# Wisdom of the crowd: mapping continuous cell states with hyperparameter-randomized ensemble clustering

Sarah Marie Goggin

B.S. Neuroscience, College of William and Mary, 2015

A Dissertation presented to the Graduate Faculty of the
University of Virginia in Candidacy for the
Degree of Doctor of Philosophy

Neuroscience Graduate Program

University of Virginia

December 2024

# Table of Contents

# Acknowledgements

I would first like to thank my advisor, Eli Zunder, for supporting my evolution as a researcher. While I initially joined the lab with the expectation of combining bench work and computational analysis, I appreciate your willingness to adapt and ultimately support my transition to a fully computational project. You maintained trust in my abilities and provided me with the independence I sought, which created an ideal environment for my scientific and personal growth.

I would also like to thank my co-advisor, Stefan Bekiranov, for your technical guidance and help ensuring the rigor and relevance of my computational approaches, and my committee members, Sarah Kucenas, John Campbell, Ali Guler, and Heather Ferris, for consistently helping me explore interesting biological applications and navigate publication strategies. I particularly enjoyed when our conversations transcended routine progress updates to become thought-provoking discussions about the big ideas and trends in biomedical research.

I am grateful to my advisor, committee, and the NGP administrators and directors including Nadia Cempre, Kim Knotts, Chris Deppmann, and Mark Beenhaker for supporting me in creating and pursuing my own unique training path.

Thank you to everyone involved in the Biomedical Data Science Training Grant, specifically Jason Papin and Nathan Sheffield for creating such a thoroughly useful training curriculum, and Kim Fitzhugh-Higgins for being a ray of sunshine amid the confusing world of research funding and administration.

To my incredible 2017 cohort, I am so grateful to have gone on this journey with such a wonderful crew - our game nights, trivia adventures, bouldering, hikes, etc. made the grad school experience so much richer than I could have hoped for.

To my lab mates Kristen Fread, Corey Williams, Amy VanDeusen - “ty” for creating a space where frustrations could be safely voiced, victories properly celebrated, and sanity somehow maintained through the rollercoaster of graduate work. Your camaraderie made all the difference in my daily experience.

To Austin Keeler - honestly just knowing you exist as a force for good in the world continues to give me faith in humanity. I've read numerous acknowledgement sections from our mutual

colleagues while drafting my own, and they all echo the same sentiment: your deep excitement about science and earnestness in everything you do has a deep and meaningful impact on those you interact with, and is something we need more of in academic science. Keep doing what you do!

To Maira Jalil - though our time working together was (relatively) brief, it coincided with a pivotal period as we both worked to conclude our thesis projects and chart our next steps beyond graduate school. I always looked forward to our meetings and would leave them feeling buoyed and motivated to keep pushing through the final arduous stretch. I hope I was able to provide the same to you.

A giant thank you to my friends, family, and family-in-law who have been my emotional anchors throughout this wild ride. To my partner Sean - words fail to capture how extraordinary it has been to share this journey with you, starting as kiddos at W&M, through our synchronized doctoral pursuits, and now as we scout our next adventures together. We've seen each other through some low lows and some high highs, particularly during grad school, and it has only made us continue to grow into a better team. To my sister - thank you for being my BFFL in the truest sense. Knowing that I have a person that has seen me through it all and with whom I will always be able to be 100% myself is such a refuge. To my dad - your self-taught transition into computer science showed me firsthand that reinvention is possible and that pursuing what genuinely interests you is worth the effort. To my mom - your genuinely unconditional love and encouragement throughout my whole life has been my bedrock, and along with the grit you have instilled in me, has enabled me to persevere through my "3E"-associated challenges and get where I am today. In your honor I will try my best to sometimes use "traditionally correct" grammar.

# Dedication

I would like to dedicate this dissertation to those I have lost along the way. To my Grandmother Sally Yobst (“Nanny”), who always encouraged me to pursue my dreams to the fullest and make my life what I want it to be, rather than follow anyone else’s expectations. And to my Uncle John (“UJ”), who always thought I could do whatever I set my mind to, and maybe even convinced me of that. Your absence is felt daily, but your influence remains constant.

# Abstract

Recent technological advances have enabled high-throughput single cell molecular profiling, generating unprecedented volumes of data that promise to revolutionize our understanding of cellular heterogeneity and its role in health and disease. However, our ability to extract meaningful biological insights from these complex datasets has not kept pace with our capacity to generate them. A fundamental challenge lies in the computational methods used to characterize cell populations, which often lack scalability, generalizability, and the ability to capture the nuanced biological reality of both discrete and continuous cellular variation.

This thesis addresses these challenges through the development of novel computational approaches for analyzing and organizing single cell data. Central to this work is ESCHR, an innovative ensemble clustering method that eliminates the need for manual parameter tuning while providing superior accuracy and robustness compared to existing approaches. ESCHR's unique hyperparameter-randomized ensemble approach not only generates high-quality discrete clustering results but also enables soft clustering to characterize regions of biological continuity and quantifies clustering uncertainty at the single cell level.

Comprehensive evaluation across a large collection of diverse single cell datasets demonstrates ESCHR's superior performance and generalizability compared to existing methods. We further showcase the method's capabilities through in-depth analysis of two distinct applications, mapping the connectivity and intermediate transitions between handwritten digits (MNIST) and between hypothalamic tanycyte subpopulations. In both cases, ESCHR successfully identified canonical discrete groups while revealing meaningful continuous structure between them. This unified approach to capturing both discrete and continuous aspects of data structure, combined with transparent uncertainty quantification, represents a significant advance in our ability to generate hypotheses from single cell data.

By emphasizing generalizability, robust performance, and interpretability while eliminating the need for manual parameter tuning, ESCHR provides a powerful framework for extracting biological insights from the growing volume of single cell data. This work contributes not only practical tools for the single cell research community but also advances our conceptual approach to understanding cellular heterogeneity.

# Chapter 1: Introduction

## 1.1 Why do we need single cell molecular profiling?

Cells are the fundamental units of life, serving as the basic building blocks from which all living organisms are constructed. This principle, first formalized in cell theory by the pioneering work of Schleiden, Schwann, and Virchow in the 19th century [1], has only grown in importance as our understanding of biology has advanced. What makes cells particularly remarkable is their ability to adopt diverse functional roles while maintaining the same genetic blueprint (approximately, barring somatic mutation). This cellular diversity is what enables complex multicellular organisms to develop and function.

The diverse functions of cells arise from their molecular phenotypes – the complete set of proteins, RNA molecules, metabolites, and other biomolecules that are present and active within each cell at any given time [2, 3]. These molecular components work together in complex networks to determine cellular behavior, from basic processes like energy metabolism to specialized functions like neurotransmitter release or antibody production. Thus, the diversity of cellular functions that we observe at the physiological level must emerge from underlying diversity in molecular phenotypes.

However, our understanding of this molecular basis of cellular diversity remains incomplete. Traditional bulk approaches to molecular profiling, which analyze entire populations of cells simultaneously, obscure the very heterogeneity that makes our organismal complexity possible. These methods provide only an average view of cellular state, potentially missing rare cell types, transitional states, and subtle variations in cellular phenotypes that may be crucial for tissue function [4]. Furthermore, bulk approaches often rely on pre-existing knowledge to isolate supposedly pure populations of cells, introducing potential bias and limiting our ability to discover new cell states or types.

To truly understand biological complexity, we need approaches that can comprehensively map the molecular phenotypic space of cells in an unbiased manner. This mapping would reveal how cells are distributed across this high-dimensional space, identifying both discrete groupings and patterns of continuous variation. By relating these molecular profiles to cellular functions, we can begin to understand how molecular diversity gives rise to functional diversity, how cells transition between different states, and how these processes may go awry in disease. High-throughput, high dimensional single cell molecular profiling technologies (which we will generally refer to as “single cell ‘omics” throughout this thesis) offer the resolution and scale

needed to create such maps, providing a path toward understanding how molecular variation gives rise to biological function.

## 1.2 The rise of single cell ‘omics

When single cell RNA sequencing (scRNA-seq) was first demonstrated in 2009 [5], it marked a pivotal moment in biological research. While very low-throughput, it demonstrated the feasibility of generating reliable transcriptional profiles from the minute amounts of RNA present in single cells. The subsequent development of Smart-seq and related approaches further improved the quality of single cell data by enabling full-length transcript coverage, allowing detection of splice variants and more accurate gene quantification [6, 7]. The technological progression rapidly accelerated with the development of droplet-based scRNA-seq methods [8, 9], which enabled high-throughput profiling of thousands of cells simultaneously. These advances dramatically reduced per-cell sequencing costs, encouraging adoption and facilitating rapid scaling of the number of cells being assayed. Building on the success of scRNA-seq, a diverse ecosystem of single cell technologies has emerged. Single cell ATAC-seq and epigenomic profiling techniques have enabled investigation of the complex regulatory landscapes that underpin cellular diversity and function [10, 11], while single cell DNA sequencing has enabled fine-grained analysis of somatic variation in contexts ranging from cancer evolution to development and aging [12].

In parallel with sequencing-based approaches, mass cytometry emerged as a powerful advancement over traditional flow cytometry and immunohistochemistry (IHC) techniques. While flow cytometry had enabled single cell protein analysis since the late 1970’s, it was limited by spectral overlap between fluorescent markers to measuring around 12-15 proteins simultaneously [13]. Mass cytometry overcame this limitation by using heavy metal isotopes as antibody labels instead of fluorophores, enabling simultaneous measurement of over 40 protein markers in individual cells [14]. This dramatic increase in the number of proteins that could be measured together provided unprecedented ability to deeply phenotype cellular populations and capture complex protein-level molecular phenotypes, though at the cost of losing the ability to retain viable cells as with traditional flow cytometry.

More recently, technological innovations have expanded in two key directions: spatial resolution and multimodal integration. Spatial 'omics techniques have emerged to map molecular profiles while preserving the precise spatial relationships between cells within their tissue context. Methods like MERFISH [15] and seqFISH+ [16] enable spatial transcriptomics, while imaging

mass cytometry extends the protein-detection capabilities of mass cytometry into a spatial context, allowing visualization of dozens of proteins within intact tissue sections [17]. In parallel, the field has evolved toward multimodal approaches that integrate multiple types of measurements from the same cell, though no current methods measure more than two modalities at once. Technologies such as CITE-seq [18] bridge the sequencing and antibody-based approaches by simultaneously capturing both RNA and protein expression from individual cells. Together, these technological advances have created a toolkit for dissecting cellular heterogeneity at unprecedented scale and depth.

## 1.3 Challenges of single cell ‘omics analysis

The advent of single cell ‘omics technologies has revolutionized our ability to dissect cellular heterogeneity, uncovering layers of complexity previously obscured by bulk measurements. However, this new frontier comes with its own set of analytical challenges. With the ability to capture vast numbers of cells across thousands of molecular features, researchers now face two interconnected problems: the inherent complexity of cellular identity and the technical challenges of analyzing high-dimensional, noisy, and often sparse data. Addressing these issues is essential to fully realize the potential of single cell ‘omics and to make meaningful biological inferences from this unprecedented scale and resolution of data.

A fundamental limitation of single cell technologies is their destructive nature: cells are lysed to extract molecular information, making it impossible to track changes within the same cell over time. This presents a significant challenge for studying dynamic biological processes such as differentiation, immune responses, and disease progression. Reconstructing temporal processes from distinct cell snapshots requires computational inference, which can then introduce uncertainty and potential inaccuracies.

A single cell dataset typically consists of molecular counts of a single modality (even in multimodal assays, the data are separated into counts for each modality), such as mRNA, protein, or chromatin accessibility. Most modalities rely on indirect measurements that require inference of the true biological target. For example, scRNA-seq measures cDNA molecules reverse transcribed from mRNA, while mass cytometry quantifies antibody binding events as a proxy for protein abundance. Others require even more complex inference, such as scATAC-seq which detects DNA fragments made accessible through nucleosome displacement to infer regulatory element activity. The raw data obtained from a single cell ‘omics experiment thus

provides as a starting point only a partial and indirect view of cellular state, which is insufficient to fully capture the complexity of cellular identity.

The variation observed in single cell molecular counts arises from three main sources: technical variation, allele-intrinsic variation, and allele-extrinsic variation [19].

- Technical variation stems from differences in cell lysis, RNA capture efficiency, and detection sensitivity [20, 21].
- Allele-intrinsic variation reflects stochastic factors inherent to molecular mechanisms, such as transcriptional bursting and variable mRNA degradation, leading to differences between otherwise identical cells or even between alleles of the same gene within a cell [21–23].
- Allele-extrinsic variation arises from external factors, such as regulatory molecules or stable chromatin states, that influence gene expression and establish differences between cell types or states [21–23].

Most studies aim to understand allele-extrinsic variation, which is often biologically meaningful, but technical and allele-intrinsic variations can obscure these signals. The noise from these sources is particularly problematic when analyzing subtle phenotypic differences or rare cell populations. Furthermore, technical variability can correlate with biological features [24], complicating efforts to disentangle genuine biological signals from artifacts.

This variation can manifest as anything from coordinated differences shared by discrete groups of cells to a complex high dimensional continuum of changes. Both of these extremes can be observed in the case of stem cell differentiation, where cells proceed through transitional states characterized by continuous variation of markers associated with both progenitor and mature populations, but there are pools of progenitors and mature cells that each have clearly distinct shared molecular profiles [25]. To make matters even more complicated, these varying degrees of discreteness and continuity are reflected differently across different molecular modalities, with some regulatory layers enforcing sharp transitions while others allow for more graded responses [26].

Single cell datasets are inherently high-dimensional, with dozens to tens of thousands of molecular features measured per cell. This richness enables detailed exploration of cellular diversity but also introduces the "curse of dimensionality" [27]. In high-dimensional spaces, it

becomes difficult to accurately define distances or relationships between cells, which can obscure biologically meaningful patterns.

Adding to these challenges is data sparsity. Single cell data matrices, especially those derived from sequencing-based methods, are often dominated by zeros. These zeros reflect some combination of technical limitations (such as incomplete molecular capture), biological noise (such as from stochastic gene expression dynamics), and true biological phenomena of interest (such as selective gene expression in differently specialized cell types). This sparsity complicates the application of traditional statistical and machine learning methods, which often assume data distributions that are not met in single cell datasets [28].

Finally, the sheer size of single cell datasets has grown dramatically. Modern technologies can generate data for hundreds of thousands or even millions of cells in a single experiment. Analyzing and interpreting these large-scale datasets requires substantial computational resources and sophisticated algorithms capable of scaling efficiently.

The core analytical challenge of single cell 'omics lies in transforming this noisy and fragmented count data into meaningful biological insights. One approach is to first identify underlying structure in how cells cluster or arrange themselves, then work backwards to understand the molecular features driving these patterns. Alternatively, we can start by identifying coordinated patterns in molecular features themselves, using these to infer cellular states and transitions. Both approaches must address the fundamental question of how molecular-level measurements connect to higher-level biological phenomena - how patterns of RNA counts, chromatin accessibility, or protein levels reflect meaningful cellular states and functions. Successfully bridging this conceptual gap is essential for translating the vast amounts of data generated by single cell technologies into advances in our understanding of biological systems.

## 1.4 Current methods for analyzing single cell data

Major technological and experimental advances over the last decades have enabled increasingly high-resolution dissection of molecular features including the genome, transcriptome, epigenome, proteome, metabolome, and more at the level of single cells. Throughput for these methods has also expanded rapidly, and now large information-rich datasets are increasingly easy to collect. However, realizing the full scope of possibilities afforded by this new capacity for data collection depends on the ability to overcome the

challenges discussed above and find ways to extract meaningful insights from these large and complex datasets. To achieve this, researchers employ two primary methodological frameworks: clustering and trajectory inference. Each addresses distinct aspects of cellular heterogeneity and functional organization and offers a different lens for understanding cellular population structures.

### 1.4.1 Clustering: defining discrete cell populations

Clustering aims to partition cells into distinct groups based on their similarity across measured features (e.g., gene expression). This approach assumes that cells within a cluster share functional properties or represent a common biological state.

Clustering methods in single cell analysis operate through a series of interconnected computational steps, each adapted from traditional methods to handle the unique characteristics of single cell data. The process begins with quality control and preprocessing, where cells are filtered based on quality metrics such as total counts, number of detected features, and proportion of mitochondrial reads (in sequencing-based methods). Normalization techniques, ranging from simple library size adjustment to more sophisticated methods accounting for technical covariates, help mitigate batch effects and technical variation [29, 30].

Feature selection identifies informative molecular markers while reducing computational burden and noise. This step often employs variance-based methods, but can also incorporate domain knowledge about marker genes or more sophisticated statistical approaches [31]. This is often followed by a subsequent step of dimensionality reduction, typically using variations of Principal Component Analysis (PCA) [30].

The final step is to perform the actual clustering, and there are a wide variety of algorithms with distinct strengths and weaknesses that have been used. Graph-based methods like Louvain or Leiden, combined with k-nearest neighbor graphs, have become popular due to their ability to handle large datasets and identify communities at multiple resolutions. Density-based methods can identify clusters of varying shapes and sizes, while hierarchical clustering can reveal relationships between populations. These and other base algorithms, such as k-means, are often modified with specialized distance metrics and weighting schemes before they are applied to single cell data [32].

Applications of clustering have been particularly successful in tumor heterogeneity analysis, where they have revealed previously unknown cancer cell states [33], and in immunology, where they have helped define new cell subtypes [34]. However, clustering methods tend to oversimplify complex biology by forcing cells into rigid categories, introducing artificial boundaries in continuous processes. Many methods also make strong assumptions about cluster properties, such as expecting spherical shapes or similar cluster sizes, making the detection of rare states or groups a particular challenge. The sensitivity of these methods to parameter choices - from feature selection to distance metrics to cluster number - further complicates their application, as decisions at each step of the analysis pipeline can substantially affect biological interpretations [35, 36, 37, 38].

### 1.4.2 Trajectory inference: mapping dynamic processes

While clustering focuses on static population structures, trajectory inference aims to reconstruct dynamic processes by arranging cells along a continuum or developmental pathway.

Trajectory inference methods reconstruct continuous cellular processes by ordering cells along developmental or state-transition paths. These methods begin with similar preprocessing steps to clustering but diverge in their core algorithms. The fundamental assumption is that cells captured at a single timepoint contain information about their developmental history or future states, allowing reconstruction of temporal processes from static measurements [39, 40].

Early methods like Monocle [41] and Wanderlust [42] introduced the concept of pseudotime ordering, where cells are arranged along a trajectory based on their molecular similarity. Modern approaches have evolved to handle more complex topologies, including branching processes and cyclical behaviors [40]. Methods like RNA velocity leverage splicing dynamics to infer directionality, providing additional information about cellular transitions [43].

The algorithmic approaches vary significantly. Some methods use minimum spanning trees or principal curves to find paths through high-dimensional space. Others employ manifold learning techniques to identify low-dimensional representations of cellular transitions. Graph-based approaches construct cellular networks and use path-finding algorithms to identify likely trajectories. More recent methods incorporate probabilistic models to account for uncertainty in trajectory inference [39, 40].

These methods have provided crucial insights in developmental biology, revealing new understanding of cell fate decisions and lineage commitment [42]. In disease research, they've helped map responses to drug treatments in cancer [44]. However, trajectory methods face several challenges. They often require substantial computational resources and can be highly sensitive to noise [39, 40]. Additionally, most of these methods are designed with specific biological processes in mind, assuming that continuous variation must reflect directional transitions like development or disease progression, rather than accommodating the broader spectrum of cellular continuity present in biological systems [26].

## 1.5 Balancing discrete and continuous perspectives: a unified approach

When analyzing single cell data, researchers must navigate the inherent tension between discrete and continuous representations of cellular identity. While clustering methods effectively identify distinct cellular populations that may represent stable functional units, trajectory inference methods capture dynamic processes and gradual transitions. Neither approach alone fully captures biological reality, as cellular identity can manifest not only as discrete cell types and dynamic transitions, but also as continuous spectra of molecular phenotypes within stable cell types. For instance, while neurons constitute a discrete cell type, their diverse functional roles are reflected in continuous variations in their molecular profiles - making both discrete and continuous representations essential for understanding cellular function. The challenge is further complicated by technical noise and stochastic biological variation, which can blur the boundaries between what might otherwise be discrete populations. Understanding how such complex population structures relate to cellular function remains a central challenge in single-cell biology. Progress requires computational methods that can flexibly capture both discrete and continuous representations while conveying uncertainty and facilitating biological interpretability. This goal motivates the methodological developments and applications presented in this thesis.

# Chapter 2: A hyperparameter-randomized ensemble approach for robust clustering across diverse datasets

A version of this chapter has been published as:

**Goggin, S.M.**, Zunder, E.R. (2024). ESCHR: a hyperparameter-randomized ensemble approach for robust clustering across diverse datasets. *Genome Biology*, 25, 242. https://doi.org/10.1186/s13059-024-03386-5

## 2.1 Abstract

Clustering is widely used for single cell analysis, but current methods are limited in accuracy, robustness, ease of use, and interpretability. To address these limitations, we developed an ensemble clustering method that outperforms other methods at hard clustering without the need for hyperparameter tuning. It also performs soft clustering to characterize continuum-like regions and quantify clustering uncertainty, demonstrated here by mapping the connectivity and intermediate transitions between handwritten digits (MNIST), and between hypothalamic tanycyte subpopulations. This hyperparameter-randomized ensemble approach improves the accuracy, robustness, ease of use, and interpretability of single cell clustering, and may prove useful in other fields as well.

