Proteomics

WILEY-VCH

Proteomics
Proteomics and Systems Biology

REVIEW OPEN ACCESS

Computational Methods for Data Integration and Imputation of Missing Values in *Omics* Datasets

Yannis Schumann[1] | Antonia Gocke[2,3] | Julia E. Neumann[2,4]

[1]IT-Department, Deutsches Elektronen-Synchroton DESY, Hamburg, Germany | [2]Center for Molecular Neurobiology (ZMNH), University Medical Center Hamburg-Eppendorf (UKE), Hamburg, Germany | [3]Core Facility Mass Spectrometric Proteomics, University Medical Center Hamburg-Eppendorf (UKE), Hamburg, Germany | [4]Institute of Neuropathology, University Medical Center Hamburg-Eppendorf (UKE), Hamburg, Germany

Correspondence: Yannis Schumann (yannis.schumann@desy.de) | Julia E. Neumann (ju.neumann@uke.de)

Received: 28 June 2024 | **Revised:** 8 November 2024 | **Accepted:** 26 November 2024

Funding: J.E.N is funded by the DFG (Emmy Noether program). A.G. is funded by the German Federal Ministry of Education and Research (BMBF) as part of project COMET.

Keywords: algorithm | data integration | missing values | *omics*

ABSTRACT

Molecular profiling of different *omic*-modalities (e.g., DNA methylomics, transcriptomics, proteomics) in biological systems represents the basis for research and clinical decision-making. Measurement-specific biases, so-called batch effects, often hinder the integration of independently acquired datasets, and missing values further hamper the applicability of typical data processing algorithms. In addition to careful experimental design, well-defined standards in data acquisition and data exchange, the alleviation of these phenomena particularly requires a dedicated data integration and preprocessing pipeline. This review aims to give a comprehensive overview of computational methods for data integration and missing value imputation for *omic* data analyses.

We provide formal definitions for missing value mechanisms and propose a novel statistical taxonomy for batch effects, especially in the presence of missing data. Based on an automated document search and systematic literature review, we describe 32 distinct data integration methods from five main methodological categories, as well as 37 algorithms for missing value imputation from five separate categories. Additionally, this review highlights multiple quantitative evaluation methods to aid researchers in selecting a suitable set of methods for their work. Finally, this work provides an integrated discussion of the relevance of batch effects and missing values in *omics* with corresponding method recommendations. We then propose a comprehensive three-step workflow from the study conception to final data analysis and deduce perspectives for future research. Eventually, we present a comprehensive flow chart as well as exemplary decision trees to aid practitioners in the selection of specific approaches for imputation and data integration in their studies.

1 | Introduction

Omic data provides a detailed characterization of a specific and high-dimensional molecular target landscape (e.g., metabolome, proteome, or transcriptome) in a biological system and can be obtained using various measurement techniques (e.g., mass spectrometry, microarrays, or next-generation sequencing). Such *omic* layers can be considered both independently as well as

Abbreviations: (sc)RNA, seq(single-cell) RNA sequencing; PCA, principal components analysis; SVD, singular value decomposition; t-SNE, t-distributed stochastic neighborhood embedding.

Yannis Schumann and Antonia Gocke shared first authorship.

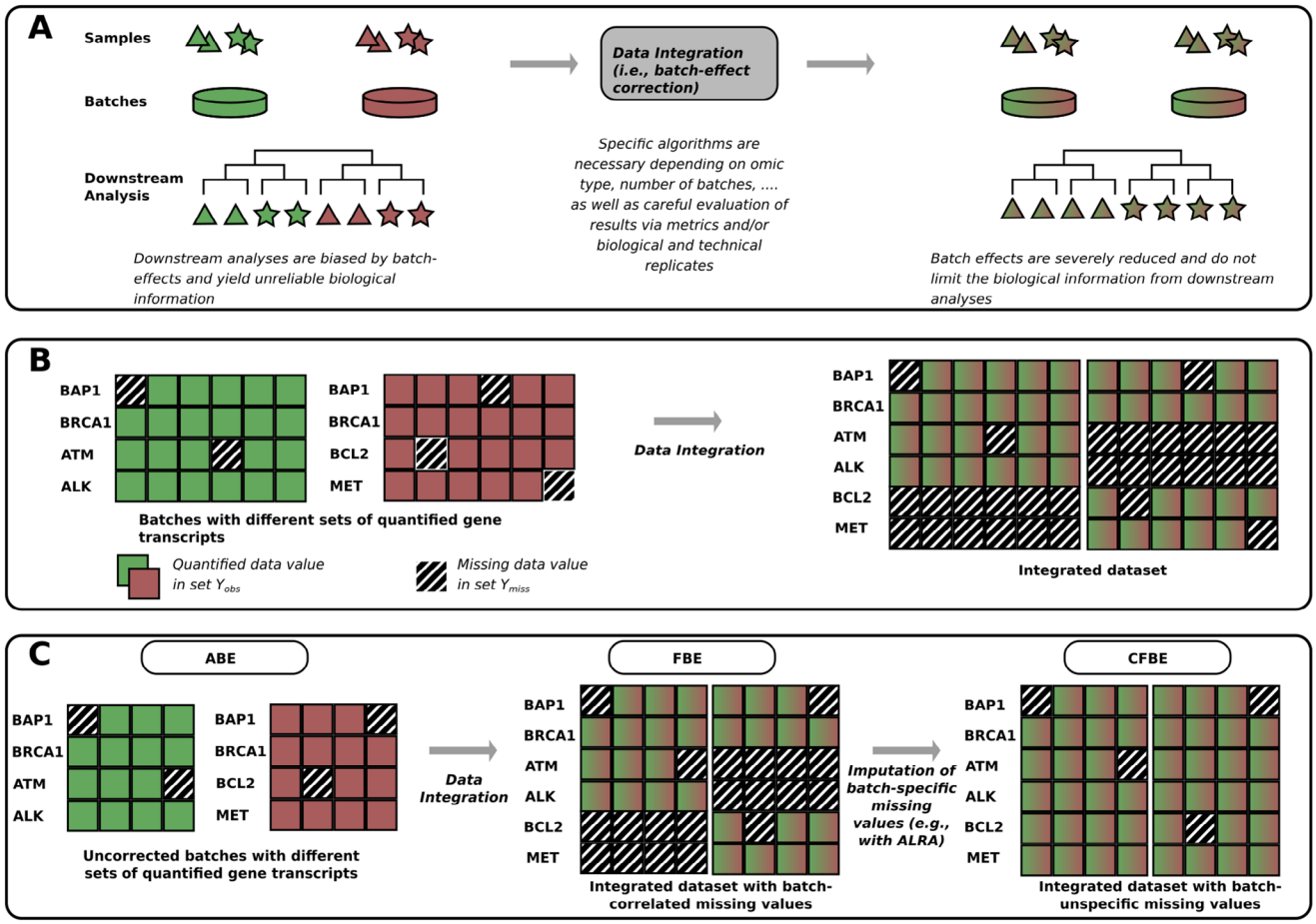

FIGURE 1 | (A) Concept sketch for data integration. (B) The total data matrix is partitioned into observed and missing subsets. Data incompleteness is additionally amplified by data integration. (C) The novel statistical terminology for batch effects in the presence of missing values introduced in this work. ABE, afflicted with batch effects; FBE, free of batch effects; CFBE, completely free of batch effects.

jointly (multi-*omic*) and frequent applications include characterization, classification, and trait prediction of a considered biological system or condition. Normally, it is assumed that *omic* data represent the "totality" of the analyzed system (limited by technical possibilities). Due to the high dimensionality, *omic* data are exclusively processed using computational methods, which are, for example, designed to perform subtype discovery, to find biomarkers/therapeutic targets, to predict survival/risk of disease progression, or even to perform clinical diagnostics. Whilst a large number of well-established methods for such (and other) tasks exists, their applicability for most *omic* types is often hampered by the prevalence of missing values and nonbiological variation in the data.

In particular, researchers typically pool their considered data from multiple measurements/data sources, since technical limitations, financial shortcomings, and sample availability often lead to small cohort sizes and hence limit the statistical strength of the individual cohorts. Often, each of the considered datasets is afflicted with specific biases (batch effects), that limit their direct comparability and thus also the direct applicability of most downstream data analysis/data processing algorithms. Therefore, *data integration* methods are necessary that apply (often *omic*-specific) statistical models to remove these batch effects and construct an integrated dataset for downstream data analysis, compare Figure 1,panel A. Note that while the term *data integration* is commonly used to describe (1) integration of data from different omic types (multi-*omics*) and (2) integration of multiple datasets of the same *omic*-type, the algorithms for these two fields of application differ greatly. In particular, this work focuses on the integration of multiple cohorts from the same *omic*-type and the reader is referred to the reviews in [1–3] for an overview of multi-*omic* data integration and to [4] for an integrated consideration of multi-*omics* with missing values.

In addition to batch effects, *omic* measurements are often afflicted with missing values, which can for instance occur due to biological, technical, or software-analytical shortcomings during data quantification. Most data-analysis methods either compute surrogate values for the unquantified data (imputation) or apply heuristic approaches for data analysis, for example, listwise deletion, the latter of which is only applicable in case of minimal data incompleteness. Note that the number of missing values may increase additionally when performing data integration, compare Figure 1,panel B. Another important challenge is to specifically account for the type of missing values (cf. Section 1.1), since individual missing values may need to be accounted for differently than dataset-specific missing values.

Although a number of surveys on both data integration [5–7] as well as on the consideration of missing values in *omics* [8, 9] have been published, these works limit their considerations to one or the other, often focus on a specific *omic* type and rarely include a dedicated discussion of statistical definitions or suitable evaluation metrics. Moreover, there is a lack of recommendations for computational workflows and methods that holistically address both challenges, as well as a lack of an integrated perspective for future research. To this end, this review contributes to the corpus of existing literature by

- Introducing a novel statistical terminology for batch effects, specifically in the presence of missing data,
- providing a comprehensive and integrated overview of computational methods for data integration and missing values in *omics* (i.e., discussing both challenges without limitation to a specific *omic* type) and establishing corresponding taxonomies based on the respective statistical approaches,
- recommending dedicated algorithms for specific use cases and proposing a *best-practice* workflow covering study design, data integration, and data imputation, as well as analysis of *omic* data in the presence of missing values,
- eliciting the limitations of existing methods and outlining potential directions of future research.

The following Section 1.1 formally defines missing values, batch effects, and the respective terminology used throughout this work. Subsequently, Section 2 describes the literature selection process for this study. Then, two sections consecutively describe Missing Values and Imputation Methods (Section 3) and Data Integration Methods (Section 4), followed by the introduction of the corresponding Section 5. The final Section 6 introduces a three-step workflow from study conceptualization to data analysis, recommends specific algorithms as starting points for further evaluations per *omic* type and provides an outlook on future research directions.

1.1 | Definitions and Systematics

This work defines *missing values* as any missing data value for a variable of an observation, that is, independent of the respective cause of missingness (e.g., biological or technical reasons). Let now the complete data matrix Y be partitioned into disjoint sets of observed and unobserved (missing) data

$$Y \equiv Y_{\text{obs}} \cup Y_{\text{miss}} \text{ with } Y_{\text{obs}} \cap Y_{\text{miss}} = \emptyset.$$

Following common practice, missing values can be categorized as missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR) as defined by Little and Rubin [10].

That is, given the conditional distribution $f(M \mid Y, \phi)$ of the missing data pattern M on Y and unknown parameters ϕ, missing data are classified as

MCAR if f does not depend on any missing or observed data, that is, $f(M|Y,\phi) = f(M|\phi) \forall Y, \phi$ (e.g., missing values in data-independent acquisition mass-spectrometry)

MAR if f depends only on the observed data Y_{obs}, that is, $f(M|Y,\phi) = f(M|Y_{\text{obs}},\phi) \forall Y_{\text{miss}}, \phi$ (e.g., missing Y-chromosome information for a female patient)

MNAR if the missing data mechanism depends on the original, numeric values of the missing data, that is, $f(M|Y,\phi) = f(M|Y_{\text{miss}},\phi) \forall Y_{\text{obs}}, \phi$ (e.g., missing values in data-dependent acquisition of a mass-spectrometer).

Note that the missing value pattern M is distinct from the unobserved numerical values Y_{miss}. In general, *omic*-types that are largely affected by a limit of detection (i.e., mass spectrometric proteome data, metabolite data) comprise a complex mixture of MAR, MCAR, MNAR missing values [11]. In sequencing data (esp. scRNAseq-data) researchers typically differentiate between biological and technical zeroes. The former are introduced due to the low expression of the transcript and can be regarded as MNAR-type missing values, whereas the latter can arise either due to a low sequencing depth or inefficient amplification/cDNA generation and are thus regarded as MCAR-type missing values [12–14]. As elaborated later in this manuscript, careful consideration of the missing value type must be taken when selecting, for example, an imputation method.

In practice, data are comprised by one or multiple cohorts (*batches*), that each represent one data acquisition event. Every batch represents a collection of observations (*samples*), each of which represents a (potentially unknown) combination of biological and experimental (e.g., treatment) conditions. When incorporated into a statistical model, these conditions are referred to as *covariates*. Note that multiple samples may be taken from the same individual (even across batches).

Measurement-specific variations between the batches are referred to as *batch effects*. Throughout this work, we define *data integration* as the process of reducing these batch effects from a given set of batches (*batch-effect correction*) with the aim to combine these into one integrated dataset with a larger number of cases (i.e., with higher statistical power in downstream analyses). Note that albeit many publications provide statistical models for batch effects, these have—to the best of our knowledge—not yet been defined formally, especially not under consideration of missing values. We thus propose to transfer the above definition of missing value patterns to batch effects. In particular, we assume that the complete data matrix Y can be expressed as a function of batch B, the covariates C and unknown parameters ω, where the latter also include all further biological and technical variation. Let then $g(Y_{obs}|B,C,\omega)$ and $h(M|B,C,\omega)$ represent the conditional distribution of the observed data Y_{obs} and the missing data pattern M on these parameters (i.e., implicitly accounting for the data matrix Y). Note that we restrict the considerations of g, h to Y_{obs} and M—while this does not imply an independence of the true (but unobserved) values Y_{miss} with respect to B, C, ω, we argue that only Y_{obs}, M are observable and hence most relevant to a practitioner. We now propose to classify data as completely free

of batch effects (CFBE), free of batch effects (FBE), and afflicted with batch effects (ABE) based on the following definition:

CFBE if g and h do not depend on B, that is, $g(Y_{\text{obs}}|B,C,\omega) = g(Y_{\text{obs}}|C,\omega) \wedge h(M|B,C,\omega) = h(M|C,\omega) \forall C,\omega$ (e.g., array data where missing intensities represent true zeroes)

FBE if *only* g is independent of B, that is, $g(Y_{obs}|B,C,\omega) = g(Y_{\text{obs}}|C,\omega) \forall Y_{\text{obs}},C,\omega \wedge \exists \text{B},M,C,\omega : h(M|B,C,\omega) \neq h(M|C,\omega)$ (e.g., integrated dataset from multiplexed mass spectrometry with tandem mass tags, in which variables are most typically either quantified or missing for all samples)

ABE otherwise.

CFBE, FBE, and ABE are exemplarily visualized in Figure 1,panel C. Note that the terms *free of batch effects* and *batch-effect correction* primarily refer to the theoretical concepts—in practice, batch effects are typically only reduced by batch-effect correction algorithms.

2 | Methods

Eligible documents for this review were selected using the Scopus advanced search tool from Elsevier. In particular, document titles were searched for a combination of any of the (target) keywords *imputation*, *data integration*, *data harmonization*/*data harmonization*, and *batch effects* with at least one of the (context) keywords *omic*, *RNA seq*/*sequencing*, *mass spectrometry*, *microarray*, *gene expression*. Furthermore, document titles and abstracts were required to contain any of the keywords *framework*, *algorithm*, or *method* and none of *multi-omic*/*multi-omics*. In total, the respective query returned 381 peer-reviewed articles and review papers written in English until May 3, 2024 (cf. Figure 2). These documents underwent critical abstract screening, which yielded 262 documents that presented novel computational methods, reported improvements/metrics/recommendations, or reviewed the respective research field themselves, while not focusing on a specific disease/dataset/biological condition or on manual or multi-*omic* data integration. Following full-text screening by scope, relevance, and methodological heterogeneity, a total of 69 computational methods from the disciplines of data integration and data imputation were considered for this review. Algorithmic content validation of the 262 abstract-screened documents was performed using text clustering, revealing four broad article clusters with content types represented by the selected 69 methods (cf. Supporting Information).

