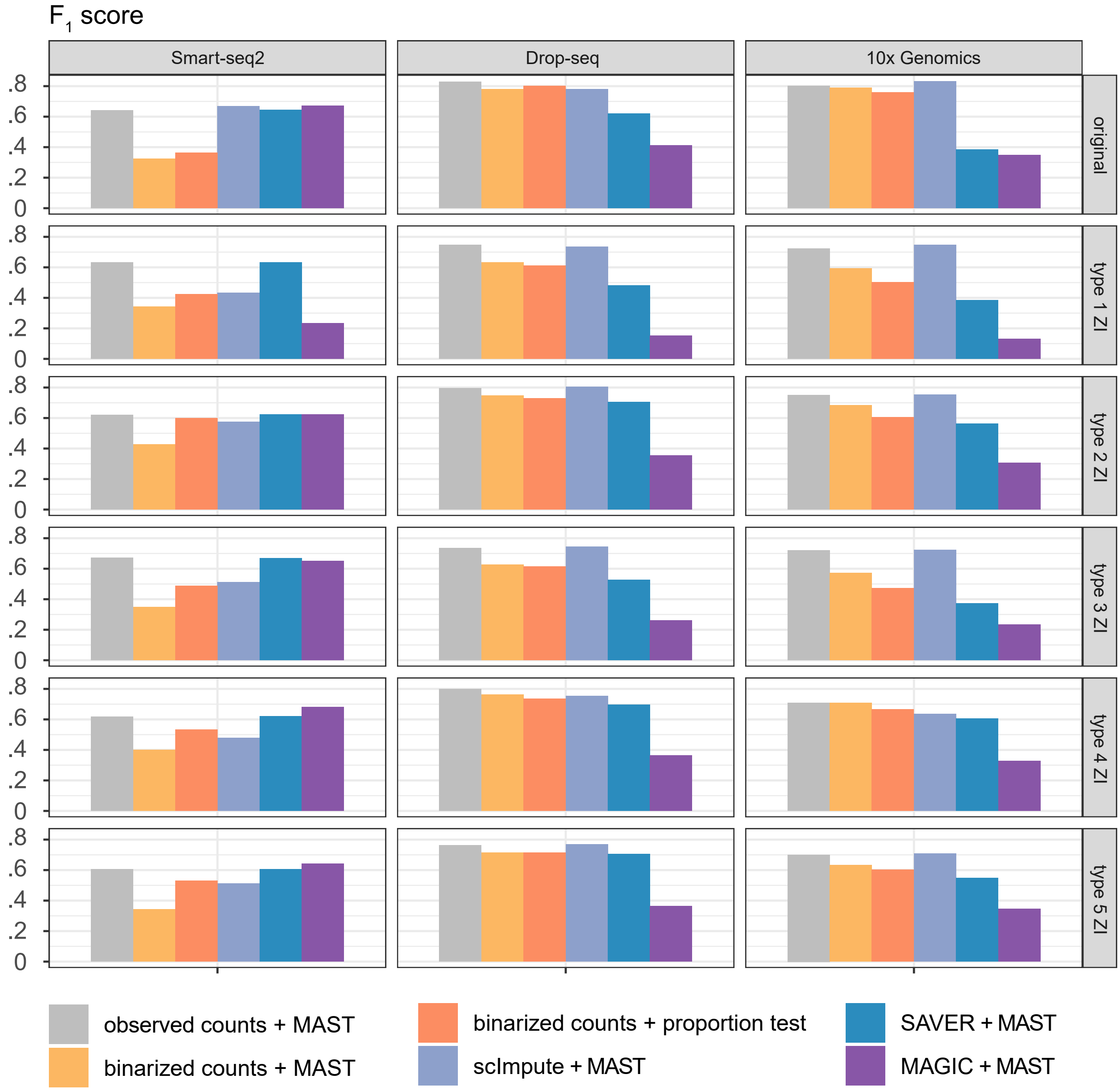


Figure 3.15: Evaluation of DE analysis on observed, binarized and imputed data. We evaluate the DE analysis on Smart-seq2, Drop-seq, and 10X Genomics data based on observed, binarized and imputed data. We perform this analysis before and after using the five masking schemes (type 1 ZI–type 5 ZI) to introduce non-biological zeros. We apply two-sample proportion test on binarzied data and MAST (in Seurat) on observed, binarized, and imputed data to perform DE analysis. We use F_1 score (given the false discovery rate 5%) to evaluate the DE results.

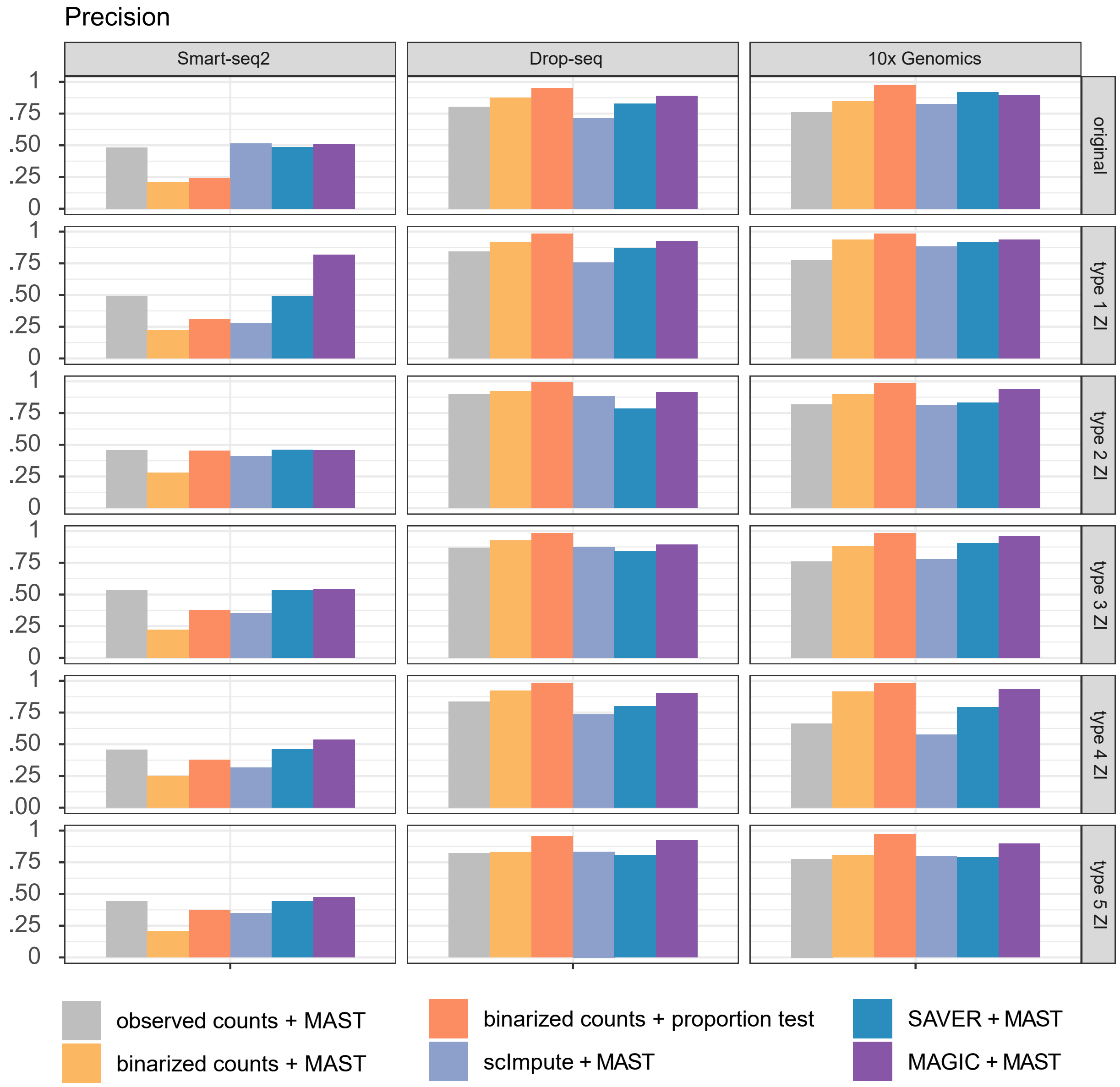


Figure 3.16: Evaluation of DE analysis on observed, binarized and imputed data. We evaluate the DE analysis on Smart-seq2, Drop-seq, and 10X Genomics data based on observed, binarized and imputed data. We perform this analysis before and after using the five masking schemes (type 1 ZI–type 5 ZI) to introduce non-biological zeros. We apply two-sample proportion test on binarzied data and MAST (in Seurat) on observed, binarized, and imputed data to perform DE analysis. We use precision (given the false discovery rate 5%) to evaluate the DE results.

