

Data-Centric Framework for Testing and Benchmarking of scRNA-seq Pipelines

Konghao Zhao, Nathan P. Whitener and Natalia Khuri

Department of Computer Science

Wake Forest University

Winston Salem, North Carolina, USA

natalia.khuri@wfu.edu

Abstract—Computational methods for the analysis of single-cell RNA sequencing data are driving advances in personalized medicine. Given the rapid integration of these complex tools into research and development pipelines, the systematic and reliable testing of computational methods is critical to ensure the integrity of derived insights. However, a significant gap exists in appropriate benchmarks and innovative frameworks for the systematic and rigorous evaluation of existing and emergent digital technologies. In this work, we present a new framework, powered by a software package called *scrnabench*, to conduct systematic testing and benchmarking of computational tools. This framework supports the development and evaluation of digital health technologies by ensuring reliable, stable, and trustworthy results. The package can be used to select the most appropriate method for data analyses, to evaluate emergent tools, and to identify the strengths and weaknesses of existing software. In addition, we leverage software engineering techniques of metamorphic testing to help uncover implementation errors, faults, and anomalies and increase trust in AI-driven techniques. The *scrnabench* package is open-source, accessible and extendable, and we demonstrate its unique features in several use-case scenarios.

Index Terms—benchmarking, metamorphic testing, single-cell RNA sequencing

I. INTRODUCTION

Single-cell RNA sequencing (scRNA-seq) has become increasingly common in biomedical research due to the advances in the integrated robotic systems and molecular barcoding. These advances enabled cost-effective transcriptional profiling of individual cells in several tissues, organs, and diseases. The integrated systems for scRNA-seq differ in throughput, sensitivity, specificity and cost. Consequently, there exists a wide array of computational methods for pre-processing and analysis of scRNA-seq data. These methods have various strengths and limitations, and prior benchmarking studies provided some insights into the best combination of various methods.

Effective, reproducible, and unbiased benchmarking of scRNA-seq methods still faces several challenges. The first challenge is the paucity of good benchmark datasets. Good benchmarks should be accessible, consistent, reproducible, diverse, scalable and relevant [1]–[3]. In addition, good benchmarks should be suitable for the nuanced assessments of the individual steps as well as of the entire data analysis pipeline [4]. Synthetic datasets are often used to evaluate

existing and emergent scRNA-seq methods because a large number of datasets of various sizes, dimensionality and complexity can be generated. Additionally, “ground truth” signals can be artificially simulated, such as specific cell-types or differentially-expressed genes, for example. Despite significant improvements, simulations of single-cell gene expression are unable to fully capture and generate non-trivial biological signals. Therefore, benchmarking studies that use synthetic datasets often overestimate performance of data analysis methods [2], [3], [5], [6]. Benchmarking with experimental data may provide better practical insights. However, experimental datasets are limited in number and size, lack “ground truths” and have been over-used [3], [7]. Over-used benchmarks are those experimental datasets that have been used for the development and tuning of existing scRNA-seq methods.

The second challenge is methodological. The majority of scRNA-seq data analysis methods use machine learning, including deep learning networks. To date, their development, validation and benchmarking have been largely model-centric. In model-centric development [8], [9], the datasets are kept unchanged, while the developers iterate the model to improve its performance. The iterations include but are not limited to different architectures, optimization techniques, cost functions, and so on. Similarly, model-centric benchmarking focuses on the comparison and ranking of pre-tuned models on fixed (static) datasets. The main objective of model-centric approaches is to evaluate the generalizability, estimated by performance in cross-validation or hold-out validation. As a result, benchmarking studies often fail to provide useful recommendations because no single method outperforms other methods on all benchmarks. Such insights are of limited value because they reveal neither the specific characteristics of the datasets nor the specific properties of the methods, yielding better performance. Moreover, model-centric validation and benchmarking does not address the expectations of scRNA-seq practitioners regarding the reliability, robustness, and trustworthiness of the results. A scRNA-seq practitioner expects that a better data transformation will improve model’s performance, or that results will not change if the dataset is re-ordered, for example.

The third challenge is due to the oracle problem [10], which makes it difficult to systematically test the implementation quality of various scRNA-seq methods. In machine learning

(ML), in particular, it is difficult to detect implementation errors because it is impossible or very expensive to verify the correctness of computed outputs for given inputs. Testing is different from formal algorithm’s analysis and performance validation. Even if some of the scRNA-seq discoveries are experimentally validated, or algorithm’s correctness has been formally proven, without systematic testing, it is impossible to guarantee that the implementation is free of defects, faults and anomalies. In scRNA-seq, the oracle problem is amplified because data analysis proceeds in a sequence of steps, such that the output of one computational method is used as the input to the next step, without safeguards. Lack of testing and over-reliance on the general-purpose implementation of ML algorithms, can lead to unreliable results, software failures, users’ frustrations, and expensive consequences.

In this work, we propose a data-centric framework for the validation, testing and benchmarking of scRNA-seq methods, and make the following contributions. First, we formulate and implement a new framework for the evaluation of scRNA-seq methods, inspired by an established software engineering technique called metamorphic testing. The purpose of the proposed framework is to uncover hidden errors and instabilities resulting from data quality issues, which are not revealed through standard validation techniques, including experimental validation. Second, we establish baseline benchmarks for the evaluation of reliability and stability of methods for cluster analysis and cell-type prediction by validating their behavior under diverse conditions. To that end, we identify and formulate metamorphic relations for the proposed framework. A metamorphic relation is a specific type of relationship between inputs and outputs of a program under test that holds even when certain perturbations are applied to the inputs [4], [11], [12]. Thus, a metamorphic relation defines an expected consistency of a program. Central to the proposed framework is the ability to rapidly generate diverse and realistic benchmarks with known “ground truth”. In our third contribution, we implement and release into the public domain, a software package called *scrnabench* (Fig. 1) for automatic generation and reproducible preparation of metamorphic benchmarks [13].