## 2.2 Introduction

Clustering is widely used for exploratory data analysis across diverse fields, where it is applied to identify dataset grouping structures in an unsupervised manner. In particular, clustering has become a workhorse tool for single cell analysis, enabling the identification and characterization of cell populations that share similar molecular profiles within heterogeneous biological samples [45]. The output of clustering analysis is often used for direct comparison of biological samples, to identify changes in the abundance or molecular state of specific cell populations. Furthermore, clustering output is frequently carried forward into additional downstream analyses such as cell type classification or trajectory analysis [40, 46, 47]. Therefore, the accuracy and reproducibility of clustering partitions is important for the quality of single cell analysis. This importance has motivated the development of hundreds [48] of clustering methods with a variety of algorithmic strategies, but there are still important shortcomings in all of these methods which reduce their effectiveness.

An ideal clustering method for single cell analysis would satisfy the following requirements:

1) Operate without the need for human input such as hyperparameter tuning. The vast majority of existing methods require selection and optimization of hyperparameters, which can significantly impact clustering quality [35–38]. Manual hyperparameter tuning is time-consuming and relies subjectively on human intuition about which groupings appear correct [49]. Automated methods have been proposed to overcome this limitation, but many are computationally inefficient, and all are biased by the criteria used for optimization [38, 50–52].

2) Perform well across diverse single cell datasets from different tissues and across multiple measurement modalities such as single cell/single-nucleus RNA sequencing (scRNA-seq and snRNA-seq), single cell Assay for Transposase-Accessible Chromatin sequencing (scATAC-seq), flow cytometry, mass cytometry, and multiplexed imaging analysis such as high-content fluorescence imaging, imaging mass cytometry (IMC), multiplexed ion beam imaging (MIBI), and multiplexed error-robust fluorescence in situ hybridization (MERSCOPE). Generalizability is a concern in existing methods; many clustering methods perform well on gold-standard single cell datasets, but do not generalize well to datasets from other tissue types or from other single cell analysis modalities which may have different or more complex distributions or structural properties [36–38, 49, 53].

3) Produce stable and consistent partitions that are robust to random sampling and minor perturbations. Existing methods do not reliably produce robust partitions when applied to complex, high dimensional single cell datasets. Meaningfully different results can be produced with different hyperparameter combinations [37], slight perturbations of a dataset [49, 53], or even when an identical dataset and hyperparameters are run multiple times due to randomization steps in most clustering algorithms (Supplementary Fig. 1a,b).

4) Capture and describe the wide variety of discrete and continuous grouping structures present in single cell datasets [26, 54]. Most existing methods implement hard clustering, which assumes a data structure with discrete, well-separated groups, but is unable to characterize overlap or continuity between groups. Alternative computational methods for trajectory inference can better capture specific types of continuum-like processes such as cell differentiation in single cell datasets, but these methods make a different set of assumptions about data structure that can be equally restrictive.

5) Quantify uncertainty at the levels of individual data points and clusters. There are many scenarios where clustering can provide useful information, but a single optimal solution to the clustering task either does not exist or cannot be determined [55]. In many cases, there is additionally no known ground truth that could define what a correct solution might look like. Therefore, measures of uncertainty are crucial to assess the reliability and aid interpretability of clustering results before using them as inputs for downstream analytical methods or for purposes such as hypothesis development or orthogonal validation of results.

6) Scale to analyze large single cell datasets with millions of cells. While many of the most commonly used methods are scalable, several that have been developed to address these key challenges for clustering have done so at the expense of scalability. Methods that improve on these other challenges can only be realistically impactful if they can produce results for the large dataset sizes that are becoming increasingly commonplace.

Recently developed clustering methods have made progress towards some of these goals. Ensemble and consensus methods represent a promising approach to improve clustering robustness by combining information from multiple diverse partitions [56–63]. Fuzzy and soft clustering allow data points to belong to multiple clusters, and can therefore be used to provide a more complete description of both continuous and discrete data structures [64, 65]. There are several methods that provide measures of stability or uncertainty at the cluster level [38, 50, 62, 66], but cell-level measures of uncertainty are rarely provided in single cell methods [67, 68]. However, none of these approaches have been able to incorporate all of the six key features described above.

To address this need for a single method that performs robustly across diverse datasets with no hyperparameter tuning and transparently communicates uncertainty, we developed a clustering algorithm that applies EnSemble Clustering with Hyperparameter Randomization (ESCHR). This algorithm requires no human input due to hyperparameter randomization, which explores a wide range of data subspaces that contribute to the final consensus clustering step. Our implementation of ESCHR in Python (https://github.com/zunderlab/eschr) can be used as a self-contained framework for clustering, or it can be integrated into commonly used single cell analysis pipelines such as the scverse ecosystem [69]. To evaluate this new method, we performed extensive benchmarking tests, which demonstrated that ESCHR outperforms both general clustering methods and the most widely used clustering methods for single cell analysis [62, 70–72], both in terms of accuracy on synthetic datasets with a known “ground truth,” and in terms of robustness on real single cell datasets encompassing diverse tissues (bone marrow, pancreas, developing and adult brain), organisms (mouse, human), cell numbers (from hundreds to millions), and measurement techniques (single cell RNA sequencing, mass cytometry, flow cytometry).

After benchmarking for accuracy and robustness, we applied ESCHR clustering to two complex real-world datasets - first to the MNIST dataset [73], a commonly used example for machine learning image analysis, and then in the single cell context to investigate the relationships between tanycyte populations in the hypothalamus, which have been previously shown to display spatial and molecular-level continuity between subtypes [74–78]. In both of these exploratory analyses, the soft cluster assignments and uncertainty scoring from ESCHR were used to identify regions of low confidence cluster assignments corresponding to transitional overlap between clusters and map the key feature transitions that define these regions.

## 2.3 Results

### 2.3.1 Overview of ESCHR clustering

To develop a robust and scalable clustering method for analysis of single cell datasets, we employed an ensemble and consensus approach, which has been shown to improve robustness across many domains of machine learning [59, 79–85]. This approach consists of two main steps: 1) generate a set of base partitions, referred to as the ensemble, and 2) use this ensemble to generate a final consensus partition. The graph-based Leiden community detection method [86] was selected as a base algorithm to generate the clustering ensemble, because it is widely used for single cell analysis, and is efficiently implemented to be scalable for large datasets [47].

A key element of successful consensus approaches is generating sufficient diversity in the ensemble [59, 80, 81, 87]. To generate this diversity, ESCHR randomizes four hyperparameters for each base partition: subsampling percentage, number of nearest neighbors, distance metric, and Leiden resolution. Within a given base partition, ESCHR first selects a subsampling percentage by random sampling from a gaussian distribution with μ scaled to dataset size (within 30-90%), and then extracts the specified subset of data from the full dataset. Next, ESCHR randomly selects values for the number of nearest neighbors (15-150) and the distance metric (euclidean or cosine) and uses these to build a k-nearest neighbors (kNN) graph for the extracted subset of data. Finally, ESCHR performs Leiden community detection on this kNN graph using a randomly selected value for the required resolution-determining hyperparameter (0.25-1.75). The ranges for randomization of these hyperparameters were optimized empirically (Supplementary Fig. 2a-f and Methods). This subsampling and randomization scheme is used

to produce diversity amongst each of the different base partitions (Fig. 1a). This diversity provides many different views of the dataset, and the full ensemble of these views provides a more comprehensive picture of the dataset grouping structure (Supplementary Fig. 3), which is less likely to be influenced by the stochastic variations present in any single view, including the full unsampled dataset. In addition to generating ensemble diversity, this hyperparameter randomization approach is what enables ESCHR to operate without the need for hyperparameter tuning at this first stage of the algorithm.

After generating a diverse ensemble of base partitions, ESCHR applies a bipartite graph clustering approach to obtain the final consensus partition. First, the base partitions are assembled into a bipartite graph, where cells are represented by one set of vertices, base clusters are represented as a second set of vertices, and each cell is connected by an edge to each of the base clusters it was assigned to throughout the ensemble (Fig. 1b). Next, ESCHR applies bipartite community detection to obtain the final consensus partition (Fig. 1b) [88]. Bipartite community detection is applied here instead of more common consensus approaches that suffer from information loss [89]. To remain hyperparameter-free without the need for human intervention in this consensus stage of the algorithm, ESCHR performs internal hyperparameter selection to determine the optimal resolution for the final consensus clustering step by selecting the medoid from a range of resolutions (Supplementary Fig. 4). After obtaining the final consensus partition, ESCHR converts the ensemble bipartite graph to a final weighted bipartite graph by collapsing all base partition cluster nodes assigned to the same consensus cluster into a single node. Cells are then connected to these consensus cluster nodes by edges with weights representing the number of times each cell was assigned to any of the base partition clusters that were collapsed into a given consensus cluster (Fig. 1b). These raw membership values are then normalized to obtain proportional soft cluster memberships, and hard cluster labels are assigned as the consensus cluster in which a cell has the highest proportional membership (Fig. 1c).

While many analysis strategies for single cell datasets require hard clustering labels, these by definition cannot convey whether a cell is at the borderline between multiple clusters or located firmly in the center of a single cluster. Hard clusters also do not provide any insight into potential continuity between clusters. Using the soft cluster memberships derived from the weighted consensus bipartite graph, ESCHR provides several additional outputs beyond hard cluster assignments that enable more comprehensive characterization of the grouping structures within

a dataset. Firstly, soft cluster memberships can be directly visualized in heatmap form to identify areas of cluster overlap at the single cell level (Fig. 1c). Importantly, these soft membership heatmap visualizations can serve as complements or even alternatives to the widely used but also widely misinterpreted [90] stochastic embedding methods (i.e. UMAP [91], t-SNE [92] ) for visualizing the complex relational structures within single cell datasets. ESCHR also produces an Uncertainty Score for every object, derived from its soft cluster membership, which quantifies regions of higher and lower certainty in hard cluster assignment (Fig. 1c). Finally, ESCHR produces a cluster-level map of the continuity structure within a dataset by using the soft cluster memberships to calculate a corrected-for-chance measure of the connectivity between each pair of hard clusters (Fig. 1c and Methods).

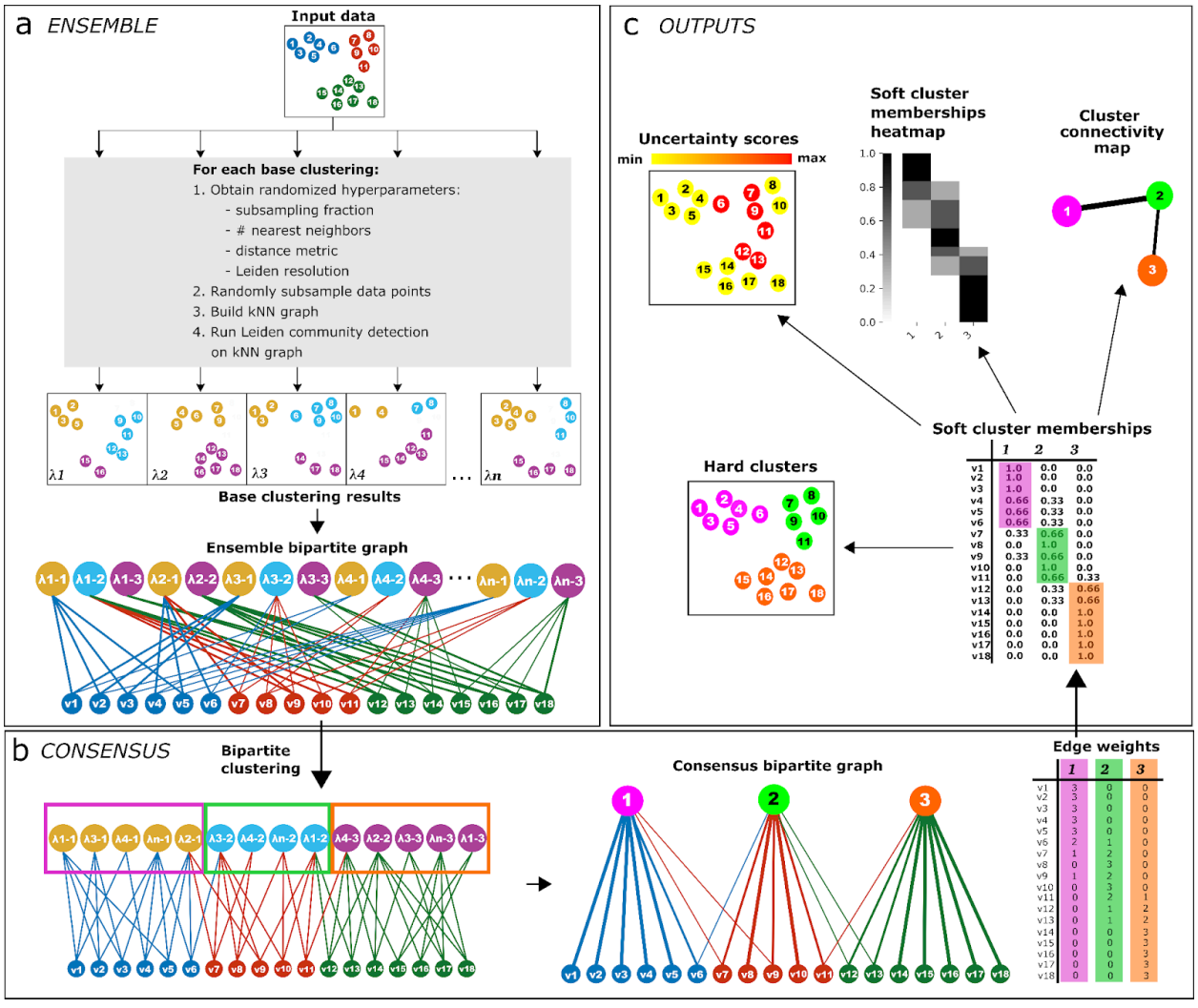



**Figure 1: ESCHR framework overview.**

(A) Starting from a preprocessed input dataset, ESCHR performs ensemble clustering using randomized hyperparameters to obtain a set of base partitions. This set of base partitions is represented using a

bipartite graph where one type of node consists of all data points and one type of node consists of all clusters from all base partitions and edges exist between data points and each base cluster they were assigned to throughout the ensemble. (B) Leiden bipartite clustering is performed on the ensemble bipartite graph. Base clusters are collapsed into their assigned consensus clusters obtained through the bipartite clustering and edge weights are summed such that each data point now has a weighted edge to each consensus cluster representing the number of base clusters it had been assigned to the were then collapsed into that consensus cluster. (C) Soft cluster memberships are obtained by scaling edge weights between 0 and 1, and can then be visualized directly in heatmap form and used to generate hard cluster assignments, per-data point uncertainty scores, and cluster connectivity maps.

### 2.3.2 ESCHR soft clustering and uncertainty scores capture diverse structural characteristics and quantify uncertainty in cluster assignments

We first sought to examine how ESCHR uncertainty scores and soft clustering could enable effective and informative analysis for datasets containing complex combinations of continuity and discreteness, and how these results compared to a wide range of alternative clustering methods used for single cell analysis or general purpose clustering (Supplementary Table 1 and Methods). For this analysis, we generated a synthetic scRNA-seq dataset containing 1000 cells and 1000 features using the DynToy package [93]. This dataset is generated by sampling “cells” from a complex trajectory model, with library size and transcript distributions per cell modeled on a real scRNA-seq dataset. Specifically, “cells” are sampled from prototypical “cell states”, where each cell has a varying probability of belonging to multiple neighboring states and the ground truth hard cluster labels are assigned as the state in which the cell has the highest percent membership. This process generates a dataset which is similar to real single cell data but provides known ground truth grouping structure and known ground truth continuity structure (Fig. 2a-b, Supplementary Fig. 7a), which is not generally available for real datasets (Supplementary Note 1).

We first compared the ESCHR hard clustering results (Fig. 2c, Supplementary Fig. 7b) and uncertainty scores (Fig. 2d) with the true hard cluster labels and the true membership percentage for those labels. While ESCHR successfully captures all of the ground truth cell states, it also adds two additional clusters (ESCHR clusters 9 and 6) between true clusters M2 and M3 and between M1 and M7. However, the ground truth membership percentages show that these regions are highly transitional, with low percentages for the maximum membership (Fig. 2b). ESCHR uncertainty scores correspond closely to this observed ground truth continuity in Figure 2b, indicating that the uncertainty scores can identify regions of uncertainty in cluster assignment due to ground truth continuity and cluster overlap. In addition to quantifying this

level of uncertainty per ”cell”, ESCHR also provides information at the cluster level about which clusters overlap, and to what extent, through direct visualization of the soft cluster memberships. This reveals overlap structure that corresponds to the ground truth patterns of transitional membership between groups, such as between ESCHR clusters 7, 9, and 1 (corresponding to true labels M2 and M3) and ESCHR clusters 1, 8, and 2 (corresponding to true labels M3, M6, and M5) (Fig. 2e).

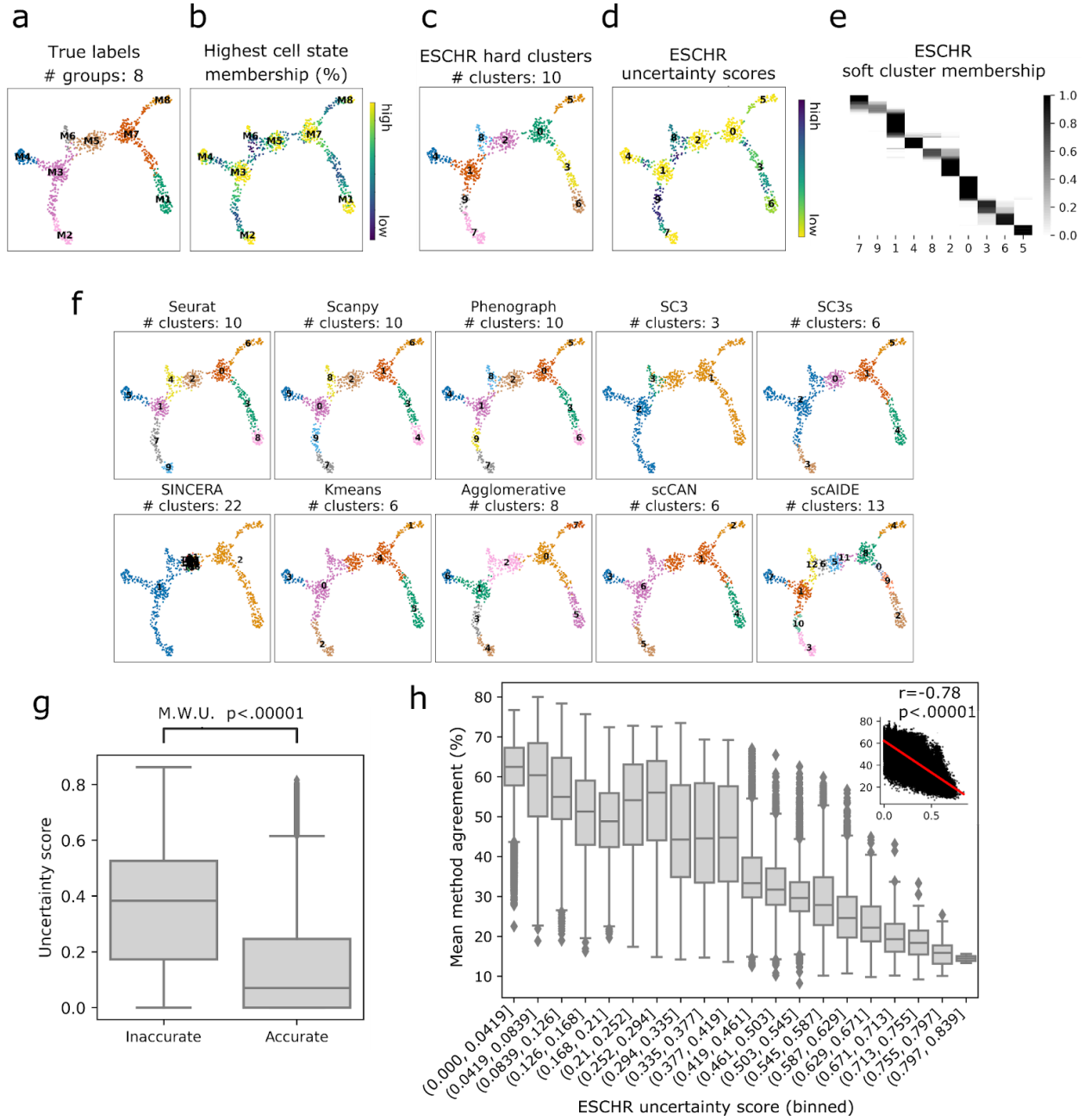



**Figure 2: Visualization of ESCHR clustering and uncertainty scores compared to other clustering methods.**

UMAP visualizations of (A) ground truth cluster labels, (B) ground truth cell state membership, (C) ESCHR hard clusters, and (D) ESCHR uncertainty scores. (E) Heatmap visualization of ESCHR soft cluster memberships. (F) UMAP visualizations of cluster assignments from selected comparison methods. Points are colored by cluster ID. (G) Box and whisker plot comparing uncertainty scores of data points from ESCHR hard clustering that were accurately assigned versus not accurately assigned. The box shows the quartiles of the dataset, whiskers extend to 1.5*IQR, plotted points are outliers. Two-sided Mann-Whitney U test was used for statistical analysis. N = 126545, 750955 for inaccurate and accurate groups respectively. (H) Comparison of ESCHR uncertainty scores versus method agreement per each individual data point. Primary box and whisker plot x-axis is binned ESCHR uncertainty scores and y-axis is the average method agreement across all pairs of methods; inset scatterplot shows raw data (i.e. not binned) with red line of best fit and Pearson correlation statistic.