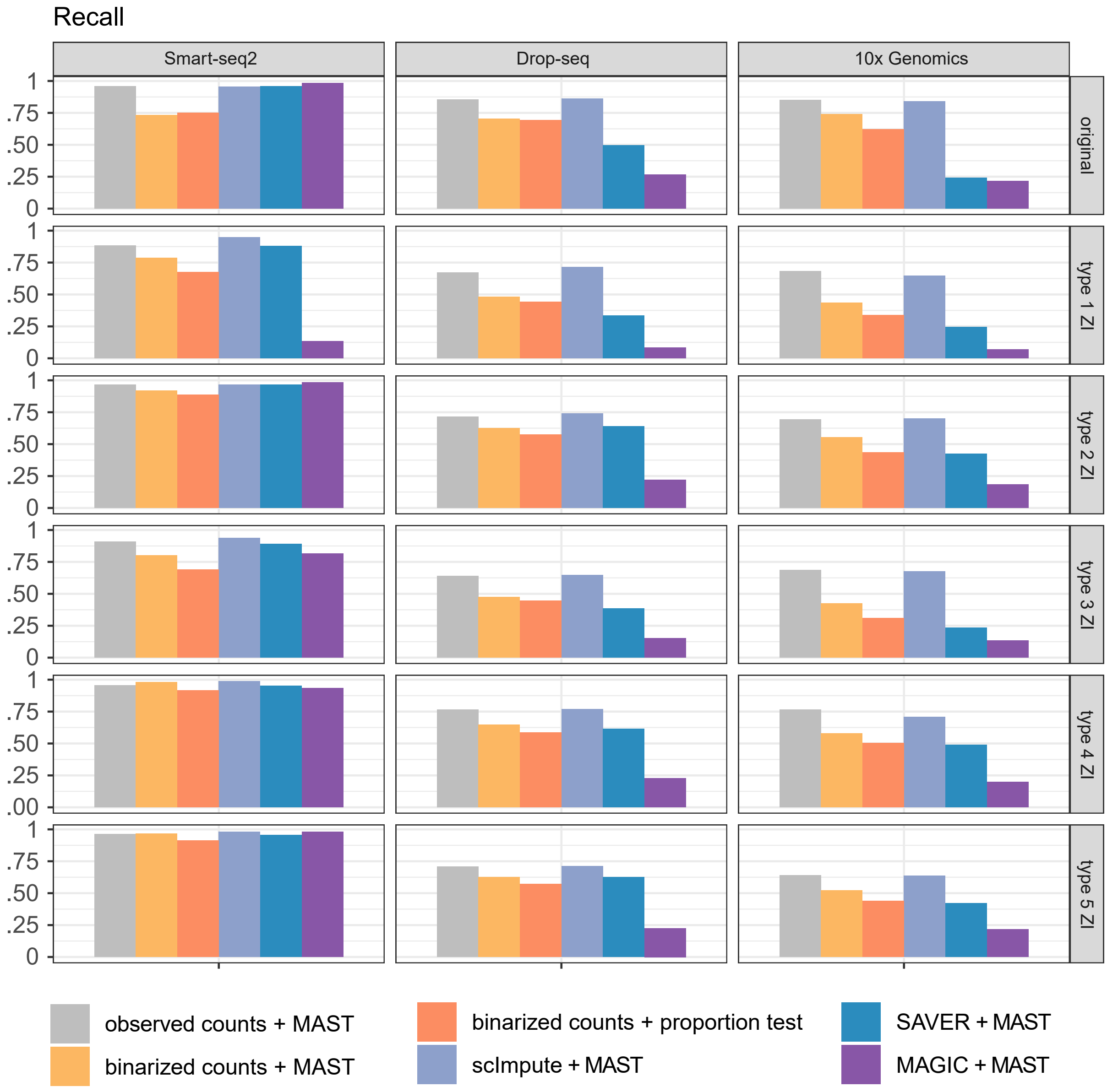


Figure 3.17: Evaluation of DE analysis on observed, binarized and imputed data. We evaluate the DE analysis on Smart-seq2, Drop-seq, and 10X Genomics data based on observed, binarized and imputed data. We perform this analysis before and after using the five masking schemes (type 1 ZI–type 5 ZI) to introduce non-biological zeros. We apply two-sample proportion test on binarzied data and MAST (in Seurat) on observed, binarized, and imputed data to perform DE analysis. We use recall (given the false discovery rate 5%) to evaluate the DE results.

CHAPTER 4

Discussion

The imputation problem in statistics, such as in survey data, is different from the imputation problem in machine learning, such as in the movie recommendation system data. In statistics, researchers care about the validity of inference on the imputed data. The state-of-the-art method, in this case, is multiple imputation. Multiple imputation provides a framework that can incorporate the uncertainty in the imputation step into the post-imputation inference. In contrast, in machine learning imputation problems, such as the film recommendation system in the Netflix data challenge, researchers or tool developers care more about the imputed values per se. They want to decide whether or not to recommend a specific film based on the imputed ratings. In this case, prediction is the key, while the uncertainty in the imputed value is not so essential as for statistical inference.

In the imputation problem for sequencing data, the goal is a mixture of inference and prediction. For inference, using scRNA-seq data analysis as an example, if researchers are interested in identifying DE genes using statistical tests, the uncertainty in the imputation step should be considered. In this case, the multiple imputation framework is still feasible for Bayesian imputation models such as SAVER and regression-based imputation models like mbImpute. In contrast, the low-rank factorization-based imputation methods, such as ALRA and softImpute, aim for the exact recovery of the original matrix, and no uncertainty is reported in their imputation results. Currently, how the uncertainty in most imputation methods affects inference tasks has not been well studied and remains a future research direction. For prediction, if researchers aim to perform unsupervised learning such as cell clustering or dimension reduction, their primary focus is on the imputed values, similar to the the movie recommendation problem, where the uncertainty of imputed values is less of a concern.

It remains an open question regarding how to evaluate imputation methods on real sequencing data. For scRNA-seq imputation methods, the most popular benchmarking approach is to artificially introduce additional zeros (missing values) into the matrix, perform imputation, and compare the imputed values of these additional zeros with the original values. However, this approach is in fact a simulation study as this artificial missing mechanism may differ significantly from the true missing mechanism. Another approach is to use external data—e.g., fluorescence in situ hybridization (FISH) data—as the ground truth and compares the imputed values with the corresponding values in the external data. However, this approach is often unachievable or restricted to a few values available in the external data; in most cases, it cannot be used to evaluate the accuracy of all imputed values. The third approach is to perform downstream analysis such as clustering on the imputed data. For example, in scRNA-seq cell clustering analysis, researchers can evaluate the clustering accuracy before and after imputation by associating cell clusters with known cell types. The problem with this approach is that it is specific to a downstream analysis and does not directly reflect the imputation accuracy. For example, an imputation method working well for cell clustering may not perform well for DE gene analysis. In order to fairly and thoroughly evaluate how different imputation methods perform, researchers need a better understanding of the missing mechanisms in sequencing data. For example, experiments can be designed to evaluate the reverse transcription efficiency of different mRNAs and see correspondingly how non-biological zeros would arise in the resulting sequencing data.

Another question to be answered is whether researchers should use direct modeling or imputation in sequencing data analysis. Many statistical models can directly quantify the over-dispersion and large proportion of zeros in sequencing data. In Chapter 3, we have introduced the zero-inflated models, which add a zero mass to the original distribution to explain additional zeros. In the microbiome field, researchers apply zero-inflated quasi-Poisson [265] and Poisson log-normal models [266] to microbiome sequencing data. However, this direct modeling approach treats all zeros equally and ignores the fact some zeros are biological and thus trustworthy, while other zeros are not, similar to our discussion about scRNA-seq data in Chapter 3. How the existence of non-biological zeros affects parameter estimation depends on the unknown generation mechanism of non-biological zeros. Furthermore, zero-inflated models often face convergence issues during model fitting. On the contrary, our mbImpute attempts to distinguish likely non-biological zeros and correct them so that non-zero-inflated models can be fitted efficiently and stably. However, questions remain about the potential biases of imputed values. Hence, future work is need to compare the direct modeling approach and the imputation approach under the same objectives relevant to sequencing data, an issue outlined in the last paragraph.