We emphasize that the objective of metamorphic testing and benchmarking is not to find the best performing method but to uncover data-dependent limitations or anomalies of existing or emergent tools. As scRNA-seq technology is poised to become a routine tool of biomedical and clinical research, computational tools must be tested not only for their ability to derive biological insights but also for their robustness and consistency. The overarching aim of this work is to bring attention to the need for the development and implementation of alternative evaluation approaches, that go beyond “competition” benchmarking. In “competition” benchmarking, emergent methods are compared to the current “best-performing” techniques that have been tuned to a handful of the datasets, thus producing the “best performance” on these specific datasets. Surpassing the performance of the state-of-the-art method provides a “necessary but not sufficient” condition of superiority. A good method should also be self-

consistent and robust to slight changes in the input data. In this work, we bring awareness to this fact, and propose a new framework for examining each method, on its own merit.

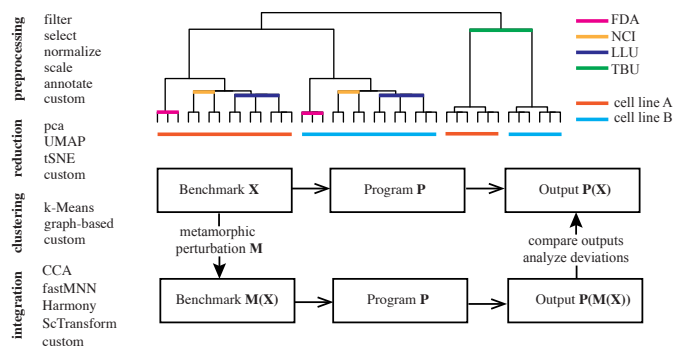


Fig. 1. Overview of *scrnabench* package data and functions. Data comprise cell line A (orange, cancer) and cell line B (blue, normal), sequenced and pre-processed in 4 centers: United States Food and Drug Administration (FDA), National Cancer Institute (NCI), Loma Linda University (LLU), and TBU International (TBU). Standard scRNA-seq data analysis methods are implemented and programmatic support is provided for data imports from custom methods and for generation of metamorphic tests.

While the scope of this work is on computational methods for the analysis of scRNA-seq data, the general principles of metamorphic testing and the data-centric architecture of the *scrnabench* framework are domain-agnostic. The modules are implemented for general applicability to the analysis of high-dimensional datasets used to advance precision medicine, such as those collected by the proteomic, metabolomic, and imaging technologies. In all applications, algorithmic reliability, consistency and validity are important. Therefore, our framework and package may be used to improve transparency and trustworthiness across the entire spectrum of biomedical data science, including drug discovery and development.

The remainder of this work is organized as follows. In Section II, we review prior and relevant work. The proposed data-centric framework is described in Section III. Results of computational experiments are presented and analyzed in Section IV. Finally, we summarize our findings and propose future extensions of this work in Section V.

II. PRIOR AND RELEVANT WORK

Single-cell transcriptomes provide fine-grained information about the abundances and types of RNA transcripts in individual cells. This information can be used to annotate cell-types and cell-states, which is of great value to studying human diseases and disorders. Detection of cell-types in sequenced scRNA data is an important task of data analysis methods. The original methods for cell-type detection rely on cluster analysis, an unsupervised ML technique. The ever-growing catalogues of sequenced cells make it possible to use supervised ML, in particular neural networks, to annotate cell types in new datasets [14]–[16].

Prior works analyzed strengths and limitations of scRNA-seq data analysis methods, and described several challenges

of effective benchmarking, including the lack of good benchmarks and of consistent and reproducible baselines [5], [17]–[30].

Methodologically, the main goal of prior scRNA-seq benchmarking has been to assess the generalizability of methods and models. Hence, benchmarking studies perform cross-validation of internal, external, or stability performance, using general-purpose validation techniques and metrics [17]–[30]. Internal validity of clustering, for example, is measured by compactness, connectedness or separation of clusters, whereas external validity is assessed by the agreement between predicted and “ground truth” cell-types, for example [31]–[33]. Validation of stability, which emphasizes reproducibility of results alongside generalizability, is most relevant to our research. A stable method should yield similar results when applied to data drawn from the same distribution. Although the importance of stability is widely acknowledged [34]–[36], stability benchmarking is rarely performed. Those that do, are based on sampling from the original data, including sampling with or without replacement and cross-validation [37]. The need for new methods and frameworks for the assessment of reliability, stability and robustness of ML workflows and applications is growing. The development of alternative benchmarking approaches has been described as one of the grand challenges of scRNA-seq data analysis [3]. Recently, for example, a Machine Learning Technology Readiness Levels framework was published [38]. The framework outlined guidelines for the implementation of ML workflows and highlighted major concerns about current development of ML applications which are trained and validated on a handful of overused, static datasets, without effective measures and testing for future scenarios and user expectations.

