Title

Toward informed batch correction for single-cell transcriptome integration

Journal

Nature Computational Science, 6(2)

ISSN

2662-8457

Authors

Li, Shuang
Lücken, Malte
Marioni, John C
et al.

Publication Date

2026-02-16

DOI

10.1038/s43588-025-00943-1

Peer reviewed

Towards informed batch correction for single-cell transcriptome integration

Shuang Li[1,2], Malte Lücken[3,4], John C. Marioni[1,5,6], Sarah A. Teichmann[2], Peng He[7]
1.Wellcome Sanger Institute, Wellcome Genome Campus, Hinxton, UK
2.Stem Cell Institute & Department of Medicine, University of Cambridge, UK
3.Institute of computational Biology, Helmholtz Munich, Neuherberg, Germany
4.Institute of Lung Health & Immunity, Helmholtz Munich; Member of the German Center for Lung Research (DZL), Munich, Germany
5.European Molecular Biology Laboratory European Bioinformatics Institute, Hinxton, Cambridge, UK
6.Genentech, South San Francisco, CA, USA
7.Department of Pathology, ImmunoX Initiative, University of California, San Francisco, CA, USA

Abstract

Over the past decade, single-cell datasets have grown in both size and complexity, enabling the construction of large-scale cell atlases. Technical variability in data generation, also known as batch effects, hinders meaningful comparisons. Although numerous batch correction algorithms have been developed, they often struggle with overcorrection or undercorrection. Here, we review commonly used data cleaning and integration methods. We envision that future frameworks will learn interpretable gene and cell representations and achieve informed modelling of technical and biological variation.

Introduction

The rapid expansion of single-cell RNA sequencing (scRNA-seq) technologies[1] has enabled deeper, more sensitive, and more robust exploration of biological systems. However, batch effects often obscure biologically meaningful signals and pose substantial challenges for downstream analyses. In the absence of a universally-agreed definition[2,3], we define batch effects as sources of variation that are irrelevant to the biological question of interest and that vary systematically across experimental batches, also referred to as nuisance variables. Drawing from common batch correction practices, we broadly classify batch effects into two categories (Fig. 1a), each addressed by a distinct class of methods.

We define Class I batch effects as universally unwanted technical artifacts that arise from known sources of variation introduced by experimental design. Till now, these include factors such as variation of cell-free RNA contamination, mRNA capture efficiency, and sequencing depth across batches. Such artifacts introduce technical noise that can obscure true biological signals and confound interpretation across experiments. Class I batch effects are often addressed during data preprocessing. These include procedures such as data normalization [4], ambient RNA correction [5], and feature selection [6]. While not always labeled as batch correction techniques, these methods effectively reduce unwanted variation across datasets [7] and facilitate more effective data integration [5]. Importantly, although data cleaning methods alone often result in undercorrection and may not fully resolve batch effects, they can offer mechanistic insights into how specific technical factors influence cell- or gene-level data distributions.

In addition, we define Class II batch effects as poorly characterized, batch-specific variation, which may stem from both experimental and biological factors. Experimental sources include variability in library preparation protocols, the use of cells versus nuclei for sequencing, and differences in tissue dissociation methods. Biological sources encompass sampling location within tissues, and inter-donor variability such as sex, age, and genetic background. Importantly, our definition of batch effects is contextual and depends on the biological question being addressed. For example, donor-to-donor variability may be considered a batch effect in studies aiming to identify conserved cell types across individuals, but may instead represent a source of biological interest in investigations of population-level heterogeneity. This class of batch effects is usually modeled by data integration methods, often referred to as batch correction methods. These approaches aim to align biologically similar cells across datasets by modeling and minimizing variation associated with batch covariates. Despite extensive benchmarking efforts[3,4,8–11], integration methods continue to face the challenge of an imbalanced trade-off—either removing true biological variability (overcorrection) or failing to fully eliminate technical noise (undercorrection), as illustrated in Figure 1b. For example, in the construction of the Human Lung Cell Atlas[12], state-of-the-art integration approaches were applied across 14 datasets. However, certain fine-grained immune cell types, such as megakaryocytes identified in the original study by Travaglini et al.,[13] and regulatory T cells identified in Braga et al.[14] became indistinguishable due to overcorrection during integration.

In this Perspective, we summarise the current state of batch correction methods, focusing on both data cleaning techniques that help mitigate batch effects and classic data integration approaches. We discuss challenges arising from the unsupervised training of integration models on a per-dataset basis, as well as limitations inherent to current batch correction practices. Finally, we highlight future directions for the field to support the development of more explicit, informed and generalisable batch correction methods.

Current batch correction methods

In this section, we summarise the development of data cleaning and data integration methods, with a focus on their underlying batch correction strategies. Data cleaning methods typically rely on general assumptions about the statistical signatures of technical artifacts—such as ambient RNA contamination or low-variance features—without requiring batch identity information. These approaches reduce noise factors that contribute to the overall batch effect by modeling these source factors directly, without aligning batches or modeling inter-batch relationships.

In contrast, data integration methods aim to reconcile datasets by modeling the relationships between samples from different batches. These approaches incorporate batch labels as inputs and typically align data in a shared latent space to minimize batch-associated variation while preserving biological signals.

We acknowledge that the boundary between these two categories is not always clear. For example, some normalization and denoising procedures are embedded within integration pipelines, making it difficult to classify them as standalone data cleaning or integration steps. In such cases, we clarify the role of each method on a case-by-case basis throughout the manuscript. Below, we outline representative methods from both categories (**Figure 2**).

Data cleaning methods

In this section we discuss the first category of batch correction methods, often known as data cleaning methods, focusing on sources of batch-related variance that can be explicitly modeled such as ambient RNA correction, sampling depth, and feature selection.

Ambient RNA correction

Ambient RNA correction is crucial [15,16] for droplet-based microfluidic devices that produce large amounts of cell-free RNA that can be incorporated into individual droplets. This issue is particularly pronounced in single-nucleus experiments [17] due to the release of cytoplasmic RNAs during the nuclei extraction step. Computational methods for ambient RNA correction may leverage empty droplets, expression patterns across cell populations.

Methods that rely on empty droplet information, such as SoupX[5], Cellbender[18], FastCAR[19], and scAR[20], share the assumption that cell-containing droplets and empty droplets capture the same ambient RNAs, but differ in their modelling strategy. FastCAR uses a simple thresholding rule to identify highly abundant genes in empty droplets and subtracts the maximum observed counts from all cells, SoupX sequentially estimates and subtracts contamination on a per-cell basis, while both Cellbender and scAR adopt Bayesian frameworks, with CellBender further accounting for barcode swapping.

When raw count matrices are unavailable, tools such as DecontX[21] and scCDC[22] can operate on filtered data. DecontX infers contamination by comparing expression profiles across clusters, whereas scCDC detects and targets only “global contamination-causing genes.”

While downstream steps like highly variable gene selection may buffer against residual ambient noise, remaining contamination can still confound interpretation, especially in single-nucleus data, highlighting the need for systematic benchmarking in these contexts[23].

Sampling depth

Major attention has been paid to variation in sampling depth, including capture efficiency and sequencing depth, which is an issue shared by both bulk and scRNA-seq. This issue has been typically addressed by normalization methods that either model count data using statistical distributions (e.g., poisson or negative binomial) or apply scaling factors and variance-stabilising transformations during preprocessing. For instance, the default data normalisation strategies in two widely used differential expression packages, edgeR[24] and limma[25], developed for bulk RNA-seq or microarrays can be applied to pseudo-bulked scRNA-seq data. edgeR models expression counts using a negative binomial distribution, while limma instead uses voom, which log transforms counts-per-million (CPM) and empirically estimates the mean-variance relationship, a process conceptually related to the delta methods.