We next evaluated the results from multiple different clustering methods and found that there was wide disagreement between the results of these different methods (Fig. 2f, Supplementary Fig. 7c). Seurat, Scanpy, and Phenograph, which are all based on either leiden or louvain as their base clustering method, all identify approximately the same clusters as ESCHR, but importantly each of these methods has selected different boundaries between these clusters. While the results from the remaining methods exhibit more diversity, it is notable that none have placed cluster boundaries within the regions of ground truth high single state membership but rather have over-clustered transitional regions or under-clustered by grouping multiple true clusters together. The regions of disagreement between the different clustering methods highlight areas that are challenging for and perhaps not well suited to the discreteness assumptions of traditional hard clustering. High ESCHR uncertainty scores and overlapping soft cluster memberships correspond to regions of disagreement between other clustering methods, providing further evidence that these metrics can help identify regions that are challenging for traditional clustering methods due to continuous data structures such as overlap between ground truth clusters.

To assess whether ESCHR uncertainty scores were similarly informative across diverse datasets, we generated an additional 4 simulated datasets using DynToy and 16 additional structurally diverse synthetic datasets which consist of randomly generated gaussian distributions varying in number of objects (5000 or 10000), number of features (20,40,50,60), number of clusters (3,8,15,20), cluster sizes, cluster standard deviations, cluster overlap, and feature anisotropy (Supplementary Figs. 5-6, Supplementary Tables 2-3). To quantitatively evaluate the utility of ESCHR uncertainty scores across our full set of 21 structurally diverse synthetic datasets with ground truth cluster labels, we first compared ESCHR uncertainty scores to the accuracy of assignment compared to ground truth labels per data point across all

datasets, and found that ESCHR uncertainty scores were significantly higher in inaccurately assigned cells (Fig. 2g). We then quantified the level of agreement between clustering assignments from all the different clustering algorithms we tested (in Fig. 2f) and used this as an alternative external indicator for per data point uncertainty and difficulty of clustering (Methods). This analysis revealed that higher ESCHR uncertainty scores were significantly negatively correlated with method agreement (Fig. 2h). Taken together, these comparisons demonstrate that ESCHR uncertainty scores identify meaningful uncertainty, and that when used in combination with the soft clustering results, they enable more in-depth interpretation of dataset structure than other methods which produce only hard cluster assignments. Furthermore, ESCHR is able to provide these high-quality insights for datasets with diverse structural characteristics without the need for human intervention such as hyperparameter tuning.

### 2.3.3 ESCHR outperforms other methods across measures of accuracy and robustness

To systematically evaluate the performance of ESCHR vs. other clustering methods on real datasets as well as synthetic ones, we performed systematic benchmarking of ESCHR against other clustering algorithms (Supplementary Table 1) using a collection of 45 published real datasets in addition to the 21 synthetic datasets described above. This collection of 45 published datasets vary widely in size (300-2,000,000 cells), source tissue (e.g. blood, bone marrow, brain), measurement type (sc/nRNA-seq, mass cytometry, flow cytometry, non-single cell datasets), and data structure (varying degrees of discreteness and continuity) (Supplementary Table 4). For our evaluation criteria, we selected two extrinsic evaluation metrics, Adjusted Rand Index (ARI) and Adjusted Mutual Information (AMI), to assess two aspects of the clustering results: 1) accuracy and 2) robustness. Extrinsic evaluation metrics measure the distance of a clustering result to some external set of labels, and our two selected metrics ARI and AMI represent different approaches to this problem, with divergent biases. ARI tends to yield higher scores in cases of similarly sized clusters and similar numbers of clusters within and between the partitions being compared, while AMI is biased towards purity and yields higher scores when there are shared pure clusters between the two partitions (Methods) [94]. Using ARI and AMI together should therefore provide a more complete comparison of clustering performance [51, 52].

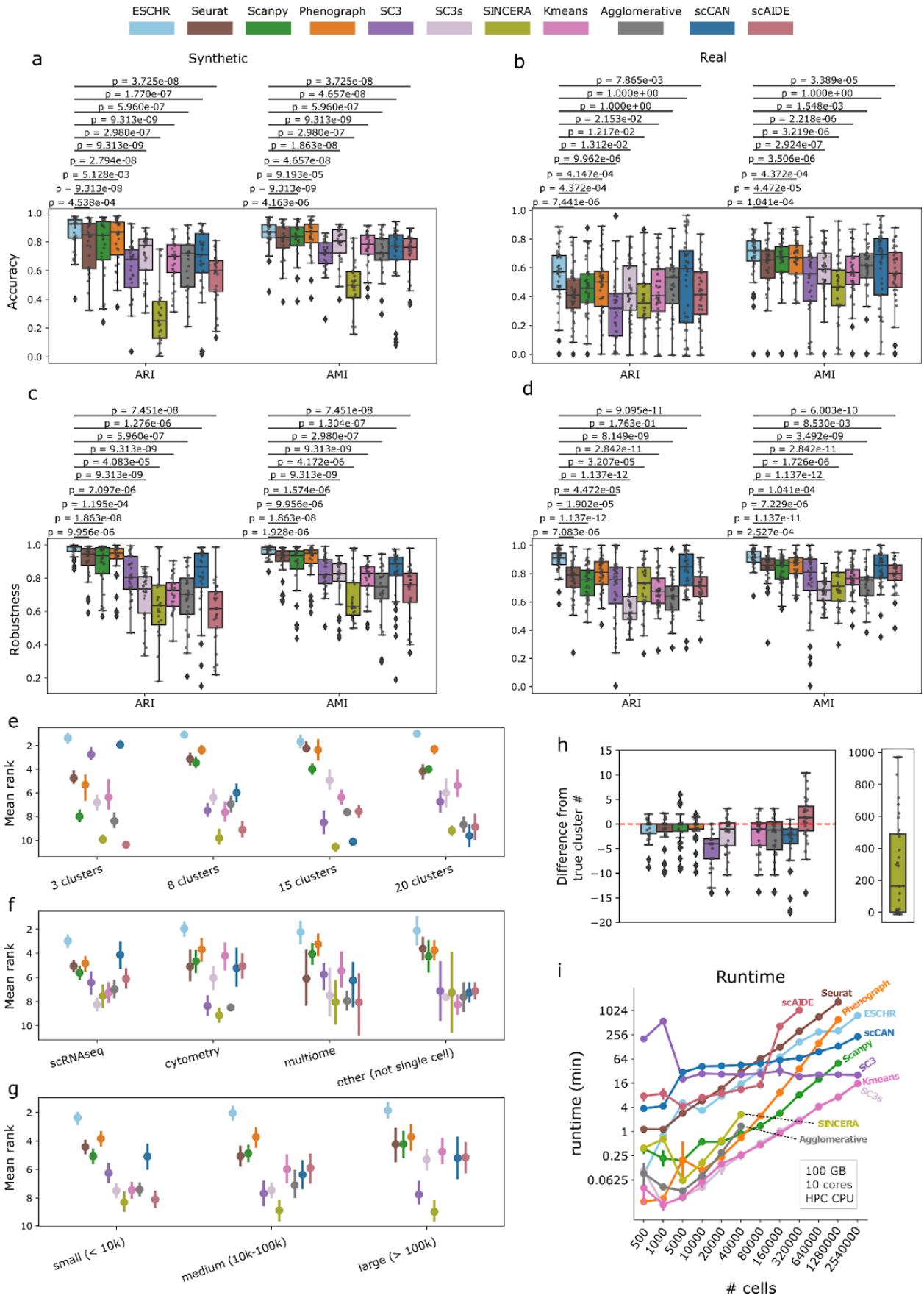

ESCHR
Seurat
Scanpy
Phenograph
SC3
SC3s
SINCERA
Kmeans
Agglomerative
scCAN
scAIDE
a
Synthetic
b
Real
c
d
Accuracy
Robustness
ARI
AMI
e
f
g
Mean rank
3 clusters
8 clusters
15 clusters
20 clusters
scRNAseq
cytometry
multiome
other (not single cell)
small (< 10k)
medium (10k-100k)
large (> 100k)
h
Difference from true cluster #
i
Runtime
runtime (min)
# cells
100 GB
10 cores
HPC CPU

**Figure 3: Systematic analysis of ESCHR clustering performance compared to competing methods on synthetic and real datasets.**

(A-D) Box and whisker plots comparing accuracy (A and B) and robustness (C and D) of results from ESCHR and all comparison methods across all synthetic (A and C) and real (B and D) benchmark datasets as measured by ARI (left) and AMI (right). Boxes show the quartiles of the dataset, whiskers extend to 1.5*IQR. Data points used in creation of box and whisker plots and shown in overlaid scatterplots are the means across 5 replicates for each dataset. Two-sided Wilcoxon signed-rank test with Bonferroni correction was used for statistical analysis comparing ESCHR to each method. N = 21 for comparisons using synthetic datasets and N = 45 for comparisons using real datasets. (E) Mean rank across all metrics shown in box-and-whisker plots for different cluster numbers of the synthetic datasets. Error bars show 1 standard deviation. (F) Mean rank across all metrics shown in box-and-whisker plots for different modalities of the real datasets. Points represent means across all replicates of all datasets in a given category and error bars show 1 standard deviation. (G) Mean rank across all metrics shown in box-and-whisker plots for different sample number bins for all real and synthetic datasets. Points represent means across all replicates of all datasets in a given category and error bars show 1 standard deviation. (H) Box plots of difference from the true cluster number for each synthetic datasets for each method. Values below zero reflect calculated cluster numbers being lower than true cluster numbers and higher than zero indicates more clusters than the true cluster number. SINCERA is shown separately due to the scale of values being 2 orders of magnitude different from all other methods. (I) Scalability comparison between ESCHR and other methods on synthetic datasets with increasing number of data points. X-axis is log scaled but labels show the unscaled values for easier interpretation. Each dot represents 5 replicates and error bars show 1 standard deviation.

When we applied these extrinsic metrics ARI and AMI to assess clustering accuracy for our collection of synthetic datasets, ESCHR outperformed all other clustering algorithms across both metrics, and this superior performance was statistically significant for all cases (Fig 3a, Supplementary Table 5, Supplementary Fig. 8b-c). We also applied ARI and AMI to benchmark clustering accuracy in non-synthetic real datasets, although it is important to note that a priori known class labels do not generally exist for real-world single cell datasets, and the various proxies accepted as ground truth labels should be interpreted with skepticism (discussed further in Supplementary Note 1). Keeping these caveats in mind, ESCHR still clustered real datasets more accurately by ARI and AMI than all methods, significantly so in all comparisons except for scCAN and Agglomerative clustering by ARI and only scCAN by AMI (Fig 3b, Supplementary Table 6, Supplementary Fig. 8b-c). Many of the ground truth labels that are widely accepted for real single cell datasets are based on a hierarchical framework of clustering or manual labeling, which could explain why agglomerative clustering performs better relative to the other methods for this particular comparison.

After benchmarking for accuracy, we next used ARI and AMI to evaluate clustering robustness, by comparing results from repeated runs with different random subsamples of a given dataset (Methods). Due to its ensemble and consensus clustering approach, we expected ESCHR to

perform well in these tests of robustness, and indeed it demonstrated superior performance to all other clustering algorithms on both synthetic and real data across both ARI and AMI metrics (Fig. 3d,e, Supplementary Fig. 8d-e). These results were significant for all comparisons except against scCAN on the real datasets by ARI (Supplementary Tables 5-6). To gain insight into the generalizability of ESCHR versus the other methods for specific dataset types, we calculated the mean rank of each clustering algorithm across all metrics for major subcategories of our collection of datasets: cluster number for synthetic datasets (for which we have reliable ground truth cluster numbers), data modality for real datasets, and sample number across all datasets. Different clustering algorithms perform better or worse for different subsets, but ESCHR is consistently ranked first or tied for first across these subcategories of both synthetic (Fig. 3e,g) and real datasets (Fig. 3f,g), indicating that its performance is more generalizable to diverse datasets than the other tested clustering algorithms.

We next evaluated the scalability of each method over a range of dataset sizes. While ESCHR generally takes the longest, this does not present a practical limitation for typical usage, as it is able to successfully complete analyses on millions of data points and the runtime scales linearly (Fig. 3g). This analysis also revealed that several of the alternative clustering algorithms we tested could not successfully run to completion for larger datasets. The dataset size limit for ESCHR is effectively the size limit of its underlying base clustering method, the Leiden algorithm implemented in Python [86]. While it is true that our method does have longer runtimes than some of the commonly used methods we compare to here, we believe it is worth the wait due to the demonstrated superior accuracy and robustness of our results, and perhaps even more importantly due to the additional insights afforded by the uncertainty scores and soft cluster membership information highlighted in Figure 2. Additionally, the manual guess-and-check hyperparameter tuning that is required to achieve desired results with other methods can be very time consuming (not to mention highly subjective), and so it is possible that in practical usage ESCHR could potentially end up providing useful results more quickly than other methods. When taken together, these quantitative evaluations demonstrate that ESCHR performs favorably compared to the other methods tested here and achieves our desired goals of providing accurate and robust results, being generalizable to a broad range of diverse datasets, and being scalable to large datasets.

### 2.3.4 ESCHR soft clustering and uncertainty scores provide increased interpretability in exploratory data analysis of the MNIST dataset

To illustrate how ESCHR can identify regions of continuity and provide insight into cluster overlap and dataset structure, we selected the MNIST dataset for further analysis. This dataset, consisting of 70,000 handwritten digits with ground truth labels, is often used for machine learning demonstrations because the images can be visualized for intuitive interpretation [73]. Other clustering algorithms set to default hyperparameters do not recapitulate the ground truth labels with high accuracy (Supplementary Fig. 9a), explained in part by the real variation that exists within the ground truth sets. For example, there are two common variations of the handwritten digit 1, and most of the clustering algorithms capture this difference. Of all the clustering algorithms tested, ESCHR clusters the MNIST dataset with the highest robustness and accuracy (Supplementary Fig. 9b), but it consistently splits the 1 and 9 digits into separate subsets (Fig. 4a,b), and in some cases it splits the digit 4 as well (Supplementary Fig. 9a). ESCHR usually produces highly consistent results from run to run thanks to its consensus clustering step, but this inconsistency around the digits 4 and 9 is suggestive of a high degree of continuity within and between these two classes (Supplementary Fig. 9c), which is highlighted by elevated ESCHR uncertainty scores in this region (Fig. 4c). The soft cluster membership heatmap also draws attention to the visual similarities between digits 3, 5, and 8, as well as the two types of handwritten 1 digits (Fig. 4d). These subset-level differences and connections between related digits motivated further investigation of the ESCHR outputs for the MNIST dataset.

To further investigate the continuity and overlap structure that was indicated by the uncertainty scores and soft cluster membership heatmap, cluster connectivity mapping was applied to identify significant overlap beyond what would be expected by random chance for the ESCHR clusters (Fig. 4e) (Methods). This revealed significant overlap between clusters “3”-”5”-”8”, “1a”-”1b”, and “4”-“9a”-“9b”. To explore the nature of the continuity structure underlying the significantly overlapping clusters “1a” and ”1b”, we devised a simple rank ordering scheme based on the soft membership values for the datapoints in these two clusters, and then used this ordering score to examine both the continuous progression of soft membership values across the rank-ordered datapoints and their density along this ordering score (Methods). This revealed that each cluster had a high density peak of “core” datapoints with a secondary smaller “transitional” peak (Fig. 4f, bottom). Individual representative MNIST digit images (Fig. 4f, top

row) and summed pixel intensities (Fig. 4f, second row) from the images within each of these regions indicate that the core “1b” images are heavily slanted whereas the core “1a” images are vertically straight, with the images from the lower density transitional peaks falling in between these extremes. The two high density peaks consisting of images with distinctly different styles of 1s explain why ESCHR and many of the other clustering methods tested identified two clusters corresponding to this single digit (Supplementary Fig. 9a), while the high degree of pixel overlap between the two styles and the presence of images with intermediate slantedness explain the high degree of continuity and significant overlap detected by ESCHR.

We next examined the more complex relationship between subsets of the digits 4 and 9. Cluster connectivity mapping indicated that there was significant overlap among all three of the ESCHR clusters “4”,“9a”, and “9b” (Fig. 4e). Additionally in the soft membership heatmap, there appear to be some cells that are overlapping all three clusters, and some cells from clusters “9a” and “9b” that overlap separately with cluster “4” and not with each other (Fig. 4d). Unlike the simpler relationship between ESCHR clusters “1a” and ”1b,” which could be analyzed by linear one-dimensional reduction, the more complex relationship around digits 4 and 9 could not be adequately captured or described along a single dimension, so principal components analysis (PCA) was applied to the ESCHR soft cluster memberships corresponding to these three clusters in order to reduce these relationships into two dimensions (Methods). Representative images selected from throughout the resulting PC space reveal that the between-cluster continuity is indeed reflecting the existence of a continuous progression through different conformations of the two digits 4 and 9 (Fig. 4f). Specifically, we can see that there is continuous progression through the 9’s based on how slanted they are, with two areas of higher density at either extreme. This explains why a clustering algorithm would be likely to split this into two clusters, albeit with a high amount of uncertainty about precisely where to make the split. The images also illustrate how the more slanted closed 4’s form a continuous transition primarily with cluster 9a and more vertically oriented closed 4’s form a continuous transition primarily with cluster 9b. This approach also allows us to identify features that are most correlated with the top two principal components. The top PC-correlated features lend further insight by identifying the specific pixels that are primarily capturing these changes in slantedness and upper loop closure (Fig. 4g). These analyses illustrate how structures within the MNIST dataset are not ideally suited for hard clustering assignment, but also how ESCHR is able to identify these structures and provide deeper insights than could be obtained by other hard clustering methods, or even beyond what is available from the ground truth class assignments.

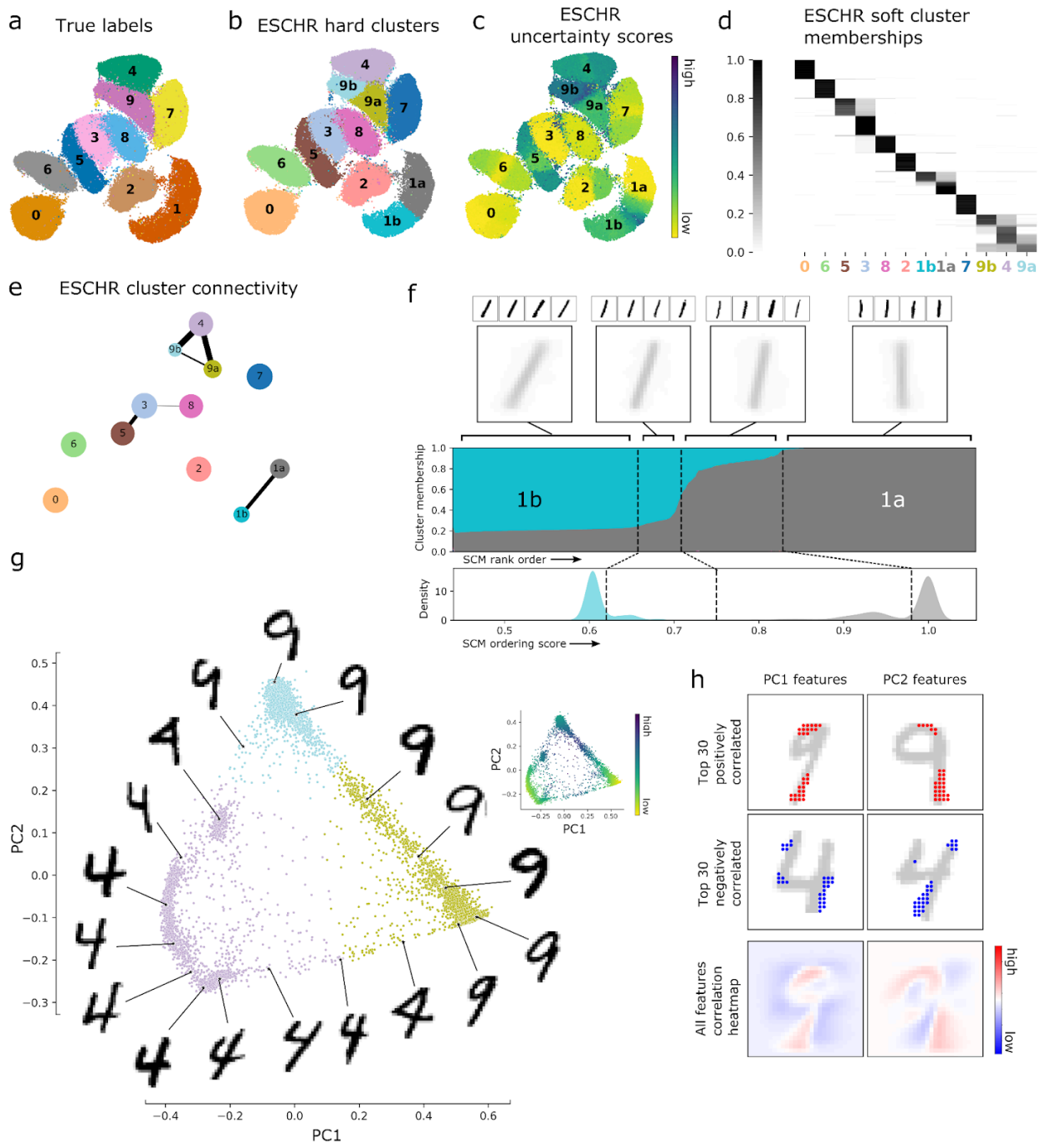



**Figure 4: ESCHR-guided exploration of the benchmarking dataset MNIST.**

(A) UMAP visualization with points colored by true class labels. (B) UMAP visualization with points colored by ESCHR hard cluster labels. (C) UMAP visualization with points colored by ESCHR uncertainty score. (D) Heatmap visualization of ESCHR soft cluster memberships. (E) Nodes represent ESCHR hard clusters and are located on the centroid of the UMAP coordinates for all data points assigned to that hard cluster. Node size is scaled to the number of data points in a given cluster. Edges exist between nodes which were determined to have significant connectivity by ESCHR cluster connectivity analysis, and edge thickness is scaled to the connectivity score. (F) Stacked bar plot showing the soft membership of datapoints in clusters 1b and 1a, ordered by increasing ESCHR soft cluster membership (SCM) rank ordering score (middle); kernel density estimation across the ordering score (bottom); dashed lines indicate boundaries between ordering score density peaks to separate “core” and “transitional” datapoints

(middle and bottom); smaller images show individual representative images and larger images show summed pixel intensities for all datapoints contained within each dashed partition (top). (G) Visualization of data points from ESCHR clusters 4, 9a, and 9b projected onto the first two principal components resulting from PCA performed on the soft membership matrix of these three clusters. Primary scatterplot shows points colored by their ESCHR hard cluster assignment, and inset scatterplot shows points colored by ESCHR uncertainty score. Images are real examples from the MNIST dataset. (H) Scatterplot points in the first two rows of plots show the pixel locations of the 30 features with the largest positive (first row, red) and 30 largest negative (second row, blue) Pearson correlation to each of the PCs. Example digit images are underlaid in light gray to aid interpretation. The final row contains heatmaps with each pixel colored according to its Pearson correlation with PC1 (left) or PC2 (right), with bright red indicating a large positive correlation and dark blue indicating a large negative correlation.