In our mbImpute work, we used a regression-based framework to perform imputation. Methodologically, how the regression-based framework differs from the low-rank factorization-based framework has not been well studied. The current matrix factorization literature aims to achieve exact recovery/imputation of the missing values based on the observed values. On the other hand, the regression-based framework assumes a probabilistic model. Both methods face the high-dimensional parameter estimation problem, but they perform penalization from different perspectives to achieve sparsity in the estimated parameters. For mbImpute's regression-based framework, we add L1 penalties to parameters. On the other hand, the low-rank factorization-based framework, as indicated by its name, puts a constraint on the rank of the matrix to be imputed to achieve feasible parameter estimation. In terms of model assumption, low-rank factorization-based imputation methods require the true matrix to be low rank, which does not necessarily hold for sequencing data due to the widespread biological diversity of individuals and cells. Empirically, we observed that softImpute, the state-of-the-art low-rank factorization method, decreases the variance of non-zero values in the imputed data, while mbImpute better preserves the variance after imputation. A new theoretical framework is yet to be developed to evaluate the preservation of variance by imputation.

Our mbImpute model employs a data pre-processing step: the log transformation of counts. For count data analysis, it is still a controversial problem regarding whether the log transformation is advantageous or disadvantageous over direct modeling of counts. Booeshaghi and Patcher (2021) pointed out two reasons for applying the log transformation to scRNA-seq data, including variance stabilization and converting the multiplicative effect to additive effect [267]. As discussed in Chapter 3, the PCR amplification in sequencing leads to exponential amplification of sequencing materials. Applying log-transformation to the sequencing data with PCR amplification will bring the values in the data to a similar scale. On the other hand, there are voices against using the log transformation on count data [268]. If researchers are dealing with UMI data that have corrected for the amplification bias, the count distributions fit well to the observed data, and no log-transformation is needed.

Bibliography

[1] Lin Liu et al. “Comparison of next-generation sequencing systems”. In: *Journal of Biomedicine and Biotechnology* 2012 (2012).

[2] Erwin L Van Dijk et al. “Ten years of next-generation sequencing technology”. In: *Trends in genetics* 30.9 (2014), pp. 418–426.

[3] Valentine Svensson et al. “Power analysis of single-cell RNA-sequencing experiments”. In: *Nature methods* 14.4 (2017), pp. 381–387.

[4] Wei Vivian Li and Jingyi Jessica Li. “An accurate and robust imputation method scImpute for single-cell RNA-seq data”. In: *Nature communications* 9.1 (2018), pp. 1–9.

[5] Zhang Xinyan, Mallick Himel, and Yi Nengjun. “Zero-inflated negative binomial regression for differential abundance testing in microbiome studies”. In: *Journal of Bioinformatics and Genomics* 2 (2) (2016). ISSN: 2530-1381. DOI: 10.18454/jbg.2016.2.2.1. URL: http://journal-biogen.org/article/view/12.

[6] Wenpin Hou et al. “A Systematic Evaluation of Single-cell RNA-sequencing Imputation Methods”. In: *bioRxiv* (2020).

[7] Donald B Rubin. “Inference and missing data”. In: *Biometrika* 63.3 (1976), pp. 581–592.

[8] Roderick JA Little and Donald B Rubin. “The analysis of social science data with missing values”. In: *Sociological Methods & Research* 18.2-3 (1989), pp. 292–326.

[9] Donald B Rubin. “Multiple imputation after 18+ years”. In: *Journal of the American statistical Association* 91.434 (1996), pp. 473–489.

[10] Rebecca R Andridge and Roderick JA Little. “A review of hot deck imputation for survey non-response”. In: *International statistical review* 78.1 (2010), pp. 40–64.

[11] José M Jerez et al. “Missing data imputation using statistical and machine learning methods in a real breast cancer problem”. In: *Artificial intelligence in medicine* 50.2 (2010), pp. 105–115.

[12] Daniel J Stekhoven and Peter Bühlmann. “MissForest—non-parametric missing value imputation for mixed-type data”. In: *Bioinformatics* 28.1 (2012), pp. 112–118.

[13] Cyril Garcia and Luca Rona. “The Netflix Challenge”. In: ().

[14] Xiangnan He et al. “Neural collaborative filtering”. In: *Proceedings of the 26th international conference on world wide web*. 2017, pp. 173–182.

[15] Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. “Spectral regularization algorithms for learning large incomplete matrices”. In: *The Journal of Machine Learning Research* 11 (2010), pp. 2287–2322.

[16] Mo Huang et al. “SAVER: gene expression recovery for single-cell RNA sequencing”. In: *Nature methods* 15.7 (2018), pp. 539–542.

[17] David Van Dijk et al. “Recovering gene interactions from single-cell data using data diffusion”. In: *Cell* 174.3 (2018), pp. 716–729.

[18] Lihua Zhang and Shihua Zhang. “Comparison of computational methods for imputing single-cell RNA-sequencing data”. In: *IEEE/ACM transactions on computational biology and bioinformatics* (2018).

[19] Ruochen Jiang, Wei Vivian Li, and Jingyi Jessica Li. “mbImpute: an accurate and robust imputation method for microbiome data”. In: *Genome biology* 22.1 (2021), pp. 1–27.

[20] Ruochen Jiang et al. “Zeros in scRNA-seq data: good or bad? How to embrace or tackle zeros in scRNA-seq data analysis?” In: *bioRxiv* (2020).

[21] Katherine R Amato. “An introduction to microbiome analysis for human biology applications”. In: *American Journal of Human Biology* 29.1 (2017), e22931.

[22] Peter J Turnbaugh et al. “An obesity-associated gut microbiome with increased capacity for energy harvest”. In: *nature* 444.7122 (2006), p. 1027.

[23] Buck S Samuel and Jeffrey I Gordon. “A humanized gnotobiotic mouse model of host–archaeal–bacterial mutualism”. In: *Proceedings of the National Academy of Sciences* 103.26 (2006), pp. 10011–10016.

[24] Jakob Stokholm et al. “Maturation of the gut microbiome and risk of asthma in childhood”. In: *Nature communications* 9.1 (2018), pp. 1–10.

[25] Alexa A Pragman et al. “The lung microbiome in moderate and severe chronic obstructive pulmonary disease”. In: *PloS one* 7.10 (2012), e47305.

[26] Elaine Holmes et al. “Understanding the role of gut microbiome–host metabolic signal disruption in health and disease”. In: *Trends in microbiology* 19.7 (2011), pp. 349–359.

[27] John Besser et al. “Next-generation sequencing technologies and their application to the study and control of bacterial infections”. In: *Clinical microbiology and infection* 24.4 (2018), pp. 335–341.

[28] M Luz Calle. “Statistical analysis of metagenomics data”. In: *Genomics & informatics* 17.1 (2019).

[29] Juan Jovel et al. “Characterization of the gut microbiome using 16S or shotgun metagenomics”. In: *Frontiers in microbiology* 7 (2016), p. 459.

[30] Benjamin J Callahan et al. “DADA2: high-resolution sample inference from Illumina amplicon data”. In: *Nature methods* 13.7 (2016), pp. 581–583.

[31] Benjamin J Callahan, Paul J McMurdie, and Susan P Holmes. “Exact sequence variants should replace operational taxonomic units in marker-gene data analysis”. In: *The ISME journal* 11.12 (2017), pp. 2639–2643.

[32] Hongzhe Li. “Microbiome, metagenomics, and high-dimensional compositional data analysis”. In: *Annual Review of Statistics and Its Application* 2 (2015), pp. 73–94.

[33] Georg Zeller et al. “Potential of fecal microbiota for early-stage detection of colorectal cancer”. In: *Molecular systems biology* 10.11 (2014).

[34] Qiang Feng et al. "Gut microbiome development along the colorectal adenoma–carcinoma sequence". In: *Nature communications* 6 (2015), p. 6528.

[35] Jun Yu et al. "Metagenomic analysis of faecal microbiome as a tool towards targeted non-invasive biomarkers for colorectal cancer". In: *Gut* 66.1 (2017), pp. 70–78.

[36] Emily Vogtmann et al. "Colorectal cancer and the human gut microbiome: reproducibility with whole-genome shotgun sequencing". In: *PloS one* 11.5 (2016).

[37] Fredrik H Karlsson et al. "Gut metagenome in European women with normal, impaired and diabetic glucose control". In: *Nature* 498.7452 (2013), pp. 99–103.