However, methods developed for bulk RNA-seq data were not optimised to handle high dropout rates characteristic of scRNA-seq data. Consequently, researchers introduced zero-inflated negative binomial and zero-inflated poisson distribution to better represent scRNA-seq distributions[26], though later studies have challenged their necessity[27–30]. Alternative transformations have since been proposed: scran[31] improves size factor estimation by pooling cells, and Hafemeister and Satija[32] apply Pearson residuals derived from a regularized negative binomial regression that pools information across genes of similar abundance. Several studies have benchmarked normalization methods and pipelines [33–35]. The most recent effort [4] provides a comprehensive theoretical and empirical evaluation of normalization strategies for single-cell RNA-seq data. Their benchmarking results reinforce the common practice of applying a log1p transformation to library size–normalized counts, which often performs as well as or better than more complex alternatives. Notably, the ratio of the pseudo-count to the normalization factor can substantially influence the transformation, particularly for lowly expressed genes [36]. While offering a theoretical perspective on how different normalization strategies affect genes across expression ranges and distributions, the authors[4] emphasize that theoretically optimized methods and hyperparameters frequently offer limited practical benefits. Although their findings favor the shifted logarithm approach—potentially at odds with other studies recommending Pearson residuals—they caution that benchmarking outcomes are highly context-dependent and may vary with dataset and downstream application.

Feature selection

As associations between subtypes of batch effects and gene programs gain increasing attention, it has been noted that feature (gene) selection strongly influences the prominence of batch effects. Selecting a small panel of marker genes, exemplified by traditional multiplex FISH (Fluorescence In Situ Hybridization), can confidently label cell states without concern for batch effects, but this approach depends on prior information and only captures a subset of the transcriptomic variation. Conversely, including all expressed genes in downstream analyses without further corrections often introduces noise and obscures meaningful cell-state features.

To balance coverage and specificity, the concept of highly variable genes (HVGs) was introduced to systematically prioritize biologically informative genes, typically those with high variance-to-mean ratios [37]. This assumes that such genes reflect biologically relevant cellular variation more than technical noise. While HVG-based selection is now a standard preprocessing step, it remains imperfect: selected genes may still be influenced by batch-specific effects or technical artifacts. Alternatively, GiniClust [38] uses Gini index to select feature genes for rare cell types. M3drop [39] can identify genes with unusually high zero prevalence. DeepTree [40] and DUBStepR [41] leverage co-expression patterns between genes to enrich signals against transcriptional noise. These methods show improvement in downstream analyses but have challenges in establishing appropriate cutoffs and often do not reflect the non-linear component of gene expression.

Beyond ad hoc unsupervised feature selection, curated gene sets are also often utilised for data cleaning and batch correction. The level of cellular stress during tissue dissociation is hallmarked by

the 'Immediate early genes' [42] such as those encoding Activator protein 1 (AP-1) (Fos and Jun gene families). Some researchers handle this using a simple linear regression against the average expression profiles of the list of 'stress genes' [43] in the hope of eliminating the stress-induced portion of batch effects. Similarly, cell cycle genes[44, 45, 46, 47, 48] are often treated as unwanted variability, though they can both obscure and reveal important biological states [49]. More recently, Zhou et al.[50] proposed a gene-level perspective on batch correction by introducing the Group Technical Effects (GTE) metric to quantify batch sensitivity at the level of individual genes. They identified highly batch-sensitive genes (HBGs) and showed that removing HBGs can reduce batch effects. This insight supports the view that gene attributes contribute to cell type/state and batch effect to different degrees and that more sophisticated feature selection/weighting strategies are needed (**Figure 3**).

Currently, there is only one benchmarking study[6] on feature selection. It systematically evaluated 13 feature selection strategies across 12 datasets. Their findings reinforce the effectiveness of unsupervised feature selection applied after variance-stabilizing transformations, such as log-normalization or Pearson residuals, approaches that remain widely used due to their consistency across tasks like clustering and classification. In contrast, marker-guided or supervised feature selection approaches showed biased and inconsistent performance across datasets, while batch-aware feature selection methods yielded mixed results. Together, this benchmarking, along with insights from HBGs, highlights the potential and challenges of integrating unsupervised strategies with curated knowledge for feature selection.

Data integration methods

While data cleaning is an essential preprocessing step for removing Class I batch effects, many batch effects in single-cell data remain in the Class II category, batch-specific, poorly characterized, and driven by diverse technical or biological conditions. To address this, data integration methods have been developed to provide more flexible, agnostic, and often non-linear frameworks capable of correcting such complex batch effects.

In this section, we categorize integration methods based on the type of input information they use to model and correct batch variation, with their respective characteristics summarized in **Table 1** and **Figure 2**. These categories include:

1. **Batch covariates conditioned methods** that incorporate user-defined batch covariates—such as sample ID, donor, or technology—via mutual nearest neighbor (MNN) identification, conditional modeling, adversarial objectives, or contrastive learning (CL) to adjust for more structured sources of variation.
2. **Biological covariates conditioned methods** that additionally incorporate biological covariates to avoid overcorrection and improve interpretability; these may use the same modeling strategies as batch covariates conditioned methods but extend them to disentangle biological and technical variation.
3. **Reference-query approaches** that leverage pre-trained models and annotated reference datasets to extract batch-corrected representations for new query data.

Batch covariates conditioned methods

The first category of integration methods aims to eliminate batch effects by leveraging user-defined batch covariates, assuming that variation associated with these covariates reflects technical rather than biological differences. These methods typically require one or multiple batch covariates as input and seek to learn batch corrected cell embeddings through MNN identification, conditional modeling, batch-aware sampling strategies [51, 52], adversarial objectives[53,54], or CL.

In 2018, Haghverdi et al. pioneered the concept of MNN [55], establishing the first dedicated single-cell batch correction method. MNN introduced cell-level batch effect correction by quantifying systematic differences between cells identified as mutual nearest neighbours across batches. Subsequent methods have adopted this foundational concept, advancing scalability, refining learning strategies, or extending the framework to cluster-level relationships. Noteworthy advancements in scalability include fastMNN [56], a computationally optimized implementation developed by the same authors, and scanorama [57] which repurposed computer vision algorithms and still remains one of the top performing methods on complex integration tasks [3]. A related concept is employed in Seurat 3 [58] which identifies "anchor points" between datasets through canonical correlation analysis. Another subset of related methods extends the cell neighbour concept from individual cells to cell clusters, such as scMerge [59], HDMC [60], and BERMUDA [61]. These approaches argue that batch correction based on paired cell cluster neighbours, rather than paired individual cell neighbours, yields both improved robustness and scalability.

A majority of methods identify batch-corrected latent representations for every cell via matrix factorization [58, 57, 62] or deep learning architectures [63, 64,65], in particular, variational autoencoders (VAEs). A prominent example is scVI, which uses a conditional VAE (CVAE) architecture and negative binomial distribution for modelling counts. Batch covariates are passed through both the encoder and decoder networks, enabling the model to condition its reconstruction on batch information. Based on this paradigm, trVAE and related methods [66] extend this framework by incorporating maximum mean discrepancy (MMD) penalties to enforce batch invariance in the latent space. However, benchmarking studies suggest that even the base CVAE architecture used in scVI offers reasonably effective batch correction in the learned cell embeddings, particularly for preserving cell type information.

An orthogonal approach is CL, which seeks to align cells of the same type across batches by minimizing distances in the latent space. This is implemented using a contrastive loss applied to carefully constructed positive and negative pairs, typically defined via data augmentation or label supervision. [67]A recent method, CONCORD[52], introduced a contrastive learning-based data integration method by integrating a probabilistic sampling strategy. This strategy ensures that mini-batch training is not confounded by batch effects and draws cells from local neighborhood graphs to preserve biologically meaningful structure.

Biological covariates conditioned methods

While unsupervised methods aim to learn cell embeddings that are independent of batch covariates, they can inadvertently obscure critical biological signals, such as cell type identity or disease-associated variation, during the integration process. To mitigate this, another category of

integration methods incorporates biological covariates, such as partial cell type labels [68], to guide the learning of cell embeddings through semi-supervised or supervised frameworks.

Early approaches rely on parametric or non-parametric regression models, offering flexible frameworks to incorporate both batch covariates and biological covariates of interest. For instance, ZINB-WaVE [26] and GASPACHO [48] can include biological covariates in the predicting variables and uses zero-inflated negative binomial and gaussian processes respectively for parametrization, thereby preserving biologically meaningful signals during correction. More recently, scANVI[69] and scPoli[70] have extended the CVAE architecture to incorporate partial cell type annotations. The method scANVI, part of the scVI-tools framework, is a probalistic model that use deep learning modules to approximate posterior distributions. scANVI enables semi-supervised learning through conditioning on labelled data and marginalizing over unlabelled data while scPoli calculates a prototype loss computed on labeled cells. Both methods report improved preservation of cell type information without compromising batch correction performance.