3 | Missing Values and Imputation Methods

Missing values have been reported for all contemporary *omic* types, including proteomics [15] and metabolomics [11], microarray-based technologies [16], and even single-cell RNA sequencing [17]. Furthermore, data integration often increases the absolute number of missing values, since some variables (e.g., gene transcripts) may be missing entirely for some of the integrated datasets. This incompleteness causes a challenge for many subsequent data analysis tasks and missing-value tolerant methods are scarce. Therefore, to not exclude these variables (e.g., through listwise deletion), researchers often

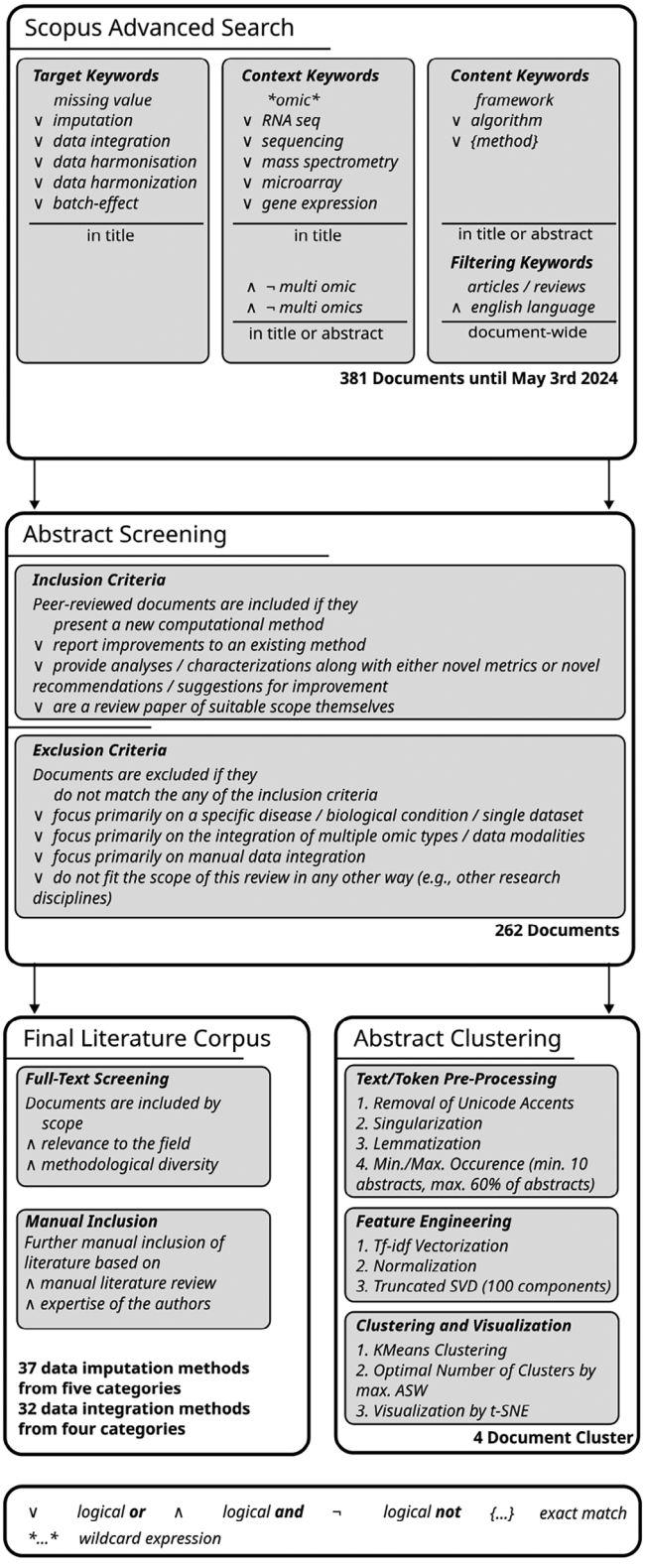

FIGURE 2 | Overview over the employed methodology for literature selection and identification of document clusters. Initial document search was performed using Scopus Advanced Search (top), followed by manual abstract screening (center) which resulted in 262 documents for full-text screening. Finally, 37 data imputation methods and 32 data integration algorithms were incorporated into this review (bottom left). Unsupervised clustering of the 262 screened articles revealed four groups of considered manuscripts.

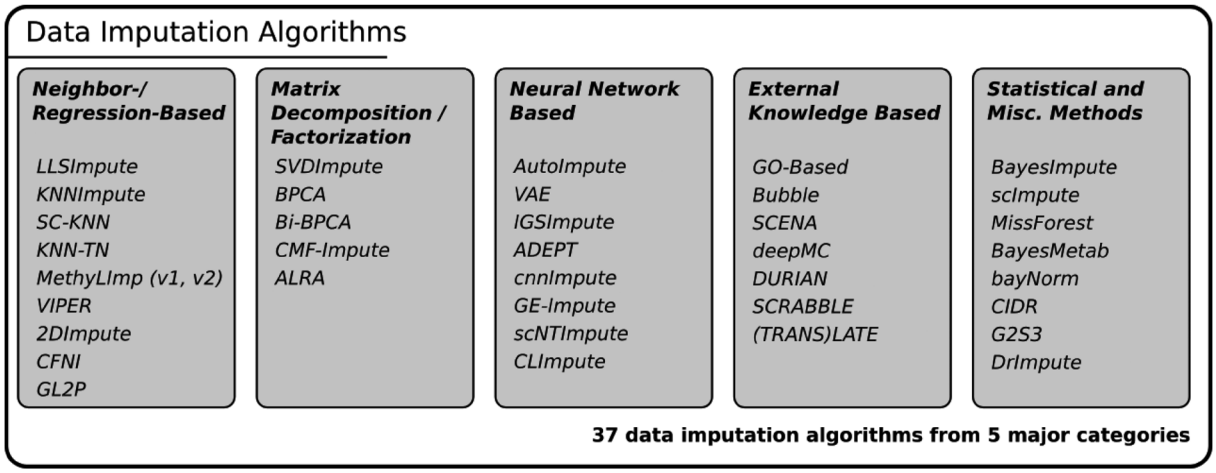

FIGURE 3 | Overview over the data imputation methods discussed in this review.

utilize imputation methods, which aim to estimate the "true" values of the missing data points. With the increasing use of single-cell techniques (including e.g., single-cell proteomics [18]) and high-throughput technologies, this approach will gain increasing relevance and this section hence provides a comprehensive overview of representative imputation methods.

Generally, imputation methods can be distinguished into single imputation (i.e., providing a single estimate for the missing data point) and multiple imputation, which aims to provide multiple potential estimates. This review focuses on single imputation, since it has been demonstrated to often provide imputation results with similar quality [19], whereas multiple imputation methods (e.g., MICE [20, 21]) may, for example exhibit model instability or weak convergence for high-dimensional data [22] such as encountered in omics analyses. Moreover, obtaining multiple imputation estimates poses an additional hindrance to intuitiveness and interpretability, for example, in scatter plots as commonly used in practice. The systematic literature review procedure described in Section 2 revealed five major methodological approaches for imputation: Neighbor- and regression-based approaches, matrix decomposition/factorization techniques, neural network methods, approaches based on external knowledge and statistical/miscellaneous procedures. This section briefly presents 37 exemplary methods from all five categories. These methods and their respective categorization are visualized in Figure 3 and Table 1 additionally presents a comprehensive summary of their software availability as well as suitable omic-types.

3.1 | Neighbor- and Regression-based Approaches

Neighbor- and regression-based approaches represent a very versatile group of imputation methods that offer a particularly large range of applicable *omic* types. They typically quantify similarity between individual observations (e.g., cells) and variables (e.g., genes) and estimate the missing data points from other data in close proximity. Note that these imputation methods directly utilize other data points for inference, which may yield artificially increased correlation values, that need to be considered when choosing any downstream data analysis method.

LLSImpute (and variations e.g., ILLSimpute/ RLLSimpute/ WLLSI/LAW-LSImpute)

With local least-squares (LLS) based imputation, Kim et al. proposed a highly influential imputation method for microarray gene expression data that uses a two-step procedure [23]. To impute a data point for a given target gene, it first selects neighboring genes based on Pearson correlation coefficients and then fits a least-squares regression model to these genes in order to estimate the missing values. Researchers have proposed several variations of the original method, among which ILLSimpute (iterated LLS imputation) iteratively refines missing value estimates using suitable candidate genes from within a distance threshold [24]. Authors further proposed L1/L2 regularized LLS regression (RLLSimpute_L1/L2) to account for highly correlated genes and overfitting [25]. Further notable variations include WLLSI (weighted LLS imputation) and LAW-LSImpute (locally auto-weighted LLS imputation) [26, 27]. Note that LLSImpute has been shown to be less suitable for data with high ratios of MNAR data and more suited for data with higher ratios of MCAR or MAR data [28].

KNNImpute (and variations, e.g., SKNNImpute, IKNNImpute)

A similarly influential model for microarray data imputation, KNNImpute (K-nearest neighbor imputation), was introduced by Troyanskaya et al., who first selected neighboring genes based on a distance metric (e.g., Euclidean distances) and estimated missing values based on distance-weighted averages of these genes [29]. Amongst other variations of KNNImpute, SKNNImpute sequentially imputes each gene, starting from the most complete one [30]. Later, IKNNImpute was proposed, which iteratively refines the missing value estimates [31]. KNN methods, like KNNImpute, have been shown to be most effective for MCAR missing values [32].

TABLE 1 | Comprehensive summary of considered imputation methods.

Name	Access	Direct link	Developed for data type
		Neighbor- and regression-based approaches	
LLSImpute	F/P	https://www.bioconductor.org/packages/release/bioc/html/pcaMethods.html	Microarray gene expression
KNNImpute	F/P	https://scikit-learn.org/stable/index.html	Microarray gene expression
SC-KNN	F/P	https://scikit-learn.org/stable/index.html	Microarray gene expression
KNN-TN	C/R	https://doi.org/10.1186/s12859-017-1547-6	Metabolome
MethyLImp/ MethyLImp2	L/R	https://www.bioconductor.org/packages/release/bioc/html/methyLImp2.html	DNA methylation
VIPER	L/R	https://github.com/ChenMengjie/VIPER	Single cell RNA seq
2Dimpute	L/R	https://github.com/zky0708/2DImputeCFNI	Single cell RNA seq
CFNI	—	https://doi.org/10.1504/IJDMB.2016.076535	DNA microarray
GL2P	—	https://doi.org/10.1016/j.compbiomed.2016.08.005	Microarray data
		Matrix Decomposition/Factorization Approaches	
SVDImpute	F/R	https://www.bioconductor.org/packages/release/bioc/html/pcaMethods.html	Microarray gene expression
BPCA	F/R	https://www.bioconductor.org/packages/release/bioc/html/pcaMethods.html	Microarray gene expression
Bi-BPCA	F/M	https://github.com/fanchi/bi-BPCA	Microarray gene expression
CMF-Impute	L/M	https://github.com/xujunlin123/CMFImpute	Single cell RNA seq
ALRA	L/R	https://github.com/KlugerLab/ALRA	Single cell RNA seq
		Neural Network Based Approaches	
AutoImpute	L/P	https://pypi.org/project/autoimpute/	Single cell RNA seq
VAE	L/P	https://github.com/gevaertlab/BetaVAEImputation	Transcriptome and methylome
IGSImpute	L/P	https://github.com/ericcombiolab/IGSimpute	Single cell RNA seq
ADEPT	L/P	https://github.com/maiziezhoulab/ADEPT	Spatial transcriptome
cnnImpute	—	https://doi.org/10.1038/s41598-024-53998-x	Single cell RNA seq
GE-Impute	L/P	https://github.com/wxbCaterpillar/GE-Impute	Single cell RNA seq
scNTImpute	L/P	https://github.com/qiyueyang-7/scNTImpute.git	Single cell RNA seq
CL-Impute	L/P	https://github.com/yuchen21-web/Imputation-for-scRNA-seq	Single cell RNA seq

(Continues)

TABLE 1 | (Continued)

Name	Access	Direct link	Developed for data type
		Approaches Based on External Knowledge	
GO-Based Imp.	(-)	https://doi.org/10.1093/bioinformatics/btk019	Microarray gene expression
Bubble	L/R+P	https://github.com/CSUBioGroup/Bubble	Single cell RNA seq
SCENA	—	https://doi.org/10.1089/cmb.2021.0403	Single cell RNA seq
deepMC	—	https://doi.org/10.1089/cmb.2019.0278	Single cell RNA seq
DURIAN	—	https://doi.org/10.1093/bib/bbac223	Single cell RNA seq
SCRABBLE	L/R+M	https://github.com/tanlabcode/SCRABBLE	Single cell RNA seq
(TRANS)LATE	L/P	https://github.com/audreyqyfu/LATE	Single cell RNA seq
		Statistical and Miscellaneous Approaches	
BayesImpute	—	https://doi.org/10.1016/j.ymeth.2023.06.004	Single cell RNA seq
scImpute	L/R	https://github.com/Vivianstats/scImpute	Single cell RNA seq
missForest	L/R	https://github.com/stekhoven/missForest	Any type of input
BayesMetab	(-)	https://doi.org/10.1186/s12859-019-3250-2	Metabolome
bayNorm	L/R	https://bioconductor.org/packages/release/bioc/html/bayNorm.html	Single cell RNA seq
CIDR	L/R	https://github.com/VCCRI/CIDR	Single cell RNA seq
G2S3	L/R, F/M	https://github.com/ZWang-Lab/G2S3	Single cell RNA seq
DrImpute	L/R	https://github.com/gongx030/DrImpute	Single cell RNA seq

Note: Access is specified to the best of our knowledge as function (F), source code (C), or library (L), followed by the language being Python (P), Matlab (M), or (R).

SC-KNN Dubey et al. introduced a particularly noteworthy variation of distance-weighted nearest-neighbor imputation of microarray data, which selects suitable neighboring genes based on a combination of K-Means and spectral clustering [33]. Note that such KNN-based methods, especially KNNImpute, have been shown to be most effective for MCAR missing values [32].

KNN-TN Focusing on metabolomic data, Shah et al. proposed to account for the detection limit when standardizing data prior to KNN imputation [33]. Note that such KNN-based methods, especially KNNImpute, have been shown to be most effective for MCAR missing values [32].

MethyLImp/ MethyLImp2 The authors of methyLImp proposed a limited-range linear regression model specifically for DNA methylation data [34]. In later work on methyLImp2, the same authors introduced a chromosome-wise parallelization and a mini-batch approach for large number of samples [35]. Note that the method requires a subset of variables without missing values, which is however typically fulfilled in DNA methylation data.

VIPER Variability-Preserving ImPutation for Expression Recovery (VIPER), models the normalized expression value of a given cell via a hard-thresholded, nonnegative regression problem with expression of other cells as independent variables [36]. These cells are preselected using a penalized regression (L1/L2) regression model.

2Dimpute Zhu et al. introduced 2Dimpute for single-cell RNA-seq data, which first distinguishes biological from technical zeros using the Jaccard-index [37]. It then identifies coexpression signatures in the data, followed by imputation of spurious zeros among these signatures and imputation of the remaining dropout values via KNN-based regression across cells.

CFNI The cluster-directed framework for neighbor-based imputation (CFNI) first performs K-Means clustering of genes, for which it uses initial estimates for the missing values from mean imputation [38]. It then determines the optimal number of gene clusters by a collection of different metrics and subsequently performs neighbor-based imputation (e.g., LLSImpute or KNNImpute) based on a gene set from the cluster of the gene to impute.

GL2P GL2P (global learning with local preservation) iteratively selects the gene with the lowest number of missing values [39]. For each such gene, GL2P then uses pairwise Euclidean distances to compute the similarity to other genes and uses these similarities as inputs to a Gaussian kernel in order to compute the weights for a linear regression model.

3.2 | Matrix Decomposition/Factorization Approaches

Other methods apply mathematical decomposition or factorization to the data matrix in order to obtain quantitative estimates for the imputed data.