One of the reasons for the lack of attention to testing and benchmarking of properties beyond generalizability, may be the oracle problem. All ML applications, including those used in scRNA-seq data analysis, suffer from the oracle problem. The oracle problem refers to situations, where it is not possible to construct input-output pairs for testing of the correctness of an implementation. In cluster analysis, for example, there is no “ground truth” not only because clusters are not known but also because there may be many, equally valid clusters in the same dataset. Therefore, it may seem impossible to determine whether an implementation of a specific cluster analysis algorithm is correct or not. Importantly, proofs of algorithm’s correctness do not guarantee that its implementation or application is correct. The situation becomes even more daunting when different models are combined into a single pipeline, in which unreliable final outputs may be generated due to stochasticity or failure in any of the pipeline’s steps. There exist, however, a large body of research about effective approaches to software testing and validation in the presence of the oracle problem.

To the best of our knowledge, our work is the first proposal for the use of a software engineering technique, known as metamorphic testing [4], [11], [12], [39], in the validation and benchmarking of scRNA-seq data analysis methods.

In addition, our proposed framework shifts the focus from model-centric to data-centric benchmarking. In doing so, we demonstrate, for example, that over-correction for the batch effects during data integration may lead to unstable cell-type predictions for between 1% to 7% of the test data without any change in the generalization errors.

III. PROPOSED DATA-CENTRIC FRAMEWORK

The framework is inspired by a software engineering technique called metamorphic testing. Its purpose is to expose and quantify inconsistencies in the outputs of methods by perturbing their inputs. At its core, is a set of metamorphic relations between inputs and outputs for a system under test. The system under test, in the context of scRNA-seq, may be a cell-type classification model, a cluster model, or a data integration method, for example. The metamorphic relationship between inputs and outputs is formulated qualitatively or quantitatively. For example, a classifier should not arbitrarily change cell-type predictions on test data when the order of cells or genes is permuted in the input dataset. This is an example of a qualitative metamorphic relation. We can also measure the differences quantitatively by computing the absolute or percent change in performance of baseline and metamorphic tests. Importantly, a metamorphic relation may not be a necessary property of an algorithm. Rather it formulates the expectations of a practitioner who uses its implementation. End user may be concerned about the change of predicted cell-types due to re-ordering of the input data, for example. However, invariance to data permutation is not a necessary property of K-Means algorithm because of tie-breaking rules when finding nearest neighbors.

We formulate several metamorphic relations and implement `scrnabench` package to automatically generate metamorphic benchmarks for tests of these relations [13]. These relations are formulated for two specific tasks of scRNA-seq data analysis, namely, cluster analysis and classification. Both of these tasks are used to detect cell-types in sequenced data. Metamorphic relation 1 (MR1) permutes cells of the original dataset randomly. MR2 applies a linear transformation to raw gene expressions by multiplying each gene expression by 2 and adding 1 to it. MR3 duplicates one randomly chosen cell in a dataset. MR4 permutes the order of genes. MR5 adds one extra gene, whose expression counts are all set to 1. MR6 changes the sign of all expression counts from positive to negative. MR7 removes one randomly selected cell from a dataset and MR8 selects a representative coresets of data [40], [41]. Qualitatively, we expect to see no change in the outputs for any of these metamorphic relations. Again, we emphasize the difference between testing for biological accuracy compared to testing for correct implementation and robustness. Consider MR6, for example. Raw counts are non-negative, and it is therefore expected that data analysis methods that rely on the non-normalized expression counts, should raise an implementation error message when there are negative values in the input. However, normalization and batch-correction can produce some negative values in the processed data. Therefore,

methods that work with normalized data should not raise an error message when some input data are negative.

In this work, we use MR1–MR6 to establish baselines for cluster analysis and MR7–MR8 for cell-type classification.

IV. RESULTS

We apply our framework to a well-known scRNA-seq benchmark available in `scrnabench` package. The benchmark comprises 48 reference datasets of full-length and partial-length sequencing of two well-characterized cell lines. These cell lines are sequenced on four different platforms (10X Genomics, C1_HT, C1 and ICELL8) and at four different centers (FDA, NCI, LLU and TBU) [42], [43]. The 10X Genomics data are processed using four different methods (`cellranger2.0`, `cellranger3.1`, `umitools` and `zumi`), and the remaining datasets are processed using three different methods (`featureCounts`, `rsem`, `kallisto`).

First, we establish performance and stability baselines for cluster analysis methods. Cluster analysis is an unsupervised machine learning technique, which is used to group cells with similar gene expression, identify differentially-expressed genes in these groups and discover their cell-types. We use a standard implementation of the general-purpose K-Means ($k=10$) algorithm and of the domain-specific Seurat clustering algorithm.

We apply PCA, tSNE and UMAP to reduce the dimension benchmarks to 2, and cluster reduced benchmarks separately. Sil scores are computed for each clustering, ranging from -1 to 1 , where values closer to 1 indicate better clustering. Results show that Seurat’s Sil scores have greater variability (Fig. 2), indicating that its performance is more influenced by the characteristics of reference benchmarks than the performance of K-Means algorithm. We observe that internal cluster validity of both baseline methods improves for all benchmarks when non-linear methods (tSNE and UMAP) are used to reduce dimensionality compared to linear PCA. Sil scores of PCA-reduced clusterings of both methods are low and even negative for some of the reference datasets (Fig. 2A). In addition, while K-Means performs equally well with both, tSNE and UMAP benchmarks, performance of Seurat clustering is clearly influenced by the choice of dimensionality reduction. It outperforms K-Means on tSNE data and underperforms on UMAP benchmarks.