[38] Junjie Qin et al. "A metagenome-wide association study of gut microbiota in type 2 diabetes". In: *Nature* 490.7418 (2012), pp. 55–60.

[39] Matteo Calgaro et al. "Assessment of single cell RNA-seq statistical methods on microbiome data". In: *BioRxiv* (2020).

[40] Barak Brill, Amnon Amir, and Ruth Heller. "Testing for differential abundance in compositional counts data, with application to microbiome studies". In: *arXiv preprint arXiv:1904.08937* (2019).

[41] Justin D Silverman et al. "Naught all zeros in sequence count data are the same". In: *BioRxiv* (2020), p. 477794.

[42] Joana Pereira-Marques et al. "Impact of host DNA and sequencing depth on the taxonomic resolution of whole metagenome sequencing for microbiome analysis". In: *Frontiers in microbiology* 10 (2019), p. 1277.

[43] Microbiome Human. "Project Consortium 2012". In: *Structure, function and diversity of the healthy human microbiome. Nature* 486 (), pp. 207–214.

[44] Jason Lloyd-Price et al. "Strains, functions and dynamics in the expanded Human Microbiome Project". In: *Nature* 550.7674 (2017), pp. 61–66.

[45] Fan Xia et al. "A logistic normal multinomial regression model for microbiome compositional data analysis". In: *Biometrics* 69.4 (2013), pp. 1053–1063.

[46] Siddhartha Mandal et al. “Analysis of composition of microbiomes: a novel method for studying microbial composition”. In: *Microbial ecology in health and disease* 26.1 (2015), p. 27663.

[47] Matthew CB Tsilimigras and Anthony A Fodor. “Compositional data analysis of the microbiome: fundamentals, tools, and challenges”. In: *Annals of epidemiology* 26.5 (2016), pp. 330–335.

[48] Sophie Weiss et al. “Normalization and microbial differential abundance strategies depend upon data characteristics”. In: *Microbiome* 5.1 (2017), p. 27.

[49] Abhishek Kaul et al. “Analysis of microbiome data in the presence of excess zeros”. In: *Frontiers in microbiology* 8 (2017), p. 2114.

[50] Lizhen Xu et al. “Assessment and selection of competing models for zero-inflated microbiome data”. In: *PloS one* 10.7 (2015).

[51] Jun Chen et al. “An omnibus test for differential distribution analysis of microbiome sequencing data”. In: *Bioinformatics* 34.4 (2018), pp. 643–651.

[52] Paul J McMurdie and Susan Holmes. “phyloseq: an R package for reproducible interactive analysis and graphics of microbiome census data”. In: *PloS one* 8.4 (2013), e61217.

[53] Michael I Love, Wolfgang Huber, and Simon Anders. “Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2”. In: *Genome biology* 15.12 (2014), p. 550.

[54] Joseph N Paulson et al. “Differential abundance analysis for microbial marker-gene surveys”. In: *Nature methods* 10.12 (2013), pp. 1200–1202.

[55] Xiaoling Peng, Gang Li, and Zhenqiu Liu. “Zero-inflated beta regression for differential abundance analysis with metagenomics data”. In: *Journal of Computational Biology* 23.2 (2016), pp. 102–110.

[56] Timothy W Randolph et al. “Kernel-penalized regression for analysis of microbiome data”. In: *The annals of applied statistics* 12.1 (2018), p. 540.

[57] Zhigang Li et al. “Conditional regression based on a multivariate zero-inflated logistic-normal model for microbiome relative abundance data”. In: *Statistics in biosciences* 10.3 (2018), pp. 587–608.

[58] Stijn Hawinkel et al. “A broken promise: microbiome differential abundance methods do not control the false discovery rate”. In: *Briefings in bioinformatics* 20.1 (2019), pp. 210–221.

[59] M Claire Horner-Devine et al. “A comparison of taxon co-occurrence patterns for macro-and microorganisms”. In: *Ecology* 88.6 (2007), pp. 1345–1353.

[60] Albert Barberán et al. “Using network analysis to explore co-occurrence patterns in soil microbial communities”. In: *The ISME journal* 6.2 (2012), pp. 343–351.

[61] Jarishma K Gokul et al. “Taxon interactions control the distributions of cryoconite bacteria colonizing a High Arctic ice cap”. In: *Molecular ecology* 25.15 (2016), pp. 3752–3767.

[62] Ilma Tapio et al. “Taxon abundance, diversity, co-occurrence and network analysis of the ruminal microbiota in response to dietary changes in dairy cows”. In: *PloS one* 12.7 (2017).

[63] James Bennett, Stan Lanning, et al. “The netflix prize”. In: *Proceedings of KDD cup and workshop*. Vol. 2007. Citeseer. 2007, p. 35.

[64] Sarat C Dass and Vijayan N Nair. “Edge detection, spatial smoothing, and image reconstruction with partially observed multivariate data”. In: *Journal of the American Statistical Association* 98.461 (2003), pp. 77–89.

[65] Friedrich Faubel, John McDonough, and Dietrich Klakow. “Bounded conditional mean imputation with Gaussian mixture models: A reconstruction approach to partly occluded features”. In: *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2009, pp. 3869–3872.

[66] Valeria Rulloni, Oscar Bustos, and Ana Georgina Flesia. "Large gap imputation in remote sensed imagery of the environment". In: *Computational Statistics & Data Analysis* 56.8 (2012), pp. 2388–2403.

[67] Jason Ernst and Manolis Kellis. "Large-scale imputation of epigenomic datasets for systematic annotation of diverse human tissues". In: *Nature biotechnology* 33.4 (2015), p. 364.

[68] Jonathan Marchini and Bryan Howie. "Genotype imputation for genome-wide association studies". In: *Nature Reviews Genetics* 11.7 (2010), pp. 499–511.

[69] George C Linderman, Jun Zhao, and Yuval Kluger. "Zero-preserving imputation of scRNA-seq data using low-rank approximation". In: *bioRxiv* (2018), p. 397588.

[70] Gökcen Eraslan et al. "Single-cell RNA-seq denoising using a deep count autoencoder". In: *Nature communications* 10.1 (2019), pp. 1–14.

[71] Cameron Martino et al. "A novel sparse compositional technique reveals microbial perturbations". In: *MSystems* 4.1 (2019).

[72] Yun Cai, Hong Gu, and Toby Kenney. "Learning microbial community structures with supervised and unsupervised non-negative matrix factorization". In: *Microbiome* 5.1 (2017), p. 110.

[73] László Zsolt Garamszegi. *Modern phylogenetic comparative methods and their application in evolutionary biology: concepts and practice.* Springer, 2014.

[74] Liam J Revell. "phytools: an R package for phylogenetic comparative biology (and other things)". In: *Methods in ecology and evolution* 3.2 (2012), pp. 217–223.

[75] Steven W Kembel et al. "Picante: R tools for integrating phylogenies and ecology". In: *Bioinformatics* 26.11 (2010), pp. 1463–1464.

[76] David Orme et al. "The caper package: comparative analysis of phylogenetics and evolution in R". In: *R package version* 5.2 (2013), pp. 1–36.

[77] Gregory B Gloor and Gregor Reid. "Compositional analysis: a valid approach to analyze microbiome high-throughput sequencing data". In: *Canadian journal of microbiology* 62.8 (2016), pp. 692–703.

[78] Jun Chen et al. "Structure-constrained sparse canonical correlation analysis with an application to microbiome data analysis". In: *Biostatistics* 14.2 (2013), pp. 244–258.

[79] Tao Wang and Hongyu Zhao. "Constructing predictive microbial signatures at multiple taxonomic levels". In: *Journal of the American Statistical Association* 112.519 (2017), pp. 1022–1031.

[80] Jian Xiao, Hongyuan Cao, and Jun Chen. "False discovery rate control incorporating phylogenetic tree increases detection power in microbiome-wide multiple testing". In: *Bioinformatics* 33.18 (2017), pp. 2873–2881.

[81] Alex D Washburne et al. "Methods for phylogenetic analysis of microbiome data". In: *Nature microbiology* 3.6 (2018), pp. 652–661.

[82] T Michael Anderson, Marc-André Lachance, and William T Starmer. "The relationship of phylogeny to community structure: the cactus yeast community". In: *The American Naturalist* 164.6 (2004), pp. 709–721.

[83] Campbell O Webb et al. "Phylogenies and community ecology". In: *Annual review of ecology and systematics* 33.1 (2002), pp. 475–505.