SVDImpute In the same publication as the popular KNNImpute method, Troyanskaya additionally proposed an imputation method based on singular value decomposition (SVD). The approach first replaces missing values by the respective gene averages, followed by SVD to compute the first principal components, from which new estimates for the missing values can be obtained via linear regression. This procedure is repeated until convergence.

SVDImpute has been tested extensively for the imputation of MNAR data, however, results seem contradictory yielding either a very positive or poor performance of the algorithm [11, 32].

BPCA The Bayesian Principal Component Analysis (BPCA) builds on top of probabilistic [40] principal components analysis (PCA) and estimates missing values in microarray-based gene expression profiles using a variational Bayes method in an iterative fashion [41]. BPCA has been shown to achieve the best results on MCAR data [28, 32].

Bi-BPCA Driven by the aim to better exploit local structure of the gene expression matrix, Meng et al. proposed a bicluster-based enhancement for the BPCA method. Their two-step method, bi-BPCA, first imputes the expression matrix by BPCA, followed by nearest neighbor-based biclustering to find similar genes and samples [42]. Final estimates for the missing values are then obtained by a second iteration of BPCA on each bicluster individually. Since Bi-BPCA relies on the same principles as BPCA it can safely be assumed that it is similarly suited for MCAR data.

CMF-Impute Based on collaborative matrix factorization (CMF), researchers proposed CMF-Impute, that decomposes the measured gene expression matrix into a product of a cell feature matrix and a gene feature matrix [43]. The authors further employ a nonnegativity constraint to the expression values to ensure biologically meaningful results.

ALRA ALRA (Adaptive Low-Rank Approximation) was explicitly designed to distinguish between biological zeros and dropouts in single-cell RNA-seq data [44]. ALRA computes a low-rank approximation of the expression matrix via SVD and then restores biological zeros by thresholding each gene based

on the low-rank approximation. Note that the method further allows to estimate the number of missed technical zeros after imputation.

3.3 | Neural Network-Based Approaches

Driven by the increasing wealth of data, researchers have leveraged neural networks for imputation of missing values. Many of these approaches rely on autoencoders and are most often designed for single-cell RNA-seq data, where the large number of observations (cells) provides sufficient data for training. However, these methods are reliant on the abundance of training data and generally come at the expense of increased computational costs [45].

AutoImpute Among the earliest methods, Talwar et al. proposed the AutoImpute algorithm [46]. The method employs an overcomplete autoencoder architecture with three fully connected layers that are trained to reconstruct the entries with nonzero counts.

VAE Other researchers proposed to use variational autoencoder (VAE) to iteratively refine missing value estimates and have successfully applied their method to transcriptomic and epigenomic (methylation) data [47]. Note that the authors also proposed a modified version with shift correction to account for detection limits.

IGSImpute IGSImpute, introduced by Xu et al., employs a denoising autoencoder with an instance-wise gene selection layer, that selects contributing genes for the latent space, and a gene–gene interaction layer for estimating cell–cell and gene–gene similarities [48].

ADEPT Although primarily designed to cluster spatial transcriptomic data, the ADEPT method (autoencoder with differentially expressed genes and imputation) also performs imputation of gene expression matrices [49]. The method first constructs a spot graph using K-nearest neighbors, which is then fed into a graph autoencoder to learn a representation of each spot in latent space. ADEPT then selects matrices with differentially expressed genes that it imputes by averaging nonzero expression values across all similar spots as identified from the latent space.

cnnImpute Researchers further proposed to use convolutional neural networks (CNNs) for data imputation [50]. Their approach, cnnImpute, identifies dropout using same mixture model as scImpute (cf. Section 3.5). It then constructs imputed values for sets of target genes with missing values from highly correlated genes using individual CNN models per gene set.

GE-Impute Wu et al. proposed to use graph-embedding-based neural networks [51]. Their method, GE-Impute, utilizes a Skip-Gram model to learn feature representations for cells in a cell–cell similarity network constructed from node2vec. Imputation is then performed based on an average over the expression of neighboring cells.

scNTImpute scNTImpute employs the mixture model introduced in scImpute (see Section 3.5) and leverages a neural topic model to model the distribution of scRNA-seq data [52]. Ultimately, the method computes the cell similarity matrix using the topic mixture per cell and estimates the missing values from expression of neighboring cells.

CLImpute This method, proposed by Shi et al., employs contrastive learning using simulated dropout as augmentation [53]. It leverages self-attention to learn a cell representation matrix and maximizes the similarity between different augmentations of the same cell (and vice versa). For imputation, the method extracts similar cells from the network and fits a least-squares regression model from these to impute the missing data under consideration.

3.4 | Approaches Based on External Knowledge

Especially for single-cell sequencing data, for which dropout is particularly prominent, researchers proposed a large number of methods that incorporate additional knowledge, for example, relationships to sample-matched bulk data. A selection of such methods is presented in the following. Note, however, that these methods rely heavily on the respective external data (e.g., bulk data or databases) and may hence reproduce their potential biases.

GO-Based Imp. Among the earliest works, Tuikkala et al. [54] proposed to incorporate gene ontology to compute the semantic dissimilarity between genes and integrate this with the classical expression level distance (i.e., Euclidean distances) to select the genes for imputation with LLS imputation (LLSImpute) as described in Section 3.1.

Bubble Bubble first identifies dropouts the same way as BayesImpute (cf. Section 3.5). It then uses a four-layer autoencoder with ReLu activation, which it trains to minimize the reconstruction error and to maximize the match to given bulk RNA-seq data [55].

SCENA Single-cell RNA-seq Correlation completion by ENsemble learning and Auxiliary information (SCENA) aims to directly correct the gene–gene correlation matrix for missing values in the raw data [56]. The method allows to integrate further sources of information, for example, gene networks or other RNA-seq data, by estimating the final corrected correlation matrix from an ensemble of these inputs.

deepMC The authors of deepMC formulate imputation as a matrix factorization method based on a three-layered neural network [57].

DURIAN Given matching single-cell and bulk RNA-seq data, DURIAN (deconvolution and multitask-regression-based imputation) first estimates the bulk cell composition via deconvolution of the bulk data using the single-cell data [58]. Imputation is then performed using a singular-value thresholded scheme constrained by a pseudobulk reference and the process is repeated for multiple iterations.

SCRABBLE SCRABBLE (single-cell RNA-seq imputation constrained by bulk RNA-seq data) requires consistent cell populations between single-cell and bulk RNA-seq data [59]. The algorithm is based on matrix regularization and optimizes the deviation of the corrected matrix error from the input, its nuclear norm and the deviation between bulk- and single-cell data.

(TRANS)LATE Badsha et al. proposed the LATE (Learning with Autoencoder) model, which employs a deep autoencoder that minimizes the reconstruction error on the measured data [60]. Missing values are treated as zeros and are added to a pseudocount of 1 to allow for log-transformation. A model variant, TRANSLATE, first trains this encoder on a reference gene expression dataset and then transfers the learned weights to the single-cell RNA-seq data.

3.5 | Statistical and Miscellaneous Approaches

A small number of notable imputation methods that were revealed by our systematic literature search could not be grouped into any of the aforementioned categories. These techniques often employ a specific statistical approach or follow an entirely different concept and are hence briefly described in the following.

BayesImpute After preprocessing of the raw count matrix, BayesImpute performs dimensionality reduction and clustering of cells [61]. The algorithm then distinguishes between dropout and biological zeros by thresholding based on the median expression rate and coefficient of variation per cell subpopulation and imputes the dropouts using the posterior mean from Bayesian estimation.

scImpute Li et al. proposed scImpute, which first identifies subpopulations in single-cell RNA-seq data by spectral clustering in PCA-transformed, preprocessed, and normalized count space [62]. The method then computes respective probabilities for dropouts/biological zeros using a two-component model with a Gamma distribution for dropouts and a normal distribution for true expression and finally performs imputation of the dropouts by nonnegative least squares regression.

missForest Stekhoven et al. proposed to leverage the RandomForest [63] algorithm for data imputation, especially for mixed categorical and continuous data [64]. The algorithm iteratively refines the missing data estimates starting from a simple method such as mean imputation has been found to perform well for proteomics and metabolomics. missForest has been extensively validated and has been shown to best function for MCAR missing values [28, 32].

BayesMetab Explicitly designed to distinguish between MNAR missing values caused by detection limits and other dropouts in metabolomics, BayesMetab uses a Bayesian approach for imputation, based on a Markov Chain Monte Carlo method to estimate the posterior distributions [65].

bayNorm bayNorm assumes a negative binomial distribution for the original counts of each gene to estimate the posterior distribution in each cell based on the observed count data of that gene [66]. Note that the authors found the method to be also capable of reducing batch effects on data and that it can be used for both single and multiple imputation.

CIDR As part of its clustering approach, the CIDR (clustering through imputation and dimensionality reduction) method fits a shared logistic model via nonlinear least squares regression to the dropout probability as function of the true expression [67]. Here, dropout candidates are identified from a cell-specific expression threshold and are imputed from a weighted average based on the estimated probability.

G2S3 G2S3 (sparse gene graph of smooth signals), proposed by Wu et al., learns a weighted gene graph across cells via primal dual approaches [68]. It then imputes the data by one or multiple matrix products with a lazy random walk matrix.

DrImpute In their work on single-cell RNA-seq data imputation, Gong et al. proposed DrImpute, that first repeatedly clusters cells and computes average expression values from cells of the same cluster, followed by obtaining the missing value estimates as mean of these estimates [69].

4 | Data Integration Methods

The systematic literature collection procedure described in Section 2 yielded 32 representative data integration methods, which can be broadly distinguished into five groups based on their core principles: Model-based, neighbor-/clustering-based, reference-based, and machine-learning-based approaches, as well as other matrix-operation-based methods. Figure 4 highlights the respective categorization of the considered methods into these five groups and Table 2 reports their corresponding access options and primary data types.

TABLE 2 | Comprehensive summary of considered data integration methods.

Name	Access	Direct Link	Developed for data type
		Model-based Approaches	
ComBat	F/R	https://rdrr.io/bioc/sva/	Microarray
Re-ComBat	L/P	https://github.com/BorgwardtLab/reComBat	Gene expression
M-ComBat	C/R	https://github.com/SteinCK/M-ComBat	Microarray gene expression
ComBat-seq	L/R	https://github.com/zhangyuqing/ComBat-seq	RNA seq
limma	L/R	https://www.bioconductor.org/packages/release/bioc/html/limma.html	Microarray data
Q-ComBat	—	https://doi.org/10.1371/journal.pone.0156594	Microarray transcriptome
Longitudinal ComBat	L/R	https://github.com/jcbeer/longCombat	Multi-scanner imaging
GMM ComBat	L/P	https://github.com/hannah-horng/generalized-combat	Radiomics
Nested ComBat	L/P	https://github.com/hannah-horng/generalized-combat	Radiomics
OPNested ComBat	L/P	https://github.com/hannah-horng/opnested-combat	Radiomics
		Neighbor- /Clustering-Based Approaches	
MNN	F/R	https://bioconductor.org/packages/release/bioc/html/scran.html	Single cell RNA seq
Scanorama	L/P	https://github.com/brianhie/scanorama	Single cell RNA seq
Harmony	L/R	https://github.com/immunogenomics/harmony	Single cell RNA seq
SCIBER	L/R	https://cran.r-project.org/web/packages/SCIBER/	Single cell RNA seq
		Reference-Based Approaches	
IRS	—	https://doi.org/10.1074/mcp.M116.065524	TMT—labelled proteome
BRIDGE	L/R	https://github.com/qingxiaa/brg	Microarray data
COCONUT	C/R	https://wiki.khatrilab.stanford.edu/sepsis	Same as ComBat
BESC	L/R	https://bio.tools/besc	Microarray data
SMNNiSMNN	L/R	https://github.com/yycunc/SMNN	Single cell RNA seq
		Machine-Learning-Based Approaches	
NormAE	L/P	https://github.com/luyiyun/NormAE	Metabolome
AD-AE	C/P	https://gitlab.cs.washington.edu/abdincer/ad-ae	Gene expression
Dual AD-AE	C/P	https://github.com/LaraCavinato/Dual-ADAE	Radiomics
Procrustes	L/P	https://github.com/BostonGene/Procrustes	RNA seq
scGAMNN	—	https://doi.org/10.1109/JBHI.2023.3311340	Single cell RNA seq
BERMAD	L/P	https://github.com/zhanglabNKU/BERMAD	Single cell RNA seq
HDMC	L/P	https://github.com/zhanglabNKU/HDMC	Single cell RNA seq
		Other Matrix-Operation-Based Approaches	
HarmonizRBERT	L/R	https://www.bioconductor.org/packages/release/bioc/html/HarmonizR.html https://www.bioconductor.org/packages/devel/bioc/html/BERT.html	Validated for microarray gene expression, metabolomics and proteomics
scBatch	L/R	https://github.com/tengfei-emory/scBatch	RNA seq
CCA (SEURAT)	L/P	https://satijalab.org/seurat/	Single cell RNA seq
LIGER	L/R	https://github.com/welch-lab/liger	Single cell RNA seq
BEclear	L/R	https://bioconductor.org/packages/release/bioc/html/BEclear.html	DNA methylation

Note: Access is specified to the best of our knowledge as function (F), source code (C), or library (L), followed by the language being Python (P), Matlab (M), or (R).

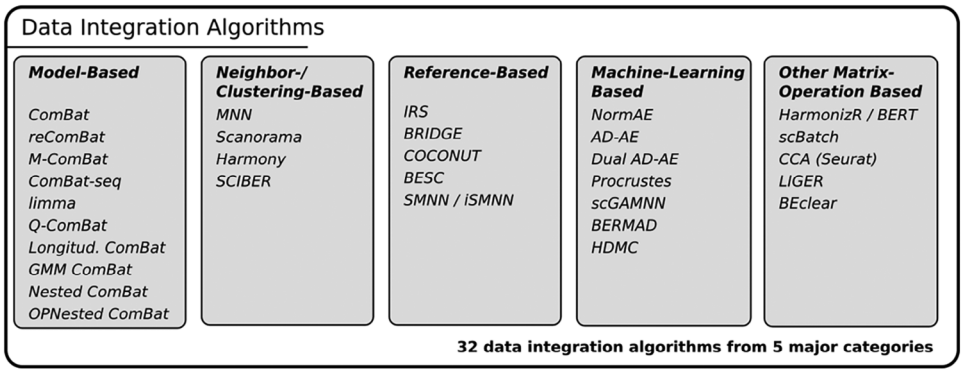

FIGURE 4 | Overview over the data integration methods discussed in this review.

4.1 | Model-Based Approaches

Model-based data integration methods primarily represent the batch effects via location-and-scale (L/S) models and estimate their parameters (typically mean and variance) from the observed data. When employing this category of data integration methods, it is imperative to understand which data type and model may be applicable. Using the wrong underlying model can lead to a bias.

ComBat Among the most popular methods for data integration, ComBat [70] employs the parametric empirical Bayes method to estimate additive and multiplicative batch effects via iterative refinement in a computationally efficient manner. The algorithm can also employ non-parametric priors and was specifically designed for batches with small sample sizes. Among other applications for the integration of microarray-data [71], radiomic features [72] and RNA-seq measurements [73], Petralia et al. used ComBat to integrate 23 batches of pediatric brain tumors from triple mass-spectrometry with tandem mass tags [74].

reComBat ComBat relies on an initial linear regression step for the standardization of input data. The authors of regularized ComBat (reComBat) argue that the design matrix of independent variables in this regression step may become singular when integrating large-number of batches [75]. They hence propose to use ElasticNet regression instead to ensure validity of the results and tested this approach on microarray and bulk RNA-seq data.

M-ComBat With M-ComBat, researchers proposed to reformulate ComBat as to adjust all data on a predefined optimal batch, for example, with superior data quality [76]. To date, such an option is provided by the official ComBat implementation as well.

ComBat-seq Zhang et al. identified the assumption of normal distributions in parametric ComBat as the main caveat for its applicability to count data [77]. They hence propose a negative binomial regression model specifically for count matrices of RNA-seq studies to better handle skewed input data and outliers.

limma The *limma* library by Ritchie et al. offers a linear regression model for batch effects, that allows to center each variable to its respective grand mean across batches [78]. Although this only corrects the location (L) and neglects the scale (S), this method has successfully been applied to different omic-types, such as proteomics [79] and methylomics [80]. Since it is based on well-established linear regression, this method requires very little runtime in practice.