### 2.3.5 ESCHR captures cell types and continuity in static adult tissue

To illustrate how ESCHR can provide additional interpretability and insight for single cell datasets, we selected an integrated scRNA-seq dataset of hypothalamic tanycytes [77] for further analysis. Tanycytes are elongated ependymoglial cells that form the ventricular layer of the third ventricle and median eminence, and have historically been classified into four subtypes (α1, α2, β1, β2) based on the hypothalamic nuclei where they project to, their spatial localization along the third ventricle, and their morphological, structural, genetic, and functional properties (Fig. 5a) [95]. More recent studies have suggested that many of these properties may exhibit substantial continuity between and within each of these subtypes [74–78, 96]. However, individual tanycyte scRNA-seq studies and an integrated analysis of these datasets all reported discrete groupings of tanycytes defined by hard clustering approaches [75, 77, 97–99], with no insight into the robustness of these assignments and whether there is overlap or continuity between them.

Initial ESCHR analysis produced hard clustering outputs that match canonical tanycyte subtypes by their RNA expression profiles (Fig. 5b-d) [77]. Subtypes β1 (expressing Fizb, Penk, Rlbp1, and Ptn) and α2 (expressing Vcan, Nr2e1, Fabp5, and Slc7a11) are represented by multiple hard clusters, while the subtypes β2 (expressing Scn7a, Cal25a1, Meat, and Lrrtm3) and α1 (expressing Mafb, Necab2, Agt, Slc17a8, and Lyz2) each correspond to a single hard cluster, indicating that there is more transcriptional diversity within the β1 and α2 populations. On top of this however, ESCHR uncertainty scores identify substantial heterogeneity within each hard cluster, including the β2 and α1 clusters (Fig. 5c), and the soft cluster memberships reveal additional levels of overlap and continuity between these canonical tanycyte subtypes (Fig. 5e). ESCHR cluster connectivity mapping (Methods) revealed significant overlap between the β1 clusters (2, 3, and 5) and each of the other three canonical subtypes (Fig. 5f). This result was

somewhat unexpected, because transcriptional continuity was previously thought to exist only between spatially neighboring tanycyte subtypes [76, 96]. A more recent study provided evidence that β1 tanycytes exhibit some transcriptional continuity with both α1 and α2 tanycytes, but also indicated that β2 tanycytes were non-overlapping and transcriptionally distinct [74]. Our analysis with ESCHR soft clustering memberships and cluster connectivity provide additional corroboratory evidence for the transcriptional continuity between β1 and α1/α2 tanycytes, but also reveal a previously uncharacterized relationship of transcriptional continuity between β1 and β2 tanycytes.

To further investigate this previously uncharacterized transcriptional overlap between β1 and β2 tanycytes, specifically between ESCHR clusters 1 and 2, we selected the subset of cells comprising the transitional zone between clusters, and rank ordered these based on whether their soft cluster membership was closer to β1 (ESCHR cluster 1) or β2 (ESCHR cluster 2) (Fig. 5g and Methods). Using this rank ordering scheme, we identified genes with expression patterns that correlate with progression through the transition zone from β2 to β1 tanycytes, either decreasing across the transition like Igfbp5 (Fig. 5h,k), peaking during the transition like Tgfb2 (Fig. 5i,l), or increasing across the transition like Crym (Fig. 5j,m). We next sought to determine whether these gene expression patterns in the transitional zone between ESCHR clusters were also observed in the spatial distribution of β2 and β1 tanycytes along the median eminence and third ventricle where these subtypes are thought to reside (Fig. 5n). To investigate this possibility, we examined the in situ hybridization (ISH) database from the Allen Mouse Brain Atlas (ABA; http://mouse.brain-map.org) [100] and observed that the overlapping expression for these three genes did in fact manifest as progressive spatial overlap spanning the anatomical regions canonically associated with β2 and β1 populations (Fig. 5o-q). Altogether, this analysis of tanycyte subtypes demonstrates the utility of ESCHR for 1) identifying robust and biologically meaningful hard cluster assignments, 2) providing insight into the overlap and continuity between cell type clusters, and 3) providing a springboard for further analysis of expression level transitions via soft cluster membership ordering.

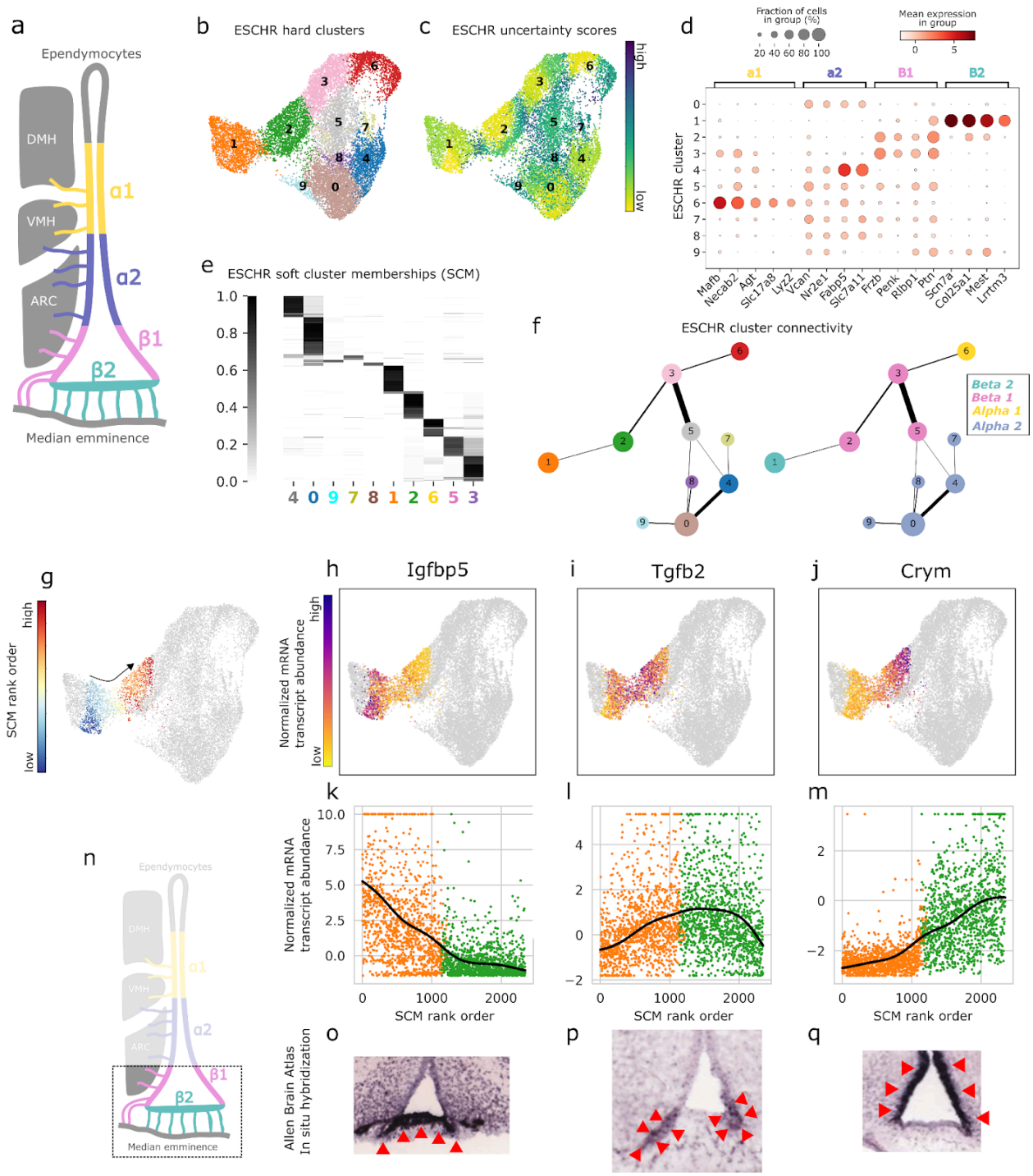



**Figure 5: ESCHR identifies continuity between and within canonical cell subtypes in static adult tissue.**

(A) Schematic illustration of canonical tanycyte subtypes in their anatomical context surrounding the third ventricle. (B) UMAP visualization with points colored by ESCHR hard cluster labels. (C) UMAP visualization with points colored by ESCHR uncertainty score. (D) Heatmap dotplot showing expression of marker genes for the canonical tanycyte subtypes across the ESCHR hard clusters. (E) Heatmap visualization of ESCHR soft cluster memberships. (F) Nodes represent ESCHR hard clusters and are located on the centroid of the UMAP coordinates for all data points assigned to that hard cluster. Node

size is scaled to the number of data points in a given cluster. Edges exist between nodes which were determined to have significant connectivity by ESCHR cluster connectivity analysis, and edge thickness is scaled to the connectivity score. Node colors map to their ESCHR hard cluster colors from panel B (left) and to the color from panel A of the canonical subtype to which they primarily belong (right). (G) UMAP visualization with the subset of points which were included in the ordering analysis colored by ESCHR soft cluster membership (SCM) rank ordering score, and all others colored gray. (H-J) UMAP visualizations where points included in the ordering analysis are colored by their expression level and all others are colored gray. (K-M) Scatterplots showing normalized mRNA abundance on the y-axis and SCM rank order on the x-axis. Expression is bounded between the 2nd and 98th percentiles. Lines show gaussian-smoothed B-splines fit to the data. (N) Schematic illustration of the anatomical region being shown in O-Q. (O-Q) In situ hybridization (ISH) of coronal brain sections, using probes specific for Igfbp5, Tgfb2, and Crym (Allen Mouse Brain Atlas). Red arrowheads indicate the areas of expression in the region of interest.

## 2.4 Discussion and conclusions

Clustering is a fundamental tool for single cell analysis, used to identify groupings of cell types or cell states that serve as the basis for direct comparisons between biological samples or between specific cell types within a biological sample, as well as numerous further downstream applications. However, it has proven challenging to generate appropriate and consistent cell groupings when using previously available clustering methods on single cell datasets, due to 1) continuity and overlap between cell types, 2) randomness and stochasticity built into the clustering algorithms, and 3) non-generalizable hyperparameter settings that were optimized for a specific dataset or data type. To overcome these limitations we developed ESCHR, a user-friendly method for ensemble clustering that captures both discrete and continuous structures within a dataset and transparently communicates the level of uncertainty in cluster assignment. Using a large collection of datasets representing a variety of measurement techniques, tissues of origin, species of origin, and dataset sizes, we benchmarked ESCHR's performance against several other clustering algorithms, demonstrating that ESCHR consistently provides the highest robustness and accuracy for clustering across all categories of this diverse dataset collection.

One of the key design features of ESCHR is our approach using hyperparameter randomization during the ensemble generation step. While this was a deliberate design choice to generate diversity amongst the base clusterings to enhance robustness and generalizability of clustering, an additional benefit is that it removes the need to manually test and select an optimized set of hyperparameters for each dataset. This design also affords several avenues for potential future improvements to the ESCHR algorithm, such as expanding the number of hyperparameters randomized in order to generate an even more diverse clustering ensemble. For example, we currently use k-nearest neighbor (kNN) graphs for the base Leiden clustering steps, but mutual

nearest neighbor (mNN) or shared nearest neighbor (sNN) have shown good performance in other frameworks [47, 101, 102], and may improve ESCHR performance if incorporated as an additional hyperparameter to vary. ESCHR may also benefit from expanding the set of distance metrics utilized. We currently restrict our analysis to euclidean and cosine distances due to their efficient implementations within our chosen fast approximate nearest neighbor (ANN) package [103]. However, recent research has demonstrated the efficacy of a broader range of distance metrics for capturing diverse data structural properties [104]. While not all of these metrics may be applicable in an ANN context, several may hold potential for enhancing the quality of our clustering outcomes. Additionally, the current version of ESCHR uses only Leiden community detection for clustering in the ensemble stage, but additional base clustering methods could be explored and potentially incorporated in future versions. Finally, our empirical identification of optimal ranges for ESCHR's numeric hyperparameters was somewhat limited by the time and memory required for running these experiments with many, sometimes large, datasets and very wide search spaces. It is therefore possible that there may be more optimal default ranges or more sophisticated regimes for hyperparameter randomization and selection that could improve ESCHR's performance.

Another key design feature of ESCHR is our soft clustering approach for generating the final consensus results. Single cell data is inherently complex and heterogeneous, and clustering methods often make assumptions about the structure of the data that may not hold in practice. For example, hard clustering methods assume discrete groups of single cells, which rarely exist in biological data [26]. Many clustering algorithms make further assumptions about the shapes and other properties of these discrete groups. In the opposite direction, toward continuity rather than discreteness, numerous methods have been developed for trajectory inference in single cell datasets [40], but these methods also make assumptions about dataset structure, for example many force a branched tree structure. ESCHR's soft cluster outputs enable unified mapping of both discrete and continuous grouping structures, without the need for assumptions about the shape and properties of the dataset. To illustrate this concept, we used ESCHR to identify tanycyte subtypes and reveal the transitional continuity between them (Fig. 5a-q, Supplementary Fig. 10), which is notable because assumptions about lineage relationships or dynamic developmental processes in this static adult tissue would be inappropriate and could lead to inaccuracies and distortion. Instead, ESCHR can identify and characterize discrete and continuous patterns simultaneously, even in the same dataset, without relying on assumptions about data shape and properties.

One of ESCHR's most useful outputs is the per-cell uncertainty score, which enables users to estimate clustering uncertainty and interpret hard clustering results more effectively. The Impossibility Theorem for clustering states that it is impossible for any clustering method to satisfy the three proposed axioms of good clustering, and therefore all clustering algorithms must make trade-offs among the desirable features, and no clustering result can be perfect [55]. Because of this, it is critical to evaluate the guaranteed uncertainty in a clustering result before using it for direct comparisons, downstream analyses, or hypothesis generation. ESCHR uncertainty scores, which are derived from the degree of cluster overlap for each datapoint as indicated by their soft cluster assignments, provide a useful proxy for this uncertainty and difficulty in cluster assignment. These scores can be visualized alongside hard cluster assignments to facilitate more discerning interpretation of clustering results. We have validated the utility of these uncertainty scores by demonstrating that (1) they identify areas of ground truth continuity due to cells transitioning between cell states in simulated scRNA-seq data (Fig. 2b,d), (2) they are significantly higher for inaccurately assigned data points (Fig. 2g), and (3) they are significantly negatively correlated with the level of agreement between clustering algorithms (Fig. 2h). Altogether, these findings demonstrate that ESCHR uncertainty scores provide meaningful insights into clustering uncertainty.

To make the advantages of ESCHR clustering easily accessible to the research community, we have made ESCHR available as a Python module on github (https://github.com/zunderlab/eschr), packaged as an extensible software framework that is compatible with the scverse suite of single cell analysis tools [69]. We have provided tutorials for how to incorporate it into existing single cell analysis workflows as well as for how to use it as a standalone analysis framework. In conclusion, our results demonstrate that ESCHR is a useful method for single cell analysis, offering robust and reproducible clustering results with the added benefits of per-cell uncertainty scores and soft clustering outputs for improved interpretability. By emphasizing ease of adoption, clustering robustness and accuracy, generalizability across a wide variety of datasets, and improved interpretability through soft clustering outputs and the quantification of uncertainty, we aim to support the responsible and informed use of clustering results in the single cell research community.

# 2.5 Methods

## 2.5.1 ESCHR Framework

ESCHR takes as input a matrix, $M$, with *n* instances (e.g. cells) as rows and *d* features (e.g. genes/proteins) as columns. It does not perform internal normalization or correction, so input data are expected to have already been preprocessed appropriately. ESCHR can be thought of in three primary steps: base clustering to generate the ensemble, consensus determination, and output/visualization.

Consistent with other published manuscripts in this domain, we will use the following notation. Let $X = \{x_1, x_2, ..., x_n\}$ denote a set of objects to be clustered, where each $x_i$ is a tuple of some $d$ -dimensional feature space for all $i = 1 \ldots n$. Let $X_s = \{x_1, x_2, ..., x_r\}$ denote a random subset of $X$ where all of $x_1, ..., x_r$ are between 1 and $n$. $\mathbb{P} = \{P_1, P_2, ..., P_m\}$ is a set of partitions, where each $P_i = \{C_1^i, C_2^i, ..., C_{q_i}^i\}$ is a partition of an independent instantiation of $X_s$ and contains $q_i$ clusters. $C_j^i$ is the $j$ th cluster of the $i$ th partition, for all $i = 1 \ldots m$ . $t = \sum_{i=1}^{m} q_i$ is the total number of clusters from all ensemble members. Where $\mathbb{P}_X$ is the set of all possible partitions with the set of objects $X$ and $\mathbb{P} \subset \mathbb{P}_X$ , the goal of clustering ensemble methods is to find a consensus partition $P^* \in \mathbb{P}_X$ which best represents the properties of each partition in $\mathbb{P}$. Additionally, the more general terminology of “instance” and “feature” will generally be used rather than domain specific terms such as cells and genes/proteins.

### Hyperparameter-randomized ensemble clustering

The ESCHR ensemble is generated with Leiden community detection as the base clustering algorithm [86]. Leiden is applied using Reichardt and Bornholdt’s Potts model with a configuration null model [105]. Diversity is generated amongst ensemble members through a combination of data subsampling and Leiden hyperparameter randomization. The subsampling percentage varies for each ensemble member and is selected from a gaussian distribution with the mean $\mu$ scaled to dataset size within the range 30 to 90. After subsampling a random subset $X_s$ from $X$, principal components analysis (PCA) is applied to generate the most informative

features for this data subspace. A default value of 30 or one less than the number of features if the number of features is less than 30 is used for the number of PCs.. In the subsequent clustering step, three numerical hyperparameters are randomized for each ensemble member: 1) $k$, the number of neighbors for building a k-nearest neighbors (kNN) graph, 2) the choice of distance metric for building the kNN graph, and 3) $r$, a resolution parameter for the modularity optimization function used in Leiden community detection. The numerical hyperparameters $k$ and $r$ are randomly selected from within empirically established ranges (Supplementary Fig. 2). The distance metric is selected between either euclidean or cosine, because these choices are efficiently implemented for fast calculation of approximate nearest neighbors (ANN) in our chosen implementation, nmslib [103]. Since each ensemble member is independent, we implemented parallelization via multiprocessing for this stage of the algorithm. Ensemble size is set at a default of 150 based on experiments demonstrating that this was sufficient to reach convergence to a stable solution for all of our diverse collection of datasets (Supplementary Fig. 2).