[84] Evan Weiher and Paul A Keddy. "Assembly rules, null models, and trait dispersion: new questions from old patterns". In: *Oikos* (1995), pp. 159–164.

[85] Cédric Arisdakessian et al. "DeepImpute: an accurate, fast, and scalable deep neural network method to impute single-cell RNA-seq data". In: *Genome biology* 20.1 (2019), pp. 1–14.

[86] T Hastie and R Mazumder. "softimpute: Matrix completion via iterative soft-thresholded svd". In: *R package version* 1 (2015), p1.

[87] Emma Allen-Vercoe and Christian Jobin. "Fusobacterium and Enterobacteriaceae: important players for CRC?" In: *Immunology letters* 162.2 (2014), pp. 54–61.

[88] Paresh Dandona, Ahmad Aljada, and Arindam Bandyopadhyay. “Inflammation: the link between insulin resistance, obesity and diabetes”. In: *Trends in immunology* 25.1 (2004), pp. 4–7.

[89] Santosh Dulal and Temitope O Keku. “Gut microbiome and colorectal adenomas”. In: *Cancer journal (Sudbury, Mass.)* 20.3 (2014), p. 225.

[90] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. “glmnet: Lasso and elastic-net regularized generalized linear models”. In: *R package version* 1.4 (2009).

[91] Wuming Gong et al. “DrImpute: imputing dropout events in single cell RNA sequencing data”. In: *BMC bioinformatics* 19.1 (2018), p. 220.

[92] Simon Jackman. “pscl: Classes and methods for R. Developed in the Political Science Computational Laboratory, Stanford University. Department of Political Science, Stanford University, Stanford, CA. R package version 1.03. 5”. In: *http://www. pscl. stanford. edu/* (2010).

[93] Markus Kalisch and Peter Bühlman. “Estimating high-dimensional directed acyclic graphs with the PC-algorithm.” In: *Journal of Machine Learning Research* 8.3 (2007).

[94] Jesse H Krijthe. “Rtsne: T-distributed stochastic neighbor embedding using Barnes-Hut implementation”. In: *R package version 0.13, URL https://github. com/jkrijthe/Rtsne* (2015).

[95] Nadja Larsen et al. “Gut microbiota in human adults with type 2 diabetes differs from non-diabetic adults”. In: *PloS one* 5.2 (2010).

[96] Ruth E Ley et al. “Obesity alters gut microbial ecology”. In: *Proceedings of the National Academy of Sciences* 102.31 (2005), pp. 11070–11075.

[97] Wei Vivian Li and Jingyi Jessica Li. “A statistical simulator scDesign for rational scRNA-seq experimental design”. In: *Bioinformatics* 35.14 (2019), pp. i41–i50.

[98] Inés Martıénez et al. “Gut microbiome composition is linked to whole grain-induced immunological improvements”. In: *The ISME journal* 7.2 (2013), pp. 269–280.

[99] Geicho Nakatsu et al. "Gut mucosal microbiome across stages of colorectal carcinogenesis". In: *Nature communications* 6.1 (2015), pp. 1–9.

[100] Ninh T Nguyen et al. "Relationship between obesity and diabetes in a US adult population: findings from the National Health and Nutrition Examination Survey, 1999–2006". In: *Obesity surgery* 21.3 (2011), pp. 351–355.

[101] Kirsten JM van Nimwegen et al. "Is the 1000 genome as near as we think? A cost analysis of next-generation sequencing". In: *Clinical chemistry* 62.11 (2016), pp. 1458–1464.

[102] Douglas Noble et al. "Risk models and scores for type 2 diabetes: systematic review". In: *Bmj* 343 (2011), p. d7163.

[103] Edoardo Pasolli et al. "Accessible, curated metagenomic data through ExperimentHub". In: *Nature methods* 14.11 (2017), p. 1023.

[104] Marlene Remely et al. "Abundance and diversity of microbiota in type 2 diabetes and obesity". In: *J Diabetes Metab* 4.253 (2013), p. 2.

[105] B Ren et al. *SparseDOSSA: Sparse data observations for simulating synthetic abundance. 2016.*

[106] Nina Sanapareddy et al. "Increased rectal microbial richness is associated with the presence of colorectal adenomas in humans". In: *The ISME journal* 6.10 (2012), pp. 1858–1868.

[107] Dwayne C Savage. "Microbial ecology of the gastrointestinal tract". In: *Annual review of microbiology* 31.1 (1977), pp. 107–133.

[108] Ivana Semova et al. "Microbiota regulate intestinal absorption and metabolism of fatty acids in the zebrafish". In: *Cell host & microbe* 12.3 (2012), pp. 277–288.

[109] Xiang Jun Shen et al. "Molecular characterization of mucosal adherent bacteria and associations with colorectal adenomas". In: *Gut microbes* 1.3 (2010), pp. 138–147.

[110] Iradj Sobhani et al. "Microbial dysbiosis in colorectal cancer (CRC) patients". In: *PloS one* 6.1 (2011).

[111] Jamie Waese, Nicholas J Provart, and David S Guttman. "Topo-phylogeny: Visualizing evolutionary relationships on a topographic landscape". In: *PloS one* 12.5 (2017), e0175895.

[112] Tingting Wang et al. "Structural segregation of gut microbiota between colorectal cancer patients and healthy volunteers". In: *The ISME journal* 6.2 (2012), pp. 320–329.

[113] Tiffany L Weir et al. "Stool microbiome and metabolome differences between colorectal cancer patients and healthy adults". In: *PloS one* 8.8 (2013).

[114] Na Wu et al. "Dysbiosis signature of fecal microbiota in colorectal cancer patients". In: *Microbial ecology* 66.2 (2013), pp. 462–470.

[115] Sicheng Wu et al. "GMrepo: a database of curated and consistently annotated human gut metagenomes". In: *Nucleic acids research* 48.D1 (2020), pp. D545–D553.

[116] Jian Xiao et al. "Predictive modeling of microbiome data using a phylogeny-regularized generalized linear mixed model". In: *Frontiers in microbiology* 9 (2018), p. 1391.

[117] Yaohua Yang et al. "Oral microbiome and obesity in a large study of low-income and African-American populations". In: *Journal of oral microbiology* 11.1 (2019), p. 1650597.

[118] Jun Chen et al. "An omnibus test for differential distribution analysis of microbiome sequencing data". In: *Bioinformatics* 34.4 (2018), pp. 643–651.

[119] Yanyun Gu et al. "Analyses of gut microbiota and plasma bile acids enable stratification of patients for antidiabetic treatment". In: *Nature communications* 8.1 (2017), pp. 1–12.

[120] Masanori Horie et al. "Comparative analysis of the intestinal flora in type 2 diabetes and nondiabetic mice". In: *Experimental animals* (2017), pp. 17–0021.

[121] WE Moore and Lillian H Moore. "Intestinal floras of populations that have a high risk of colon cancer." In: *Applied and environmental microbiology* 61.9 (1995), pp. 3202–3207.

[122] Cécily Lucas, Nicolas Barnich, and Hang Thi Thu Nguyen. “Microbiota, inflammation and colorectal cancer”. In: *International journal of molecular sciences* 18.6 (2017), p. 1310.

[123] Christine T Peterson et al. “Immune homeostasis, dysbiosis and therapeutic modulation of the gut microbiota”. In: *Clinical & Experimental Immunology* 179.3 (2015), pp. 363–377.

[124] Shaoguang Wu et al. “A human colonic commensal promotes colon tumorigenesis via activation of T helper type 17 T cell responses”. In: *Nature medicine* 15.9 (2009), pp. 1016–1022.

[125] Jun Wang and Huijue Jia. “Metagenome-wide association studies: fine-mining the microbiome”. In: *Nature Reviews Microbiology* 14.8 (2016), pp. 508–522.

[126] He Gao et al. “Polysaccharide from fermented Momordica charantia L. with Lactobacillus plantarum NCU116 ameliorates type 2 diabetes in rats”. In: *Carbohydrate polymers* 201 (2018), pp. 624–633.

[127] Keisuke Kosumi et al. “The amount of Bifidobacterium genus in colorectal carcinoma tissue in relation to tumor characteristics and clinical outcome”. In: *The American journal of pathology* 188.12 (2018), pp. 2839–2852.

[128] Asadollahi Parisa et al. “Anti-cancer effects of Bifidobacterium species in colon cancer cells and a mouse model of carcinogenesis”. In: *PloS one* 15.5 (2020), e0232930.