Q-ComBat In a comparison of different data integration methods for longitudinal gene expression data, Müller et al. found quantile normalization followed by ComBat normalization (Q-ComBat) to perform best [81].

Longitudinal ComBat Q-ComBat considers each measurement as independent, which was identified as a weakness by the authors of longitudinal ComBat [82]. They proposed an adaptation of ComBat's L/S model for longitudinal data that accounts for the inherent correlation between acquisitions of the same sample during the course of a longitudinal study and validated their method on radiomic features generated from functional MRI images.

GMM ComBat Horng et al. assumed that confounding variables that cause unwanted variations may sometimes be unknown.

They hence suggested to identify groups by repeatedly fitting a two-component Gaussian Mixture Model (GMM) to the observed data and correcting for the identified groups as batches. The method was successfully validated on radiomic data [83].

Nested ComBat In the same publication [83], the authors of GMM ComBat proposed Nested ComBat that allows to sequentially remove any number of batch effects by repeated application of ComBat. The respective order of batch effects is optimized exhaustively by minimizing the number of batch-effect afflicted features and individual batch effects may be corrected repeatedly.

OPNested ComBat In later work by the same group, Nested ComBat was combined with Gaussian Mixture Models. The new model, OPNested ComBat, iterates over all possible permutations of batch effects and uses the mixture models to either identify further batch effects or to identify covariates to preserve during batch-effect correction [84].

4.2 | Neighbor- /Clustering-Based Approaches

Typical data integration tasks arise during the consideration of a specific biological condition or biological system. Hence, the batches to integrate typically share distributional properties, for example, common cell types occurring in independently acquired single-cell RNA-seq datasets. Neighbor- and clustering-based approaches leverage these similarities by identifying matched observations (e.g., cells) or groups of observations (e.g., cell types) across the cohorts. Note that these matches are often based on unsupervised heuristics (e.g., a low pairwise distance) and that reference-based methods may offer a more supervised and exact approach.

MNN In pioneering work by Haghverdi et al., it was proposed to compute batch-effect correction vectors for single-cell RNA-seq data based on mutual nearest neighbors (MNNs) [85]. The method first standardizes the data using a cosine normalization, followed by MNN estimation. For each pair of batches, local linear batch-effect correction vectors are then obtained from matched cells using Gaussian smoothing.

Scanorama The Scanorama algorithm for single-cell RNA-seq data first performs randomized SVD for dimensionality reduction, followed approximate nearest neighbors search based on hyperplane locality sensitive hashing and random projection trees [86]. In contrast to MNN, which iteratively aggregates batches in a pairwise fashion, Scanorama is insensitive to the input order and is computationally very efficient. Note that the algorithm can both perform data integration in the low-dimensional space, as well as compute batch-effect corrected data in the original high-dimensional space.

Harmony The Harmony algorithm employs iterative batch-effect correction of single-cell data [87]. It first projects the data to a lower-dimensional embedding by PCA, followed by repeated maximum diversity clustering (Cosine-distance-based K-Means with penalization of clusters with low batch-diversity) and a mixture model based linear correction step until convergence.

SCIBER The single-cell integrator and batch-effect remover (SCIBER) method aims to correct batch effects with respect to a predefined reference batch. It first performs K-Means clustering in each batch to identify clusters and matches these across batches based on differential gene expression. The expression of each cell is then decomposed into a mixture of these matched groups, followed by corresponding transfer to the reference batch.

4.3 | Reference-Based Approaches

In contrast to the aforementioned approaches, in which anchor points (neighbors and shared clusters) are constructed algorithmically, reference-based approaches employ external knowledge about technical or biological properties shared by the batches. This also enables researchers to combine batches with more unequal distributions of biological groups or conditions. Importantly, these methods strictly mandate the existence of identical or at minimum very similar samples across all batches to yield reliable data integration and researchers must plan for a corresponding study design.

IRS Plubell and Wilmarth proposed to include one or multiple standards (internal reference samples, IRS) in each plex of mass spectrometric measurements with tandem mass tags [88]. Normalization factors for each variable are computed as to scale the reference mean per batch to their geometric average across batches. This technique is currently common practice in mass-spectrometry based proteomics and has, for example, been used by Krug et al. for the investigation of treatment-naïve primary breast cancers [89].

BRIDGE With a similar approach researchers proposed BRIDGE, an empirical Bayes method for batch-effect correction in longitudinal (esp. gene expression) studies [90]. BRIDGE employs a modified L/S model, the parameters of which are estimated from a subset of shared (bridging) samples between different timepoints.

COCONUT In a study on the classification of bacterial and viral infections, Sweeney et al. introduced a novel variation of ComBat,

COCONUT, that utilizes user-defined samples of similar biological properties to infer parameters of COMBAT's L/S model that are subsequently transferred for batch-effect correction of the remaining samples [91].

BESC Assuming that potential sources of batch effects are associated with a unique signature, batch-effect signature correction (BESC) aims to minimize the squared residuals between the measurements and a linear combination of such predefined signatures. The signatures are represented by unit vectors with zero mean and can be constructed from biological or technical replicates across batches. BESC has been validated on microarray gene expression data.

SMNN and iSMNN While most methods based on MNNs and matched clusters are based on heuristics for matching observations (e.g., cells or cell clusters) across batches, SMNN allows users to specify marker genes and respective cell types in single-cell RNA-seq data to ensure correct inter-batch matching before adjusting the data based on MNNs [92]. These neighbors are computed on the unadjusted input data, which was identified as main caveat of SMNN, such that the same authors proposed iterative SMNN (iSMNN) in later work [93]. iSMNN repeatedly computes MNNs and corrects for the respective batch effects.

4.4 | Machine-Learning-Based Approaches

Over the last decade, the increase of both general data availability and batch sizes has allowed researchers to make use of neural networks and other machine-learning methods to develop novel data integration techniques. Typical methods are often based on autoencoders and adversarial learning, where the former represents a pair of stacked neural networks that first project the input data to a low-dimensional latent space (*encoder* network) and then reconstruct the input (*decoder* network) with minimal loss of information. While autoencoders thus represent a specific network architecture, adversarial learning refers to a training procedure in which multiple neural networks (typically two) compete to achieve adversarial objectives (e.g., reconstructing the measured data from a latent space and being unable to predict confounders from this latent space). Note that many machine-learning methods are primarily trained to remove batch effects in the latent space, which hence often needs to be used for any downstream tasks. Similar to the machine-learning based approaches for imputation, data integration methods from this category generally require large cohorts and sufficient computational resources.

NormAE Among the earliest methods, normalization autoencoders (NormAE) for batch-effect removal in liquid-chromatography-based mass spectrometry for metabolomics were proposed [94]. NormAE uses an adversarial classifier, that is, trained to predict the batch of origin and other known sources of unwanted variation (e.g., injection order) from the latent representation, while the encoder network is trained to maximize the loss of the classifier. The decoder network is trained to recover the original peak intensities from the latent representation given the labels for batch and the other confounders.

AD-AE Motivated by the goal of generating confounder-robst and generalizable latent representations, AD-AE, an adversarial deconfounding autoencoder, was introduced. Here, the adversarial classifier is trained to predict one or multiple confounders from the latent representation [95]. In contrast to NormAE, the decoder does not utilize the confounder label to reconstruct the input. AD-AE was validated on microarray- and sequencing based transcriptomics.

Dual AD-AE More recently, Cavinato et al. built upon AD-AE by proposing a dual AD-AE model specifically for radiomics, in which two independent adversarial networks are used for the prediction of center and scanner [96].

Procrustes In addition to the aforementioned autoencoder-based approaches, Procrustes leverages linear correlation between expression of individual genes and coexpression of specific genes by using Elastic-Net regression for computing integrated data based on either individual or multiple genes [97].

scGAMNN Motivated by the success of nearest-neighbor based methods for integration of single-cell data (see below), researchers combined such techniques with autoencoder-based data integration. scGAMNN, in particular, trains a joint autoencoder based on graph convolutional network (GCN) layers for the expression matrix and the MNN adjacency matrix [98]. Of note, the method uses an additional distance-based loss for the MNNs in latent space and does hence not require any adversarial networks.

BERMAD BERMAD, as introduced by Zhan et al., is specifically designed to avoid over- or under-correction of batch effects in single-cell RNA-seq data. The method trains an independent autoencoder per batch and additionally optimizes a transfer loss (maximum mean discrepancy) between the most similar cell clusters of the batches to enforce matched output distributions.

HDMC Hierarchical distribution matching and contrastive learning (HDMC) combines the traditional reconstruction loss from autoencoders, a contrastive loss in the latent space (reducing distance of similar clusters and vice versa for noisy clusters) and an adversarial loss from the prediction of the batch of origin after an adversarial layer [99]. Note that

the strength of the contrastive loss is gradually increased to improve stability during early training epochs.

4.5 | Other Matrix-Operation-Based Approaches

Although all data integration methods operate on matrices of variables and observations, some approaches could not be grouped into any of the prior categories and are hence reported in the following.

HarmonizR and BERT Since many of the previously reported data integration methods are not primarily designed for the integration of data with large amounts of missing values (especially those introduced by the process of data integration itself), our group introduced the HarmonizR algorithm, that employs matrix dissection to identify eligible sub-matrices for downstream correction with the aforementioned ComBat and limma methods [100]. Intrinsic loss of numerical values by the matrix dissection and limited support of user-defined references and covariates is a central shortcoming of HarmonizR, such that we further proposed batch-effect reduction trees (BERT) to overcome these limitations while still offering the same tolerance to missing values. Note that BERT is so far only available as a software publication [101]. BERT and HarmonizR are particularly suited for mass-spectrometry-based and microarray-based *omic* data (e.g., proteomics/metabolomics and transcriptomics, respectively).

scBatch The scBatch algorithm employs linear transformations to a count matrix from single-cell RNA-sequencing as to yield a batch-effect free Pearson-correlation matrix, which is computed using QuantNorm [102, 103]. scBatch is primarily designed for balanced study designs.

CCA (Seurat) Originally, the popular Seurat package for analysis of single-cell data employed canonical correlation analysis (CCA) to learn shared gene correlation structure, followed by a comparison to a principal component analysis to identify and optionally remove subpopulations with expression patterns that are not well explained by this correlation structure [104]. The method then uses dynamic time warping to iteratively integrate the data per batch by aligning the canonical correlation vectors. Note that the method integrates the data into a conserved low-dimensional space. Of interest, in a later version of Seurat (v3), the method was extended to utilize MNNs to compute correction vectors based on cells with similar biological state.

LIGER LIGER, as introduced by Welch et al., employs integrative nonnegative matrix factorization to express each observation by dataset-specific and shared factors [105]. The approach then constructs a shared factor neighborhood graph in the factor space and normalizes the factor loadings to a predefined reference.

BEclear Originally designed for DNA-methylation data, BEclear aims to correct only batch-effect afflicted genes, for the identification of which it employs a Kolmogorov-Smirnov test to compare distributions between batches [106]. It then applies a cutoff to the difference of gene medians, selects batches with strong batch effects and corrects the respective identified genes by a latent factor model based on matrix factorization.

5 | Evaluation Methods

Reliable validation metrics/methods are important to researchers and practitioners alike to shed light on the success of both data integration and imputation. While most evaluation techniques for the former are capable of estimating output quality in both characterization of novel approaches as well as during research-specific data integration tasks, most metrics for data imputation require knowledge of the correct values of the missing data. This is however most commonly not the case in real-world research projects, such that researchers often rely on validation with artificially introduced missing values or on recommendations obtained from large-scale comparative studies to decide on a suitable imputation method.

For data integration, the most common evaluation techniques represent either manually analyzed visualizations of the integrated data, quantitative approaches borrowed from the fields of statistics and machine-learning, as well as task-specific performance measures, compare Figure 5.

Visualization, in particular, represents the most intuitive category. Most commonly, researchers project the data (raw and integrated) into a two- or three-dimensional space using techniques such as PCA, t-distributed stochastic neighborhood embedding (t-SNE) [107] or uniform manifold approximation and projection (UMAP) and visualize the observations using scatterplots [108]. Improved overlap between biologically similar groups of observations from different input batches indicates effective batch-effect

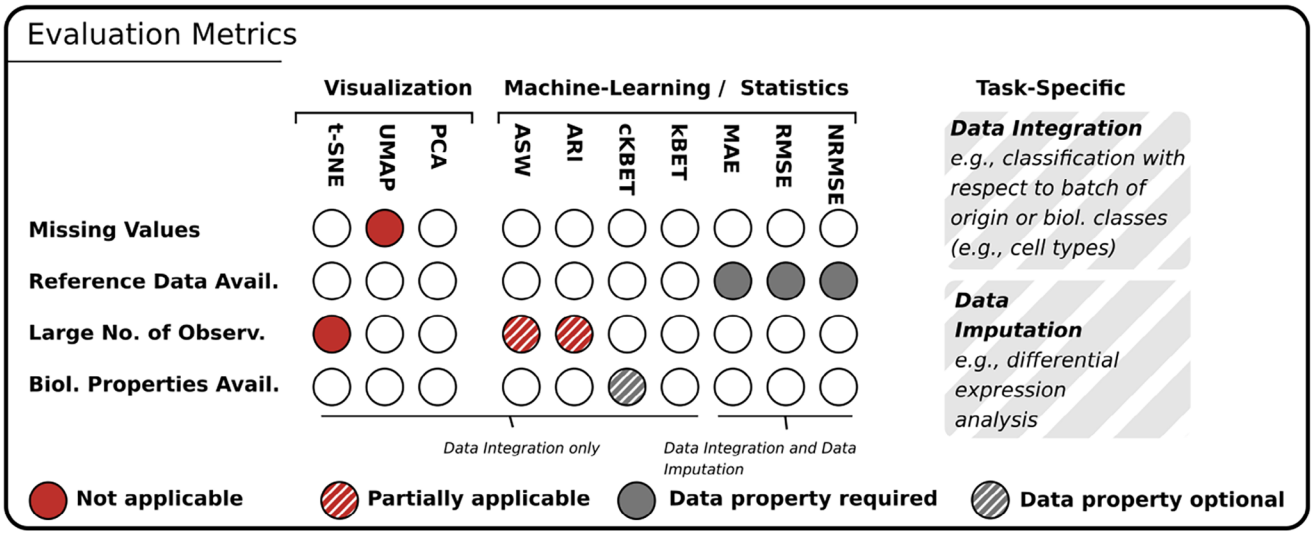

FIGURE 5 | Overview over representative evaluation metrics for data integration and data imputation, as well indications for requirements and limitations based on literature and experiences of the authors.

correction. PCA is employed particularly often due to its computational efficiency, as well as to the inherent quantification of the explained variance, which can aid researchers to estimate the reliability of the visualization. In contrast, the t-SNE method does computationally not scale well to large number of observations (e.g., cell types or samples). It is of interest that UMAP generally requires complete data, whereas t-SNE and PCA can both be computed based on pairwise operations. In particular, t-SNEs can be computed from arbitrary distance matrices (such as from pairwise Euclidean distances), whereas the nonlinear iterative partial least squares (NIPALS) algorithm [109] can be used for PCA. As an example, the latter was employed by Godbole et al. for both validating data integration with HarmonizR and for the identification of medulloblastoma types in mass-spectrometry based proteomics [110].