### Bipartite graph clustering and consensus determination

Bipartite graph clustering was used to obtain consensus clusters from the ESCHR ensemble. This approach was selected because methods that compute consensus using unipartite projection graphs of either instance or cluster pairwise relations suffer from information loss [89]. For these calculations, the biadjacency matrix is defined as: $B = \begin{bmatrix} 0 & A^T \\ A & 0 \end{bmatrix}$ where $A$ is an $n \times t$ connectivity matrix whose rows correspond to instances {1 . . . n} and columns correspond to the ensemble clusters {1 . . . t}. $A_{i,j}$ is an indicator that takes value 1 if instance $i$ belongs to the $j$ -th cluster and 0 otherwise. Using this, we then create a bipartite graph $G = (V, W)$. The weights matrix $W = B$, and $V = V_1 \cup V_2$, where $V_1$ contains $n$ vertices each representing an instance of the data set $X$; $V_2$ contains $t$ vertices each representing a cluster of the ensemble (see Fig. 1a “Ensemble bipartite graph”). Given our bipartite graph $G$, we can define a community structure on $G$ as a partition $P_1 = \left\{C_1, C_2, ..., C_{k_1}\right\}$ containing pairwise disjoint subsets of $V_1$ and $P_2 = \left\{D_1, D_2, ..., D_{k_2}\right\}$ containing pairwise disjoint subsets of $V_2$, such that all $V_I$ nodes in a specific $C_i$ are more connected to a particular subset of $V_2$ than the rest of the nodes in $V_I$ are, and likewise (but opposite) for a given $D_j$ of $V_2$. Optimal $P_1$ and $P_2$ are computed with the Leiden

algorithm for bipartite community detection with the Constant Potts Model quality function [88, 106]. This approach was designed to overcome the resolution limit of previous bipartite community detection approaches [107, 108]. There is one hyperparameter for this approach, the resolution $\gamma$, which indirectly influences the number of clusters for P1 and P2 by modulating the density of connections within and between communities [106]. To avoid the need for external hyperparameter tuning, we implemented an internal hyperparameter selection strategy at this stage. First, ESCHR generates a set of potential consensus labelings across an internally-specified range of $\gamma$ values. Since ARI can be used as a similarity measure between two different clustering results, ESCHR then calculates the pairwise ARI between each of the final consensus labelings generated using each different $\gamma$ value. Finally, ESCHR selects the result that has the highest sum of similarity to all other results from the set of potential consensus labelings (the medoid) to return as the final consensus result. In experiments to validate this approach, we found that the number of and memberships in the final consensus hard clusters are robust to the setting of this resolution parameter, indicating that more extensive optimization is not required (Supplementary Fig. 4d-e). To obtain the final consensus result, we collapse the base ensemble clusters contained in $V_2$ into the $P_2$ meta-cluster to which they were assigned. This results in each vertex of $V_1$ having a weighted edge to each meta-cluster equal to the sum of its edges with constituent base clusters of $V_2$. The resulting weighted bipartite graph $G^*$ therefore represents the final consensus clustering $P^*$, with $n$ vertices representing the instances, $q^*$ vertices representing the final consensus clusters, and weighted edges representing the membership of instance $i$ in each of the $q^*$ clusters of $P^*$.

### Hard and soft clustering outputs

Let $\Theta \in \mathbb{R}\, n \times q^*$ be a nonnegative matrix where each row, $\Theta_i := (\Theta_{i1}, \ldots, \Theta_{ik_2})$, contains nonnegative numbers that sum to less than or equal to one, representing the membership of instance $i$ in each of the $q^*$ clusters of $P^*$. $\Theta_{ij}$ is calculated by dividing the weight of the edge between instance $i$ and consensus cluster $D_j$ by the sum of all edge weights for instance $i$. We refer to this matrix as the soft membership matrix and to each row as the association vector $v$ for each instance. To determine hard clustering assignments, each instance is assigned to the meta-cluster with the highest entry in its association vector $v$, with ties broken randomly. A "core" cell of a given cluster $j$ will have $\Theta_{ij} = 1$ and zeros elsewhere, while a "transitional"

instance may have up to $q^*$ non-zero membership values. To describe the degree to which a given instance is “core” versus” transitional”, we define an “uncertainty score”, $\Omega$, for each instance as the highest membership value in its association vector ( $\Omega = max(v)$). We can additionally calculate the mean of all instance memberships in a given cluster to yield a measure of each cluster’s discreteness, which we call the “cluster stability score” $s = \frac{1}{n}\Sigma_{i\epsilon n}\theta_{i,j}$ .

### 2.5.2 Cluster connectivity mapping

To map the connectivity structure of clusters, we first calculate the sum-of-squares-and-cross-products matrix (SSCP) of the soft membership matrix $\Theta$, which is calculated as $S = \Theta'\Theta$ and then consider $S_{i,j}$ to be an uncorrected measure for connectivity between consensus clusters $i$ and $j$. To correct for connectivity that may result from random chance, we first estimate a null distribution of connectivity scores accounting for the following attributes of $\Theta$: (1) the association vector $v$ for a given instance are proportions and can sum to no more than 1 (with cells summing to less than one being potentially outliers) and (2) the distribution of values is not uniformly distributed and will be differently skewed for different datasets depending on overall levels of continuity or discreteness. In practice, we achieve this by independently shuffling the association vector, $v$ , for each instance to generate a random sample. We then calculate the SSCP for 500 iterations of this randomization procedure. Using this empirical null distribution, we then calculate a p-value for each observed edge and prune edges that do not meet a default alpha value cutoff of 0.05. Thus the final corrected connectivity is defined as the ratio of the cross-product of instance memberships between a given two clusters normalized to the cross-product of instance memberships expected under constrained randomization.

### 2.5.3 Exploring soft cluster continuity for the MNIST dataset

To visualize the transition between clusters 1a and 1b (Fig. 4f) that was identified by connectivity mapping of ESCHR clustering results, we devised a simple approach for creating a one-dimensional ordering of the instances in a transitional zone based on their membership in the connected clusters of interest. Specifically, the ordering score, $\delta_i$, of cell $i$ having $m_j$ membership in the cluster at position $j$ along the cluster path of interest was calculated as: $\delta_i = \Sigma_{j=1}^{N} m_j \cdot j$ , where $N$ is the number of clusters in the path. To obtain the relevant cells of

interest for ordering, we used the following criteria: cells were included if they had (1) >90% membership in either one of the 2 clusters of interest or (2) >5% membership in both clusters and a combined membership of >80% in the 2 clusters. We then visualized the progression of cluster memberships using a stacked bar plot of rank ordered data points. The ordered data points were then partitioned into “core” datapoints and “transitional” datapoints for each cluster based on bimodality observed in the ordering scores for the cells assigned to each hard cluster.

While a linear ordering approach could in principle be used to create an ordering across a path of more than two connected clusters, it would likely only be effective in cases where connectivity mapping identifies a linear path of successively connected clusters. In cases such as the example in Figure 4 where connectivity mapping identified a ring of 3 connected clusters (9a, 9b, 4), this approach will generally not work as well since a group of more than two clusters with nonlinear connectivity may exhibit more complex continuity structures than could be captured with a simple linear ordering. We therefore devised another method for distilling the core continuity structure for cases of greater than two clusters and nonlinear connectivity paths. We first performed principal components analysis (PCA) on the columns of the soft membership matrix $\Theta$ that correspond to the hard clusters selected for analysis, thereby capturing the primary axes of variation contained within these soft memberships. We then projected the data onto the first two PCs and used this to gain insight into the continuity structure by (1) visualizing the data points belonging to the relevant hard clusters projected into the space of these first two PCs and (2) identifying and exploring the features most highly correlated with these PCs.

### 2.5.4 Exploring soft cluster continuity for the Tany-Seq dataset

To visualize transitional zones between connected clusters in the Tany-Seq dataset [77], we applied the same one-dimensional linear ordering approach that was used to examine clusters 1a and 1b of the MNIST dataset in Figure 4f. To identify marker features associated with the one-dimensional soft cluster transition paths, we calculated the Pearson correlation between each feature and the vector of cluster memberships for each cluster in the path. Features were then selected based on their correlation with each of the clusters individually and based on the sum of their correlations across the clusters. The three genes in Figure 5h-j were selected from the top ten features identified through each of these methods based on their expression patterns and the availability of in situ hybridization images of sufficient quality in the Allen Mouse Brain Atlas [100]. To handle outliers for the expression heatmap UMAP plots and the expression scatterplots in Figure 5h-j, values were thresholded to fall between the 2nd percentile and 98th

percentile. The curves overlaid on the expression scatter plots in these panels were generated by first fitting B-splines with degree 3 (cubic) to the points included in the scatterplot. To generate a smoothed curve, a gaussian kernel with sigma of 10 was applied on the results of the spline function evaluated at 100 evenly spaced points within the range of the number of points included in the scatter plot. This is approximately equivalent to the behavior for large data sizes of the `geom_smooth` function from the R package ggplot [109].

### 2.5.5 Clustering evaluation metrics

Extrinsic evaluation metrics measure the distance of a clustering result to some external set of labels. When these labels are ground truth class labels, we can consider these to be measures of accuracy. However, they can also be used in other contexts such as with an “external” set of clustering labels. There are numerous metrics that can be used to measure this distance between a given set of predicted cluster labels and a ground truth or other set of external labels. Each of these metrics introduces some type of bias in evaluating the accuracy and robustness of clustering results, as discussed further below. To diversify these biases, we selected 2 metrics from different categories: Adjusted Rand Index (ARI) from the category of methods that employ peer-to-peer correlation and Adjusted Mutual Information (AMI) from the information theoretic measures [94]. We use both ARI and AMI to evaluate accuracy and robustness in our systematic benchmarking in Figure 3 and in Supplementary Figures 2 and 7.

The ARI is the corrected-for-chance version of the Rand index, which measures the agreement between two sets of partition labels $U$ and $V$ [110]. The ARI is defined as:

$$ARI = \frac{\binom{n}{2}(a+d) - [(a+b)(a+c)+(c+d)(b+d)]}{\binom{n}{2}^2 - [(a+b)(a+c)+(c+d)(b+d)]}$$

where $a$ is the number of pairs of two objects in the same group in both $U$ and $V$; $b$ is the number of pairs of two objects in different groups in both $U$ and $V$; $c$ is the number of pairs of two objects in the same group in $U$ but in different groups in $V$; and $d$ is the number of pairs of two objects in different groups in $U$ but in the same group in $V$. Random clusterings have an expected score of zero and identical partitions have a score of 1. ARI is biased towards solutions containing (1) balanced clusters (i.e. similar size clusters within each partition) and (2) similar cluster numbers and sizes between the two partitions [52]. ARI was calculated using the implementation in sklearn (v 1.0.1).

AMI is the corrected-for-chance version of Mutual Information, which quantifies the amount of information that can be obtained about one random variable (in this application, a list of cluster labels) by observing the other random variable (another list of cluster labels) [111]. Let $C = \{C_1, C_2, ..., C_{tc}\}$ and $G = \{G_1, G_2, ..., G_{tg}\}$ be the predicted and ground truth labels on a dataset with n cells. AMI is then defined as:

$$AMI(C, G) = \frac{I(C,G) - E\{I(C,G)\}}{max\{H(C),H(G)\} - E\{I(C,G)\}}$$

Here $I(C, G)$ represents the mutual information between $C$ and $G$ and is defined as:

$$I(C, G) = \sum_{p=1}^{tc} \sum_{q=1}^{tg} \left|C_p \cap G_q\right| log \frac{n\left|C_p \cap G_q\right|}{\left|C_p\right| \times \left|G_q\right|}$$

. $H(C)$ and $H(G)$ are the entropies: $H(C) = -\sum_{p=1}^{tc} \left|C_p\right| log \frac{\left|C_p\right|}{n}$ and $H(G) = -\sum_{p=1}^{tg} \left|G_p\right| log \frac{\left|G_p\right|}{n}$. $E\{I(C, G)\}$ is the expected mutual information between two random clusters. Random clusterings have an expected score of zero and identical partitions have a score of 1. AMI is biased towards solutions containing pure clusters, with a “pure cluster” being defined as a cluster in one set of labels that contains instances from only one cluster of the other set of labels to which it is being compared [52]. AMI was calculated using the implementation in sklearn (v 1.0.1).

### 2.5.6 Systematic Benchmarking

For benchmarking ESCHR, we selected the following clustering algorithms for comparison: (1&2) K-means and agglomerative hierarchical clustering (from scikit-learn version 1.0.1) [112], (3) SC3 (version 1.10.1 from Bioconductor) [62], (4) SC3s (version 0.1.1 through Scanpy) [63], (5) Seurat (version 4.1.1 from CRAN) [113], (6) SINCERA (version 1.0 from https://github.com/xu-lab/SINCERA) [114], (7) Scanpy (version 1.8.2 from Anaconda) [71], (8) Phenograph (version 1.5.7) [101], (9) scCAN (version 1.0 from https://github.com/bangtran365/scCAN) [115], and (10) scAIDE (version 1.0 https://github.com/tinglabs/scAIDE) [116].

Clustering algorithms were excluded from our benchmarking comparison if they did not meet the following selection criteria: (1) software freely available; (2) code publicly available; (3) can run on multiple data modalities (e.g. not scRNA-seq-specific); (4) no unresolved errors during install or implementation; (5) does not require additional user input during the algorithm (other than

prior information); and (6) able to complete analysis of datasets with >= 100,000 data points and 2,000 features.

For included methods, we followed the instructions and tutorials provided by the authors of each software package. For the K-means, SC3s, and Agglomerative methods, which require pre-specification of cluster number, we calculated distortion scores over a range of cluster numbers for K-means clustering and used the elbow method to select the optimal cluster number for use across all three methods. Default values were used for all other hyperparameters for each tool, as is common practice for most realistic use cases [35, 117]. ESCHR was also run with all default settings, which is the intended usage. For all benchmarking analyses, the memory was set to 100GB of RAM on the University of Virginia (UVA) Rivanna High Performance Computing (HPC) cluster.

No random seeds were intentionally fixed, but from inspecting the respective codebases we believe it is likely that there remained internally-fixed random seeds for some functions within some of the tested methods. Many common methods have internally-fixed random seeds and/or default hyperparameters with fixed random seeds. This practice may mask a lack of robustness of these methods, and should only legitimately serve to replicate exact analyses when that is desired by the end user.

To assess accuracy of methods in clustering our synthetic and image datasets, which have ground truth labels, we used the two extrinsic evaluation metrics defined above (ARI, AMI). For these purposes, each of five independent runs of a given method was scored against the ground truth labels. Since it is nearly universal in papers describing new single cell analysis methods, we also applied this analysis to evaluate accuracy for our collection of “real” datasets using available published labels. However we stress that we do not think this is a reliable or effective measure for evaluating clustering methods, as we detail further in Supplementary Note 1.

We also used the extrinsic metrics ARI and AMI to evaluate the stability and reproducibility of hard clustering results. In line with standard practice for benchmarking stability/robustness [118], we performed repeated runs with 5 random subsamples (90%) of each dataset for every method. This simulates slight differences in data collection and/or preprocessing, and if the clustering is capturing a true underlying structure rather than overfitting to noise it should be

detected regardless of the exact set of cells that are sampled for the analysis. We then calculated pairwise scores for each metric between each of the 5 independent runs of a respective method and then took the mean across replicate pairs to obtain the final score per dataset-method.

To calculate both method and replicate “agreement scores” for comparison with ESCHR uncertainty scores (Fig. 2j), we first constructed contingency matrices between all pairs of replicates and methods and mapped the cluster labels from the result with more clusters to the result with fewer clusters. Using the shared labels between a given pair of clustering results we could then calculate per-instance agreement (binary) within the pair of results. The final per-instance score was calculated as the mean agreement across all possible combinations.

We similarly calculated the F-measure per true cluster per method (Figure 2 c,g) by constructing a contingency matrix between the true cluster labels and the hard cluster labels predicted by each method. Using this mapping we then calculated the harmonic mean of precision and recall separately for each true cluster and its best-matched predicted cluster.

### 2.5.7 Statistical analyses

Statistical comparisons were performed using the “scipy.stats” and “statannotations” python packages [119, 120]. The two-sided Wilcoxon signed-rank test with Bonferroni correction was used to compare the performance of ESCHR versus each alternative method in the systematic benchmarking panels shown in Figure 3. Comparisons were calculated using dataset means across replicates for all tested datasets. N = 21 for comparisons using synthetic datasets and N = 45 for comparisons using real datasets. The two-sided Mann-Whitney-Wilcoxon was used for comparing uncertainty scores between accurately and inaccurately assigned cells in Figure 2g, with N = 126545 and N = 750955 for inaccurate and accurate groups respectively. The resulting p-value was below the threshold of calculation in a standard python computing environment and was reported as zero, so we have reported this as $p<0.00001$ in the figure.

### 2.5.8 Datasets

All “real” datasets are publicly available. The 45 datasets used in our study are described in Supplementary Table 2, including information and links to download preprocessed datasets. The only processing step we performed was log2 transformation for scRNA-seq datasets and

arcsinH transformation for mass cytometry datasets if a given dataset was not yet scaled. The MNIST dataset was downloaded from keras datasets [121] and was preprocessed as recommended by the accompanying documentation.

Synthetic gaussian datasets were generated using sklearn “make_blobs” [112] using various combinations of object number, feature number, cluster number, cluster size, and cluster standard deviations. Anisotropic transformations were then applied to the resulting datasets by multiplying pairs of features (an n x 2 subset of the full data matrix) by different 2 x 2 matrices filled with random values between -2 and 2. These datasets are available at: https://doi.org/10.5281/zenodo.12746558.

Simulated scRNA-seq datasets with 1000 “cells” and 1000 “genes" were generated using the DynToy package, which simulates different complex trajectory models based on real single cell gene expression data [93]. These datasets are available at: https://doi.org/10.5281/zenodo.12786322

ISH images were downloaded from the ABA portal (http://mouse.brain-map.org) [122] and are freely available.

### 2.5.9 Software

The ESCHR python package can be downloaded from PyPi (TBD), github (https://github.com/zunderlab/eschr). The version used for making the figures in this manuscript is noted in the documentation and can be found on zenodo (link TBD). ESCHR is compatible with standard python single cell data structures and can be easily incorporated into scverse workflows or used as a standalone framework.

# Chapter 3: Concluding remarks and future directions

## 3.1 Overview and Key Contributions

Clustering analysis is widely used to group objects by similarity, but for complex datasets such as those produced by single-cell analysis, the currently available methods are limited by accuracy, robustness, ease of use, and interpretability. ESCHR was specifically designed to address these core challenges, and our results demonstrate significant advances in several key areas.

First, ESCHR provides a novel solution to one of the most persistent challenges in single cell analysis: the tension between discrete and continuous perspectives of cellular organization [54]. Traditional hard clustering methods assume the presence of discrete groups, which are rare in biological data [26]. Additionally, many clustering approaches impose further assumptions about the shapes, sizes, or other properties of these discrete groups. To complement these discrete perspectives, significant effort has been dedicated to developing methods that capture continuity in tissues or cells. However the focus has been almost entirely on contexts where cells are undergoing dynamic, directional changes, such as during development or disease. As such, these methods often make assumptions about the structure of continuity based on the expectation of an underlying dynamic process (e.g. branching structures). Meanwhile, the pervasive continuity within ostensibly "static" tissues has been largely neglected, even though understanding this continuity may be crucial for deciphering the function of tissues or their constituent cell types and states [26]. ESCHR's weighted bipartite graph representation of the grouping structure in a dataset enables simultaneous mapping of both discrete and continuous structures, and importantly, this unified approach achieves its results without making assumptions about the shapes or properties of areas of continuity or discreteness. This capability was demonstrated in our analysis of tanycyte cells, where ESCHR successfully identified the four canonical discrete subtypes while simultaneously revealing substantial continuity both within and between these populations (Chapter 2, Figure 5). This balanced perspective provides a more nuanced and biologically relevant view of cellular organization than traditional approaches that favor either discrete or continuous interpretations.

Another significant advancement lies in ESCHR's uncertainty quantification. The fundamental limitations of clustering are well established, as demonstrated by Kleinberg's Impossibility Theorem which proves that no clustering algorithm can simultaneously satisfy all desirable clustering properties [55]. This inherent limitation makes it crucial to understand and communicate the uncertainty in clustering results before using them for downstream analyses or

hypothesis generation. ESCHR addresses this challenge through its provision of per-data-point uncertainty scores, which we have rigorously validated through multiple approaches. We have validated these uncertainty scores by demonstrating that they are significantly higher for inaccurately assigned data points (Chapter 2, Figure 2g) and that they are strongly positively correlated with the level of disagreement between the results from different methods (Chapter 2, Figure 2h). These findings confirm that ESCHR's uncertainty scores provide meaningful insights into clustering uncertainty, enabling researchers to make more informed decisions in their analyses.

The challenge of method selection and parameter tuning has also been effectively addressed through ESCHR's ensemble hyperparameter randomization approach. Single cell datasets vary widely in their characteristics, and no single parameter setting or method can be optimal across all cases [35, 36]; [37]; [38]. ESCHR's hyperparameter-randomized ensemble approach eliminates the need for manual parameter tuning while providing robust results across diverse datasets. This was demonstrated through our comprehensive evaluation across a large collection of datasets representing various modalities, sizes, tissues, species, and intrinsic structures, where ESCHR consistently provided accurate and robust results regardless of the dataset's specific characteristics (Chapter 2, Figure 3).

Finally, ESCHR contributes new means for interpretation and visualization of clustering results. While stochastic embedding methods like UMAP [91] and t-SNE [92] are widely used for visualizing single cell data, they can be misleading if not properly interpreted [90]. It is common practice to color a UMAP or t-SNE 2D layout by cluster labels or other metadata. Visualizing ESCHR's uncertainty scores alongside hard cluster labels can improve transparency and interpretability, allowing researchers to better understand the underlying structure of their data and the reliability of the clustering results and how they map to the 2D layout. Additionally, soft cluster membership heatmaps offer a completely separate alternative visualization method, effectively capturing the complex relational structures within single cell datasets without the risk of overinterpreting spurious patterns that can arise from 2D layouts.

By addressing these fundamental challenges while maintaining compatibility with existing workflows through integration with the scverse ecosystem [69], ESCHR represents a significant step forward in single cell analysis methodology. Its success in providing robust, interpretable results while explicitly acknowledging and quantifying uncertainty sets a new standard for responsible and informed utilization of clustering in single cell research.