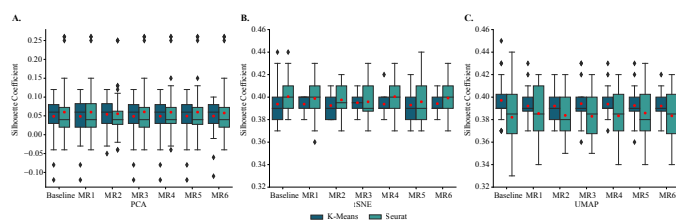


Fig. 2. Metamorphic benchmarking of baseline cluster analysis algorithms. Shown are box plots of Silhouette coefficients of K-Means (dark green) and Seurat clustering (light green). Reference (baseline) and perturbed datasets are reduced using (A.) PCA, (B.) tSNE and (C.) UMAP. The scale of the y-axis of PCA is different than the scales of tSNE and UMAP.

To examine the stability and reliability of these baselines, we apply our data-centric framework. Six metamorphic tests of MR1–MR6 are created for each reference benchmark, and resulting metamorphic datasets are processed using the same `scrnabench` workflows. Importantly, we keep all random seeds constant in K-Means and Seurat clustering to eliminate stochasticity in their initializations. We compare the differences in the distributions of baseline and metamorphic Sil scores using paired T-test. We expect to see no significant change in the results of MR1–MR6. Notably, K-Means is a general purpose algorithm, and Seurat clustering works with normalized data. Hence, we expect that they will cluster data regardless of their sign and produce no change in MR6 testing. Indeed, our expectation for MR1–MR6 is supported by the results of metamorphic tests, and the distributions of Sil scores are not impacted by input perturbations (Fig. 2). Although some differences are seen in the metamorphic results compared with the baseline, they are not statistically significant. K-Means results remain more consistent than Seurat clustering, having a more compact distribution of Sil scores. For UMAP benchmarks of 10X Genomics, for example, Sil scores of Seurat vary from 0.32 to 0.44, with low performance on benchmarks preprocessed using `zumi` and `umitools` compared with high performance on `cellranger2.0` and `cellranger3.1`. The `scrnabench` package outputs the total number of clusters in each benchmark, in addition to Sil scores. In K-Means the number of clusters is fixed to 10, and Seurat clustering detects the number of clusters automatically. Interestingly, Seurat finds fewer clusters in PCA-reduced benchmarks than in tSNE and UMAP benchmarks. On average, Seurat detects 7.4, 15.5, and 14.1 clusters in PCA, tSNE and UMAP. In addition, we observe a strong positive correlation between the number of cells and the number of clusters detected by Seurat clustering, with Pearson correlation coefficients of about 0.88 for PCA benchmarks, 0.95 for tSNE benchmarks and 0.92 for UMAP benchmarks. Taken together, these results demonstrate that K-Means algorithm should be used as a baseline cluster analysis method in benchmarking studies due to its performance, stability and reliability.

Second, we establish performance and stability baselines for data integration methods, which are commonly used to project data from different sources onto a common representation with the goal of increasing the statistical power of data analysis. In this experiment, we integrate pairs of reference benchmarks from two different cell lines, sequenced on the same platform, at the same center, and preprocessed similarly. As an example, we integrate together normal and cancer datasets of 10X Genomics sequenced at NCI and preprocessed using `cellranger2.0`. These integrations are suitable for the evaluation of batch effect correction and preservation of biological variability. We compute Integration Method Selection (IMS) score for each integrated dataset, which is a metric of uncorrected batch effects [5]. Higher IMS score indicates lower integration quality. We also integrate paired metamorphic (MR1–MR6) benchmarks and compute their IMS scores. Our results show that MR1–MR5 are satisfied by all four integration methods

and the differences of the distributions of baseline and metamorphic IMS scores are not statistically significant (Fig. 3). Therefore, all four integration methods are stable. However, we found that most integration methods do not test for invalid inputs and integrate data regardless of the sign of raw counts. FastMNN, however, terminates with a fault when presented with negative gene expressions.

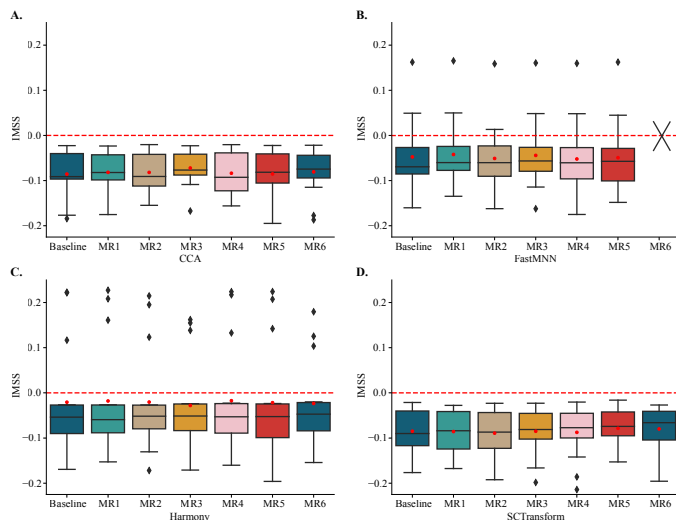


Fig. 3. Metamorphic benchmarking of integration methods. Shown are IMS scores of integration methods (A.) CCA, (B.) FastMNN, (C.) Harmony, and (D.) SCTransform. IMS scores above the baseline zero (red line) indicate that presence of batch effect in the integrated data.