Despite the common use of visualization techniques, they lack a quantification of data integration success, which might, for example, be necessary to decide on an optimal strategy before proceeding to downstream analysis. The machine-learning community has developed several such metrics for clustering (i.e., when only batch/biological labels are known) or regression (i.e., if the correct numerical values are known e.g., from technical replicates). Among the former, researchers often compute the average silhouette width (ASW) with respect to both batch and/or biological condition, which is computed as the difference between the mean intra-cluster distance and the mean nearest-cluster distance divided by the maximum of the two [111]. Here, an ASW decrease for the batch label indicates successful data integration and vice versa for the biological condition. In a similar fashion, observations can be manually clustered before and after data integration to compute the Rand index, which quantifies the agreement to predefined groups (e.g., the biological conditions or batch of origin). Higher Rand scores indicate better clustering by the respective groups. Most studies report the adjusted Rand index (ARI), which represents the Rand index corrected for chance [112]. Furthermore, the kBET metric was introduced that employs a χ^2-based statistical test for "mixedness" in fixed-size neighborhoods of single-cell data [113]. A low average rejection rate indicates good data integration. Recently, the cKBET method was proposed that explicitly considers cell type labels (known or inferred by clustering), which can be particularly suited for data integration tasks with unequal distributions across batches [114]. Note that the ASW, the ARI, and all regression metrics are well suited for incomplete data given a suitable distance metric for clustering and distance computations (e.g., pairwise Euclidean distances). Importantly, each of these methods quantifies different aspects of the raw and integrated data and researchers should consider more than one. In particular, a recent benchmark study on batch-effect correction in single-cell RNA-seq data used the ASW, ARI, kBET and LISI [5] for quantification and different methods performed best per metric, although some methods performed generally favorable compared to others [115]. Given pairs of matching samples across batches (e.g., references from BRIDGE, IRS or BERT), scientists and practitioners can employ regression metrics for the evaluation of data integration such as the mean absolute error (MAE), the root mean square error (RMSE) or the normalized root mean square error (NRMSE). These methods can also be used to validate data imputation, which however requires knowledge of the true value of the dropout data. In typical experimental settings, the latter may arise either from repeated measurements of the same sample (such that randomly missing data from one measurement may have been quantified in another measurement) or from simulation studies, where nonmissing data are set missing (e.g., using MCAR or MNAR mechanisms) and subsequently imputed with the aim to recover the original values.

Finally, both data integration as well as imputation methods may be also evaluated by the performance of downstream tasks on the processed data, which requires domain-specific knowledge of the researcher. For data integration, in particular, researchers often employ classification of observations (i.e., predicting either the original batch or any biological condition of interest) and monitor the change in suitable classification metrics [116] (e.g., accuracy, balanced accuracy, AUC score) caused by batch-effect correction. Examples for other specific tasks include segmentation in

radiomics (quantified for instance by intersection over union or Dice metric) or quality of coexpression networks [73] (e.g., quantified by matching gene ontology terms) and researchers may define further tasks depending on the considered problem. Similarly, successful data imputation should also yield improved downstream analysis, such as differential expression analysis, clustering and cell trajectory analysis for single-cell RNA-seq data.

6 | Discussion and Recommendations

With the increasing data availability arising from technical improvements to high-throughput measurement techniques and the establishment of FAIR data principles [117], data integration is becoming a key part of daily research and even clinical applications (e.g., DNA-methylation-based classification of tumors in the central nervous system [118]). Yet, the trend for high throughput and single-cell technologies also introduces large number of missing values to the data—A problem that is often additionally amplified by data integration, since genes or other features might not have been quantified in each batch. Although these two aspects (data integration and missing values) are interconnected, previous review papers have typically focused on one or the other and this work complements the field by providing a comprehensive and integrated discussion of both data integration and imputation. In particular, we identified four primary groups among the qualified 262 documents (cf. Section 2 and Supporting Information) and systematically address all these groups within this review.

Unresolved batch effects may affect study results and limit the validity of statistical analyses. For example, the Mouse ENCODE consortium studied gene expression data of mice and humans and found that samples primarily grouped by species and not by tissue [119]. This surprising finding was however demonstrated to be caused by batch effects and study design in later work by Gilad et al., who used ComBat to integrate the data and recover the expected clustering by tissue [120]. On the contrary, as highlighted by Goh et al., batch-effect correction can also over-correct data leading to biased, overly confident or exaggerated differences between cohort subpopulations (e.g., biological groups) [7, 121]. This work introduced 32 qualified data integration methods from five major categories, as well as three relevant groups of techniques for their evaluation. The aim was to provide an extensive overview over all relevant classes of methods, although not every method could be integrated into the manuscript for reasons of brevity (including e.g., RUV and SVA as well their adaptations for sequencing data [122–125], resPAN [126] or BERMUDA [127]).

In two benchmark papers on batch-effect correction methods for single-cell RNAseq data, Harmony was highlighted as the generally best-performing method and might hence represent a suitable starting point for further analysis [115, 128]. Tran et al. further recommended to test LIGER and Seurat (v3) if the results of Harmony were not satisfactory. For microarray-based gene expression studies, authors [129] have recommended ComBat for data integration and variations of this approach (see above) that allow to further specialize the respective L/S model to the considered problem. In a review paper on data integration of imaging data [130], ComBat and its variations were also recommended for the harmonization of radiomic features, especially for small to moderate sample sizes and in longitudinal studies.

Generally, choosing an optimal method for a specific data integration task at hand is challenging. Users must carefully choose a suitable algorithm for the data at hand, for example, considering not only the specific *omic* type, but also the distribution of groups across batches and the study design (longitudinal or cross-sectional, bulk or single-cell), as well as the expected missing value types. Especially in mass spectrometric proteome measurements, the multitude of developed techniques (e.g., BioIDs [131], multiplexing, DIA/DDA measurements) presents various distinct underlying assumptions, leading to different methods being of relevance for the data processing, a detailed discussion of which is however beyond the scope of this cross-omic review. The introduced methods from Section Evaluation Methods and the accompanying description of their merits and demerits may aid researchers deciding on an appropriate method. Interactive tools such as DBnorm (for metabolomics), proBatch (for mass-spectrometry-based proteomics) or malbacR (mass-spectrometry based lipidomics, metabolomics, proteomics and nuclear magnetic resonance data, especially for integration with pmartR [132]) can aid users with less programming experience in quickly testing a broad range of methods [132–135]. Although successful data integration increases the number of considered samples and leads to improved and reliable research outcomes, the correct application of suitable batch-effect correction algorithms is challenging, and their respective results need to be rigorously validated by a mixture of the methods described above. Individual metrics solely capture aspects of the corrected data and an integrated consideration of multiple methods may better represent the whole picture [115]. If applicable, biological or technical references (e.g., replicates) may provide a particularly unbiased estimate for success of batch-effect correction. Where appropriate, researchers should additionally aim to minimize any confounding effects (e.g., class imbalances) by study design.

It is important to note that data integration may amplify the prevalence of missing values, since variables may be missing for entire batches. This particular missing data pattern represents a major challenge for many data integration methods (in particular also for the popular ComBat method), and authors have hence advocated against these methods for mass-spectrometry, that often exhibits this specific type of dropout [134]. To date, only few data integration algorithms are explicitly designed for missing data, although some established methods are conceptually extendable to incomplete data either by employing pairwise operations (i.e., using shared subsets of quantified variables for each pair of samples) such as the IRS method, or by manually removing variables with insufficient data in any of the batches via listwise deletion (e.g., for ComBat). Package developers should aim to account for this to make algorithms more easily applicable to incomplete data (i.e., without manual package modification by the applying researcher). In contrast, HarmonizR and BERT leverage matrix dissection and hierarchical algorithms respectively, to directly extend the applicability of ComBat and limma to arbitrarily incomplete data and they may hence represent suitable and easy-to-use methods for highly incomplete datasets (e.g., mass-spectrometry based proteomics or metabolomics). For

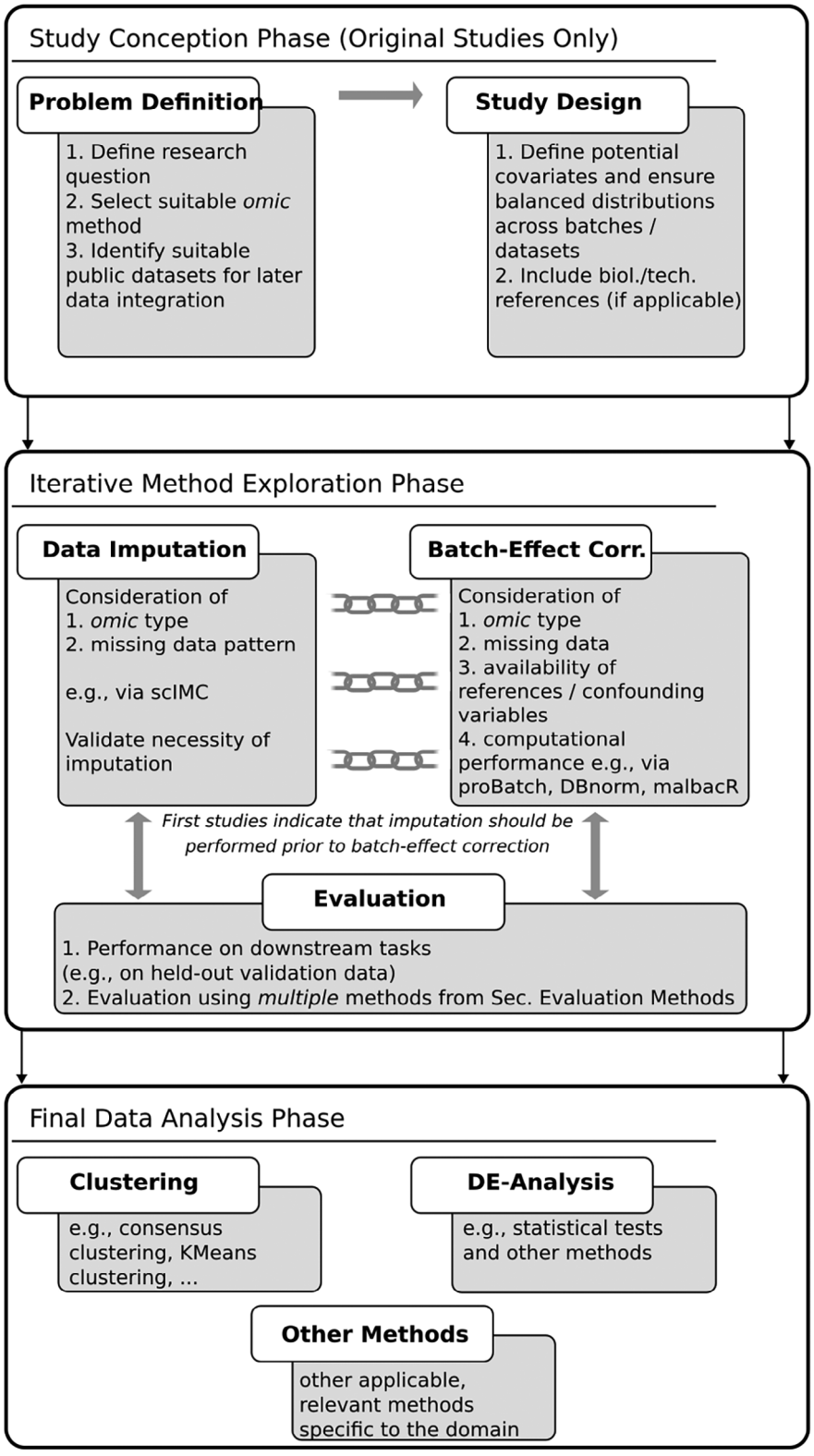

FIGURE 6 | The proposed three-step workflow for integrated consideration of batch-effect correction and missing value imputation from study conception phase (top), to the method exploration phase (center) and the final data analysis step in which practitioners apply domain-specific data analyses methods to the final batch-effect corrected and imputed data (ellipses at bottom, nonexhaustive list). Note that the selected data imputation and batch-effect correction methods are iteratively refined to optimize the selected evaluation metrics.

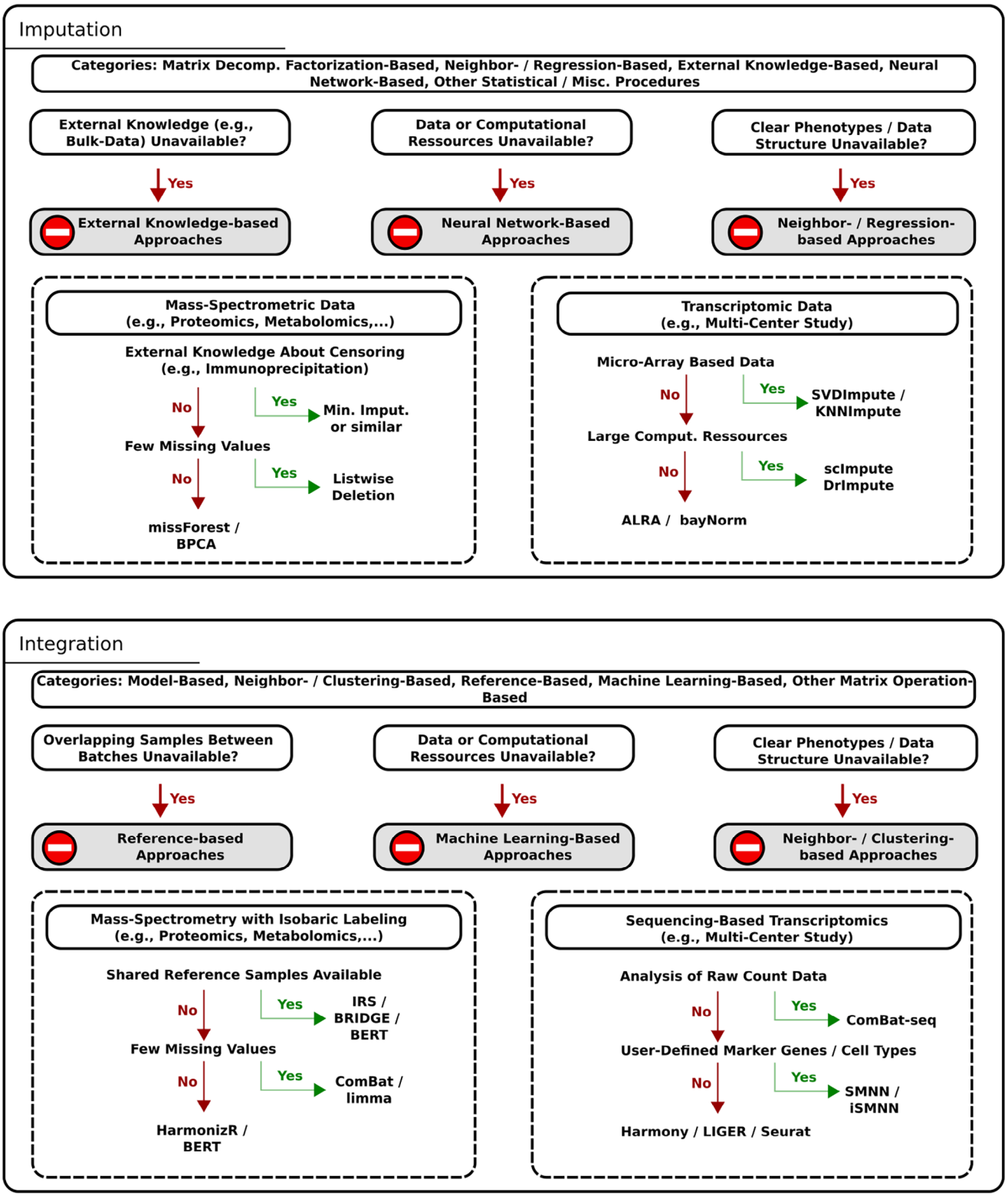

FIGURE 7 | Top: Imputation methods and appropriate criteria for elimination of methodological categories as described in this paper, as well as decision trees with specific recommendations for representative data types and study designs. Bottom: Corresponding panels for data integration methods.

example, Navolić et al. used HarmonizR to integrate incomplete data from ablation-based spatial proteomics of the embryonic mouse head [136]. With respect to our newly defined statistical terminology for batch effects in the context of incomplete data (cf. Section 1.1), all methods discussed in this section aim to create FBE data. In conjunction with suitable data imputation methods, however, they may as well be used to create CFBE data.

From this perspective, this review also introduces 37 representative imputation methods from qualified literature revealed by the

systematic search described in Section 2. These methods can be grouped into five major categories and span multiple major *omic* types. Similar to the data integration methods discussed above, not all methods and reviews could be included into this review for the sake of brevity and inclusion criteria (e.g., SAVER [137] and PIMMS-VAE [138], as well as refs. [139–141]).