With the rise of population-scale scRNA-seq datasets and high-throughput perturbation screens, the goals of data integration have expanded. Merely preserving discrete cell type identities, while still important, is no longer sufficient. More subtle biological signals—such as inter-individual variability or disease-associated shifts—have become increasingly relevant for translational applications like patient stratification and drug response prediction. Recent integration methods are beginning to reflect this shift. For instance, scPoli, though primarily designed for label-guided integration and cell type transfer, also learns sample-level embeddings, which can possibly capture richer biological variation such as disease state or donor-specific effects. Similarly, mrVI[71], another extension of scVI, models cell embeddings with baseline (sample-invariant) cell states and sample-level covariates within a causal generative framework, producing a sample-sample distance matrix per cell.

More broadly, a growing number of methods aim to learn more interpretable representations. These include models that learn cell-level embedding predictive of biological covariates (inVAE[72]), or modularize latent spaces to represent specific covariates (CellDISECT[73], biolord[74], scDisInFact[75]. These approaches represent a shift toward more interpretable, and causally informed data integration strategies that better accommodate complex sources of biological variation.

Reference-query approaches

The third category of integration methods adopts a reference-query mapping framework, which leverages pre-trained, annotated reference atlases to extract batch-corrected biological signals from new datasets. With the emergence of large, high-quality atlases[12] and growing evidence supporting the effectiveness of reference-based mapping for integration and disease-associated cell state discovery[76], this approach has gained substantial traction. Here, we discuss reference-query mapping methods, particularly those involving foundation models.

The use of reference datasets and transfer learning is well-established in fields such as computer vision and natural language processing, and similar strategies are increasingly being adopted in single-cell genomics. Early examples include tools such as Azimuth (from Seurat)[77], singleR [78], and CellTypist[79]. scArches builds on this strategy by minimizing the computational cost of using and

updating reference models and uniquely enabling seamless compatibility with community-contributed pretrained models across diverse architectures.

Community curated datasets and recent architectural advances such as transformers have laid the groundwork for foundation models capable of learning generalizable gene and cell embeddings. Notably, Geneformer[80] was among the first to apply transformer architectures, originally developed for NLP and computer vision, to single-cell data. Trained on tens of millions of cells, Geneformer learns universal gene embeddings that can be fine-tuned for tasks such as regulatory network inference.

Transformer-based gene-centric foundation models primarily learn semantic representations of genes and obtain cell embeddings by aggregating gene embeddings (Geneformer [80], scFoundation [81]) or via a class token (scBERT [82], scGPT [83], UCE [84]) or both (GeneCompass [85], scPRINT [86]). More recently, Tabula [87], introduces a joint training objective that combines gene-wise reconstruction with cell-wise contrastive learning, enabling the simultaneous optimization of both cell and gene embeddings. In gene-centric foundation models, batch correction is typically treated as an optional downstream task, addressed during fine-tuning rather than pretraining.

In contrast, SCimilarity [88] is a cell-centric foundation model that integrates biological supervision into pretraining. It [88] incorporates author-provided cell type labels across a diverse dataset across organs and diseases in its pretraining stage using contrastive learning, enabling more biologically informed representations. [87]

Recent benchmarking studies have highlighted the importance of unbiased assessment of zero-shot capabilities for foundation models. Kedzierska et al. [89] applied scGPT and Geneformer, as well as baseline methods including Harmony and scVI to 5 scRNAseq datasets and followed the Luecken et al.[3] strategy for assessing data integration performance. Their results revealed that the performance of foundation models varied across datasets and did not consistently outperform baseline methods. Liu et al. [90] reviewed seven foundation models and two other data integration methods and benchmarked their batch correction performance on 11 scRNAseq datasets. For batch correction, they observed that Harmony frequently gave the best performance. These findings suggest that the generalizability of foundation models across batches largely stems from the fine-tuning phase, rather than being intrinsically enabled by pre-trained embeddings, challenging the notion that pretrained representations inherently capture batch-invariant structure.

Current gap

Despite rapid advances of data cleaning and integration methods, current batch correction practices have neither eradicated over- and under-correction issues, nor reached reasonable levels of interpretability which are key for thorough data interrogation and future experimental design.

First, there is a lack of guidance on batch covariate selection. It is not uncommon for individual studies to include overlapping covariates, such as simultaneously using dataset identifiers, experimental technology and sample IDs, resulting in unnecessarily complex models and running the risk of overcorrection. This issue is further exacerbated by the lack of robust and standardized evaluation metrics for biological signal preservation. Sikkema et al. [12] addressed this challenge in the

construction of the Human Lung Cell Atlas by prototyping a principal component regression approach for relevant batch covariates[12]. This data-informed strategy for batch covariate selection was shown to effectively improve batch correction performance but its generalizability across datasets and modalities remains to be validated. Till now, the default and widely adopted approach continues to be the use of sample identifiers as the primary, if not only, batch covariate.

Second, a common practice in data integration methods is to fit a model from scratch for each integration project, raising concerns about the validity and robustness of the downstream discoveries. Most current data integration methods infer batch related variance solely from the input data, often using small sample sizes and imbalanced experimental design, which limits their ability in learning robust cell embeddings. Repeatedly training individual models on small, isolated datasets without leveraging or contributing to existing atlases is also an inefficient use of analytical resources. One way to overcome this challenge is through transfer learning from a pre-trained model or an existing high-quality integrated atlases[12,76,97]. However, such atlases are still relatively scarce, limiting the broader adoption of this approach.

Third, current data integration models offer limited insight into the origin and impact of batch effects, or the nature of common technical sources of variability. For example, 5' scRNA-seq, necessary for VDJ-seq (Variable, Diversity, Joining-sequencing) and CITE-seq (Cellular Indexing of Transcriptomes and Epitopes by Sequencing) favours slightly different genes than 3' scRNA-seq[98]. Single nuclei RNA sequencing (snRNA-seq) which is required for certain tissues may deplete specific genes[99] (Fig. 3). Despite these known technical biases, it is often unclear which genes have been corrected for which batch covariates by current data integration methods. Moreover, there is little guidance on how experimental design can be further optimized to minimize batch effects, highlighting a need for greater transparency and interpretability in integration pipelines.

Moreover, existing datasets lack comprehensive metadata information for developing more sophisticated methods. Metadata containing technical information for current publicly available datasets are often error-prone, incomplete and sometimes absent[100]. Even more challenging is the scarcity of biological metadata such as age, disease state, infection state, and relatedness that contribute to the overall variations in the single cell data, often limited by privacy concerns and gaps in data collection. This disorganization and inconsistency in metadata severely compromise the training of novel batch correction models that rely on database-wide datasets.

Compounding these challenges, current single-cell datasets, especially those with diseased cohorts are still in small sizes. This limits the development and application of transformer-based data integration methods, despite promising early results. This stands in contrast with natural language processing (NLP), where large and homogeneous datasets enable strong performance by fine-tuning and reinforcement learning with human feedback (RLHF). For single-cell biology to achieve comparable advances, larger and more diverse datasets are needed, capable of capturing the complexity of biological variation in a manner analogous to how large language models (LLMs) represent human language.

Future perspectives

Current batch effect correction methodologies are hampered by a lack of standardized practices, limited interpretability, and incomplete metadata in many publicly available datasets. Addressing these challenges will require a multifaceted approach. Here, we offer recommendations for improving current batch correction practices, developing future data integration methods, establishing better evaluation frameworks, and conducting retrospective studies.We also encourage community-driven standards for knowledge curation to gradually convert Class II effects to Class I effects.

First, transfer learning from well-annotated cell atlases should be adopted as a standard practice. By leveraging pre-integrated datasets as references, researchers can reduce reliance on dataset-specific integration, while preserving key biological signals. This is particularly relevant for projects with limited biological controls or underrepresented conditions. Dann et al[76] recently demonstrated that leveraging existing atlases to identify disease-associated cell states can enhance batch effect correction, even as atlases evolve. Establishing such practices as a standard step in data analysis pipelines could greatly enhance consistency and interpretability across studies.