## 3.2 Technical Improvements and Software Development

### 3.2.1 Alternative ensemble configurations

Our study has undertaken a rigorous evaluation of diverse alternatives for multiple aspects of the ESCHR algorithmic framework, but we recognize that there is potential for further methodological enhancement. For instance, we currently base our ensemble Leiden clusterings on k-nearest neighbor (kNN) graphs, but mutual nearest neighbor (mNN) or shared nearest neighbor (sNN) have demonstrated good results in other frameworks and may produce improved results [101] [102] [47]. ESCHR may also benefit from expanding the set of distance metrics utilized. We currently restrict our analysis to euclidean and cosine distances due to their efficient implementations within our chosen fast approximate nearest neighbor (ANN) package [103]. However, recent research has demonstrated the efficacy of a broader range of distance metrics for capturing diverse data structural properties [104]. While not all of these metrics may be applicable in an ANN context, several may hold potential for enhancing the quality of our clustering outcomes. Additionally, this published version of ESCHR uses only Leiden community detection for clustering in the ensemble stage, but additional base clustering methods could be explored and potentially incorporated in future versions. Finally, our empirical identification of optimal ranges for ESCHR's numeric hyperparameters was somewhat limited by the time and memory required for running these experiments with many, sometimes large, datasets and very wide search spaces. It is therefore possible that there may be more optimal default ranges or more sophisticated regimes for hyperparameter randomization and selection that could improve ESCHR's performance. Further exploration of the effects of these and other hyperparameters is an exciting avenue for future development.

### 3.2.2 Improving runtime and memory performance

There are several promising avenues for performance optimization that could further enhance the computational efficiency and scalability of the ESCHR method. The current implementation already incorporates important optimizations, including the use of Zarr [123] arrays for efficient out-of-core dataset storage and multiprocessing for parallel execution of base clusterings, but additional enhancements could further improve performance.

A primary focus for optimization is the implementation of GPU acceleration using CUDA through the RAPIDS ecosystem [124]. The computation of k-nearest neighbor graphs, which forms a

fundamental step in ESCHR's workflow, is particularly well-suited for GPU parallelization. By leveraging cuGraph and cuML from the RAPIDS suite, we could significantly accelerate both the KNN graph construction and the subsequent community detection operations. This would be especially impactful for larger datasets where these computations currently represent substantial bottlenecks. Preliminary analysis suggests that GPU acceleration could potentially reduce computation time by an order of magnitude for these steps.

Building upon our existing use of Zarr arrays for dataset storage, we propose extending this efficient storage strategy to the ensemble results themselves. Currently, while the input data is efficiently managed through Zarr and the base clustering computations are parallelized through multiprocessing, the ensemble results require significant memory during the consensus calculation phase. By implementing Zarr storage for the ensemble results, we could reduce peak memory usage during consensus calculation.

These optimizations would be implemented with careful consideration of maintaining ESCHR's current accuracy and robustness while improving its computational efficiency. All optimizations would be thoroughly validated against our existing benchmark datasets to ensure that performance improvements do not come at the cost of quality.

## 3.3 Extensions and Future Applications

### 3.3.1 Mining soft clustering patterns for biological insights

While ESCHR's soft clustering output currently provides valuable information about regions of continuity between clusters, there are opportunities to develop more sophisticated analyses of these patterns to gain deeper biological insights. The weighted relationships between cells and clusters captured in our soft clustering results contain rich information about both fine-grained population structure and patterns of molecular variation that has yet to be fully exploited.

By analyzing patterns of shared cluster membership across cells, we could potentially identify substructures within regions of continuity that aren't apparent in the hard clustering results. This could help distinguish between different types of continuity – for example, differentiating between gradual phenotypic drift versus distinct intermediate states that bridge major cell types. For instance, cells that share similar patterns of partial membership across multiple clusters might represent distinct intermediate states or transition points. These patterns could be

particularly informative when correlated back with the features of the data (e.g. gene expression), potentially revealing signatures associated with specific patterns of cluster membership.

We could also take a “features first” approach and directly map patterns of molecular phenotypic changes across regions of continuity. This would enable us to identify modules of molecular features that exhibit similar patterns across the phenotypic space, such as those that vary smoothly across a given continuous region versus those that show more discrete transitions. This approach could reveal the specific molecular programs that underlie cellular heterogeneity, helping to distinguish between coordinated program changes and more stochastic variation in gene expression. Such analysis could be particularly valuable for understanding how cells navigate phenotypic spaces and for identifying potential regulatory mechanisms that maintain or modify cell states.

While we have applied some of these analytical approaches in a targeted manner to specific examples in our study, a key challenge moving forward is to develop robust, computationally efficient implementations that can be applied across diverse datasets. By formalizing these analyses and incorporating them directly into the ESCHR framework, we aim to make these sophisticated interpretative tools readily available to the broader research community.

## 3.4 Final Conclusions

Looking forward, the extensions and improvements outlined in this chapter will further enhance ESCHR's utility for the single cell research community. From performance optimizations that will enable analysis of larger datasets to advanced interpretability tools that will provide deeper biological insights, these developments aim to ensure that ESCHR continues to evolve alongside the rapidly advancing field of single cell analysis.

By emphasizing generalizability, transparency about uncertainty, and ease of adoption while providing novel insights into cellular population structures, ESCHR advances not only the technical capabilities of the field but also our conceptual framework for understanding cellular heterogeneity. As single cell technologies continue to evolve and generate increasingly complex datasets, such robust and interpretable methods will be essential for translating this wealth of data into biological understanding.

# References

1. Mazzarello P. A unifying concept: the history of cell theory. Nat Cell Biol. 1999;1:E13-5.

2. Karr JR, Sanghvi JC, Macklin DN, Gutschow MV, Jacobs JM, Bolival B, et al. A whole-cell computational model predicts phenotype from genotype. Cell. 2012;150:389–401.

3. Kelbauskas L, Ashili S, Zeng J, Rezaie A, Lee K, Derkach D, et al. Platform for combined analysis of functional and biomolecular phenotypes of the same cell. Sci Rep. 2017;7:44636.

4. Altschuler SJ, Wu LF. Cellular heterogeneity: do differences make a difference? Cell. 2010;141:559–63.

5. Tang F, Barbacioru C, Wang Y, Nordman E, Lee C, Xu N, et al. mRNA-Seq whole-transcriptome analysis of a single cell. Nat Methods. 2009;6:377–82.

6. Ramsköld D, Luo S, Wang Y-C, Li R, Deng Q, Faridani OR, et al. Full-length mRNA-Seq from single-cell levels of RNA and individual circulating tumor cells. Nat Biotechnol. 2012;30:777–82.

7. Picelli S, Björklund ÅK, Faridani OR, Sagasser S, Winberg G, Sandberg R. Smart-seq2 for sensitive full-length transcriptome profiling in single cells. Nat Methods. 2013;10:1096–8.

8. Klein AM, Mazutis L, Akartuna I, Tallapragada N, Veres A, Li V, et al. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. Cell. 2015;161:1187–201.

9. Macosko EZ, Basu A, Satija R, Nemesh J, Shekhar K, Goldman M, et al. Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets. Cell. 2015;161:1202–14.

10. Buenrostro JD, Wu B, Litzenburger UM, Ruff D, Gonzales ML, Snyder MP, et al. Single-cell chromatin accessibility reveals principles of regulatory variation. Nature. 2015;523:486–90.

11. Karemaker ID, Vermeulen M. Single-Cell DNA Methylation Profiling: Technologies and Biological Applications. Trends Biotechnol. 2018;36:952–65.

12. Evrony GD, Hinch AG, Luo C. Applications of Single-Cell DNA Sequencing. Annu Rev Genomics Hum Genet. 2021;22:171–97.

13. Giaretti W. Origins of ... flow cytometry and applications in oncology. J Clin Pathol. 1997;50:275–7.

14. Bandura DR, Baranov VI, Ornatsky OI, Antonov A, Kinach R, Lou X, et al. Mass cytometry: technique for real time single cell multitarget immunoassay based on inductively coupled plasma time-of-flight mass spectrometry. Anal Chem. 2009;81:6813–22.

15. Wang G, Moffitt JR, Zhuang X. Multiplexed imaging of high-density libraries of RNAs with MERFISH and expansion microscopy. Sci Rep. 2018;8:4847.

16. Eng C-HL, Lawson M, Zhu Q, Dries R, Koulena N, Takei Y, et al. Transcriptome-scale super-resolved imaging in tissues by RNA seqFISH. Nature. 2019;568:235–9.

17. Chang Q, Ornatsky OI, Siddiqui I, Loboda A, Baranov VI, Hedley DW. Imaging Mass Cytometry. Cytometry A. 2017;91:160–9.

18. Stoeckius M, Hafemeister C, Stephenson W, Houck-Loomis B, Chattopadhyay PK,

Swerdlow H, et al. Simultaneous epitope and transcriptome measurement in single cells. Nat Methods. 2017;14:865–8.

19. Wagner A, Regev A, Yosef N. Revealing the vectors of cellular identity with single-cell genomics. Nat Biotechnol. 2016;34:1145–60.

20. Kolodziejczyk AA, Kim JK, Svensson V, Marioni JC, Teichmann SA. The technology and biology of single-cell RNA sequencing. Mol Cell. 2015;58:610–20.

21. Kim JK, Kolodziejczyk AA, Ilicic T, Teichmann SA, Marioni JC. Characterizing noise structure in single-cell RNA-seq distinguishes genuine from technical stochastic allelic expression. Nat Commun. 2015;6:8687.

22. Swain PS, Elowitz MB, Siggia ED. Intrinsic and extrinsic contributions to stochasticity in gene expression. Proc Natl Acad Sci USA. 2002;99:12795–800.

23. Raj A, van Oudenaarden A. Nature, nurture, or chance: stochastic gene expression and its consequences. Cell. 2008;135:216–26.

24. Ilicic T, Kim JK, Kolodziejczyk AA, Bagger FO, McCarthy DJ, Marioni JC, et al. Classification of low quality cells from single-cell RNA-seq data. Genome Biol. 2016;17:29.

25. Brackston RD, Lakatos E, Stumpf MPH. Transition state characteristics during cell differentiation. PLoS Comput Biol. 2018;14:e1006405.

26. Cembrowski MS, Menon V. Continuous Variation within Cell Types of the Nervous System. Trends Neurosci. 2018;41:337–48.

27. Bellman RE. Dynamic Programming. 1957.

28. Lähnemann D, Köster J, Szczurek E, McCarthy DJ, Hicks SC, Robinson MD, et al. Eleven grand challenges in single-cell data science. Genome Biol. 2020;21:31.

29. Stegle O, Teichmann SA, Marioni JC. Computational and analytical challenges in single-cell transcriptomics. Nat Rev Genet. 2015;16:133–45.

30. Heumos L, Schaar AC, Lance C, Litinetskaya A, Drost F, Zappia L, et al. Best practices for single-cell analysis across modalities. Nat Rev Genet. 2023;24:550–72.

31. Yang P, Huang H, Liu C. Feature selection revisited in the single-cell era. Genome Biol. 2021;22:321.

32. Petegrosso R, Li Z, Kuang R. Machine learning and statistical methods for clustering single-cell RNA-sequencing data. Brief Bioinformatics. 2020;21:1209–23.

33. Tirosh I, Suva ML. Cancer cell states: Lessons from ten years of single-cell RNA-sequencing of human tumors. Cancer Cell. 2024;42:1497–506.

34. Papalexi E, Satija R. Single-cell RNA sequencing to explore immune cell heterogeneity. Nat Rev Immunol. 2018;18:35–45.

35. Schneider I, Cepela J, Shetty M, Wang J, Nelson AC, Winterhoff B, et al. Use of “default” parameter settings when analyzing single cell RNA sequencing data using Seurat: a biologist’s perspective. JTGG. 2020.

36. Lu Y, Phillips CA, Langston MA. A robustness metric for biological data clustering algorithms. BMC Bioinformatics. 2019;20 Suppl 15:503.

37. Krzak M, Raykov Y, Boukouvalas A, Cutillo L, Angelini C. Benchmark and Parameter Sensitivity Analysis of Single-Cell RNA Sequencing Clustering Methods. Front Genet. 2019;10:1253.

38. Tang M, Kaymaz Y, Logeman BL, Eichhorn S, Liang ZS, Dulac C, et al. Evaluating single-cell cluster stability using the Jaccard similarity index. Bioinformatics. 2021;37:2212–4.

39. Wang L, Zhang Q, Qin Q, Trasanidis N, Vinyard M, Chen H, et al. Current progress and potential opportunities to infer single-cell developmental trajectory and cell fate. Current Opinion in Systems Biology. 2021;26:1–11.

40. Saelens W, Cannoodt R, Todorov H, Saeys Y. A comparison of single-cell trajectory inference methods. Nat Biotechnol. 2019;37:547–54.

41. Trapnell C, Cacchiarelli D, Grimsby J, Pokharel P, Li S, Morse M, et al. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. Nat Biotechnol. 2014;32:381–6.

42. Bendall SC, Davis KL, Amir E-AD, Tadmor MD, Simonds EF, Chen TJ, et al. Single-cell trajectory detection uncovers progression and regulatory coordination in human B cell development. Cell. 2014;157:714–25.

43. La Manno G, Soldatov R, Zeisel A, Braun E, Hochgerner H, Petukhov V, et al. RNA velocity of single cells. Nature. 2018;560:494–8.

44. Bassez A, Vos H, Van Dyck L, Floris G, Arijs I, Desmedt C, et al. A single-cell map of intratumoral changes during anti-PD1 treatment of patients with breast cancer. Nat Med. 2021;27:820–32.

45. Svensson V, da Veiga Beltrame E, Pachter L. A curated database reveals trends in single-cell transcriptomics. Database (Oxford). 2020;2020.

46. Xie B, Jiang Q, Mora A, Li X. Automatic cell type identification methods for single-cell RNA sequencing. Comput Struct Biotechnol J. 2021;19:5874–87.

47. Kiselev VY, Andrews TS, Hemberg M. Challenges in unsupervised clustering of single-cell RNA-seq data. Nat Rev Genet. 2019;20:273–82.

48. Zappia L, Theis FJ. Over 1000 tools reveal trends in the single-cell RNA-seq analysis landscape. Genome Biol. 2021;22:301.

49. Gibson G. Perspectives on rigor and reproducibility in single cell genomics. PLoS Genet. 2022;18:e1010210.

50. Patterson-Cross RB, Levine AJ, Menon V. Selecting single cell clustering parameter values using subsampling-based robustness metrics. BMC Bioinformatics. 2021;22:39.

51. Renedo-Mirambell M, Arratia A. Identifying bias in network clustering quality metrics. PeerJ Comput Sci. 2023;9:e1523.

52. Amigó E, Gonzalo J, Artiles J, Verdejo F. A comparison of extrinsic clustering evaluation

metrics based on formal constraints. Inf Retr Boston. 2009;12:461–86.

53. Freytag S, Tian L, Lönnstedt I, Ng M, Bahlo M. Comparison of clustering tools in R for medium-sized 10x Genomics single-cell RNA-sequencing data. [version 2; peer review: 3 approved]. F1000Res. 2018;7:1297.

54. Tanay A, Regev A. Scaling single-cell genomics from phenomenology to mechanism. Nature. 2017;541:331–8.

55. Kleinberg J. An Impossibility Theorem for Clustering. Advances in neural information processing systems. 2002;15.

56. Huh R, Yang Y, Jiang Y, Shen Y, Li Y. SAME-clustering: Single-cell Aggregated Clustering via Mixture Model Ensemble. Nucleic Acids Res. 2020;48:86–95.

57. Burton RJ, Cuff SM, Morgan MP, Artemiou A, Eberl M. GeoWaVe: geometric median clustering with weighted voting for ensemble clustering of cytometry data. Bioinformatics. 2023;39.

58. Tsoucas D, Yuan G-C. GiniClust2: a cluster-aware, weighted ensemble clustering method for cell-type detection. Genome Biol. 2018;19:58.

59. Sagi O, Rokach L. Ensemble learning: A survey. WIREs Data Mining Knowl Discov. 2018;8.

60. Wan S, Kim J, Won KJ. SHARP: hyperfast and accurate processing of single-cell RNA-seq data via ensemble random projection. Genome Res. 2020;30:205–13.

61. Risso D, Purvis L, Fletcher RB, Das D, Ngai J, Dudoit S, et al. clusterExperiment and RSEC: A Bioconductor package and framework for clustering of single-cell and other large gene expression datasets. PLoS Comput Biol. 2018;14:e1006378.

62. Kiselev VY, Kirschner K, Schaub MT, Andrews T, Yiu A, Chandra T, et al. SC3: consensus clustering of single-cell RNA-seq data. Nat Methods. 2017;14:483–6.

63. Quah FX, Hemberg M. SC3s: efficient scaling of single cell consensus clustering to millions of cells. BMC Bioinformatics. 2022;23:536.

64. Zhu L, Lei J, Klei L, Devlin B, Roeder K. Semisoft clustering of single-cell data. Proc Natl Acad Sci USA. 2019;116:466–71.

65. Peters G, Crespo F, Lingras P, Weber R. Soft clustering – Fuzzy and rough approaches and their extensions and derivatives. International Journal of Approximate Reasoning. 2013;54:307–22.

66. Kanter I, Dalerba P, Kalisky T. A cluster robustness score for identifying cell subpopulations in single cell gene expression datasets from heterogeneous tissues and tumors. Bioinformatics. 2019;35:962–71.

67. Chen Z, Goldwasser J, Tuckman P, Liu J, Zhang J, Gerstein M. Forest Fire Clustering for single-cell sequencing combines iterative label propagation with parallelized Monte Carlo simulations. Nat Commun. 2022;13:3538.

68. Tasic B, Menon V, Nguyen TN, Kim TK, Jarsky T, Yao Z, et al. Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. Nat Neurosci. 2016;19:335–46.

69. Virshup I, Bredikhin D, Heumos L, Palla G, Sturm G, Gayoso A, et al. The scverse project provides a computational ecosystem for single-cell omics data analysis. Nat Biotechnol. 2023;41:604–6.

70. Stuart T, Butler A, Hoffman P, Hafemeister C, Papalexi E, Mauck WM, et al. Comprehensive Integration of Single-Cell Data. Cell. 2019;177:1888-1902.e21.

71. Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. Genome Biol. 2018;19:15.

72. Guo M, Xu Y. Single-Cell Transcriptome Analysis Using SINCERA Pipeline. Methods Mol Biol. 2018;1751:209–22.

73. Deng L. The MNIST database of handwritten digit images for machine learning research. IEEE Signal Process Mag. 2012;29:141–2.

74. Brunner M, Lopez-Rodriguez D, Messina A, Thorens B, Santoni F, Langlet F. Pseudospatial transcriptional gradient analysis of hypothalamic ependymal cells: towards a new tanycyte classification. BioRxiv. 2023.

75. Campbell JN, Macosko EZ, Fenselau H, Pers TH, Lyubetskaya A, Tenen D, et al. A molecular census of arcuate hypothalamus and median eminence cell types. Nat Neurosci. 2017;20:484–96.

76. Langlet F. Tanycyte gene expression dynamics in the regulation of energy homeostasis. Front Endocrinol (Lausanne). 2019;10:286.

77. Sullivan AI, Potthoff MJ, Flippo KH. Tany-Seq: Integrated Analysis of the Mouse Tanycyte Transcriptome. Cells. 2022;11.

78. Fong H, Kurrasch DM. Developmental and functional relationships between hypothalamic tanycytes and embryonic radial glia. Front Neurosci. 2022;16:1129414.

79. Dietterich TG. Ensemble Methods in Machine Learning. In: Multiple Classifier Systems. Berlin, Heidelberg: Springer Berlin Heidelberg; 2000. p. 1–15.

80. Ghosh J, Acharya A. Cluster ensembles. WIREs Data Mining Knowl Discov. 2011;1:305–15.

81. Strehl A, Ghosh J. Cluster Ensembles - A Knowledge Reuse Framework for Combining Multiple Partitions. JMLR (J Mach Learn Res). 2002;3:583–617.

82. Ben-Hur A, Elisseeff A, Guyon I. A stability based method for discovering structure in clustered data. Pac Symp Biocomput. 2002;:6–17.

83. Monti S, Tamayo P, Mesirov J, Golub T. Consensus Clustering: A Resampling-Based Method for Class Discovery and Visualization of Gene Expression Microarray Data. Mach Learn. 2003;52:91–118.

84. Fred A. Finding consistent clusters in data partitions. In: Kittler J, Roli F, editors. Multiple Classifier Systems. Berlin, Heidelberg: Springer Berlin Heidelberg; 2001. p. 309–18.

85. Naegle KM, Welsch RE, Yaffe MB, White FM, Lauffenburger DA. MCAM: multiple clustering analysis methodology for deriving hypotheses and insights from high-throughput proteomic datasets. PLoS Comput Biol. 2011;7:e1002119.

86. Traag VA, Waltman L, van Eck NJ. From Louvain to Leiden: guaranteeing well-connected communities. Sci Rep. 2019;9:5233.

87. Topchy A, Jain AK, Punch W. Combining multiple weak clusterings. In: Third IEEE International Conference on Data Mining. IEEE Comput. Soc; 2003. p. 331–8.

88. Traag VA. leidenalg Documentation. Release 0102.dev9+gdc8ec1a.d20230927. Section 4.1.2 Bipartite:15–6.

89. Fern XZ, Brodley CE. Solving cluster ensemble problems by bipartite graph partitioning. In: Twenty-first international conference on Machine learning - ICML '04. New York, New York, USA: ACM Press; 2004. p. 36.