We observe that CCA and SCTransform remove batch effects in all benchmarks, while Harmony and FastMNN fail to correct batch effects in some of them (Fig. 3). Integrated data are often used for supervised cell-type prediction and here, we apply our proposed data-centric framework to examine how the quality of integrated datasets impacts the performance and stability of cell-type classifiers. Prior works showed that feed-forward neural networks (FFNN) can accurately predict cell-types [16], and we implement a dense FFNN classifier, comprising 1 hidden layer with varied number of hidden nodes and a rectified linear unit (ReLU) activation function. The number of nodes of the input layer is the dimension of the dataset. The number of nodes of the output layer equals the number of cell-type labels, and a softmax activation function is used to generate probabilities for each predicted cell-type. Probabilities are converted to categorical class labels using an argmax function. FFNN training uses RMSprop optimizer with a learning rate of 1×10^{-4} and a categorical cross-entropy as a loss function.

To test for stability and reliability of FFNN classifiers trained with integrated data, each integrated benchmark is first shuffled and an equal number of cells of each cell-type is selected from the shuffled dataset. Balanced benchmarks are split into training (80%) and test (20%) subsets. Balancing the data allows us to use accuracy as an unbiased metric of performance. To initialize model's parameters and control for randomness, we run one iteration of gradient descent and save

network weights. Next, we apply MR7 by randomly removing one cell from the training subset of size N . A model is initialized with saved weights and trained with the remaining $N - 1$ cells. Metamorphic models are used to make predictions for the test subset. We compute prediction accuracy and also count the number of cells whose predicted labels change from baseline predictions. We repeat MR7 test 100 times, record the maximum number of changed predictions, and report them as percentage of the total number cells in the test dataset.

We observe that an increase in FFNN complexity, measured by the number of hidden nodes/parameters, improves the classification accuracy in FastMNN and Harmony integrations but not CCA or SCTransform (Fig. 4). Stability varies between integration methods. FFNN classifiers of Harmony data are most stable and accurate. Their stability does not change when FFNN complexity increases and fewer than 1% of predicted labels flip in metamorphic tests compared with baseline. Around 1% of predicted labels change when more complex neural networks are used with FastMNN benchmarks. However, classifiers trained with CCA and SCTransform data are unstable and close to 4% of predicted cell labels flip in metamorphic testing of FFNN models with 1024 hidden nodes.

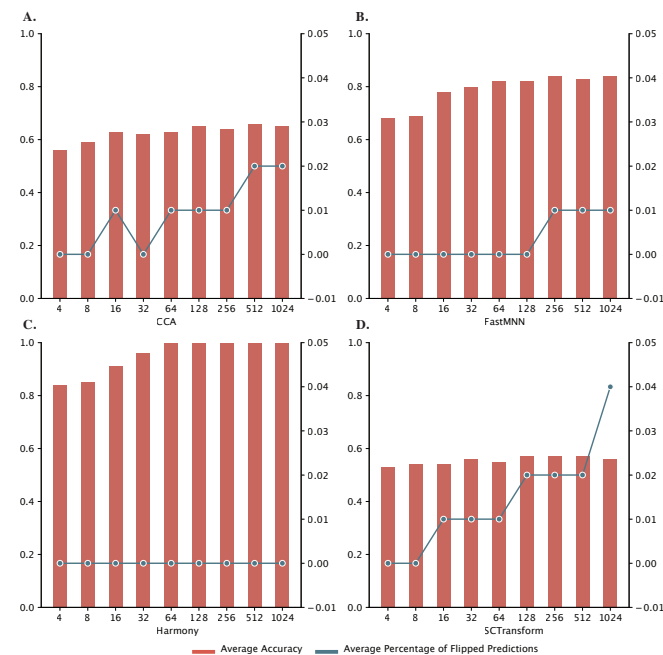


Fig. 4. Metamorphic testing of FFNN classifiers. Shown are barplots of classification performance (left y-axis) and line plots of percentage of flipped predictions (right y-axis) in FFNN classifiers trained with integrated data. Number of nodes in the hidden layer is shown on the x-axis. (A.) CCA, (B.) FastMNN, (C.) Harmony, and (D.) SCTransform.

To test the extent of instability caused by the integration quality, we add a second hidden layer to FFNNs and repeat MR7 tests. For each original number of nodes in the first hidden layer, we double the number of nodes in the second hidden layer. Thus, a model with first hidden layer with 1024