Second, data integration methods can be further improved by learning interpretable embeddings for both cells and genes. Cell-centric models like drVI and CellDisect, focus on disentangling technical and biological variation, often producing cell-level embeddings that directly support tasks such as cell clustering, cell type classification, and differential expression analysis. These models are often straightforward to evaluate, as cells can be directly linked to donor identity or experimental condition. In parallel, gene-centric transformer-based models learn embeddings for a fixed biological vocabulary, the named genes, unlocking the potential to capture fundamental gene-gene relationships. These representations can be directly connected to prior knowledge stored in existing literature and curated resources such as gene ontologies, pathway databases and gene set collections. Ideally, future data integration methods will bridge these strengths and provide meaningful representations in both cell and gene levels, thereby enabling more informed batch correction performance.

Third, benchmarking frameworks require substantial updates to keep pace with recent advances in data integration, particularly foundation models and methods that model sample-level covariates or disentangled embeddings. Current evaluation metrics often assume discrete labels and overlook the continuous nature of biological variation, such as developmental gradients or disease severity. Thus, models that perform well on traditional benchmarks can fail to preserve subtle and secondary signals. Newer metrics such as scGraph[101] and Milo scores[6,102] aim to address these limitations,but these approaches require broader community validation and adoption. Additionally, emerging integration methods that model sample-level biological variation introduce new benchmarking challenges. Evaluating properties such as disentanglement and causal effect estimation may help clarify model behavior and guide downstream validation experiments. For foundation models, systematic benchmarking should evaluate batch correction performance across both zero-shot and

task-specific fine-tuning stages, while also considering factors such as tokenization schemes and class imbalance [103,104]. These assessments could offer valuable insights into how design choices influence generalizability and biological fidelity.

Moreover, retrospective meta-analyses offer a promising route toward more interpretable and systematic approaches for batch correction. Curated gene sets that disentangle cell state, cell cycle, and technical noise, along with recent gene-level analyses such as the Group Technical Effects (GTE) framework [50], underscore the value of linking gene programs to known sources of variability (**Figure 3**). As AI tools increasingly facilitate the integration of heterogeneous knowledge sources, the opportunity to codify batch-related patterns is expanding. However, much of the contextual information needed for this remains inconsistently recorded or absent from public datasets. Meta-analysis studies can help close this gap – potentially converting some groups of Class II batch effects to Class I batch effects. Expert curation should play a central role in both conducting these studies and interpreting their findings. In turn, results from such meta-analyses could yield curated gene sets tailored to specific batch scenarios. [105]The results can also flag challenging scenarios where expert opinions are most critical for expert-in-the-loop feedback systems [105] in future foundation model construction.

Finally, future data and analyses must be shared in standardized, FAIR (Findable, Accessible, Interoperable, and Reusable) compliant formats. For example, collaboration between the Human Cell Atlas (HCA) and Cell Ontology is crucial in bridging data-driven insights and structured, community-maintained knowledge resources[106]. By anchoring model development and evaluation to evolving ontologies and shared community frameworks [107], we can ensure that batch correction strategies remain interpretable, comparable, and widely usable.

In summary, the past decade has brought substantial progress in both data cleaning and data integration methods, each addressing batch effects from distinct yet complementary perspectives. Data cleaning approaches are grounded in specific assumptions about technical artifacts and aim to correct well-characterized sources of variation. Data integration methods, by contrast, often deliver better correction performance but can lack interpretability. Looking ahead, we envision that the most effective strategies will integrate the strengths of both paradigms—combining the interpretability and specificity of data cleaning with the scalability and flexibility of modern integration techniques. Achieving this will require not only algorithmic innovation, but also systematic, expert-guided efforts in metadata curation, benchmarking, and knowledge transfer. Collectively, these developments will lay the groundwork for more robust reference atlases and enable reference-based analyses that are resilient to batch effects—ultimately reducing the need for post hoc batch correction.

Acknowledgements

We acknowledge the Teichlab, the He Lab and the EuroBioC2024 members for feedback, and Christina Theodoris and Hanchen Wang for discussion. We acknowledge Guoyang San and Wuhan Anhui Culture Co. for making the illustration of the tree of batch methods. S.L. is supported by

funding given to S.A.T. from the Ageing Biology Foundation. M.L., S.A.T. and J.C.M. is supported by the CZI data ecosystem grant. P.H. is supported by funding from the Department of Pathology and ImmunoX at UCSF.

Author Contributions

S. L. and P. H. conceptualized the study and drafted the manuscript. M. L., J. C. M., and S. A. T. provided feedback on the conceptual framework and manuscript.

Ethics declaration

Competing interests

J.C.M has been an employee of Genentech, Inc. since September 2022. S.A.T. is a scientific advisory board member of ForeSite Labs, OMass Therapeutics, Qiagen, Xaira Therapeutics, a co-founder and equity holder of TransitionBio and Ensocell Therapeutics, a non-executive director of 10x Genomics and a part-time employee of GlaxoSmithKline.

References

1. Svensson, V., Vento-Tormo, R. & Teichmann, S. A. Exponential scaling of single-cell RNA-seq in the past decade. *Nat. Protoc.* **13**, 599–604 (2018).
2. Leek, J. T. *et al.* Tackling the widespread and critical impact of batch effects in high-throughput data. *Nat. Rev. Genet.* **11**, 733–739 (2010).
3. Luecken, M. D. *et al.* Benchmarking atlas-level data integration in single-cell genomics. *Nat. Methods* **19**, 41–50 (2022).
4. Ahlmann-Eltze, C. & Huber, W. Comparison of transformations for single-cell RNA-seq data. *Nat. Methods* **20**, 665–672 (2023).
5. Young, M. D. & Behjati, S. SoupX removes ambient RNA contamination from droplet-based single-cell RNA sequencing data. *Gigascience* **9**, (2020).
6. Zappia, L. *et al.* Feature selection methods affect the performance of scRNA-seq data integration and querying. *Nat. Methods* **22**, 834–844 (2025).
7. Zhang, B. *et al.* A human embryonic limb cell atlas resolved in space and time. *Nature* (2023) doi:10.1038/s41586-023-06806-x.
8. Nguyen, H. C. T., Baik, B., Yoon, S., Park, T. & Nam, D. Benchmarking integration of single-cell

differential expression. *Nat. Commun.* **14**, 1570 (2023).

9. Xi, N. M. & Li, J. J. Benchmarking Computational Doublet-Detection Methods for Single-Cell RNA Sequencing Data. *Cell Syst* **12**, 176-194.e6 (2021).

10. Chazarra-Gil, R., van Dongen, S., Kiselev, V. Y. & Hemberg, M. Flexible comparison of batch correction methods for single-cell RNA-seq using BatchBench. *Nucleic Acids Res.* **49**, e42 (2021).

11. Tran, H. T. N. *et al.* A benchmark of batch-effect correction methods for single-cell RNA sequencing data. *Genome Biol.* **21**, 12 (2020).

12. Sikkema, L. *et al.* An integrated cell atlas of the lung in health and disease. *Nat. Med.* **29**, 1563–1577 (2023).

13. Travaglini, K. J. *et al.* A molecular cell atlas of the human lung from single-cell RNA sequencing. *Nature* **587**, 619–625 (2020).

14. Vieira Braga, F. A. *et al.* A cellular census of human lungs identifies novel cell states in health and in asthma. *Nat. Med.* **25**, 1153–1163 (2019).

15. Smith, K. S. *et al.* Lack of evidence for the transitional cerebellar progenitor. *Nature* vol. 643 E1–E8 (2025).

16. Luo, Z. *et al.* Multimodal validation of the existence of transitional cerebellar progenitors in the human fetal cerebellum. *bioRxiv* 2025.06.01.657310 (2025) doi:10.1101/2025.06.01.657310.

17. Koenitzer, J. R., Wu, H., Atkinson, J. J., Brody, S. L. & Humphreys, B. D. Single-Nucleus RNA-Sequencing Profiling of Mouse Lung. Reduced Dissociation Bias and Improved Rare Cell-Type Detection Compared with Single-Cell RNA Sequencing. *Am. J. Respir. Cell Mol. Biol.* **63**, 739–747 (2020).