90. Kobak D, Berens P. The art of using t-SNE for single-cell transcriptomics. Nat Commun. 2019;10:5416.

91. McInnes L, Healy J, Melville J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv. 2018.

92. van der Maaten L, Hinton G. Visualizing Data using t-SNE. JMLR. 2008;9:2579–605.

93. Cannoodt R, Saelens W, Deconinck L, Saeys Y. Spearheading future omics analyses using dyngen, a multi-modal simulator of single cells. Nat Commun. 2021;12:3942.

94. Palacio-Niño J-O, Berzal F. Evaluation Metrics for Unsupervised Learning Algorithms. arXiv. 2019.

95. Rodríguez EM, Blázquez JL, Pastor FE, Peláez B, Peña P, Peruzzo B, et al. Hypothalamic tanycytes: a key component of brain-endocrine interaction. Int Rev Cytol. 2005;247:89–164.

96. Langlet F. Targeting Tanycytes: Balance between Efficiency and Specificity. Neuroendocrinology. 2020;110:574–81.

97. Yoo S, Kim J, Lyu P, Hoang TV, Ma A, Trinh V, et al. Control of neurogenic competence in mammalian hypothalamic tanycytes. Sci Adv. 2021;7.

98. Chen R, Wu X, Jiang L, Zhang Y. Single-Cell RNA-Seq Reveals Hypothalamic Cell Diversity. Cell Rep. 2017;18:3227–41.

99. Deng G, Morselli LL, Wagner VA, Balapattabi K, Sapouckey SA, Knudtson KL, et al. Single-Nucleus RNA Sequencing of the Hypothalamic Arcuate Nucleus of C57BL/6J Mice After Prolonged Diet-Induced Obesity. Hypertension. 2020;76:589–97.

100. Lein ES, Hawrylycz MJ, Ao N, Ayres M, Bensinger A, Bernard A, et al. Genome-wide atlas of gene expression in the adult mouse brain. Nature. 2007;445:168–76.

101. Levine JH, Simonds EF, Bendall SC, Davis KL, Amir ED, Tadmor MD, et al. Data-Driven Phenotypic Dissection of AML Reveals Progenitor-like Cells that Correlate with Prognosis. Cell. 2015;162:184–97.

102. Xu C, Su Z. Identification of cell types from single-cell transcriptomes using a novel clustering method. Bioinformatics. 2015;31:1974–80.

103. Boytsov L, Naidan B. Engineering Efficient and Effective Non-metric Space Library. In: Brisaboa N, Pedreira O, Zezula P, editors. Similarity search and applications. Berlin, Heidelberg:

Springer Berlin Heidelberg; 2013. p. 280–93.

104. Watson ER, Mora A, Taherian Fard A, Mar JC. How does the structure of data impact cell-cell similarity? Evaluating how structural properties influence the performance of proximity metrics in single cell RNA-seq data. Brief Bioinformatics. 2022;23.

105. Reichardt J, Bornholdt S. Statistical mechanics of community detection. Phys Rev E. 2006;74.

106. Traag VA, Van Dooren P, Nesterov Y. Narrow scope for resolution-limit-free community detection. Phys Rev E. 2011;84.

107. Calderer G, Kuijjer ML. Community Detection in Large-Scale Bipartite Biological Networks. Front Genet. 2021;12:649440.

108. Xu Y, Chen L, Li B, liu W. Density-based modularity for evaluating community structure in bipartite networks. Inf Sci (Ny). 2015;317:278–94.

109. Wickham H. ggplot2: Elegant Graphics for Data Analysis. 2nd edition. Cham: Springer; 2016.

110. Hubert L, Arabie P. Comparing partitions. J of Classification. 1985;2:193–218.

111. Vinh N, Epps J, Bailey J. Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. Journal of Machine Learning Research. 2010;:2837–54.

112. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011;12:2825–30.

113. Butler A, Hoffman P, Smibert P, Papalexi E, Satija R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nat Biotechnol. 2018;36:411–20.

114. Guo M, Wang H, Potter SS, Whitsett JA, Xu Y. SINCERA: A Pipeline for Single-Cell RNA-Seq Profiling Analysis. PLoS Comput Biol. 2015;11:e1004575.

115. Tran B, Tran D, Nguyen H, Ro S, Nguyen T. scCAN: single-cell clustering using autoencoder and network fusion. Sci Rep. 2022;12:10267.

116. Xie K, Huang Y, Zeng F, Liu Z, Chen T. scAIDE: clustering of large-scale single-cell RNA-seq data reveals putative and rare cell types. NAR Genom Bioinform. 2020;2:lqaa082.

117. Rodriguez MZ, Comin CH, Casanova D, Bruno OM, Amancio DR, Costa L da F, et al. Clustering algorithms: A comparative approach. PLoS ONE. 2019;14:e0210236.

118. Weber LM, Saelens W, Cannoodt R, Soneson C, Hapfelmeier A, Gardner PP, et al. Essential guidelines for computational method benchmarking. Genome Biol. 2019;20:125.

119. Jones E, Oliphant T, Peterson P, others. SciPy: Open source scientific tools for Python. Computer software. 2001.

120. Charlier F, Weber M, Izak D, Harkin E, Magnus M, Lalli J, et al. Statannotations: v0.5. Zenodo. 2022.

121. Chollet F, Others. Keras. 2015.

122. Allen Mouse Brain Atlas. https://mouse.brain-map.org/. Accessed 2 Mar 2023.

123. Miles A, Kirkham J, Durant M, Bourbeau J, Onalan T, Hamman J, et al. zarr-developers/zarr-python: v2.4.0. Zenodo. 2020.

124. RAPIDS Development Team. RAPIDS: Libraries for End to End GPU Data Science. Computer software. 2023.

125. Yanai I, Lercher M. A hypothesis is a liability. Genome Biol. 2020;21:231.

126. Tyler SR, Bunyavanich S, Schadt EE. PMD Uncovers Widespread Cell-State Erasure by scRNAseq Batch Correction Methods. BioRxiv. 2021.

127. Zeisel A, Muñoz-Manchado AB, Codeluppi S, Lönnerberg P, La Manno G, Juréus A, et al. Brain structure. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. Science. 2015;347:1138–42.

# Appendix I: Supplementary Materials for “A hyperparameter-randomized ensemble approach for robust clustering across diverse datasets”

# Supplementary Notes

## 1. On the difficulty of identifying a “ground truth” to assess clustering accuracy on single cell datasets

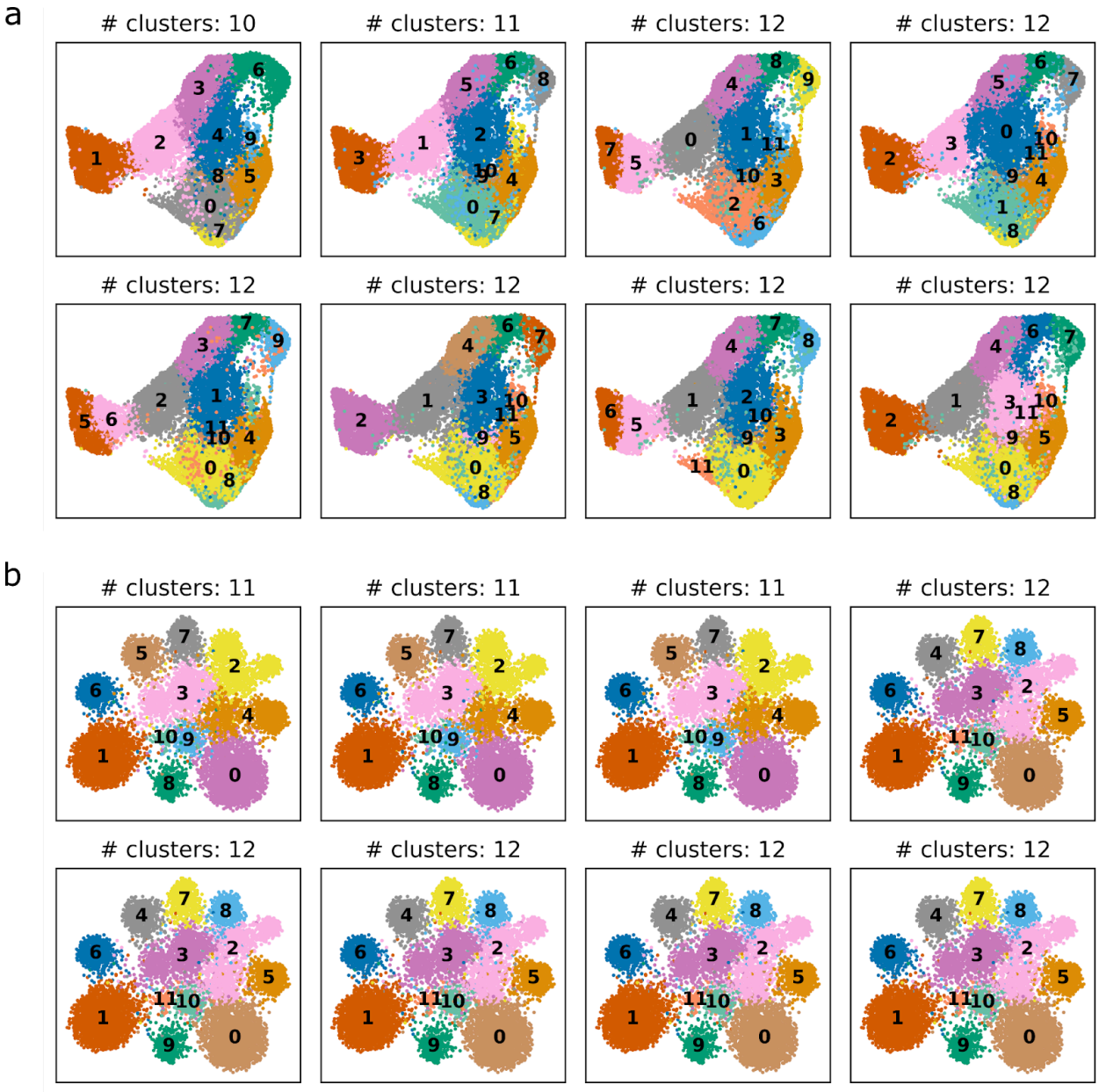
Clustering accuracy can be evaluated with extrinsic validation metrics to compare a set of cluster labels to a set of ground truth class labels. This requires the existence of a priori known ground truth class labels, which generally do not exist for real-world single cell datasets. Nevertheless, clustering accuracy is widely used as a key metric for evaluating single cell clustering methods, and many publications featuring new single cell clustering methods have claimed to measure the accuracy of their clustering results using a variety of different annotations as proxies for ground truth. These “ground truth” annotations are frequently based on either manual annotation and/or previous clustering results, and we are skeptical about the validity of these human-guided ground truth assignments. There are some rare exceptions where ground truth class labels in single cell data are unambiguous and clear cut, such as unique cell lines or drug treatments. But in most cases, we propose that the ground truth annotations used to benchmark clustering accuracy in single cell datasets are inappropriate, and not a useful yardstick to measure the “goodness” of clustering.

Besides cell lines and drug treatments, all other examples of “ground truth” annotations of single cell data that we’re aware of are problematic. Previous clustering results, even those obtained with human-optimized parameter tuning to give a result that looks best the human eye on a tSNE or UMAP plot, are still just one result from a single clustering algorithm, and should not be the yardstick by which all other clustering algorithms are judged. Even expert-guided “ground truth” annotations on a cell-by-cell basis still rely on our imperfect knowledge of functional markers, and rely on human intuition which doesn’t perform well in high-dimensional space. An important additional point which has been made by others is that manual annotation is inherently limited to only capturing already-known biology and will miss novel findings [125] [126].

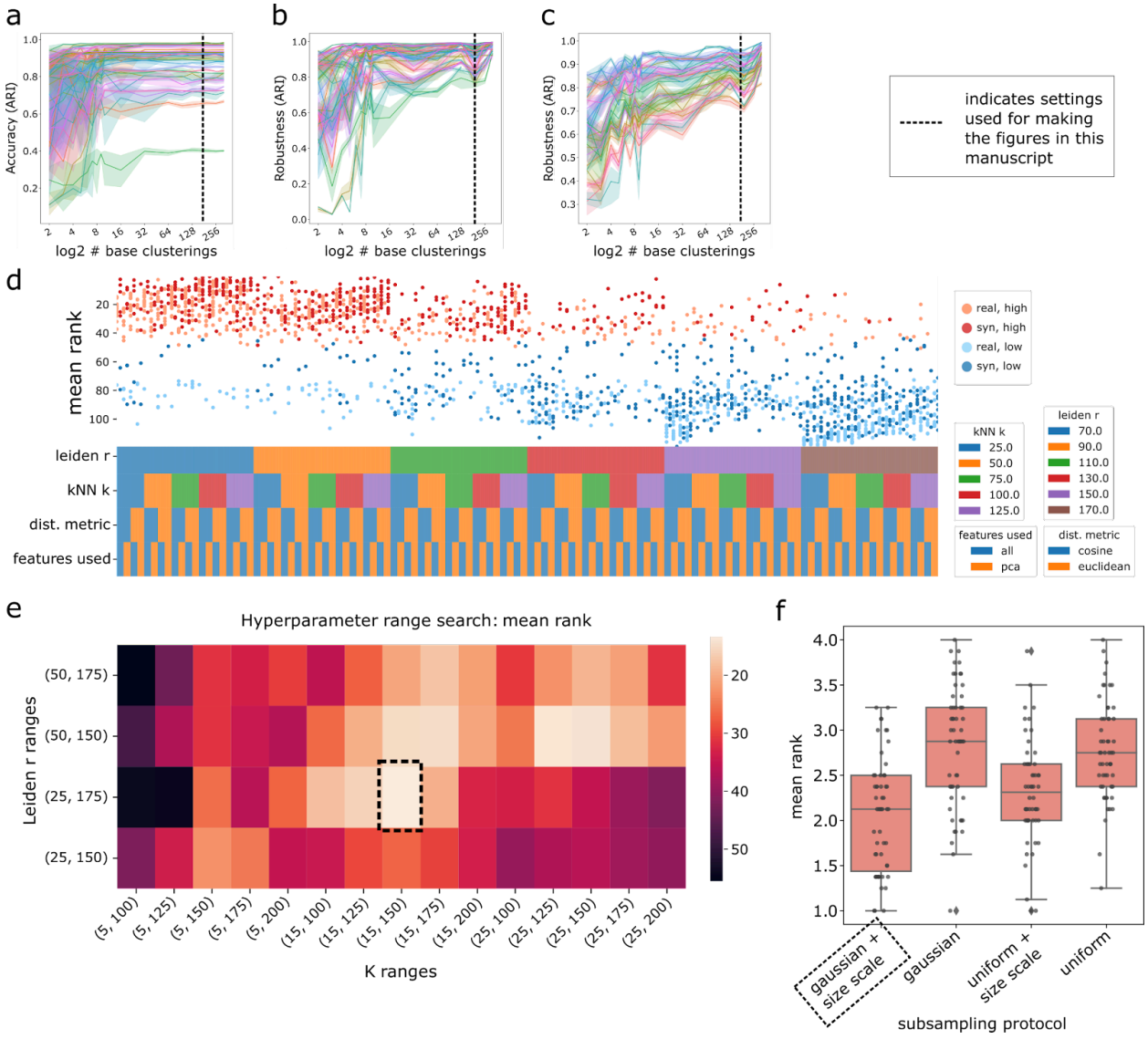
Another annotation that has previously been used as ground truth is patient source of tissue samples, but if these samples include a heterogenous mixture of cells then it is not reasonable to expect that the ground truth that clustering should capture would be differences between patients rather than between cell types shared by all patients. Developmental stage or collection day of differentiating cell types is also commonly considered to be a “gold standard” annotation, but considering that cell types often persist over varying time spans of development or other dynamic biological processes, it is a flawed assumption to believe that clustering should ideally be capturing differences between stages or collection days. Other studies use some versions of “sorting” by an alternate data modality as true labels. A common example of this is using flow cytometry-based sorting using a small set of cell surface proteins to select out different populations of cells and then using these population labels as ground truth for assessing clustering of scRNA-seq of the cells. While this can certainly be useful for a researcher aiming to explore mRNA expression specifically within cells defined by protein markers, there is no reason to believe that the actual true best clustering based on one modality should capture the same

population structure that is defined by another modality, and this is therefore also not a valid basis for assessing the accuracy of a clustering result in the context of evaluating a new method.

# Supplementary Figures

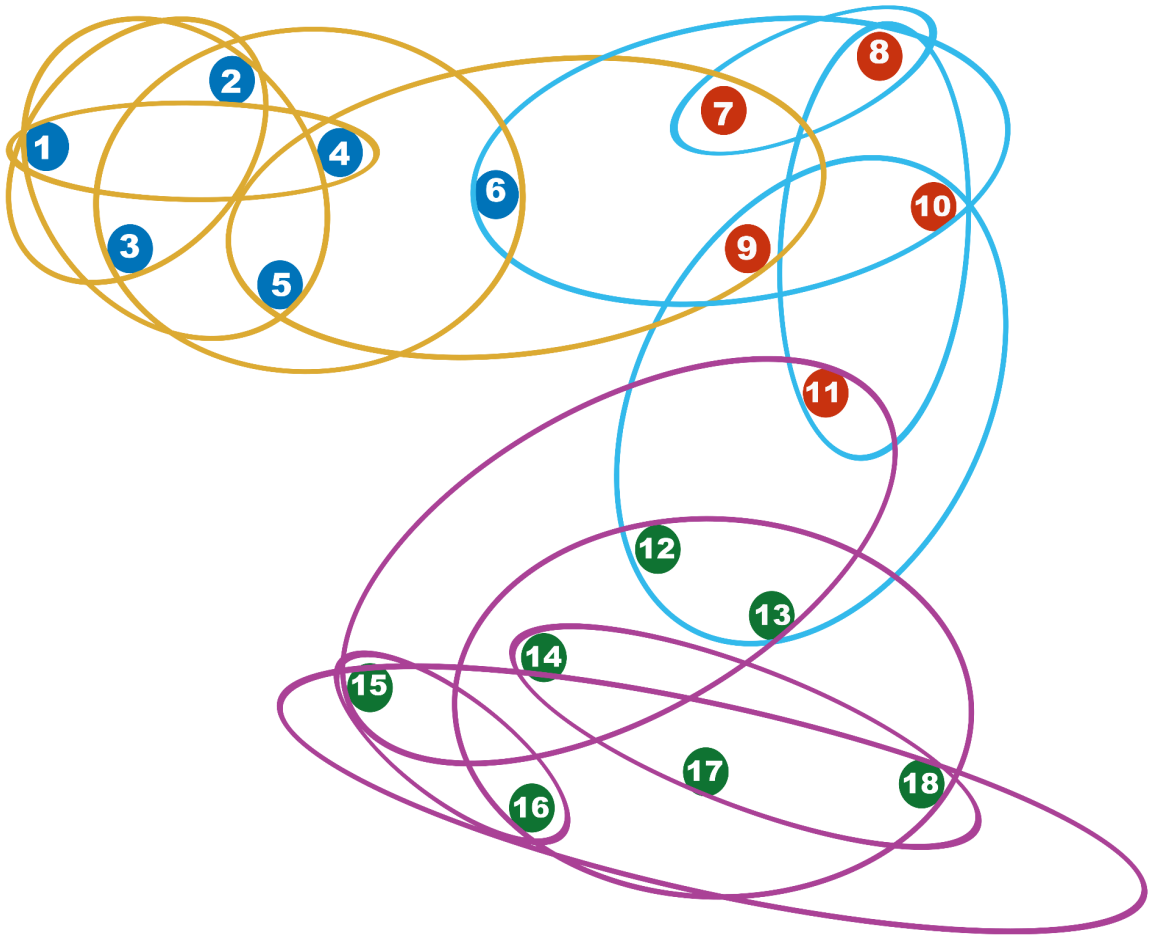



**Supplementary Figure 1: Inherent randomness in the commonly used Leiden algorithm.** UMAP visualizations of 8 replicates of Leiden clustering run with identical hyperparameters on identical input k-NN graphs. (A) shows the Tany-seq scRNA-seq dataset [77] which was explored in Figure 4 and (B) shows this analysis applied to the synthetic gaussian 10 cluster dataset that was used as one of the examples in Figure 2. Points are colored by cluster labels and numerical cluster labels are placed on the centroid of the cluster.