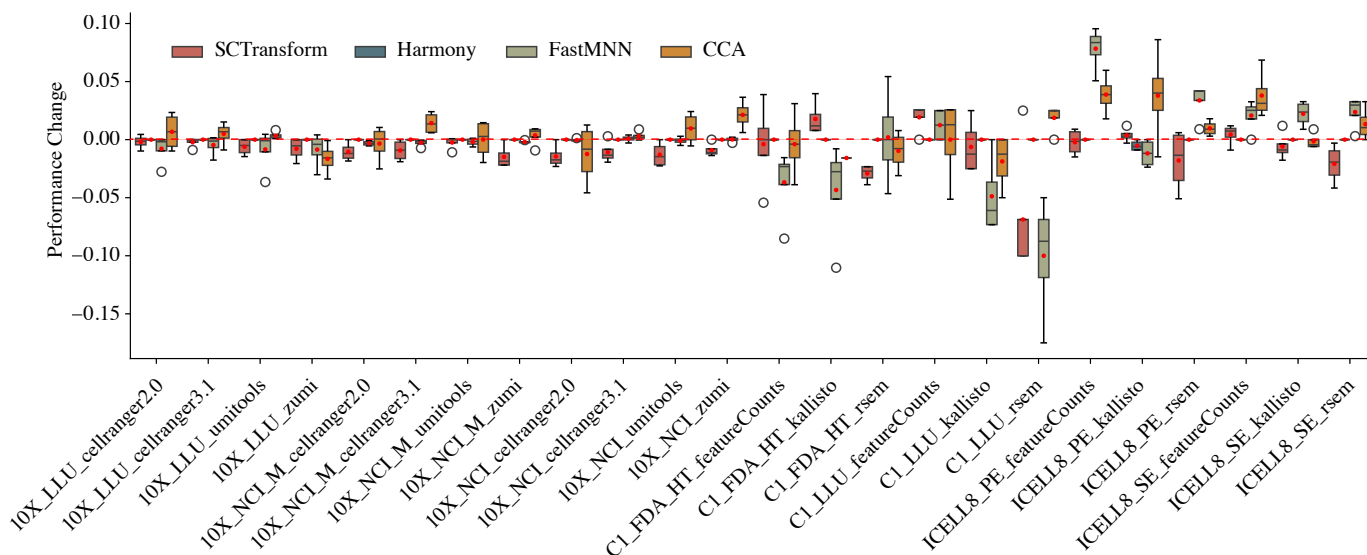


Fig. 5. MR8 Testing of Data Integration Methods. Shown are boxplots of change in test accuracy of models trained with coresets compared to the models trained with the full dataset. Each box represents the averaged results of models trained with 4 different coresets.

nodes has a second layer with 2048 nodes. Our experiments with these more complex networks show no improvements in accuracy over simpler FFNNs but rather an increase in the percentage of flipped cell-type labels, for all integration methods except Harmony. The percentages of flipped predictions for the most complex FFNN model are 8% for CCA, 3% for FastMNN, and 11% for SCTransform. Similar results are observed for FFNN models with two hidden layers, where the number of nodes in the second layer is halved.

Notably, these changes are caused by the removal of one, single cell from the training dataset (MR7). To see if these results are supported by a more traditional approach to stability validation, namely sampling, we perform metamorphic tests of MR8. Specifically, we sample four different coresets from each integrated benchmark, and train classifiers on coresets rather than the full datasets. By construction, models trained with coresets should perform similarly to models trained with full datasets.

Our MR8 results demonstrate that Harmony is the only method that preserves all biological variability, while the other three methods overcorrect for batch effects. This finding is supported by the results of MR7, which show lower classification accuracy of CCA, FastMNN and SCTransform (Fig. 4). In MR8 testing, models trained with Harmony coresets have performance similar to baseline, with an average accuracy close to 1. For benchmarks integrated by FastMNN, models trained with the full dataset have an average classification accuracy of 0.831 ± 0.078 , and models trained with coresets do not exhibit significant difference in performance and test accuracy is 0.831 ± 0.082 . We observe, significant overcorrections in CCA and SCTransform integrations, in baseline and metamorphic benchmarks.

Stability varies significantly between different sequencing technologies (Fig. 5). For example, coreset-based models of

ICELL8 benchmarks have better predictive performance than their full datasets, while accuracy of coreset-based models decreases for the C1, and C1_HT benchmarks. We observe that the performance and the size of the 10X benchmarks are larger than those of other sequencing technologies. To examine whether our results are confounded by size, we downsample 10X benchmarks to the size of ICCELL8 and repeat MR8 tests. We find that the results do not change, and therefore, the observed behavior of integration methods is due to the properties of benchmarks of different sequencing technologies.

Based on MR7–MR8 tests of performance and stability of data integration and classification methods, we recommend Harmony as the baseline benchmark, with the caveat of it not testing for invalid data inputs.

Some limitations of this work must be noted. First, the description of the scrnabench package and use-cases of metamorphic testing focused on four scRNA-seq data analysis tasks, namely, dimensionality reduction, cluster analysis, data integration, and cell classification. While the demonstrations of the framework and scrnabench package provide a robust proof-of-concept, we recognize the need to extend our approach to other scRNA-seq data analysis tasks, such as trajectory inference and differential expression analysis, for example. Our future work will focus on defining new metamorphic relations, appropriate for these tasks.

Second, future work should address the current scalability limitation of the scrnabench package. Specifically, our current implementation of metamorphic benchmarks may face challenges when performing benchmarking and testing of atlas-level datasets, comprising millions of cells. Our future work will include efforts to improve parallel processing to enable benchmarking of pipelines that deal with large-scale cell atlases.

Third, we recognize that the adoption of scrnabench depends

on its interoperability with established community platforms. To address this, we will implement API wrappers to allow other scRNA-seq benchmarking platforms, such as scIB and Seurat, for example, to call scrnabench's testing modules directly. This will enable users of existing platforms to incorporate our metamorphic tests in their native programming environment.

Finally, we emphasize that while this work focused on single-cell RNA sequencing data analysis pipelines, the ideas of metamorphic testing and the data-centric design of the scrnabench framework are not limited to this domain. Our framework and package may be used to test and benchmark dimensionality reduction, cluster analysis, classification and data integration of other high-dimensional datasets, including proteomics, metabolomics, spatial transcriptomics and high-throughput imaging data.