18. Fleming, S. J. *et al.* Unsupervised removal of systematic background noise from droplet-based single-cell experiments using CellBender. *Nat. Methods* **20**, 1323–1335 (2023).

19. Berg, M. *et al.* FastCAR: fast correction for ambient RNA to facilitate differential gene expression analysis in single-cell RNA-sequencing datasets. *BMC Genomics* **24**, 722 (2023).

20. Sheng, C. *et al.* Probabilistic machine learning ensures accurate ambient denoising in droplet-

based single-cell omics. *bioRxiv* 2022.01.14.476312 (2022) doi:10.1101/2022.01.14.476312.

21. Yang, S. *et al.* Decontamination of ambient RNA in single-cell RNA-seq with DecontX. *Genome Biol.* **21**, 57 (2020).
22. Wang, W. *et al.* scCDC: a computational method for gene-specific contamination detection and correction in single-cell and single-nucleus RNA-seq data. *Genome Biol.* **25**, 136 (2024).
23. Janssen, P. *et al.* The effect of background noise and its removal on the analysis of single-cell expression data. *Genome Biol.* **24**, 140 (2023).
24. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26**, 139–140 (2010).
25. Ritchie, M. E. *et al.* limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res.* **43**, e47 (2015).
26. Risso, D., Perraudeau, F., Gribkova, S., Dudoit, S. & Vert, J.-P. A general and flexible method for signal extraction from single-cell RNA-seq data. *Nat. Commun.* **9**, 284 (2018).
27. Jiang, R., Sun, T., Song, D. & Li, J. J. Statistics or biology: the zero-inflation controversy about scRNA-seq data. *Genome Biol.* **23**, 31 (2022).
28. Kim, T. H., Zhou, X. & Chen, M. Demystifying “drop-outs” in single-cell UMI data. *Genome Biol.* **21**, 196 (2020).
29. Sarkar, A. & Stephens, M. Separating measurement and expression models clarifies confusion in single-cell RNA sequencing analysis. *Nat. Genet.* **53**, 770–777 (2021).
30. Svensson, V. Droplet scRNA-seq is not zero-inflated. *Nat. Biotechnol.* **38**, 147–150 (2020).
31. Lun, A. T. L., Bach, K. & Marioni, J. C. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. *Genome Biol.* **17**, 75 (2016).
32. Hafemeister, C. & Satija, R. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biol.* **20**, 296 (2019).
33. Heumos, L., Schaar, A.C., Lance, C. et al. Best practices for single-cell analysis across modalities. Nat Rev Genet (2023).

34. Vieth, B., Parekh, S., Ziegenhain, C., Enard, W. & Hellmann, I. A systematic evaluation of single cell RNA-seq analysis pipelines. *Nat. Commun.* **10**, 4667 (2019).

35. Cole, M. B. *et al.* Performance assessment and selection of normalization procedures for single-cell RNA-seq. *Cell Syst.* **8**, 315-328.e8 (2019).

36. Booeshaghi, A. S. & Pachter, L. Normalization of single-cell RNA-seq counts by log(x + 1)† or log(1 + x). *Bioinformatics* **37**, 2223–2224 (2021).

37. Brennecke, P. *et al.* Accounting for technical noise in single-cell RNA-seq experiments. *Nat. Methods* **10**, 1093–1095 (2013).

38. Jiang, L., Chen, H., Pinello, L. & Yuan, G.-C. GiniClust: detecting rare cell types from single-cell gene expression data with Gini index. *Genome Biol.* **17**, 144 (2016).

39. Andrews, T. S. & Hemberg, M. M3Drop: dropout-based feature selection for scRNASeq. *Bioinformatics* **35**, 2865–2867 (2019).

40. He, P. *et al.* A human fetal lung cell atlas uncovers proximal-distal gradients of differentiation and key regulators of epithelial fates. *Cell* **185**, 4841-4860.e25 (2022).

41. Ranjan, B. *et al.* DUBStepR is a scalable correlation-based feature selection method for accurately clustering single-cell data. *Nat. Commun.* **12**, 5849 (2021).

42. Wu, Y. E., Pan, L., Zuo, Y., Li, X. & Hong, W. Detecting Activated Cell Populations Using Single-Cell RNA-Seq. *Neuron* **96**, 313-329.e6 (2017).

43. Neuschulz, A. *et al.* A single-cell RNA labeling strategy for measuring stress response upon tissue dissociation. *Mol. Syst. Biol.* **19**, e11147 (2023).

44. Tirosh, I. *et al.* Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. *Science* **352**, 189–196 (2016).

45. Park, J.-E. *et al.* A cell atlas of human thymic development defines T cell repertoire formation. *Science* **367**, (2020).

46. Buettner, F. *et al.* Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nat. Biotechnol.* **33**, 155–160 (2015).

47. Scialdone, A. *et al.* Computational assignment of cell-cycle stage from single-cell transcriptome data. *Methods* **85**, 54–61 (2015).
48. Kumasaka, N. *et al.* Mapping interindividual dynamics of innate immune response at single-cell resolution. *Nat. Genet.* **55**, 1066–1075 (2023).
49. Luecken, M. D. & Theis, F. J. Current best practices in single-cell RNA-seq analysis: a tutorial. *Mol. Syst. Biol.* **15**, e8746 (2019).
50. Zhou, Y., Sheng, Q., Wang, G., Xu, L. & Jin, S. Quantifying batch effects for individual genes in single-cell data. *Nat. Comput. Sci.* 1–9 (2025).
51. Polański, K. *et al.* BBKNN: fast batch alignment of single cell transcriptomes. *Bioinformatics* **36**, 964–965 (2020).
52.
53. Dincer, A. B., Janizek, J. D. & Lee, S.-I. Adversarial deconfounding autoencoder for learning robust gene expression embeddings. *Bioinformatics* **36**, i573–i582 (2020).
54. Ge, S., Wang, H., Alavi, A., Xing, E. & Bar-Joseph, Z. Supervised adversarial alignment of single-cell RNA-seq data. *J. Comput. Biol.* **28**, 501–513 (2021).
55. Haghverdi, L., Lun, A. T. L., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat. Biotechnol.* **36**, 421–427 (2018).
56. Lun, A. fastMNN description. https://marionilab.github.io/FurtherMNN2018/theory/description.html (2019).
57. Hie, B., Bryson, B. & Berger, B. Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. *Nat. Biotechnol.* **37**, 685–691 (2019).
58. Stuart, T. *et al.* Comprehensive Integration of Single-Cell Data. *Cell* **177**, 1888-1902.e21 (2019).
59. Lin, Y. *et al.* scMerge leverages factor analysis, stable expression, and pseudoreplication to merge multiple single-cell RNA-seq datasets. *Proc. Natl. Acad. Sci. U. S. A.* **116**, 9775–9784 (2019).

60. Wang, X., Wang, J., Zhang, H., Huang, S. & Yin, Y. HDMC: a novel deep learning-based framework for removing batch effects in single-cell RNA-seq data. *Bioinformatics* **38**, 1295–1303 (2022).

61. Wang, T. *et al.* BERMUDA: a novel deep transfer learning method for single-cell RNA sequencing batch correction reveals hidden high-resolution cellular subtypes. *Genome Biol.* **20**, 165 (2019).

62. Welch, J. D. *et al.* Single-cell multi-omic integration compares and contrasts features of brain cell identity. *Cell* **177**, 1873-1887.e17 (2019).

63. Li, X. *et al.* Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis. *Nat. Commun.* **11**, 2338 (2020).

64. Lakkis, J. *et al.* A joint deep learning model enables simultaneous batch effect correction, denoising, and clustering in single-cell transcriptomics. *Genome Res.* **31**, 1753–1766 (2021).

65. Moinfar, A. A. & Theis, F. J. Unsupervised deep disentangled representation of single-cell omics. *bioRxiv* 2024.11.06.622266 (2024) doi:10.1101/2024.11.06.622266.

66. Amodio, M. *et al.* Exploring single-cell data with deep multitasking neural networks. *Nat. Methods* **16**, 1139–1145 (2019).