[129] Sepideh Bahmani, Negar Azarpira, and Elham Moazamian. “Anti-colon cancer activity of Bifidobacterium metabolites on colon cancer cell line SW742”. In: *The Turkish Journal of Gastroenterology* 30.9 (2019), p. 835.

[130] Qing Wang et al. “Administration of Bifidobacterium bifidum CGMCC 15068 modulates gut microbiota and metabolome in azoxymethane (AOM)/dextran sulphate sodium (DSS)-induced colitis-associated colon cancer (CAC) in mice”. In: *Applied microbiology and biotechnology* 104.13 (2020), pp. 5915–5928.

[131] Miguel Gueimonde et al. "Qualitative and quantitative analyses of the bifidobacterial microbiota in the colonic mucosa of patients with colorectal cancer, diverticulitis and inflammatory bowel disease". In: *World journal of gastroenterology: WJG* 13.29 (2007), p. 3985.

[132] Cinderella A Fahmy et al. "Bifidobacterium longum suppresses murine colorectal cancer through the modulation of oncomirs and tumor suppressor mirnas". In: *Nutrition and cancer* 71.4 (2019), pp. 688–700.

[133] Eric Z Chen and Hongzhe Li. "A two-part mixed-effects model for analyzing longitudinal microbiome compositional data". In: *Bioinformatics* 32.17 (2016), pp. 2611–2617.

[134] Mehdi Layeghifard, David M Hwang, and David S Guttman. "Disentangling interactions in the microbiome: a network perspective". In: *Trends in microbiology* 25.3 (2017), pp. 217–228.

[135] Peter J Turnbaugh et al. "The human microbiome project". In: *Nature* 449.7164 (2007), pp. 804–810.

[136] Kameron Y Sugino, Nigel Paneth, and Sarah S Comstock. "Michigan cohorts to determine associations of maternal pre-pregnancy body mass index with pregnancy and infant gastrointestinal microbial communities: late pregnancy and early infancy". In: *PloS one* 14.3 (2019), e0213733.

[137] Qian Yang et al. "The roles of 27 genera of human gut microbiota in ischemic heart disease, type 2 diabetes mellitus, and their risk factors: a Mendelian randomization study". In: *American Journal of Epidemiology* 187.9 (2018), pp. 1916–1922.

[138] Lawrence A David et al. "Diet rapidly and reproducibly alters the human gut microbiome". In: *Nature* 505.7484 (2014), pp. 559–563.

[139] Malo Le Boulch et al. "The MACADAM database: a MetAboliC pAthways DAtabase for Microbial taxonomic groups for mining potential metabolic capacities of archaeal and bacterial taxonomic groups". In: *Database* 2019 (2019).

[140] Donald T McKnight et al. “Methods for normalizing microbiome data: an ecological perspective”. In: *Methods in Ecology and Evolution* 10.3 (2019), pp. 389–400.

[141] Susannah J Salter et al. “Reagent and laboratory contamination can critically impact sequence-based microbiome analyses”. In: *BMC biology* 12.1 (2014), pp. 1–12.

[142] Angela Glassing et al. “Inherent bacterial DNA contamination of extraction and sequencing reagents may affect interpretation of microbiota in low bacterial biomass samples”. In: *Gut pathogens* 8.1 (2016), pp. 1–12.

[143] Jake Jervis-Bardy et al. “Deriving accurate microbiota profiles from human samples with low bacterial content through post-sequencing processing of Illumina MiSeq data”. In: *Microbiome* 3.1 (2015), pp. 1–11.

[144] Philipp Kirstahler et al. “Genomics-based identification of microorganisms in human ocular body fluid”. In: *Scientific reports* 8.1 (2018), pp. 1–14.

[145] Lisa Karstens et al. “Controlling for contaminants in low-biomass 16S rRNA gene sequencing experiments”. In: *MSystems* 4.4 (2019).

[146] Nicole M Davis et al. “Simple statistical identification and removal of contaminant sequences in marker-gene and metagenomics data”. In: *Microbiome* 6.1 (2018), pp. 1–14.

[147] Bradley Efron and Trevor Hastie. *Computer age statistical inference.* Vol. 5. Cambridge University Press, 2016.

[148] R Poudel et al. “Microbiome networks: a systems framework for identifying candidate microbial assemblages for disease management”. In: *Phytopathology* 106.10 (2016), pp. 1083–1096.

[149] Li Chen et al. “GMPR: A robust normalization method for zero-inflated count data with application to microbiome sequencing data”. In: *PeerJ* 6 (2018), e4600.

[150] Ohad Manor and Elhanan Borenstein. “MUSiCC: a marker genes based framework for metagenomic normalization and accurate profiling of gene abundances in the microbiome”. In: *Genome biology* 16.1 (2015), p. 53.

[151] Lucas Schiffer et al. "HMP16SData: Efficient Access to the Human Microbiome Project through Bioconductor". In: *American Journal of Epidemiology* (2019). DOI: 10.1093/aje/kwz006.

[152] Ruochen Jiang. "mbImpute: an accurate and robust imputation method for microbiome data". In: *R package version 0.1, URL: https://github.com/ruochenj/mbImpute* (2015).

[153] Ruochen Jiang, Wei Vivian Li, and Jingyi Jessica Li. "mbImpute: an accurate and robust imputation method for microbiome data". In: *Zenodo* (2021). DOI: 10.5281/zenodo.4840266.

[154] Siyuan Ma et al. "A Statistical Model for Describing and Simulating Microbial Community Profiles". In: *bioRxiv* (2021).

[155] Antoine-Emmanuel Saliba et al. "Single-cell RNA-seq: advances and future challenges". In: *Nucleic acids research* 42.14 (2014), pp. 8845–8860.

[156] Serena Liu and Cole Trapnell. "Single-cell transcriptome sequencing: recent advances and remaining challenges". In: *F1000Research* 5 (2016).

[157] Vladimir Yu Kiselev, Tallulah S Andrews, and Martin Hemberg. "Challenges in unsupervised clustering of single-cell RNA-seq data". In: *Nature Reviews Genetics* 20.5 (2019), pp. 273–282.

[158] Sophie Tritschler et al. "Concepts and limitations for learning developmental trajectories from single cell genomics". In: *Development* 146.12 (2019), dev170506.

[159] Evan Z Macosko et al. "Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets". In: *Cell* 161.5 (2015), pp. 1202–1214.

[160] Robert Salomon et al. "Droplet-based single cell RNAseq tools: a practical guide". In: *Lab on a Chip* 19.10 (2019), pp. 1706–1727.

[161] Grace XY Zheng et al. "Massively parallel digital transcriptional profiling of single cells". In: *Nature communications* 8.1 (2017), pp. 1–12.

[162] Simone Picelli et al. “Full-length RNA-seq from single cells using Smart-seq2”. In: *Nature protocols* 9.1 (2014), pp. 171–181.

[163] Alex A Pollen et al. “Low-coverage single-cell mRNA sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex”. In: *Nature biotechnology* 32.10 (2014), p. 1053.

[164] Xiannian Zhang et al. “Comparative analysis of droplet-based ultra-high-throughput single-cell RNA-seq systems”. In: *Molecular cell* 73.1 (2019), pp. 130–142.

[165] Fang Wang et al. “SCMarker: ab initio marker selection for single cell transcriptome profiling”. In: *PLoS computational biology* 15.10 (2019), e1007445.

[166] Rahul Satija et al. “Spatial reconstruction of single-cell gene expression data”. In: *Nature biotechnology* 33.5 (2015), pp. 495–502.

[167] Vladimir Yu Kiselev et al. “SC3: consensus clustering of single-cell RNA-seq data”. In: *Nature methods* 14.5 (2017), pp. 483–486.

[168] Minzhe Guo et al. “SINCERA: a pipeline for single-cell RNA-Seq profiling analysis”. In: *PLoS computational biology* 11.11 (2015), e1004575.

[169] Yu-Jui Ho et al. “Single-cell RNA-seq analysis identifies markers of resistance to targeted BRAF inhibitors in melanoma cell populations”. In: *Genome research* 28.9 (2018), pp. 1353–1363.

[170] Amit Zeisel et al. “Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq”. In: *Science* 347.6226 (2015), pp. 1138–1142.

[171] Roser Vento-Tormo et al. “Single-cell reconstruction of the early maternal–fetal interface in humans”. In: *Nature* 563.7731 (2018), pp. 347–353.

[172] Adam P Croft et al. “Distinct fibroblast subsets drive inflammation and damage in arthritis”. In: *Nature* 570.7760 (2019), pp. 246–251.