V. CONCLUSION

Motivated by the need for alternative approaches for benchmarking of scRNA-seq data analysis pipelines that go beyond assessments of generalizability, we formulated and implemented a data-centric framework for testing and validation of existing and emergent methods and their implementations. Drawing upon software engineering technique called metamorphic testing, we designed and implemented a software package to automatically generate diverse, representative and realistic metamorphic benchmarks and prepare them for data analysis in a standardized manner. We establish baseline performance and stability of cluster analysis, data integration and cell-type classification methods. The scrnabench package presented here, may make benchmarking studies more reproducible, sustainable and fair, thus, increasing the transparency and trustworthiness of scRNA-seq data and results. In the spirit of the original release of the reference scRNA-seq data [43], we hope that scrnabench can become a community-supported benchmarking platform of scRNA-seq reference materials and that it will be extended to entirely new directions of method development.

REFERENCES

- [1] M. R. Aniba, O. Poch, and J. D. Thompson, "Issues in bioinformatics benchmarking: the case study of multiple sequence alignment," *Nucleic acids research*, vol. 38, no. 21, pp. 7353–7363, 2010.
- [2] L. M. Weber, W. Saelens, R. Cannoodt, C. Sonesson, A. Hafpelmeier, P. P. Gardner *et al.*, "Essential guidelines for computational method benchmarking," *Genome biology*, vol. 20, pp. 1–12, 2019.
- [3] D. Lähnemann, J. Köster, E. Szczurek, D. J. McCarthy, S. C. Hicks, M. D. Robinson *et al.*, "Eleven grand challenges in single-cell data science," *Genome biology*, vol. 21, no. 1, pp. 1–35, 2020.
- [4] X. Xie, Z. Zhang, T. Y. Chen, Y. Liu, P.-L. Poon, and B. Xu, "Mettle: A metamorphic testing approach to assessing and validating unsupervised machine learning systems," *IEEE Transactions on Reliability*, vol. 69, no. 4, pp. 1293–1322, 2020.
- [5] K. Zhao, S. Bhandari, N. P. Whitener, J. M. Grayson, and N. Khuri, "An ensemble machine learning approach for benchmarking and selection of scRNA-seq integration methods," in *Proceedings of the 14th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, ser. BCB '23. New York, NY, USA: Association for Computing Machinery, 2023.
- [6] H. L. Crowell, S. X. Morillo Leonardo, C. Sonesson, and M. D. Robinson, "The shaky foundations of simulating single-cell rna sequencing data," *Genome Biology*, vol. 24, no. 1, p. 62, 2023.
- [7] B. Löwes, C. Chauve, Y. Ponty, and R. Giegerich, "The bralibase dent—a tale of benchmark design and interpretation," *Briefings in bioinformatics*, vol. 18, no. 2, pp. 306–311, 2017.
- [8] M. H. Jarrahi, A. Memariani, and S. Guha, "The principles of data-centric ai," *Commun. ACM*, vol. 66, no. 8, p. 84–92, jul 2023.
- [9] D. Zha, K.-H. Lai, F. Yang, N. Zou, H. Gao, and X. Hu, "Data-centric ai: Techniques and future perspectives," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2023, p. 5839–5840.
- [10] E. T. Barr, M. Harman, P. McMinn, M. Shahbaz, and S. Yoo, "The oracle problem in software testing: A survey," *IEEE Transactions on Software Engineering*, vol. 41, no. 5, pp. 507–525, 2015.
- [11] S. Segura, G. Fraser, A. B. Sanchez, and A. Ruiz-Cortés, "A survey on metamorphic testing," *IEEE Transactions on Software Engineering*, vol. 42, no. 9, p. 805–824, Sep 2016.
- [12] F. U. Rehman and C. Izurieta, "An approach for verifying and validating clustering based anomaly detection systems using metamorphic testing," in *2022 IEEE International Conference On Artificial Intelligence Testing (AITest)*. IEEE, 2022, pp. 12–18.
- [13] N. Whitener, "Scrnabench: A package for metamorphic benchmarking of scRNA-seq data analysis methods," Jun 2022. [Online]. Available: <https://github.com/NWhitener/scrnabench>
- [14] C. Angermueller, H. J. Lee, W. Reik, and O. Stegle, "Deepcp: accurate prediction of single-cell dna methylation states using deep learning," *Genome Biology*, vol. 18, no. 1, p. 67, Apr 2017.
- [15] J. Alquicira-Hernandez, A. Sathe, H. P. Ji, Q. Nguyen, and J. E. Powell, "scpred: accurate supervised method for cell-type classification from single-cell rna-seq data," *Genome Biology*, vol. 20, no. 1, p. 264, Dec 2019.
- [16] F. Ma and M. Pellegrini, "Actinn: automated identification of cell types in single cell rna sequencing," *Bioinformatics (Oxford, England)*, vol. 36, no. 2, p. 533–538, Jan 2020.
- [17] C. Sonesson and M. D. Robinson, "Bias, robustness and scalability in single-cell differential expression analysis," *Nature methods*, vol. 15, no. 4, pp. 255–261, 2018.
- [18] T. Abdelaal, L. Michielsen, D. Cats, D. Hoogduin, H. Mei, M. J. Reinders *et al.*, "A comparison of automatic cell identification methods for single-cell rna sequencing data," *Genome biology*, vol. 20, pp. 1–19, 2019.
- [19] W. Saelens, R. Cannoodt, H. Todorov, and Y. Saeys, "A comparison of single-cell trajectory inference methods," *Nature biotechnology*, vol. 37, no. 5, pp. 547–554, 2019.
- [20] L. Tian, X. Dong, S. Freytag, K.-A. Lê Cao, S. Su, A. JalalAbadi *et al.*, "Benchmarking single cell rna-sequencing analysis pipelines using mixture control experiments," *Nature methods*, vol. 16, no. 6, pp. 479–487, 2019.
- [21] B. Vieth, S. Parekh, C. Ziegenhain, W. Enard, and I. Hellmann, "A systematic evaluation of single cell rna-seq analysis pipelines," *Nature communications*, vol. 10, no. 1, p. 4667, 2019.
- [22] Y. You, L. Tian, S. Su, X. Dong, J. S. Jabbari, P. F. Hickey *et al.*, "Benchmarking umi-based single-cell rna-seq preprocessing workflows," *Genome biology*, vol. 22, no. 1, pp. 1–32, 2021.
- [23] A. Cheng, G. Hu, and W. V. Li, "Benchmarking cell-type clustering methods for spatially resolved transcriptomics data," *Briefings in Bioinformatics*, 2022.
- [24] S. Junttila, J. Smolander, and L. L. Elo, "Benchmarking methods for detecting differential states between conditions from multi-subject single-cell rna-seq data," *bioRxiv*, 2022.
- [25] J. Gagnon, L. Pi, M. Ryals, Q. Wan, W. Hu, Z. Ouyang *et al.*, "Recommendations of scRNA-seq differential gene expression analysis based on comprehensive benchmarking," *Life*, vol. 12, no. 6, p. 850, 2022.
- [26] B. Li, W. Zhang, C. Guo, H. Xu, L. Li, M. Fang *et al.*, "Benchmarking spatial and single-cell transcriptomics integration methods for transcript distribution prediction and cell type deconvolution," *Nature Methods*, pp. 1–9, 2022.
- [27] C. Dai, Y. Jiang, C. Yin, R. Su, X. Zeng, Q. Zou *et al.*, "scimc: a platform for benchmarking comparison and visualization analysis of scRNA-seq data imputation methods," *Nucleic acids research*, vol. 50, no. 9, pp. 4877–4899, 2022.
- [28] X. Cao, L. Xing, E. Majd, H. He, J. Gu, and X. Zhang, "A systematic evaluation of supervised machine learning algorithms for cell pheno-