67. Richter, T., Bahrami, M., Xia, Y., Fischer, D. S. & Theis, F. J. Delineating the effective use of self-supervised learning in single-cell genomics. *Nat. Mach. Intell.* **7**, 68–78 (2024).

68. Xu, C. *et al.* Automatic cell-type harmonization and integration across Human Cell Atlas datasets. *Cell* **186**, 5876-5891.e20 (2023).

69. Xu, C. *et al.* Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Mol. Syst. Biol.* **17**, e9620 (2021).

70. De Donno, C. *et al.* Population-level integration of single-cell datasets enables multi-scale analysis across samples. *Nat. Methods* **20**, 1683–1692 (2023).

71. Boyeau, Pierre, Justin Hong, Adam Gayoso, Martin Kim, José L. McFaline-Figueroa, Michael I. Jordan, Elham Azizi, Can Ergen, and Nir Yosef. "Deep generative modeling of sample-level heterogeneity in single-cell genomics." Nature Methods (2025): 1-11.

72. Aliee, H. *et al.* inVAE: Conditionally invariant representation learning for generating multivariate single-cell reference maps. *bioRxiv* 2024.12.06.627196 (2024) doi:10.1101/2024.12.06.627196.
73. Megas, S. *et al.* Integrating multi-covariate disentanglement with counterfactual analysis on synthetic data enables cell type discovery and counterfactual predictions. *bioRxiv* 2025.06.03.657578 (2025) doi:10.1101/2025.06.03.657578.
74. Piran, Z., Cohen, N., Hoshen, Y. & Nitzan, M. Disentanglement of single-cell data with biolord. *Nat. Biotechnol.* (2024) doi:10.1038/s41587-023-02079-x.
75. Zhang, Z., Zhao, X., Bindra, M., Qiu, P. & Zhang, X. scDisInFact: disentangled learning for integration and prediction of multi-batch multi-condition single-cell RNA-sequencing data. *Nat. Commun.* **15**, 912 (2024).
76. Dann, E. *et al.* Precise identification of cell states altered in disease using healthy single-cell references. *Nat. Genet.* **55**, 1998–2008 (2023).
77. Stuart, T. *et al.* Comprehensive Integration of Single-Cell Data. *Cell* **177**, 1888-1902.e21 (2019).
78. Aran, D. *et al.* Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat. Immunol.* **20**, 163–172 (2019).
79. Domínguez Conde, C. *et al.* Cross-tissue immune cell analysis reveals tissue-specific features in humans. *Science* **376**, eabl5197 (2022).
80. Theodoris, C. V. *et al.* Transfer learning enables predictions in network biology. *Nature* **618**, 616–624 (2023).
81. Hao, M. *et al.* Large-scale foundation model on single-cell transcriptomics. *Nat. Methods* **21**, 1481–1491 (2024).
82. Yang, F. *et al.* scBERT as a large-scale pretrained deep language model for cell type annotation of single-cell RNA-seq data. *Nat. Mach. Intell.* **4**, 852–866 (2022).
83. Cui, H. *et al.* scGPT: toward building a foundation model for single-cell multi-omics using generative AI. *Nat. Methods* (2024) doi:10.1038/s41592-024-02201-0.
84. Rosen, Y. *et al.* Universal cell embeddings: A foundation model for cell biology. *bioRxiv*

2023.11.28.568918 (2023) doi:10.1101/2023.11.28.568918.

85. Yang, X. *et al.* GeneCompass: deciphering universal gene regulatory mechanisms with a knowledge-informed cross-species foundation model. *Cell Res.* **34**, 830–845 (2024).

86. Kalfon, J., Samaran, J., Peyré, G. & Cantini, L. scPRINT: pre-training on 50 million cells allows robust gene network predictions. *Nat. Commun.* **16**, 3607 (2025).

87. Ding, J. *et al.* Toward a privacy-preserving predictive foundation model of single-cell transcriptomics with federated learning and tabular modeling. *bioRxiv* 2025.01.06.631427 (2025) doi:10.1101/2025.01.06.631427.

88. Heimberg, G. *et al.* A cell atlas foundation model for scalable search of similar human cells. *Nature* **638**, 1085–1094 (2025).

89. Kedzierska, K. Z., Crawford, L., Amini, A. P. & Lu, A. X. Zero-shot evaluation reveals limitations of single-cell foundation models. *Genome Biol.* **26**, 101 (2025).

90. Liu, T., Li, K., Wang, Y., Li, H. & Zhao, H. Evaluating the utilities of foundation models in single-cell data analysis. *bioRxivorg* 2023.09.08.555192 (2024) doi:10.1101/2023.09.08.555192.

91. Johnson, W. E., Li, C. & Rabinovic, A. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics* **8**, 118–127 (2007).

92. Korsunsky, I. *et al.* Fast, sensitive and accurate integration of single-cell data with Harmony. *Nat. Methods* **16**, 1289–1296 (2019).

93. Lotfollahi, M., Wolf, F. A. & Theis, F. J. scGen predicts single-cell perturbation responses. *Nat. Methods* **16**, 715–721 (2019).

94. Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nat. Methods* **15**, 1053–1058 (2018).

95. Lotfollahi, M., Naghipourfar, M., Theis, F. J. & Wolf, F. A. Conditional out-of-distribution generation for unpaired data using transfer VAE. *Bioinformatics* **36**, i610–i617 (2020).

96. Zeng, Y. *et al.* CellFM: a large-scale foundation model pre-trained on transcriptomics of 100 million human cells. *Nat. Commun.* **16**, 4679 (2025).

97. Lotfollahi, M., Yuhan Hao, Theis, F. J. & Satija, R. The future of rapid and automated single-cell data analysis using reference mapping. *Cell* **187**, 2343–2358 (2024).

98. Hsu, J. *et al.* Comparing 10x Genomics single-cell 3' and 5' assay in short-and long-read sequencing. *bioRxiv* 2022.10.27.514084 (2022) doi:10.1101/2022.10.27.514084.

99. Thrupp, N. *et al.* Single-Nucleus RNA-Seq Is Not Suitable for Detection of Microglial Activation Genes in Humans. *Cell Rep.* **32**, 108189 (2020).

100. Huang, Yu-Ning, Pooja Vinod Jaiswal, Anushka Rajes, Anushka Yadav, Dottie Yu, Fangyun Liu, Grace Scheg et al. "The systematic assessment of completeness of public metadata accompanying omics studies in the Gene Expression Omnibus data repository." Genome Biology 26, no. 1 (2025): 274.

101. Wang, H., Leskovec, J. & Regev, A. Limitations of cell embedding metrics assessed using drifting islands. *Nat. Biotechnol.* 1–4 (2025).

102. Dann, E., Henderson, N. C., Teichmann, S. A., Morgan, M. D. & Marioni, J. C. Differential abundance testing on single-cell data using k-nearest neighbor graphs. *Nat. Biotechnol.* **40**, 245–253 (2022).

103. Alsabbagh, A. R. *et al.* Foundation models meet imbalanced single-cell data when learning cell type annotations. *bioRxiv* 2023.10.24.563625 (2023) doi:10.1101/2023.10.24.563625.

104. Maan, H. *et al.* Characterizing the impacts of dataset imbalance on single-cell data integration. *Nat. Biotechnol.* **42**, 1899–1908 (2024).

105. Cheng, F., Keller, M. S., Qu, H., Gehlenborg, N. & Wang, Q. Polyphony: An interactive transfer learning framework for single-cell data analysis. *IEEE Trans. Vis. Comput. Graph.* **29**, 591–601 (2023).

106. Osumi-Sutherland, D. *et al.* Cell type ontologies of the Human Cell Atlas. *Nat. Cell Biol.* **23**, 1129–1135 (2021).