Although multiple works have found that data imputation needs to be applied with care, many downstream data analysis methods or data preprocessing techniques (e.g., the ComBat method for batch-effect correction) require complete data and hence necessitate data imputation. Note that it is generally recommended in literature to filter each batch to variables that were quantified in at least 50–80 % of all samples before missing value imputation, which further reduces the observed numerical values per batch [11, 142]. Kokla et al. compared nine different imputation methods for mass-spectrometry based metabolomics and found missForest-based imputation to perform best [32, 143]. Similarly, Bramer et al. recommended missForest-based imputation for isobaric labeling-based shotgun proteomics, but elaborated that imputation should be applied with consideration to data with small sample sizes [144]. For label-free quantitative proteomics, Lazar et al. argue that the optimal imputation strategy should be selected based on the missing value mechanism and develop guidelines for the choice and application of imputation in proteomics [143]. For single-cell RNA-seq data, Hou et al. compared 18 different imputation method with respect to a diverse set of downstream tasks and found that the considered methods exhibited largely varying performance for each of the tasks [145]. For mass-spectrometry based metabolomic data Wei et al. compared nine different imputation methods and found that Quantile Regression Imputation of Left-Censored data showed the best performance and determined different types of missing values for different metabolomic analyses (MCAR/MAR in nontargeted GC/MS; MNAR in targeted LC/MS metabolomics) [11]. However, we generally found only few studies that explicitly account for different missing value types in (characterization) data imputation and thus advocate, that researchers should carefully select and validate suitable imputation methods for their data. For this, software tools such as the single-cell Imputation Methods Comparison platform (scIMC), may aid them to rapidly explore a broad range of methods [146].

To date, only few studies have investigated the mutual effects of data integration and missing value imputation. Hui et al. compared different mean imputation strategies (from all data, from the same batch, from a different batch) prior to batch-effect correction and found that batch-sensitive imputation methods yield better signal-to-noise ratios than batch-naïve imputation [147]. In later work, the same authors further elaborate that batch-effect associated missing values represent an additional challenge for imputation and suggest iterative batch-wise imputation as potential solution [148]. With respect to our newly defined terminology for batch-effect correction and missing data, this highlights the relevance of CFBE properties in modern *omic* data analyses. In an investigation of HarmonizR, ComBat and missForest-based imputation, Voss and Schlumbohm et al. elaborated that imputation can be error-prone, but if necessary, it should be applied before batch-effect correction [100]. If these mutual effects of batch-effect correction and missing value imputation cause challenges in data analyses, researchers may consider imputation-free methods as an alternative. Amongst other methods [149–151], BERT and HarmonizR, for example facilitate missing-value-tolerant batch-effect correction; ACF [152] allows for classification with missing data; and ProtRank [153] allows for differential expression analysis on incomplete proteomic data.

In summary, both data integration as well as missing-value imputation may be necessary to allow for high-quality downstream data analysis. In practice, there is no optimal set of methods across all omic types and datasets and researchers often follow an iterative exploration scheme, in which they repeatedly employ a combination of algorithms and validate the respective results using multiple methods and downstream tasks. During this explorative validation, multiple aspects such as missing data pattern, experimental design and even *omic* type[1] may need to be considered. We provide a formalization of this typical process into a three-step workflow under consideration of data integration and missing values starting from the study conception, followed by an iterative method exploration phase, and finishing with the final data analysis, compare Figure 6. Yet, the large number of methods (each with different statistical assumptions and aims) represent an increasing challenge for researchers and interactive tools, such as scIMC, malbacR, or proBatch will hence gain more relevance in the future. However, such tools are so far often limited to subsets of omic types and the field is still lacking a fully comprehensive tool, which might represent an attractive target for future research. Thus, we developed a comprehensive decision flow chart as to aid users in eliminating methods or method types, which are not applicable to their data, compare Figure 7. Furthermore, researchers should aim to prospectively better characterize the interconnection between batch-effect correction and imputation, which is only poorly understood to date. Here, researchers should especially aim to develop tools, similar to the ALRA approach [44], that are capable of diagnosing and categorizing the missing values according to the mechanism that generated them. In addition to unraveling the generally understudies field of missing value types in *omics*, these tools will enable the design of the aforementioned more suitable imputation strategies for the imputation of missing values. Finally, future research should establish a broader set of missing-value tolerant methods for datasets for which high-quality imputation is not possible.

Author Contributions

Yannis Schumann conducted the literature review, created the figures, and wrote the initial draft of the manuscript. Antonia Gocke provided substantial critical feedback and contributions to the manuscript text. Julia E. Neumann supervised the study and wrote the manuscript.

Acknowledgements

The authors have nothing to report.

Open access funding enabled and organized by Projekt DEAL.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Endnotes

[1]Note that many methods are applicable to various *omic* types (e.g., ComBat), but some methods are particularly suited for certain data properties that are correlated with such types. As an example, batch-effect correction based on mutual nearest neighbors is particularly suited for large number of observations as often found in single-cell data. Hence, despite the general applicability of various methods, the *omic* type still represents an important criterion during the iterative exploration phase presented in Figure 6.

References

1. N. Vahabi and G. Michailidis, "Unsupervised Multi-Omics Data Integration Methods: A Comprehensive Review," *Frontiers in Genetics* 13 (2022): 854752, https://doi.org/10.3389/fgene.2022.854752.

2. I. Subramanian, S. Verma, S. Kumar, A. Jere, and K. Anamika, "Multi-Omics Data Integration, Interpretation, and Its Application," *Bioinformatics and Biology Insights* 14 (2020), https://doi.org/10.1177/1177932219899051.

3. M. Kang, E. Ko, and T. B. Mersha, "A Roadmap for Multi-Omics Data Integration Using Deep Learning," *Briefings in Bioinformatics* 23, no. 1 (2022): bbab454, https://doi.org/10.1093/bib/bbab454.

4. J. E. Flores, M. D. Claborne, D. Z. Weller, M. B. Webb-Robertson, M. K. Waters, and M. L. Kramer, "Missing data in multi-omics integration: Recent advances through artificial intelligence," *Frontiers in Artificial Intelligence* 6, 2023, https://doi.org/10.3389/frai.2023.1098308.

5. Y. Nan, J. Del Ser, S. Walsh, et al., "Data Harmonisation for Information Fusion in Digital Healthcare: A State-of-the-Art Systematic Review, Meta-Analysis and Future Research Directions," *Information Fusion* 82 (2022): 99–122, https://doi.org/10.1016/j.inffus.2022.01.001.

6. S. X. Phua, K. P. Lim, and W. W. B. Goh, "Perspectives for Better Batch Effect Correction in Mass-Spectrometry-Based Proteomics," *Computational and Structural Biotechnology Journal* 20 (2022): 4369–4375, https://doi.org/10.1016/j.csbj.2022.08.022.

7. W. W. B. Goh, W. Wang, and L. Wong, "Why Batch Effects Matter in Omics Data, and How to Avoid Them," *Trends in Biotechnology* 35, no. 6 (2017): 498–507, https://doi.org/10.1016/j.tibtech.2017.02.012.

8. Z. Basharat, S. Majeed, H. Saleem, I. A. Khan, and A. Yasmin, "An Overview of Algorithms and Associated Applications for Single Cell RNA-Seq Data Imputation," *Current Genomics* 22, no. 5 (2020): 319–327, https://doi.org/10.2174/1389202921999200716104916.

9. B.-J. Webb-Robertson, H. K. Wiberg, M. M. Matzke, et al., "Review, Evaluation, and Discussion of the Challenges of Missing Value Imputation for Mass Spectrometry-Based Label-Free Global Proteomics," *Journal of Proteome Research* 14, no. 5 (2015): 1993–2001, https://doi.org/10.1021/pr501138h.

10. R. J. A. Little and D. B. Rubin, "Introduction," In *Statistical Analysis With Missing Data*, 1–23 (Hoboken: Wiley, 2002), https://doi.org/10.1002/9781119013563.ch1.

11. R. Wei, J. Wang, M. Su, et al., "Missing Value Imputation Approach for Mass Spectrometry-based Metabolomics Data," *Scientific Reports* 8, no. 1 (2018): 663, https://doi.org/10.1038/s41598-017-19120-0.

12. R. Jiang, T. Sun, D. Song, and J. J. Li, "Statistics or Biology: The Zero-Inflation Controversy About scRNA-seq Data," *Genome Biology* 23, no. 1 (2022): 31, https://doi.org/10.1186/s13059-022-02601-5.

13. Y.i Qiao, X. Huang, P. J. Moos, et al., "A Bayesian Framework to Study Tumor Subclone–Specific Expression by Combining Bulk DNA and Single-Cell RNA Sequencing Data," *Genome Research* 34, no. 1 (2024): 94–105, https://doi.org/10.1101/gr.278234.123.

14. D. Wang, M. Quesnel-Vallieres, S. Jewell, et al., "A Bayesian Model for Unsupervised Detection of RNA Splicing Based Subtypes in Cancers," *Nature Communications* 14, no. 1 (2023): 63, https://doi.org/10.1038/s41467-022-35369-0.

15. S. Taylor, M. Ponzini, M. Wilson, and K. Kim, "Comparison of Imputation and Imputation-Free Methods for Statistical Analysis of Mass Spectrometry Data With Missing Data," *Briefings in Bioinformatics* 23, no. 1 (2022): bbab353, https://doi.org/10.1093/bib/bbab353.

16. J. Schuchhardt, "Normalization Strategies for cDNA Microarrays," *Nucleic Acids Research* 28, no. 10 (2000): 47e–47e, https://doi.org/10.1093/nar/28.10.e47.

17. S. C. Hicks, F. W. Townes, M. Teng, and R. A. Irizarry, "Missing Data and Technical Variability in Single-Cell RNA-Sequencing Experiments," *Biostatistics* 19, no. 4 (2018): 562–578, https://doi.org/10.1093/biostatistics/kxx053.

18. H. M. Bennett, W. Stephenson, C. M. Rose, and S. Darmanis, "Single-Cell Proteomics Enabled by Next-Generation Sequencing or Mass Spectrometry," *Nature Methods* 20, no. 3 (2023): 363–374, https://doi.org/10.1038/s41592-023-01791-5.

19. M. Javanbakht, J. Lin, A. Ragsdale, S. Kim, S. Siminski, and P. Gorbach, "Comparing Single and Multiple Imputation Strategies for Harmonizing Substance Use Data Across HIV-Related Cohort Studies," *BMC Medical Research Methodology* 22, no. 1 (2022): 90, https://doi.org/10.1186/s12874-022-01554-4.

20. M. J. Azur, E. A. Stuart, C. Frangakis, and P. J. Leaf, "Multiple Imputation by Chained Equations: What is it and How Does it Work?" *International Journal of Methods in Psychiatric Research* 20, no. 1 (2011): 40–49, https://doi.org/10.1002/mpr.329.

21. T. Raghunathan, J. Lepkowski, J. Van Hoewyk, and P. Solenberger, "A Multivariate Technique for Multiply Imputing Missing Values Using a Sequence of Regression Models," *Survey Methodology* (2001): 85–95.

22. I. R. White, P. Royston, and A. M. Wood, "Multiple Imputation Using Chained Equations: Issues and Guidance for Practice," *Statistics in Medicine* 30, no. 4 (2011): 377–399, https://doi.org/10.1002/sim.4067.

23. H. Kim, G. H. Golub, and H. Park, "Missing Value Estimation for DNA Microarray Gene Expression Data: Local Least Squares Imputation," *Bioinformatics* 21, no. 2 (2005): 187–198, https://doi.org/10.1093/bioinformatics/bth499.

24. Z. Cai, M. Heydari, and G. Lin, "Iterated Local Least Squares Microarray Missing Value Imputation," *Journal of Bioinformatics and Computational Biology* 04, no. 05 (2006): 935–957, https://doi.org/10.1142/S0219720006002302.

25. A. Wang, Y. Chen, N. An, J. Yang, L. Li, and L. Jiang, "Microarray Missing Value Imputation: A Regularized Local Learning Method," *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 16, no. 3 (2019): 980–993, https://doi.org/10.1109/TCBB.2018.2810205.

26. W. K. Ching, M. Li, N. K. Tsing, et al., "A Weighted Local Least Squares Imputation Method for Missing Value Estimation in Microarray Gene Expression Data," *International Journal of Data Mining and Bioinformatics* 4, no. 3 (2010): 331, https://doi.org/10.1504/IJDMB.2010.033524.

27. Z. Yu, T. Li, S.-J. Horng, Y. Pan, H. Wang, and Y. Jing, "An Iterative Locally Auto-Weighted Least Squares Method for Microarray Missing Value Estimation," *IEEE Transactions on NanoBioscience* 16, no. 1 (2017): 21–33, https://doi.org/10.1109/TNB.2016.2636243.

28. L. Jin, Y. Bi, C. Hu, et al., "A Comparative Study of Evaluating Missing Value Imputation Methods in Label-Free Proteomics," *Scientific Reports* 11, no. 1 (2021): 1760, https://doi.org/10.1038/s41598-021-81279-4.

29. O. Troyanskaya, M. Cantor, G. Sherlock, et al., "Missing Value Estimation Methods for DNA Microarrays," *Bioinformatics* 17, no. 6 (2001): 520–525, https://doi.org/10.1093/bioinformatics/17.6.520.

30. K. Y. Kim, B. J. Kim, and G. S. Yi, "Reuse of Imputed Data in Microarray Analysis Increases Imputation Efficiency," *BMC Bioinformat-*

ics [Electronic Resource] 5, no. 1 (2004): 160, https://doi.org/10.1186/1471-2105-5-160.

31. L. P. Brás and J. C. Menezes, "Improving Cluster-Based Missing Value Estimation of DNA Microarray Data," *Biomolecular Engineering* 24, no. 2 (2007): 273–282, https://doi.org/10.1016/j.bioeng.2007.04.003.

32. M. Kokla, J. Virtanen, M. Kolehmainen, J. Paananen, and K. Hanhineva, "Random Forest-Based Imputation Outperforms Other Methods for Imputing LC-MS Metabolomics Data: A Comparative Study," *BMC Bioinformatics [Electronic Resource]* 20, no. 1 (2019): 492, https://doi.org/10.1186/s12859-019-3110-0.

33. J. S. Shah, S. N. Rai, A. P. Defilippis, B. G. Hill, A. Bhatnagar, and G. N. Brock, "Distribution Based Nearest Neighbor Imputation for Truncated High Dimensional Data With Applications to Pre-Clinical and Clinical Metabolomics Studies," *BMC Bioinformatics [Electronic Resource]* 18, no. 1 (2017): 114, https://doi.org/10.1186/s12859-017-1547-6.

34. P. Di Lena, C. Sala, A. Prodi, and C. Nardini, "Missing Value Estimation Methods for DNA Methylation Data," *Bioinformatics* 35, no. 19 (2019): 3786–3793, https://doi.org/10.1093/bioinformatics/btz134.

35. A. Plaksienko, P. Di Lena, C. Nardini, and C. Angelini, "methyLImp2: Faster Missing Value Estimation for DNA Methylation Data," *Bioinformatics* 40, no. 1 (2024): btae001, https://doi.org/10.1093/bioinformatics/btae001.

36. M. Chen and X. Zhou, "VIPER: Variability-Preserving Imputation for Accurate Gene Expression Recovery in Single-Cell RNA Sequencing Studies," *Genome Biology* 19, no. 1 (2018): 196, https://doi.org/10.1186/s13059-018-1575-1.

37. K. Zhu and D. Anastassiou, "2DImpute: Imputation in Single-Cell RNA-Seq Data From Correlations in Two Dimensions," *Bioinformatics* 36, no. 11 (2020): 3588–3589, https://doi.org/10.1093/bioinformatics/btaa148.

38. P. Keerin, W. Kurutach, and T. Boongoen, "A Cluster-Directed Framework for Neighbour Based Imputation of Missing Value in Microarray Data," *International Journal of Data Mining and Bioinformatics* 15, no. 2 (2016): 165, https://doi.org/10.1504/IJDMB.2016.076535.

39. Y. Chen, A. Wang, H. Ding, et al., "A Global Learning With Local Preservation Method for Microarray Data Imputation," *Computers in Biology and Medicine* 77 (2016): 76–89, https://doi.org/10.1016/j.compbiomed.2016.08.005.

40. M. E. Tipping and C. M. Bishop, "Probabilistic Principal Component Analysis," *Journal of the Royal Statistical Society Series B: Statistical Methodology* 61, no. 3 (1999): 611–622, https://doi.org/10.1111/1467-9868.00196.

41. S. Oba, M.-A. Sato, I. Takemasa, M. Monden, K.-I. Matsubara, and S. Ishii, "A Bayesian Missing Value Estimation Method for Gene Expression Profile Data," *Bioinformatics* 19, no. 16 (2003): 2088–2096, https://doi.org/10.1093/bioinformatics/btg287.