[173] Peijie Lin, Michael Troup, and Joshua WK Ho. “CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data”. In: *Genome biology* 18.1 (2017), p. 59.

[174] Zhe Sun et al. “DIMM-SC: a Dirichlet mixture model for clustering droplet-based single cell transcriptomic data”. In: *Bioinformatics* 34.1 (2018), pp. 139–146.

[175] Christopher Yau et al. “pcaReduce: hierarchical clustering of single cell transcriptional profiles”. In: *BMC bioinformatics* 17.1 (2016), p. 140.

[176] Tallulah S Andrews and Martin Hemberg. “M3Drop: dropout-based feature selection for scRNASeq”. In: *Bioinformatics* 35.16 (2019), pp. 2865–2867.

[177] Cole Trapnell et al. “The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells”. In: *Nature biotechnology* 32.4 (2014), p. 381.

[178] Zhicheng Ji and Hongkai Ji. “TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis”. In: *Nucleic acids research* 44.13 (2016), e117–e117.

[179] Kelly Street et al. “Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics”. In: *BMC genomics* 19.1 (2018), p. 477.

[180] Xiaojie Qiu et al. “Reversed graph embedding resolves complex single-cell trajectories”. In: *Nature methods* 14.10 (2017), p. 979.

[181] Junyue Cao et al. “The single-cell transcriptional landscape of mammalian organogenesis”. In: *Nature* 566.7745 (2019), pp. 496–502.

[182] Wouter Saelens et al. “A comparison of single-cell trajectory inference methods”. In: *Nature biotechnology* 37.5 (2019), pp. 547–554.

[183] Charlotte Soneson and Mark D Robinson. “Bias, robustness and scalability in single-cell differential expression analysis”. In: *Nature methods* 15.4 (2018), p. 255.

[184] Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. “edgeR: a Bioconductor package for differential expression analysis of digital gene expression data”. In: *Bioinformatics* 26.1 (2010), pp. 139–140.

[185] Trung Nghia Vu et al. “Beta-Poisson model for single-cell RNA-seq data analyses”. In: *Bioinformatics* 32.14 (2016), pp. 2128–2135.

[186] Zhun Miao et al. “DEsingle for detecting three types of differential expression in single-cell RNA-seq data”. In: *Bioinformatics* 34.18 (2018), pp. 3223–3224.

[187] Tomi Suomi et al. “ROTS: An R package for reproducibility-optimized statistical testing”. In: *PLoS computational biology* 13.5 (2017), e1005562.

[188] Greg Finak et al. “MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data”. In: *Genome biology* 16.1 (2015), pp. 1–13.

[189] Keegan D Korthauer et al. “A statistical approach for identifying differential distributions in single-cell RNA-seq experiments”. In: *Genome biology* 17.1 (2016), p. 222.

[190] Peter V Kharchenko, Lev Silberstein, and David T Scadden. “Bayesian approach to single-cell differential expression analysis”. In: *Nature methods* 11.7 (2014), pp. 740–742.

[191] Stephanie C Hicks et al. “Missing data and technical variability in single-cell RNA-sequencing experiments”. In: *Biostatistics* 19.4 (2018), pp. 562–578.

[192] Koen Van den Berge et al. “Observation weights unlock bulk RNA-seq tools for zero inflation and single-cell applications”. In: *Genome biology* 19.1 (2018), pp. 1–17.

[193] Jiarui Ding et al. “Systematic comparison of single-cell and single-nucleus RNA-sequencing methods”. In: *Nature biotechnology* (2020), pp. 1–10.

[194] David van Dijk et al. “MAGIC: A diffusion-based imputation method reveals gene-gene interactions in single-cell RNA-sequencing data”. In: *BioRxiv* (2017), p. 111591.

[195] Wei Vivian Li and Jingyi Jessica Li. “An accurate and robust imputation method scImpute for single-cell RNA-seq data”. In: *Nature communications* 9.1 (2018), pp. 1–9.

[196] Emma Pierson and Christopher Yau. “ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis”. In: *Genome biology* 16.1 (2015), pp. 1–10.

[197] Wuming Gong et al. “DrImpute: imputing dropout events in single cell RNA sequencing data”. In: *BMC bioinformatics* 19.1 (2018), pp. 1–10.

[198] Mo Huang et al. “SAVER: gene expression recovery for single-cell RNA sequencing”. In: *Nature methods* 15.7 (2018), pp. 539–542.

[199] Divyanshu Talwar et al. “AutoImpute: Autoencoder based imputation of single-cell RNA-seq data”. In: *Scientific reports* 8.1 (2018), pp. 1–11.

[200] Jonathan Ronen and Altuna Akalin. “netSmooth: Network-smoothing based imputation for single cell RNA-seq”. In: *F1000Research* 7 (2018).

[201] Md Bahadur Badsha et al. “Imputation of single-cell gene expression with an autoencoder neural network”. In: *Quantitative Biology* (2020), pp. 1–17.

[202] Gökcen Eraslan et al. “Single-cell RNA-seq denoising using a deep count autoencoder”. In: *Nature communications* 10.1 (2019), pp. 1–14.

[203] Aanchal Mongia, Debarka Sengupta, and Angshul Majumdar. “McImpute: Matrix completion based imputation for single cell RNA-seq data”. In: *Frontiers in genetics* 10 (2019), p. 9.

[204] Chong Chen et al. “scRMD: Imputation for single cell RNA-seq data via robust matrix decomposition”. In: *Bioinformatics* 36.10 (2020), pp. 3156–3161.

[205] Mary Qu Yang et al. “MISC: missing imputation for single-cell RNA sequencing data”. In: *BMC systems biology* 12.7 (2018), p. 114.

[206] Wenhao Tang et al. “bayNorm: Bayesian gene expression recovery, imputation and normalization for single-cell RNA-sequencing data”. In: *Bioinformatics* 36.4 (2020), pp. 1174–1181.

[207] Rebecca Elyanow et al. “netNMF-sc: leveraging gene–gene interactions for imputation and dimensionality reduction in single-cell expression analysis”. In: *Genome research* 30.2 (2020), pp. 195–204.

[208] Marmar Moussa and Ion I Măndoiu. “Locality sensitive imputation for single cell RNA-Seq data”. In: *Journal of Computational Biology* 26.8 (2019), pp. 822–835.

[209] Tao Peng et al. “SCRABBLE: single-cell RNA-seq imputation constrained by bulk RNA-seq data”. In: *Genome biology* 20.1 (2019), p. 88.

[210] Yungang Xu et al. “scIGANs: single-cell RNA-seq imputation using generative adversarial networks”. In: *Nucleic acids research* 48.15 (2020), e85–e85.

[211] Romain Lopez et al. “Deep generative modeling for single-cell transcriptomics”. In: *Nature methods* 15.12 (2018), pp. 1053–1058.

[212] Valentine Svensson. “Droplet scRNA-seq is not zero-inflated”. In: *Nature Biotechnology* 38.2 (2020), pp. 147–150.

[213] Peng Qiu. “Embracing the dropouts in single-cell RNA-seq analysis”. In: *Nature communications* 11.1 (2020), pp. 1–9.

[214] Luke Zappia, Belinda Phipson, and Alicia Oshlack. “Splatter: simulation of single-cell RNA sequencing data”. In: *Genome biology* 18.1 (2017), pp. 1–15.

[215] Bruce Alberts et al. *Molecular biology of the cell*. Garland Science, Taylor and Francis Group, 2018.

[216] Arjun Raj et al. “Stochastic mRNA synthesis in mammalian cells”. In: *PLoS Biol* 4.10 (2006), e309.

[217] Alvaro Sanchez and Ido Golding. “Genetic determinants and cellular constraints in noisy gene expression”. In: *Science* 342.6163 (2013), pp. 1188–1193.

[218] David M Suter et al. “Mammalian genes are transcribed with widely different bursting kinetics”. In: *science* 332.6028 (2011), pp. 472–474.

[219] François Spitz and Eileen EM Furlong. “Transcription factors: from enhancer binding to developmental control”. In: *Nature reviews genetics* 13.9 (2012), pp. 613–626.

[220] Sachi Inukai, Kian Hong Kock, and Martha L Bulyk. “Transcription factor–DNA binding: beyond binding site motifs”. In: *Current opinion in genetics & development* 43 (2017), pp. 110–119.

[221] Samuel A Lambert et al. “The human transcription factors”. In: *Cell* 172.4 (2018), pp. 650–665.