107. Metadata. Human Cell Atlas Data Portal https://data.humancellatlas.org/metadata. (2025)

Figures

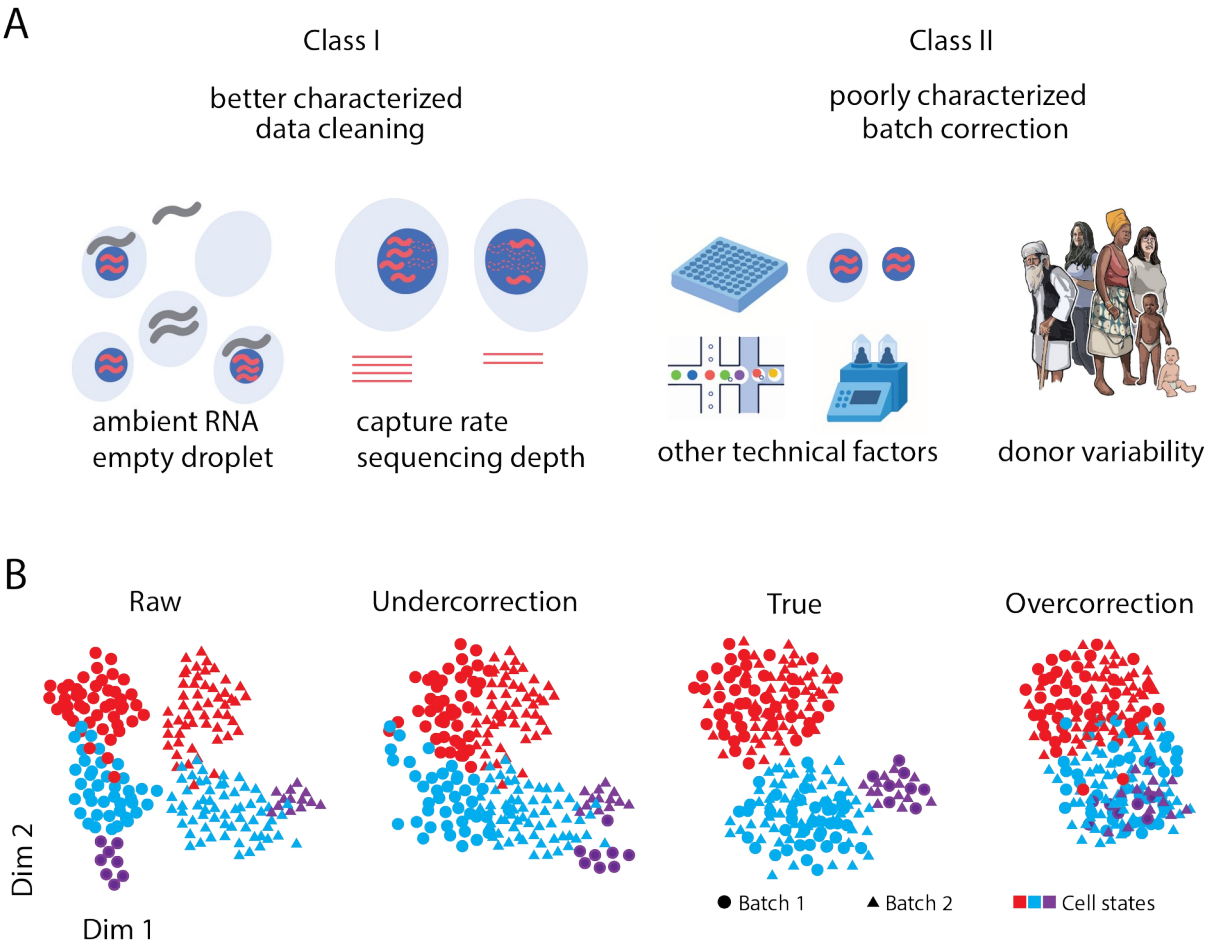
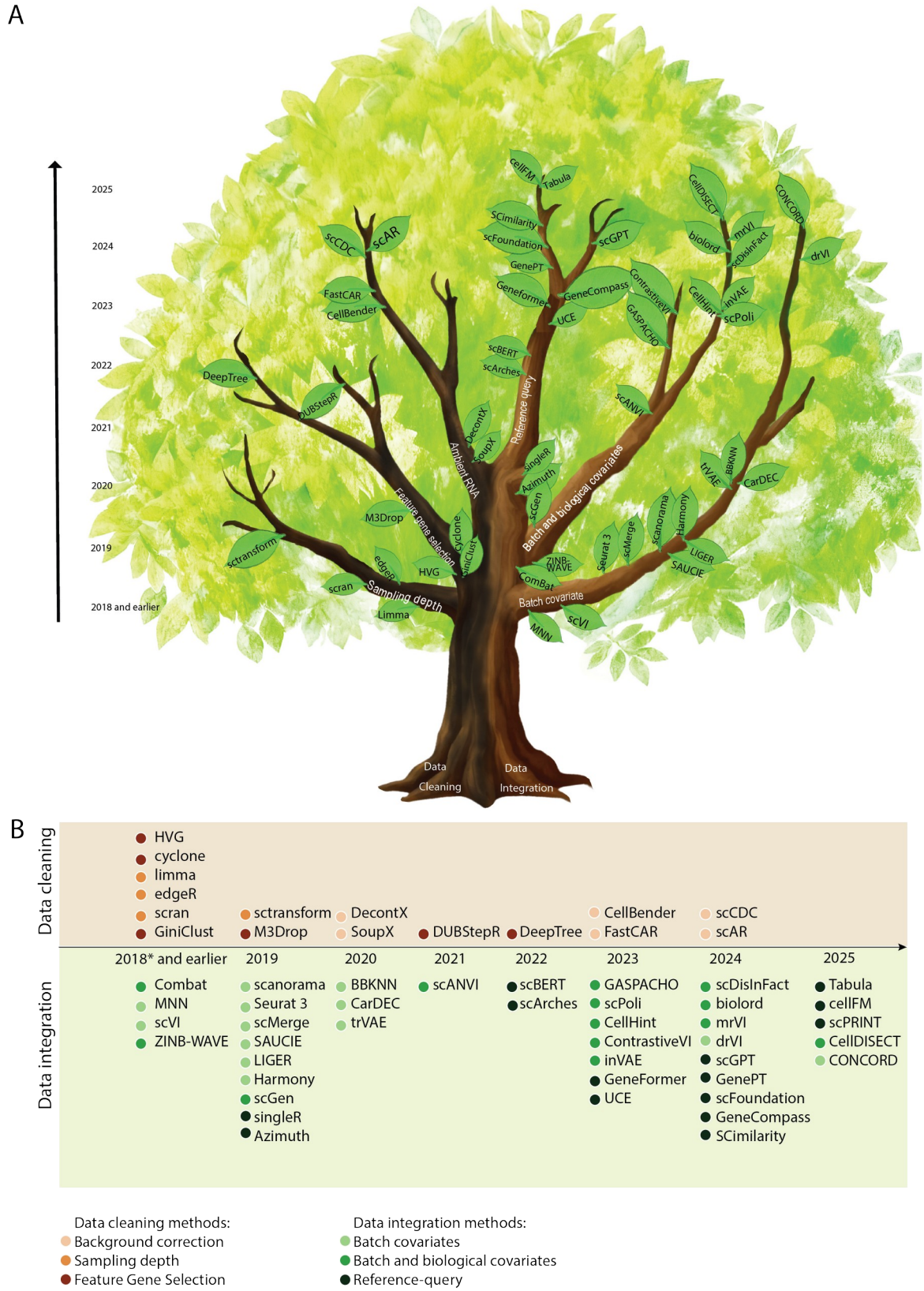


Figure 1. Overview of the batch effect categories and limitations of correction methods. A) Classes of batch effects and their correction methods. Typically, Class I includes ambient RNA, empty droplet calling, capture efficiency and sequencing depth; Class II includes other technical factors such as plate vs droplet platforms, single-cell vs single-nucleus sequencing, sample processing protocols, and more broadly donor variability. B) Illustration of batch correction outcomes in a two-dimensional feature space. *Raw*: Uncorrected data, showing batch-driven separation of biologically similar cells. *Undercorrection*: Incomplete removal of batch effects, resulting in residual technical variation across batches. *Overcorrection*: Excessive correction that removes true biological differences, leading to artificial alignment of distinct cell types. *True*: Ideal outcome in which batch effects are effectively removed while preserving genuine biological variation. *Dim 1*: Dimension 1. *Dim 2*: Dimension 2.