42. F. Meng, C. Cai, and H. Yan, "A Bicluster-Based Bayesian Principal Component Analysis Method for Microarray Missing Value Estimation," *IEEE Journal of Biomedical and Health Informatics* 18, no. 3 (2014): 863–871, https://doi.org/10.1109/JBHI.2013.2284795.

43. J. Xu, L. Cai, B.o Liao, W. Zhu, and J. Yang, "CMF-Impute: An Accurate Imputation Tool for Single-Cell RNA-seq Data," *Bioinformatics* 36, no. 10 (2020): 3139–3147, https://doi.org/10.1093/bioinformatics/btaa109.

44. G. C. Linderman, J. Zhao, M. Roulis, et al., "Zero-Preserving Imputation of Single-Cell RNA-Seq Data," *Nature Communications* 13, no. 1 (2022): 192, https://doi.org/10.1038/s41467-021-27729-z.

45. Y. Sun, J. Li, Y. Xu, T. Zhang, and X. Wang, "Deep Learning Versus Conventional Methods for Missing Data Imputation: A Review and Comparative study," *Expert Systems with Applications* 227 (2023): 120201, https://doi.org/10.1016/j.eswa.2023.120201.

46. D. Talwar, A. Mongia, D. Sengupta, and A. Majumdar, "AutoImpute: Autoencoder Based Imputation of Single-Cell RNA-seq Data," *Scientific Reports* 8, no. 1 (2018): 16329, https://doi.org/10.1038/s41598-018-34688-x.

47. Y. L. Qiu, H. Zheng, and O. Gevaert, "Genomic Data Imputation With Variational Auto-Encoders," *GigaScience* 9, no. 8 (2020), https://doi.org/10.1093/gigascience/giaa082.

48. K. Xu, C. Cheong, W. P. Veldsman, A. Lyu, W. K. Cheung, and L. Zhang, "Accurate and Interpretable Gene Expression Imputation on scRNA-Seq Data Using IGSimpute," *Briefings in Bioinformatics* 24, no. 3 (2023), https://doi.org/10.1093/bib/bbad124.

49. Y. Hu, Y. Zhao, C. T. Schunk, Y. Ma, T. Derr, and X. M. Zhou, "ADEPT: Autoencoder With Differentially Expressed Genes and Imputation for Robust Spatial Transcriptomics Clustering," *Iscience* 26, no. 6 (2023): 106792, https://doi.org/10.1016/j.isci.2023.106792.

50. W. Zhang, B. Huckaby, J. Talburt, S. Weissman, and M. Q.u Yang, "cnnImpute: Missing Value Recovery for Single Cell RNA Sequencing Data," *Scientific Reports* 14, no. 1 (2024): 3946, https://doi.org/10.1038/s41598-024-53998-x.

51. X. Wu and Y. Zhou, "GE-Impute: Graph Embedding-Based Imputation for Single-Cell RNA-seq Data," *Briefings in Bioinformatics* 23, no. 5 (2022), https://doi.org/10.1093/bib/bbac313.

52. Y. Qi, S. Han, L. Tang, and L. Liu, "Imputation Method for Single-Cell RNA-Seq Data Using Neural Topic Model," *GigaScience* 12 (2022), https://doi.org/10.1093/gigascience/giad098.

53. Y. Shi, J. Wan, X. Zhang, and Y. Yin, "CL-Impute: A Contrastive Learning-Based Imputation for Dropout Single-Cell RNA-Seq Data," *Computers in Biology and Medicine* 164 (2023): 107263, https://doi.org/10.1016/j.compbiomed.2023.107263.

54. J. Tuikkala, L. Elo, O. S. Nevalainen, and T. Aittokallio, "Improving Missing Value Estimation in Microarray Data With Gene Ontology," *Bioinformatics* 22, no. 5 (2006): 566–572, https://doi.org/10.1093/bioinformatics/btk019.

55. S. Chen, X. Yan, R. Zheng, and M. Li, "Bubble: A Fast Single-Cell RNA-Seq Imputation Using an Autoencoder Constrained by Bulk RNA-Seq Data," *Briefings in Bioinformatics* 24, no. 1 (2023): bbac580, https://doi.org/10.1093/bib/bbac580.

56. L. Gan, G. Vinci, and G. I. Allen, "Correlation Imputation for Single-Cell RNA-Seq," *Journal of Computational Biology* 29, no. 5 (2022): 465–482, https://doi.org/10.1089/cmb.2021.0403.

57. A. Mongia, D. Sengupta, and A. Majumdar, "deepMc: Deep Matrix Completion for Imputation of Single-Cell RNA-Seq Data," *Journal of Computational Biology* 27, no. 7 (2020): 1011–1019, https://doi.org/10.1089/cmb.2019.0278.

58. M. Karikomi, P. Zhou, and Q. Nie, "DURIAN: An Integrative Deconvolution and Imputation Method for Robust Signaling Analysis of Single-Cell Transcriptomics Data," *Briefings in Bioinformatics* 23, no. 4 (2022), bbac223, https://doi.org/10.1093/bib/bbac223.

59. T. Peng, Q. Zhu, P. Yin, and K. Tan, "SCRABBLE: Single-cell RNA-seq Imputation Constrained by Bulk RNA-seq Data," *Genome Biology* 20, no. 1 (2019): 88, https://doi.org/10.1186/s13059-019-1681-8.

60. M. B. Badsha, R. Li, B. Liu, et al., "Imputation of Single-Cell Gene Expression With an Autoencoder Neural Network," *Quantitative Biology* 8, no. 1 (2020): 78–94, https://doi.org/10.1007/s40484-019-0192-7.

61. S. Chen, R. Zheng, L. Tian, F.-X. Wu, and M. Li, "A Posterior Probability Based Bayesian Method for Single-Cell RNA-Seq Data Imputation," *Methods* 216 (2023): 21–38, https://doi.org/10.1016/j.ymeth.2023.06.004.

62. W. V. Li and J. J. Li, "An Accurate and Robust Imputation Method scImpute for Single-Cell RNA-Seq Data," *Nature Communications* 9, no. 1 (2018): 997, https://doi.org/10.1038/s41467-018-03405-7.

63. L. Breiman, "Random Forests," *Machine Learning* 45, no. 1 (2001): 5–32, https://doi.org/10.1023/A:1010933404324.

64. D. J. Stekhoven and P. Bühlmann, "MissForest—Non-Parametric Missing Value Imputation for Mixed-Type Data," *Bioinformatics* 28, no. 1 (2012): 112–118, https://doi.org/10.1093/bioinformatics/btr597.

65. J. Shah, G. N. Brock, and J. Gaskins, "BayesMetab: Treatment of Missing Values in Metabolomic Studies Using a Bayesian Modeling Approach," *BMC Bioinformatics [Electronic Resource]* 20, no. S24 (2019): 673, https://doi.org/10.1186/s12859-019-3250-2.

66. W. Tang, F. Bertaux, P. Thomas, et al., "bayNorm: Bayesian Gene Expression Recovery, Imputation and Normalization for Single-Cell RNA-Sequencing Data," *Bioinformatics* 36, no. 4 (2020): 1174–1181, https://doi.org/10.1093/bioinformatics/btz726.

67. P. Lin, M. Troup, and J. W. K. Ho, "CIDR: Ultrafast and Accurate Clustering Through Imputation for Single-Cell RNA-Seq Data," *Genome Biology* 18, no. 1 (2017): 59, https://doi.org/10.1186/s13059-017-1188-0.

68. W. Wu, Y. Liu, Q. Dai, X. Yan, and Z. Wang, "G2S3: A Gene Graph-Based Imputation Method for Single-Cell RNA Sequencing Data," *PLOS Computational Biology* 17, no. 5 (2021): e1009029, https://doi.org/10.1371/journal.pcbi.1009029.

69. W. Gong, I.-Y. Kwak, P. Pota, N. Koyano-Nakagawa, and D. J. Garry, "DrImpute: Imputing Dropout Events in Single Cell RNA Sequencing Data," *BMC Bioinformatics [Electronic Resource]* 19, no. 1 (2018): 220, https://doi.org/10.1186/s12859-018-2226-y.

70. W. E. Johnson, C. Li, and A. Rabinovic, "Adjusting Batch Effects in Microarray Expression Data Using Empirical Bayes Methods," *Biostatistics* 8, no. 1 (2007): 118–127, https://doi.org/10.1093/biostatistics/kxj037.

71. M. J. Larsen, M. Thomassen, Q. Tan, K. P. Sørensen, and T. A. Kruse, "Microarray-Based RNA Profiling of Breast Cancer: Batch Effect Removal Improves Cross-Platform Consistency," *BioMed Research International* 2014 (2014): 1–11, https://doi.org/10.1155/2014/651751.

72. M. Ligero, O. Jordi-Ollero, K. Bernatowicz, et al., "Minimizing Acquisition-Related Radiomics Variability by Image Resampling and Batch Effect Correction to Allow for Large-Scale Data Analysis," *European Radiology* 31, no. 3 (2021): 1460–1470, https://doi.org/10.1007/s00330-020-07174-0.

73. A. Vandenbon, "Evaluation of Critical Data Processing Steps for Reliable Prediction of Gene Co-Expression From Large Collections of RNA-Seq Data," *PLoS ONE* 17, no. 1 (2022): e0263344, https://doi.org/10.1371/journal.pone.0263344.

74. F. Petralia, N. Tignor, B. Reva, et al., "Integrated Proteogenomic Characterization Across Major Histological Types of Pediatric Brain Cancer," *Cell* 183, no. 7 (2020): 1962–1985.e31.e31, https://doi.org/10.1016/j.cell.2020.10.044.

75. M. F. Adamer, S. C. Brüningk, A. Tejada-Arranz, F. Estermann, M. Basler, and K. Borgwardt, "reComBat: Batch-Effect Removal in Large-Scale Multi-Source Gene-Expression Data Integration," *Bioinformatics Advances* 2, no. 1 (2022), https://doi.org/10.1093/bioadv/vbac071.

76. C. K. Stein, P. Qu, J. Epstein, et al., "Removing Batch Effects From Purified Plasma Cell Gene Expression Microarrays With Modified ComBat," *BMC Bioinformatics [Electronic Resource]* 16, no. 1 (2015): 63, https://doi.org/10.1186/s12859-015-0478-3.

77. Y. Zhang, G. Parmigiani, and W. E. Johnson, "ComBat-Seq: Batch Effect Adjustment for RNA-Seq Count Data," *NAR Genomics and Bioinformatics* 2, no. 3 (2020): lqaa078, https://doi.org/10.1093/nargab/lqaa078.

78. M. E. Ritchie, B. Phipson, D.i Wu, et al., "Limma Powers Differential Expression Analyses for RNA-Sequencing and Microarray Studies," *Nucleic Acids Research* 43, no. 7 (2015): e47–e47, https://doi.org/10.1093/nar/gkv007.

79. G. L. Eagle, J. M. J. Herbert, J. Zhuang, et al., "Assessing Technical and Biological Variation in SWATH-MS-Based Proteomic Analysis of Chronic Lymphocytic Leukaemia Cells," *Scientific Reports* 11, no. 1 (2021): 2932, https://doi.org/10.1038/s41598-021-82609-2.

80. S. Hajebi Khaniki, F. Shokoohi, H. Esmaily, and M. A. Kerachian, "Analyzing Aberrant DNA Methylation in Colorectal Cancer Uncovered Intangible Heterogeneity of Gene Effects in the Survival Time of Patients," *Scientific Reports* 13, no. 1 (2023): 22104, https://doi.org/10.1038/s41598-023-47377-1.

81. C. Müller, A. Schillert, C. Röthemeier, et al., "Removing Batch Effects From Longitudinal Gene Expression—Quantile Normalization Plus ComBat as Best Approach for Microarray Transcriptome Data," *PLoS ONE* 11, no. 6 (2016): e0156594, https://doi.org/10.1371/journal.pone.0156594.

82. J. C. Beer, N. J. Tustison, P. A. Cook, et al., "Longitudinal ComBat: A Method for Harmonizing Longitudinal Multi-Scanner Imaging Data," *Neuroimage* 220 (2020): 117129, https://doi.org/10.1016/j.neuroimage.2020.117129.

83. H. Horng, A. Singh, B. Yousefi, et al., "Generalized ComBat Harmonization Methods for Radiomic Features With Multi-Modal Distributions and Multiple Batch Effects," *Scientific Reports* 12, no. 1 (2022): 4493, https://doi.org/10.1038/s41598-022-08412-9.

84. H. Horng, A. Singh, B. Yousefi, et al., "Improved Generalized ComBat Methods for Harmonization of Radiomic Features," *Scientific Reports* 12, no. 1 (2022): 19009, https://doi.org/10.1038/s41598-022-23328-0.

85. L. Haghverdi, A. T. L. Lun, M. D. Morgan, and J. C. Marioni, "Batch Effects in Single-Cell RNA-Sequencing Data are Corrected by Matching Mutual Nearest Neighbors," *Nature Biotechnology* 36, no. 5 (2018): 421–427, https://doi.org/10.1038/nbt.4091.

86. B. Hie, B. Bryson, and B. Berger, "Efficient Integration of Heterogeneous Single-Cell Transcriptomes Using Scanorama," *Nature Biotechnology* 37, no. 6 (2019): 685–691, https://doi.org/10.1038/s41587-019-0113-3.

87. I. Korsunsky, N. Millard, J. Fan, et al., "Fast, Sensitive and Accurate Integration of Single-Cell Data With Harmony," *Nature Methods* 16, no. 12 (2019): 1289–1296, https://doi.org/10.1038/s41592-019-0619-0.

88. D. L. Plubell, P. A. Wilmarth, Y. Zhao, et al., "Extended Multiplexing of Tandem Mass Tags (TMT) Labeling Reveals Age and High Fat Diet Specific Proteome Changes in Mouse Epididymal Adipose Tissue," *Molecular & Cellular Proteomics* 16, no. 5 (2017): 873–890, https://doi.org/10.1074/mcp.M116.065524.

89. K. Krug, E. J. Jaehnig, S. Satpathy, et al., "Proteogenomic Landscape of Breast Cancer Tumorigenesis and Targeted Therapy," *Cell* 183, no. 5 (2020): 1436–1456.e31.e31, https://doi.org/10.1016/j.cell.2020.10.036.

90. Q. Xia, J. A. Thompson, and D. C. Koestler, "Batch Effect Reduction of Microarray Data With Dependent Samples Using an Empirical Bayes Approach (BRIDGE)," *Statistical Applications in Genetics and Molecular Biology* 20, no. 4–6 (2021): 101–119, https://doi.org/10.1515/sagmb-2021-0020.

91. T. E. Sweeney, H. R. Wong, and P. Khatri, "Robust Classification of Bacterial and Viral Infections via Integrated Host Gene Expression Diagnostics," *Science Translational Medicine* 8, no. 346 (2016): 346ra91, https://doi.org/10.1126/scitranslmed.aaf7165.

92. Y. Yang, G. Li, H. Qian, K. C. Wilhelmsen, Y. Shen, and Y. Li, "SMNN: Batch Effect Correction for Single-Cell RNA-Seq Data via Supervised Mutual Nearest Neighbor Detection," *Briefings in Bioinformatics* 22, no. 3 (2021), https://doi.org/10.1093/bib/bbaa097.

93. Y. Yang, G. Li, Y. Xie, et al., "iSMNN: Batch Effect Correction for Single-Cell RNA-Seq Data via Iterative Supervised Mutual Nearest Neighbor Refinement," *Briefings in Bioinformatics* 22, no. 5 (2021), https://doi.org/10.1093/bib/bbab122.

94. Z. Rong, Q. Tan, L. Cao, et al., "NormAE: Deep Adversarial Learning Model to Remove Batch Effects in Liquid Chromatography Mass Spectrometry-Based Metabolomics Data," *Analytical Chemistry* 92, no. 7 (2020): 5082–5090, https://doi.org/10.1021/acs.analchem.9b05460.

95. A. B. Dincer, J. D. Janizek, and S.u-I.n Lee, "Adversarial Deconfounding Autoencoder for Learning Robust Gene Expression Embeddings," *Bioinformatics* 36, no. Supplement 2 (2020): i573–i582, https://doi.org/10.1093/bioinformatics/btaa796.