[222] Pawel Paszek. “Modeling stochasticity in gene regulation: characterization in the terms of the underlying distribution function”. In: *Bulletin of Mathematical Biology* 69.5 (2007), pp. 1567–1601.

[223] Jean Peccoud and Bernard Ycart. “Markovian modeling of gene-product synthesis”. In: *Theoretical population biology* 48.2 (1995), pp. 222–234.

[224] Jong Kyoung Kim and John C Marioni. “Inferring the kinetics of stochastic gene expression from single-cell RNA-sequencing data”. In: *Genome biology* 14.1 (2013), pp. 1–12.

[225] Jessica Schwaber, Stacey Andersen, and Lars Nielsen. “Shedding light: the importance of reverse transcription efficiency standards in data interpretation”. In: *Biomolecular detection and quantification* 17 (2019), p. 100077.

[226] Stephen Bustin et al. “Variability of the reverse transcription step: practical implications”. In: *Clinical Chemistry* 61.1 (2015), pp. 202–212.

[227] Abhishek Kaul et al. “Analysis of microbiome data in the presence of excess zeros”. In: *Frontiers in microbiology* 8 (2017), p. 2114.

[228] Randall K Saiki et al. “Primer-directed enzymatic amplification of DNA with a thermostable DNA polymerase”. In: *Science* 239.4839 (1988), pp. 487–491.

[229] James Eberwine et al. “Analysis of gene expression in single live neurons”. In: *Proceedings of the National Academy of Sciences* 89.7 (1992), pp. 3010–3014.

[230] Fuchou Tang, Kaiqin Lao, and M Azim Surani. “Development and applications of single-cell transcriptome analysis”. In: *Nature methods* 8.4 (2011), S6–S11.

[231] Yu Fu et al. “Elimination of PCR duplicates in RNA-seq and small RNA-seq using unique molecular identifiers”. In: *Bmc Genomics* 19.1 (2018), p. 531.

[232] Po-Yuan Tung et al. “Batch effects and the effective design of single-cell gene expression studies”. In: *Scientific reports* 7 (2017), p. 39921.

[233] Katsuyuki Shiroguchi et al. “Digital RNA sequencing minimizes sequence-dependent bias and amplification noise with optimized single-molecule barcodes”. In: *Proceedings of the National Academy of Sciences* 109.4 (2012), pp. 1347–1352.

[234] Rita S Cha and William G Thilly. “Specificity, efficiency, and fidelity of PCR”. In: *PCR Methods Appl* 3.3 (1993), pp. 18–29.

[235] Juliane C Dohm et al. "Substantial biases in ultra-short read data sets from high-throughput DNA sequencing". In: *Nucleic acids research* 36.16 (2008), e105.

[236] Tom Smith, Andreas Heger, and Ian Sudbery. "UMI-tools: modeling sequencing errors in Unique Molecular Identifiers to improve quantification accuracy". In: *Genome research* 27.3 (2017), pp. 491–499.

[237] Daniel Aird et al. "Analyzing and minimizing PCR amplification bias in Illumina sequencing libraries". In: *Genome biology* 12.2 (2011), pp. 1–14.

[238] Hannah R Dueck et al. "Assessing characteristics of RNA amplification methods for single cell RNA sequencing". In: *BMC genomics* 17.1 (2016), pp. 1–22.

[239] F William Townes et al. "Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model". In: *Genome biology* 20.1 (2019), pp. 1–16.

[240] Abhishek K Sarkar and Matthew Stephens. "Separating measurement and expression models clarifies confusion in single cell RNA-seq analysis". In: *BioRxiv* (2020).

[241] Lingxue Zhu et al. "A unified statistical framework for single cell and bulk RNA sequencing data". In: *The annals of applied statistics* 12.1 (2018), p. 609.

[242] Maryam Zand and Jianhua Ruan. "Network-based single-cell RNA-seq data imputation enhances cell type identification". In: *Genes* 11.4 (2020), p. 377.

[243] Di Ran et al. "scDoc: correcting drop-out events in single-cell RNA-seq data". In: *Bioinformatics* 36.15 (2020), pp. 4233–4239.

[244] David Lähnemann et al. "Eleven grand challenges in single-cell data science". In: *Genome biology* 21.1 (2020), pp. 1–35.

[245] Tallulah S Andrews and Martin Hemberg. "False signals induced by single-cell imputation". In: *F1000Research* 7 (2018).

[246] Davide Risso et al. "A general and flexible method for signal extraction from single-cell RNA-seq data". In: *Nature communications* 9.1 (2018), pp. 1–17.

[247] Saiful Islam et al. "Quantitative single-cell RNA-seq with unique molecular identifiers". In: *Nature methods* 11.2 (2014), p. 163.

[248] Tianyi Sun et al. “scDesign2: a transparent simulator that generates high-fidelity single-cell gene expression count data with gene correlations captured”. In: *Genome biology* 22.1 (2021), pp. 1–37.

[249] Xinzhou Ge et al. “Clipper: p-value-free FDR control on high-throughput data from two conditions”. In: *bioRxiv* (2021), pp. 2020–11.

[250] Matthew Amodio et al. “Exploring single-cell data with deep multitasking neural networks”. In: *Nature methods* (2019), pp. 1–7.

[251] David I Warton. “Why you cannot transform your way out of trouble for small counts”. In: *Biometrics* 74.1 (2018), pp. 362–368.

[252] Andrew. *You should (usually) log transform your positive data.* August 21st, 2019. URL: https://statmodeling.stat.columbia.edu/2019/08/21/you-should-usually-log-transform-your-positive-data/.

[253] Charity W Law et al. “voom: Precision weights unlock linear model analysis tools for RNA-seq read counts”. In: *Genome biology* 15.2 (2014), R29.

[254] Yao He et al. “DISC: a highly scalable and accurate inference of gene expression and structure for single-cell transcriptomes using semi-supervised deep learning”. In: *Genome Biology* 21.1 (2020), pp. 1–28.

[255] Kai Simon and Georg Lausen. “ViPER: augmenting automatic information extraction with visual perceptions”. In: *Proceedings of the 14th ACM international conference on Information and knowledge management.* 2005, pp. 381–388.

[256] Yumei Li et al. “A large-sample crisis? Exaggerated false positives by popular differential expression methods”. In: *bioRxiv* (2021).

[257] Ruoxin Li and Gerald Quon. “scBFA: modeling detection patterns to mitigate technical noise in large-scale single-cell genomics data”. In: *Genome biology* 20.1 (2019), p. 193.

[258] Victoria Moignard et al. “Decoding the regulatory network of early blood development from single-cell gene expression measurements”. In: *Nature biotechnology* 33.3 (2015), pp. 269–276.

[259] Haifen Chen et al. “Single-cell transcriptional analysis to uncover regulatory circuits driving cell fate decisions in early mouse development”. In: *Bioinformatics* 31.7 (2015), pp. 1060–1066.

[260] Chee Yee Lim et al. “BTR: training asynchronous Boolean models using single-cell expression data”. In: *BMC bioinformatics* 17.1 (2016), pp. 1–18.

[261] Aman Agrawal et al. “Scalable probabilistic PCA for large-scale genetic variation data”. In: *PLoS Genetics* 16.5 (2020), e1008773.

[262] Shawn C Baker et al. “The external RNA controls consortium: a progress report”. In: *Nature methods* 2.10 (2005), p. 731.

[263] FDA SEQC et al. “A Comprehensive Multi-Center Cross-platform Benchmarking Study of Single-cell RNA Sequencing Using Reference Samples”. In: *bioRxiv* (2020).

[264] Tallulah S Andrews et al. “Tutorial: guidelines for the computational analysis of single-cell RNA sequencing data”. In: *Nature Protocols* (2020), pp. 1–9.

[265] Michael B. Sohn and Hongzhe Li. “A GLM-based latent variable ordination method for microbiome samples”. In: *Biometrics* 74.2 (2018), pp. 448–457.

[266] Julien Chiquet, Mahendra Mariadassou, and Stéphane Robin. “The Poisson-lognormal model as a versatile framework for the joint analysis of species abundances”. In: *Frontiers in Ecology and Evolution* 9 (2021), p. 188.

[267] A Sina Booeshaghi and Lior Pachter. “Normalization of single-cell RNA-seq counts by log (x+ 1) or log (1+ x)”. In: *Bioinformatics* 37.15 (2021), pp. 2223–2224.

[268] Robert O’Hara and Johan Kotze. “Do not log-transform count data”. In: *Nature Precedings* (2010), pp. 1–1.