669

670 **Figure 2. Overview of major computational methods for mitigating batch effects.**

A) *Vertical layout*: Methods are arranged along a vertical timeline according to their publication date. Data integration methods (left) and data cleaning methods (right) are grouped into subcategories using tree-like branching structures to reflect methodological lineages and distinctions.**B)** *Horizontal layout*: An alternative view of the same methods arranged along a horizontal timeline. Data cleaning methods (top) and data integration methods (bottom) are divided into subcategories, with each subcategory represented by a distinct colour.

Figure 2

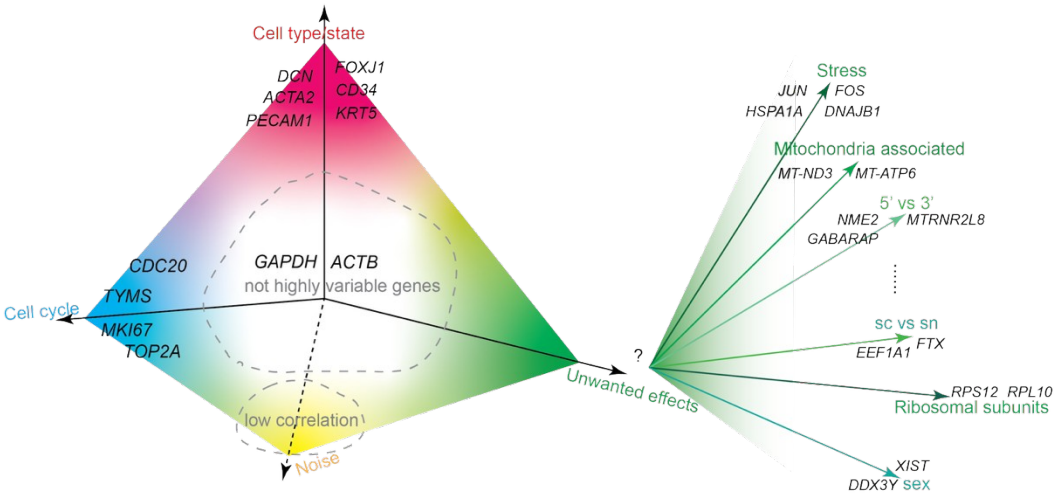


Figure 3. Conceptual diagram of a gene's contribution to the overall transcriptomic signature. Genes are mapped into a multidimensional space (left) with weightings in the direction of cell type/state, cell cycle, noise, or technical effects. Representative genes in different regions of the hyperspace are highlighted in italic. The dashed circle in the centre roughly represents where highly variable genes are depleted. The technical effects are further decoupled into dimensions (right) based on the specific sources. *sc*: single cell. *sn*: single nucleous.

Tables

Table 1: Summary of data integration methods. Each row summaries a method. The Usability notes* column mean: 4 means published packages with documentation and tutorials, demonstrated continued maintenance through code updates or ecosystem; 3 means recently published packages with documentation and tutorials with code updates within last year; 2 means published packages with documentation and tutorials, but no code updates within last year; 1 means published packages or scripts with minimum documentation.

Name	Publication year	API language	Data preprocessing	Input batch variables	Input biological variables	Output	GPU acceleration	Usability notes*

							applicable?	
Combat [91]	2007	R	Raw count matrix	One categorical	Multiple categorical or continuous	expression matrix	no	4
MNN [55]	2018	R	Library size corrected & log1p normalized expression matrix	One categorical		expression matrix	no	2
fastMNN [56]	2018	R	Library size corrected & log1p normalized expression matrix	One categorical		expression matrix	no	4
scanorama [57]	2019	python	Library size corrected & log1p normalized expression matrix	One categorical		cell embedding, expression matrix	no	4
Seurat 3 [58]	2019	R	Library size corrected & log1p normalized expression matrix	One categorical		expression matrix	no	4
scMerge [59]	2019	R	Library size corrected & log1p normalized expression matrix	One categorical	Cell type label	expression matrix	no	3
HDMC [60]	2021	require both python and R	Custom preprocessing scripts	One categorical		cell embedding	yes	1
BERMUDA [61]	2019	require both python and R	Custom preprocessing scripts	One categorical		cell embedding	yes	1
ZINB-WAVE [26]	2018	R	Raw count matrix	Multiple categorical or continuous	Multiple categorical or continuous	cell embedding	no	3
GASPACHO [48]	2023	R	Raw count matrix	Multiple categorical or continuous	Multiple categorical or continuous	cell embedding	no	2
LIGER [62]	2019	R	Custom preprocessing scripts	One categorical		cell embedding	no	4
BBKNN [51]	2020	python (R via reticulate	Cell embedding	One categorical		KNN graph	no	2

		optional)						
SAUCIE [66]	2019	python	Cell embedding or normalized matrix	One categorical		cell embedding, expression matrix	yes	2
Harmony [92]	2019	R	Cell embedding	Multiple categorical		cell embedding	no	4
scGen [93]	2019	python	Library size corrected & log1p normalized expression matrix	One categorical	cell type label	cell embedding, expression matrix	yes	2
scVI [94]	2018	python	Raw count matrix	Multiple categorical or continuous		cell embedding, expression matrix	yes	4
trVAE [95]	2020	python	Raw count matrix or normalized data	One categorical		cell embedding, expression matrix	yes	4
AD-AE [53]	2020	python	Custom preprocessing scripts	One categorical or continuous		cell embedding	yes	1
DESC [63]	2020	python	Library size corrected & log1p normalized & scaled expression matrix			cell embedding	yes	2
CarDEC [64]	2021	python	Custom preprocessing scripts	One categorical		cell embedding, expression matrix	yes	2
scDGN [54]	2021	python	Normalized expression matrix	One categorical	Cell type label	cell embedding	yes	1
scANVI [69]	2021	python	Raw count matrix	Multiple categorical or continuous	Cell type label	cell embedding and cell type predictions	yes	4
scPoli [70]	2023	python	Raw count matrix	Multiple categorical or continuous	Cell type label	cell embedding, cell type predictions and sample embedding	yes	4
InVAE [72]	2023	python	Raw count matrix	Multiple categorical or continuous	Multiple categorical or continuous	cell embedding	yes	3
biolord [74]	2024	python	Raw count matrix	Multiple categorical or continuous	Multiple categorical or continuous	cell embedding, expression matrix	yes	3
scDisinFact [75]	2024	python	raw count matrix	One categorical	One categorical	cell embedding, expression matrix	yes	3

CellHint [68]	2023	python	library size corrected & log1p normalized expression matrix	One categorical	Cell type label	KNN graph and cell type predictions	no	3
CellDisect [73]	2025	python	raw count matrix	Multiple categorical or continuous	Multiple categorical or continuous	cell embedding, expression matrix	yes	3
mrVI [71]	2024	python	raw count matrix	Multiple categorical or continuous	One categorical	cell embedding, sample-sample distance matrix per cell	yes	4
drVI [65]	2024	python	raw count matrix	Multiple categorical or continuous		cell embedding	yes	4
GeneFormer [80]	2023	python	custom tokenization	None (pretrained), optional downstream		gene and cell embeddings	yes	4
scBERT [82]	2022	python	custom tokenization	None (pretrained), optional downstream		gene and cell embeddings	yes	2
scGPT [83]	2024	python	custom tokenization	None (pretrained), optional downstream		gene and cell embeddings	yes	3
cellFM [96]	2025	python	custom tokenization	None (pretrained), optional downstream		gene and cell embeddings	yes	3
scFoundation [81]	2024	python	custom preprocessing scripts	None (pretrained), optional downstream		gene and cell embeddings	yes	3
GeneCompass [85]	2024	python	custom preprocessing scripts	None (pretrained), optional downstream		gene and cell embeddings	yes	3
SCimilarity [88]	2024	python	custom preprocessing scripts	Study ID (pretrained)	Cell type label	cell embedding	yes	3
UCE [84]	2023	python	custom tokenization	None (pretrained), optional downstream		cell embedding	yes	3
Tabula [87]	2025	python	custom tokenization	None (pretrained), optional		gene and cell embeddings	yes	3

				downstream				
CONCORD [52]	2025	python	library size corrected & log1p normalized expression matrix	One categorical		cell embeddings	yes	3