96. L. Cavinato, M. C. Massi, M. Sollini, M. Kirienko, and F. Ieva, "Dual Adversarial Deconfounding Autoencoder for Joint Batch-Effects Removal From Multi-Center and Multi-Scanner Radiomics Data," *Scientific Reports* 13, no. 1 (2023): 18857, https://doi.org/10.1038/s41598-023-45983-7.

97. N. Kotlov, K. Shaposhnikov, C. Tazearslan, et al., "Procrustes is a Machine-Learning Approach that Removes Cross-Platform Batch Effects From Clinical RNA Sequencing Data," *Communications Biology* 7, no. 1 (2024): 392, https://doi.org/10.1038/s42003-024-06020-z.

98. B. Zhang, H. Wu, Y. Wang, C. Xuan, and J. Gao, "scGAMNN: Graph Antoencoder-Based Single-Cell RNA Sequencing Data Integration Algorithm Using Mutual Nearest Neighbors," *IEEE Journal of Biomedical and Health Informatics* 27, no. 11 (2023): 5665–5674, https://doi.org/10.1109/JBHI.2023.3311340.

99. X. Wang, J. Wang, H. Zhang, S. Huang, and Y. Yin, "HDMC: A Novel Deep Learning-Based Framework for Removing Batch Effects in Single-Cell RNA-Seq Data," *Bioinformatics* 38, no. 5 (2022): 1295–1303, https://doi.org/10.1093/bioinformatics/btab821.

100. H. Voß, S. Schlumbohm, P. Barwikowski, et al., "HarmonizR Enables Data Harmonization Across Independent Proteomic Datasets With Appropriate Handling of Missing Values," *Nature Communications* 13, no. 1 (2022): 3523, https://doi.org/10.1038/s41467-022-31007-x.

101. Y. Schumann and S. Schlumbohm (2023). "BERT," Software Publication in Bioconductor, https://doi.org/10.18129/B9.bioc.BERT.

102. T. Fei and T. Yu, "scBatch: Batch-Effect Correction of RNA-Seq Data Through Sample Distance Matrix Adjustment," *Bioinformatics* 36, no. 10 (2020): 3115–3123, https://doi.org/10.1093/bioinformatics/btaa097.

103. T. Fei, T. Zhang, W. Shi, and T. Yu, "Mitigating the Adverse Impact of Batch Effects in Sample Pattern Detection," *Bioinformatics* 34, no. 15 (2018): 2634–2641, https://doi.org/10.1093/bioinformatics/bty117.

104. A. Butler, P. Hoffman, P. Smibert, E. Papalexi, and R. Satija, "Integrating Single-Cell Transcriptomic Data Across Different Conditions, Technologies, and Species," *Nature Biotechnology* 36, no. 5 (2018): 411–420, https://doi.org/10.1038/nbt.4096.

105. J. D. Welch, V. Kozareva, A. Ferreira, C. Vanderburg, C. Martin, and E. Z. Macosko, "Single-Cell Multi-Omic Integration Compares and Contrasts Features of Brain Cell Identity," *Cell* 177, no. 7 (2019): 1873–1887.e17, https://doi.org/10.1016/j.cell.2019.05.006.

106. R. Akulenko, M. Merl, and V. Helms, "BEclear: Batch Effect Detection and Adjustment in DNA Methylation Data," *PLoS ONE* 11, no. 8 (2016): e0159921, https://doi.org/10.1371/journal.pone.0159921.

107. L. van der Maaten and G. Hinton, "Visualizing Data Using t-SNE," *Journal of Machine Learning Research* 9 (2008): 2579–2605.

108. L. Mcinnes, J. Healy, N. Saul, and L. Großberger, "UMAP: Uniform Manifold Approximation and Projection," *Journal of Open Source Software* 3, no. 29 (2018): 861, https://doi.org/10.21105/joss.00861.

109. H. Wold, "Estimation of Principal Components and Related Models by Iterative Least Squares," *Multivariate Analysis*, 391–420. (New York: Academic Press, 1966).

110. S. Godbole, H. Voß, and A. Gocke, et al., "Multiomic Profiling of Medulloblastoma Reveals Subtype-Specific Targetable Alterations at the Proteome and N-Glycan Level," *Nature Communications* 15 (2024): 6237.

111. P. J. Rousseeuw, "Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis," *Journal of Computational and Applied Mathematics* 20 (1987): 53–65, https://doi.org/10.1016/0377-0427(87)90125-7.

112. L. Hubert and P. Arabie, "Comparing Partitions," *Journal of Classification* 2, no. 1 (1985): 193–218, https://doi.org/10.1007/BF01908075.

113. M. Büttner, Z. Miao, F. A. Wolf, S. A. Teichmann, and F. J. Theis, "A Test Metric for Assessing Single-Cell RNA-Seq Batch Correction," *Nature Methods* 16, no. 1 (2019): 43–49, https://doi.org/10.1038/s41592-018-0254-1.

114. Y. Zhao, Y. Guo, and L. Li, "cKBET: Assessing Goodness of Batch Effect Correction for Single-Cell RNA-Seq," *Frontiers of Computer Science* 18, no. 1 (2024): 181901, https://doi.org/10.1007/s11704-022-2111-8.

115. H. T. N. Tran, K. S. Ang, M. Chevrier, et al., "A Benchmark of Batch-Effect Correction Methods for Single-Cell Rna Sequencing Data," *Genome Biology* 21, no. 1 (2020): 12, https://doi.org/10.1186/s13059-019-1850-9.

116. Z. Đ. Vujovic, "Classification Model Evaluation Metrics," *International Journal of Advanced Computer Science and Applications* 12, no. 6 (2021), https://doi.org/10.14569/IJACSA.2021.0120670.

117. M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, et al., "The FAIR Guiding Principles for Scientific Data Management and Stewardship," *Scientific Data* 3, no. 1 (2016): 160018, https://doi.org/10.1038/sdata.2016.18.

118. D. Capper, D. T. W. Jones, M. Sill, et al., "DNA Methylation-Based Classification of Central Nervous System Tumours," *Nature* 555, no. 7697 (2018): 469–474, https://doi.org/10.1038/nature26000.

119. F. Yue, Y. Cheng, A. Breschi, et al., "A Comparative Encyclopedia of DNA Elements in the Mouse Genome," *Nature* 515, no. 7527 (2014): 355–364, https://doi.org/10.1038/nature13992.

120. Y. Gilad and O. Mizrahi-Man, "A Reanalysis of Mouse ENCODE Comparative Gene Expression Data," *F1000Research* 4 (2015): 121, https://doi.org/10.12688/f1000research.6536.1.

121. V. Nygaard, E. A. Rødland, and E. Hovig, "Methods that Remove Batch Effects While Retaining Group Differences May Lead to Exaggerated Confidence in Downstream Analyses," *Biostatistics* 17, no. 1 (2016): 29–39, https://doi.org/10.1093/biostatistics/kxv027.

122. J. A. Gagnon-Bartsch and T. P. Speed, "Using Control Genes to Correct for Unwanted Variation in Microarray Data," *Biostatistics* 13, no. 3 (2012): 539–552, https://doi.org/10.1093/biostatistics/kxr034.

123. J. T. Leek and J. D. Storey, "Capturing Heterogeneity in Gene Expression Studies by Surrogate Variable Analysis," *PLoS Genetics* 3, no. 9 (2007): e161, https://doi.org/10.1371/journal.pgen.0030161.

124. D. Risso, J. Ngai, T. P. Speed, and S. Dudoit, "Normalization of RNA-Seq Data Using Factor Analysis of Control Genes or Samples," *Nature Biotechnology* 32, no. 9 (2014): 896–902, https://doi.org/10.1038/nbt.2931.

125. J. T. Leek, "svaseq: Removing Batch Effects and Other Unwanted Noise from Sequencing Data," *Nucleic Acids Research* 42, no. 21 (2014): e161–e161, https://doi.org/10.1093/nar/gku864.

126. Y. Wang, T. Liu, and H. Zhao, "ResPAN: A Powerful Batch Correction Model for scRNA-Seq Data Through Residual Adversarial Networks," *Bioinformatics* 38, no. 16 (2022): 3942–3949, https://doi.org/10.1093/bioinformatics/btac427.

127. T. Wang, T. S. Johnson, W. Shao, et al., "BERMUDA: A Novel Deep Transfer Learning Method for Single-Cell RNA Sequencing Batch Correction Reveals Hidden High-Resolution Cellular Subtypes," *Genome Biology* 20, no. 1 (2019): 165, https://doi.org/10.1186/s13059-019-1764-6.

128. J. Arevalo, R. van Dijk, A. E. Carpenter, and S. Singh, "Evaluating Batch Correction Methods for Image-Based Cell Profiling," *BioRxiv* 15 (2024), https://doi.org/10.1101/2023.09.15.558001.

129. C. Chen, K. Grennan, J. Badner, et al., "Removing Batch Effects in Analysis of Expression Microarray Data: An Evaluation of Six Batch Adjustment Methods," *PLoS ONE* 6, no. 2 (2011): e17238, https://doi.org/10.1371/journal.pone.0017238.

130. F. Hu, A. A. Chen, H. Horng, et al., "Image Harmonization: A Review of Statistical and Deep Learning Methods For Removing Batch Effects and Evaluation Metrics for Effective Harmonization," *Neuroimage* 274 (2023): 120125, https://doi.org/10.1016/j.neuroimage.2023.120125.

131. X. Liu, K. Salokas, R. G. Weldatsadik, L. Gawriyski, and M. Varjosalo, "Combined Proximity Labeling and Affinity Purification–Mass Spectrometry Workflow for Mapping and Visualizing Protein Interaction Networks," *Nature Protocols* 15, no. 10 (2020): 3182–3211, https://doi.org/10.1038/s41596-020-0365-x.

132. K. G. Stratton, B.-J. M. Webb-Robertson, L. A. Mccue, et al., "pmartR: Quality Control and Statistics for Mass Spectrometry-Based Biological Data," *Journal of Proteome Research* 18, no. 3 (2019): 1418–1425, https://doi.org/10.1021/acs.jproteome.8b00760.

133. N. Bararpour, F. Gilardi, C. Carmeli, et al., "DBnorm as an R Package for the Comparison and Selection of Appropriate Statistical Methods for Batch Effect Correction in Metabolomic Studies," *Scientific Reports* 11, no. 1 (2021): 5657, https://doi.org/10.1038/s41598-021-84824-3.

134. J. Čuklina, C. H. Lee, E. G. Williams, et al., "Diagnostics and Correction of Batch Effects in Large-Scale Proteomic Studies: A Tutorial," *Molecular Systems Biology* no. 8 (2021): 17, https://doi.org/10.15252/msb.202110240.

135. D. T. Leach, K. G. Stratton, J. Irvahn, R. Richardson, B.-J.o M. Webb-Robertson, and L. M. Bramer, "malbacR: A Package for Standardized Implementation of Batch Correction Methods for Omics Data," *Analytical Chemistry* 95, no. 33 (2023): 12195–12199, https://doi.org/10.1021/acs.analchem.3c01289.

136. J. Navolic, M. Moritz, H. Voß, et al., "Direct 3D Sampling of the Embryonic Mouse Head: Layer-Wise Nanosecond Infrared Laser (NIRL) Ablation From Scalp to Cortex for Spatially Resolved Proteomics," *Analytical Chemistry* 95, no. 47 (2023): 17220–17227, https://doi.org/10.1021/acs.analchem.3c02637.

137. M.o Huang, J. Wang, E. Torre, et al., "SAVER: Gene Expression Recovery for Single-Cell RNA Sequencing," *Nature Methods* 15, no. 7 (2018): 539–542, https://doi.org/10.1038/s41592-018-0033-z.

138. H. Webel, L. Niu, A. B. Nielsen, et al., "Imputation of Label-Free Quantitative Mass Spectrometry-Based Proteomics Data Using Self-Supervised Deep Learning," *Nature Communications* 15, no. 1 (2024): 5405, https://doi.org/10.1038/s41467-024-48711-5.

139. W. Kong, H. W. H. Hui, H. Peng, and W. W. B. Goh, "Dealing With Missing Values in Proteomics Data," *Proteomics* 22 (2022): 24, https://doi.org/10.1002/pmic.202200092.

140. T. S. Andrews and M. Hemberg, "False Signals Induced by Single-Cell Imputation," *F1000Research* 7 (2019): 1740, https://doi.org/10.12688/f1000research.16613.2.

141. Y. V. Karpievitch, A. R. Dabney, and R. D. Smith, "Normalization and Missing Value Imputation for Label-Free LC-MS Analysis," *BMC Bioinformatics [Electronic Resource]* 13, no. S16 (2012): S5, https://doi.org/10.1186/1471-2105-13-S16-S5.

142. H. J. Kim, T. Kim, N. J. Hoffman, et al., "PhosR Enables Processing and Functional Analysis of Phosphoproteomic Data," *Cell Reports* 34, no. 8 (2021): 108771, https://doi.org/10.1016/j.celrep.2021.108771.

143. C. Lazar, L. Gatto, M. Ferro, C. Bruley, and T. Burger, "Accounting for the Multiple Natures of Missing Values in Label-Free Quantitative Proteomics Data Sets to Compare Imputation Strategies," *Journal of Proteome Research* 15, no. 4 (2016): 1116–1125, https://doi.org/10.1021/acs.jproteome.5b00981.

144. L. M. Bramer, J. Irvahn, P. D. Piehowski, K. D. Rodland, and B.-J. M. Webb-Robertson, "A Review of Imputation Strategies for Isobaric Labeling-Based Shotgun Proteomics," *Journal of Proteome Research* 20, no. 1 (2021): 1–13, https://doi.org/10.1021/acs.jproteome.0c00123.

145. W. Hou, Z. Ji, H. Ji, and S. C. Hicks, "A Systematic Evaluation of Single-Cell RNA-Sequencing Imputation Methods," *Genome Biology* 21, no. 1 (2020): 218, https://doi.org/10.1186/s13059-020-02132-x.

146. C. Dai, Y.i Jiang, C. Yin, et al., "scIMC: A Platform for Benchmarking Comparison and Visualization Analysis of scRNA-Seq Data Imputation Methods," *Nucleic Acids Research* 50, no. 9 (2022): 4877–4899, https://doi.org/10.1093/nar/gkac317.

147. H. W. H. Hui, W. Kong, H. Peng, and W. W. B. Goh, "The Importance of Batch Sensitization in Missing Value Imputation," *Scientific Reports* 13, no. 1 (2023): 3003, https://doi.org/10.1038/s41598-023-30084-2.

148. W. W. B. Goh, H. W. H. Hui, and L. Wong, "How Missing Value Imputation Is Confounded With Batch Effects and What you Can Do About It," *Drug Discovery Today* 28, no. 9 (2023): 103661, https://doi.org/10.1016/j.drudis.2023.103661.

149. S. Chrétien, C. Guyeux, B. Conesa, et al., "A Bregman-Proximal Point Algorithm for Robust Non-Negative Matrix Factorization With Possible Missing Values and Outliers—Application to Gene Expression Analysis," *BMC Bioinformatics [Electronic Resource]* 17, no. S8 (2016): 284, https://doi.org/10.1186/s12859-016-1120-8.

150. S. Plancade, M. Berland, M. Blein-Nicolas, O. Langella, A. Bassignani, and C. Juste, "A Combined Test for Feature Selection on Sparse Metaproteomics Data—An Alternative to Missing Value Imputation," *PeerJ* 10 (2022): e13525, https://doi.org/10.7717/peerj.13525.

151. D.-W. Kim, K.-Y. Lee, K. H. Lee, and D. Lee, "Towards Clustering of Incomplete Microarray Data Without the use of Imputation," *Bioinformatics* 23, no. 1 (2007): 107–113, https://doi.org/10.1093/bioinformatics/btl555.

152. Y. Schumann, J. E. Neumann, and P. Neumann, "Robust Classification Using Average Correlations as Features (ACF)," *BMC Bioinformatics [Electronic Resource]* 24, no. 1 (2023): 101, https://doi.org/10.1186/s12859-023-05224-0.

153. M. Medo, D. M. Aebersold, and M. Medová, "ProtRank: Bypassing the Imputation of Missing Values in Differential Expression Analysis of Proteomic Data," *BMC Bioinformatics [Electronic Resource]* 20, no. 1 (2019): 563, https://doi.org/10.1186/s12859-019-3144-3.

Supporting Information

Additional supporting information can be found online in the Supporting Information section.