- type classification using single-cell rna sequencing data,” *Frontiers in genetics*, vol. 13, 2022.
- [29] L. Yu, Y. Cao, J. Y. Yang, and P. Yang, “Benchmarking clustering algorithms on estimating the number of cell types from single-cell rna-sequencing data,” *Genome biology*, vol. 23, no. 1, pp. 1–21, 2022.
- [30] M. D. Luecken, M. Büttner, K. Chaichoompu, A. Danese, M. Interlandi, M. F. Müller *et al.*, “Benchmarking atlas-level data integration in single-cell genomics,” *Nature methods*, vol. 19, no. 1, pp. 41–50, 2022.
- [31] R. Dubes and A. K. Jain, “Validity studies in clustering methodologies,” *Pattern recognition*, vol. 11, no. 4, pp. 235–254, 1979.
- [32] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, “Cluster validity methods: part i,” *ACM Sigmod Record*, vol. 31, no. 2, pp. 40–45, 2002.
- [33] J. Handl, J. Knowles, and D. B. Kell, “Computational cluster validation in post-genomic data analysis,” *Bioinformatics*, vol. 21, no. 15, pp. 3201–3212, 2005.
- [34] S. Ben-David, U. Von Luxburg, and D. Pál, “A sober look at clustering stability,” in *International conference on computational learning theory*. Springer, 2006, pp. 5–19.
- [35] U. Von Luxburg *et al.*, “Clustering stability: an overview,” *Foundations and Trends in Machine Learning*, vol. 2, no. 3, pp. 235–274, 2010.
- [36] S. Ben-David and L. Reyzin, “Data stability in clustering: A closer look,” *Theoretical Computer Science*, vol. 558, pp. 51–61, 2014.
- [37] C. Hennig, “Cluster-wise assessment of cluster stability,” *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 258–271, 2007.
- [38] A. Lavin, C. M. Gilligan-Lee, A. Visnjic, S. Ganju, D. Newman, S. Ganguly *et al.*, “Technology readiness levels for machine learning systems,” *Nature Communications*, vol. 13, no. 1, p. 6039, 2022.
- [39] S. Yang, D. Towey, and Z. Q. Zhou, “Metamorphic exploration of an unsupervised clustering program,” in *2019 IEEE/ACM 4th International Workshop on Metamorphic Testing (MET)*, 2019, pp. 48–54.
- [40] Y. Chen, M. Welling, and A. Smola, “Super-samples from kernel herding,” in *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, ser. UAI’10. Arlington, Virginia, USA: AUAI Press, 2010, p. 109–116.
- [41] O. Bachem, M. Lucic, and A. Krause, “Coresets for nonparametric estimation - the case of dp-means,” in *Proceedings of the 32nd International Conference on Machine Learning*. PMLR, Jun. 2015, p. 209–217.
- [42] W. Chen, Y. Zhao, X. Chen, Z. Yang, X. Xu, Y. Bi *et al.*, “A multicenter study benchmarking single-cell rna sequencing technologies using reference samples,” *Nature Biotechnology*, vol. 39, no. 9, pp. 1103–1114, 2021.
- [43] X. Chen, Z. Yang, W. Chen, Y. Zhao, A. Farmer, B. Tran, V. Furtak, M. Moos, W. Xiao, and C. Wang, “A multi-center cross-platform single-cell rna sequencing reference dataset,” *Scientific data*, vol. 8, no. 1, pp. 1–11, 2021.