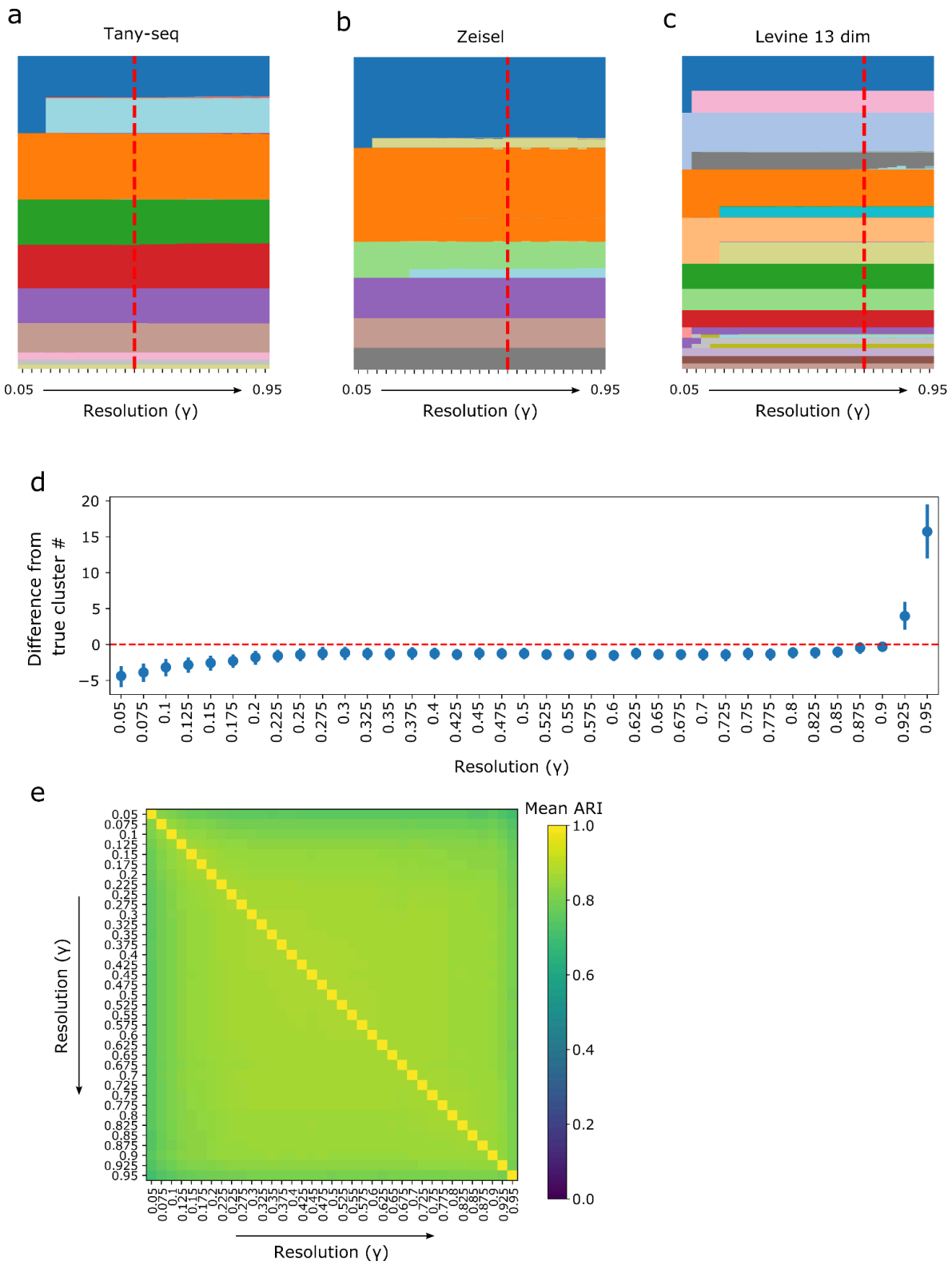



**Supplementary Figure 2: Establishing default settings for ensemble hyperparameter randomization.** (A-C) Clustering quality metrics (y-axis) improve with increasing ensemble size (x-axis), evaluated in synthetic datasets for accuracy by ARI (A) and robustness by ARI (B), and in real datasets for robustness by ARI (C). Each dataset is represented by a different color of line. Dark lines show the mean and lighter colored surrounding band shows 1 standard deviation from across 5 replicates of ESCHR hard clustering results. X-axis is log2 scaled but the text labels show the unscaled values to aid interpretation. (D) Composite visualization illustrating the variability of optimal hyperparameters across different datasets. Dark and light red dots represent the highest scoring 10 hyperparameter combinations for each synthetic dataset and real dataset respectively. Dark and light blue dots represent the lowest scoring 10 hyperparameter combinations for each synthetic dataset and real dataset respectively. The vertical position of the dots represents the mean rank across accuracy ARI and AMI and robustness ARI and AMI and the horizontal position indicates the combination of hyperparameters represented by a given point, as specified by the overlapping colorbars. (E)

Heatmap showing the mean rank across accuracy ARI and AMI and robustness ARI and AMI and across all synthetic datasets analyzed with different ranges set for the 2 numeric hyperparameters, k number of neighbors for building the k-NN graph on the x-axis and r resolution for Leiden clustering on the y-axis. Dashed box indicates the highest scoring combination, which was selected for use in the version of ESCHR that was used to generate the figures in this manuscript. (F) Box and whisker plot showing the mean rank calculated as described for panels D and E for each of 4 different subsampling protocols. Individual dots represent each of 5 replicates of each of the 20 synthetic datasets. Note that lower rank indicates better performance.

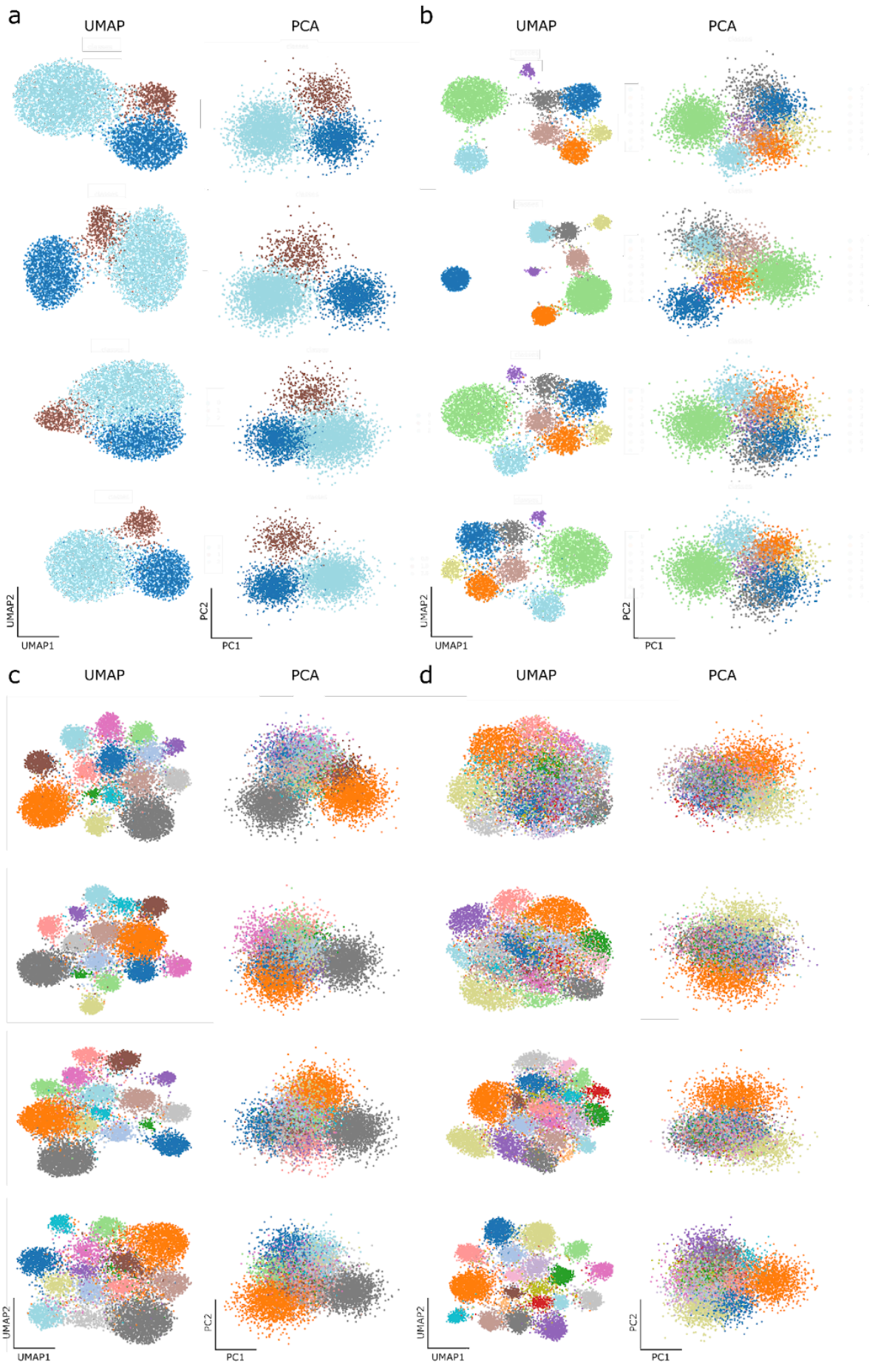



**Supplementary Figure 3: Combined visualization of ensemble base clusterings.** The base clusterings from Fig. 1a are displayed as ovals, with the same coloring as in the “Base clustering results” section of Fig. 1a (light brown, light blue, purple). The data points are displayed as filled circles, with labels and coloring matched to the “Input data” section of Fig. 1a (labels 1-18, colors blue, red, and green).

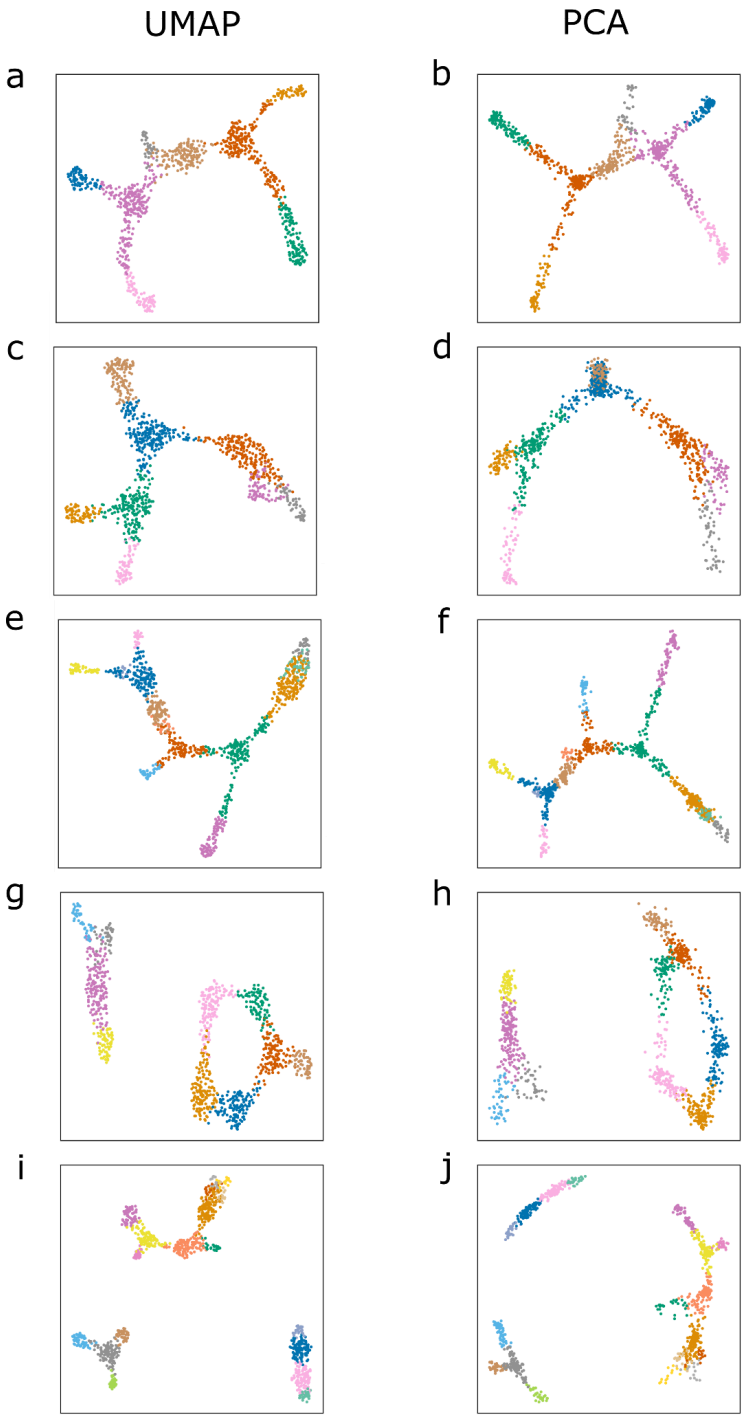



**Supplementary Figure 4: Validation of internal hyperparameter selection at consensus stage.** Sorted heatmap visualizations indicating ESCHR hard cluster groupings (y-axis, sorted group labels together) across different resolutions (on x-axis) for the consensus stage bipartite

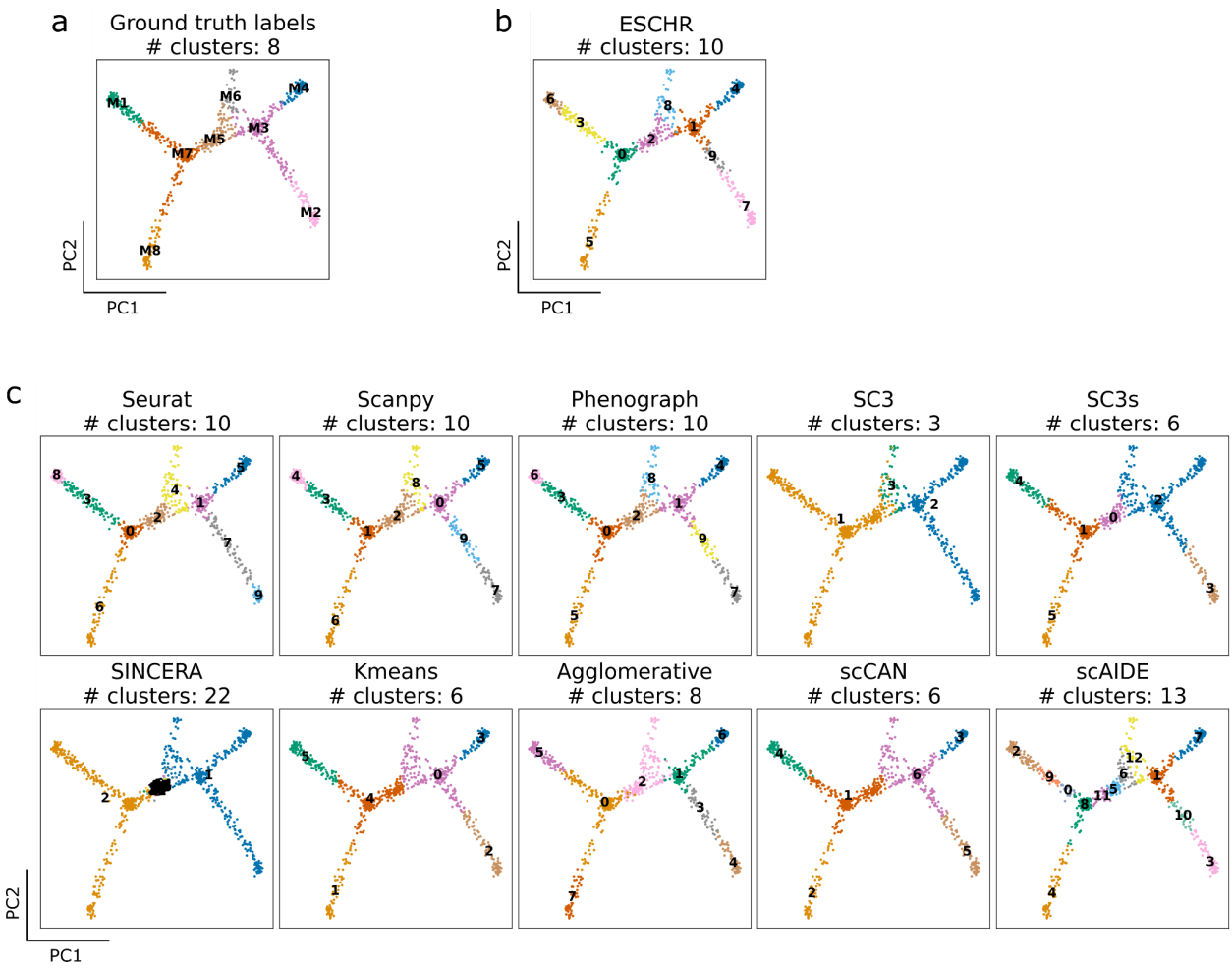
clustering. Example datasets shown here include the Tany-seq scRNA-seq dataset [77] (A), Zeisel scRNA-seq dataset [127] (B), Levine 13 dim mass cytometry dataset [101] (C). The red dashed line indicates the resolution that was selected by the ESCHR internal optimization protocol in each case. (D) Point plot showing the difference from true cluster number across 16 synthetic datasets at each possible value for the resolution hyperparameter gamma for final consensus bipartite clustering. Points represent means across all replicates of all datasets and error bars show 1 standard deviation. (E) Heatmap of mean ARI between consensus labelings from different resolutions within the same set of synthetic datasets used to generate panel D.


a
UMAP
PCA
b
UMAP
PCA
UMAP2
UMAP1
PC2
PC1
UMAP2
UMAP1
PC2
PC1
c
UMAP
PCA
d
UMAP
PCA
UMAP2
UMAP1
PC2
PC1
UMAP2
UMAP1
PC2
PC1

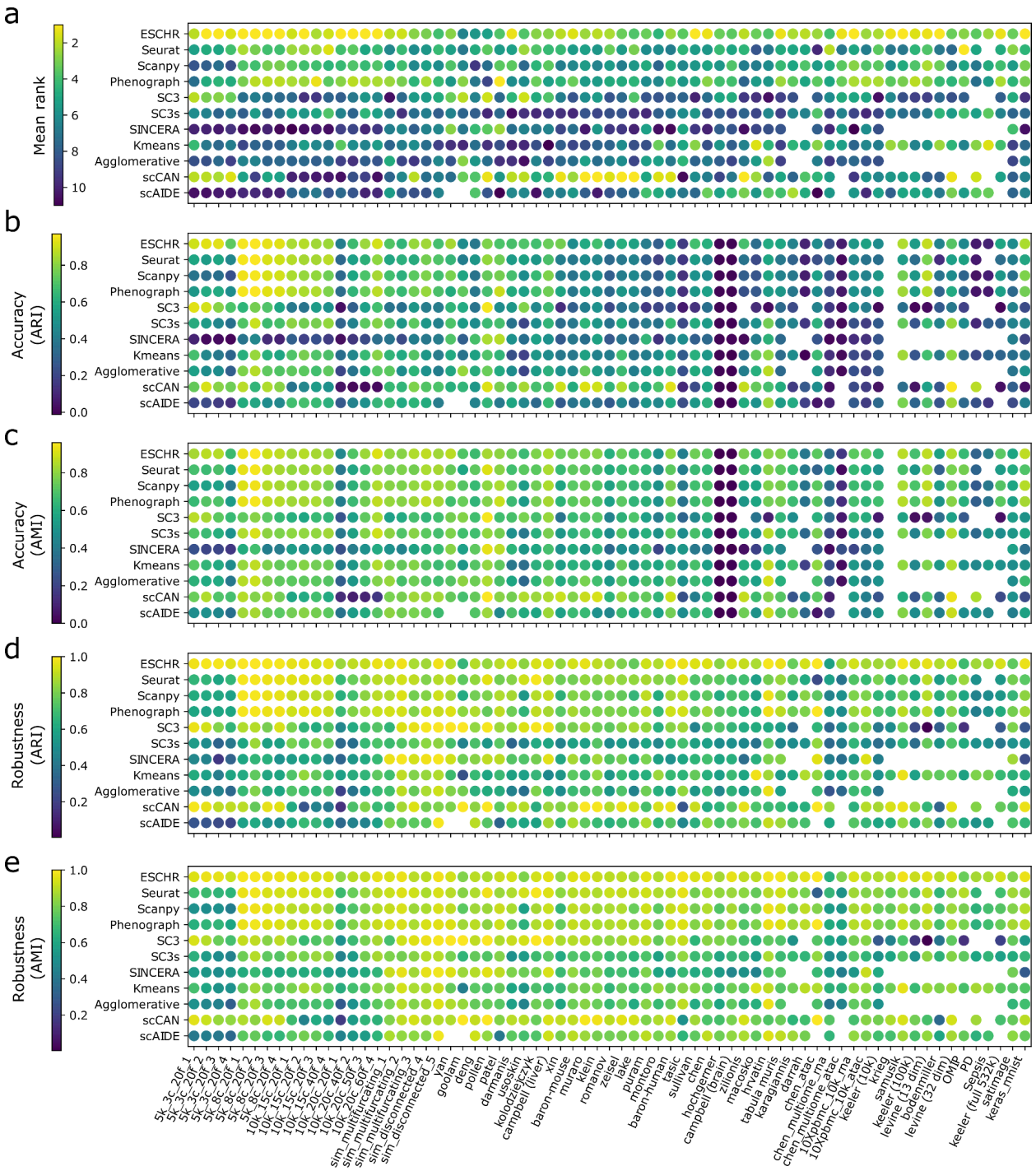
**Supplementary Figure 5: 2D visualizations of synthetic gaussian datasets.** UMAP (first column) and PCA (second column) dimensionality reduced 2D visualization of the synthetic gaussian datasets used for benchmark testing. (A) contains the datasets with three ground truth clusters, (B) contains the datasets with 8 ground truth clusters, (C) contains the datasets with 15 ground truth clusters, (D) contains the datasets with 20 ground truth clusters.




**Supplementary Figure 6: 2D visualizations of simulated scRNA-seq datasets.** UMAP (first column) and PCA (second column) dimensionality reduced 2D visualization of the synthetic gaussian datasets used in Figures 2 and 3. (A-B) contains the dataset labeled dyntoy_multi_1 in corresponding Supplementary Table 4, (C-D) contains the dataset dyntoy_multi_2, (E-F) contains the dataset dyntoy_multi_3, (G-H) contains the dataset dyntoy_discon_1, (I-J) contains the dataset dyntoy_discon_2.

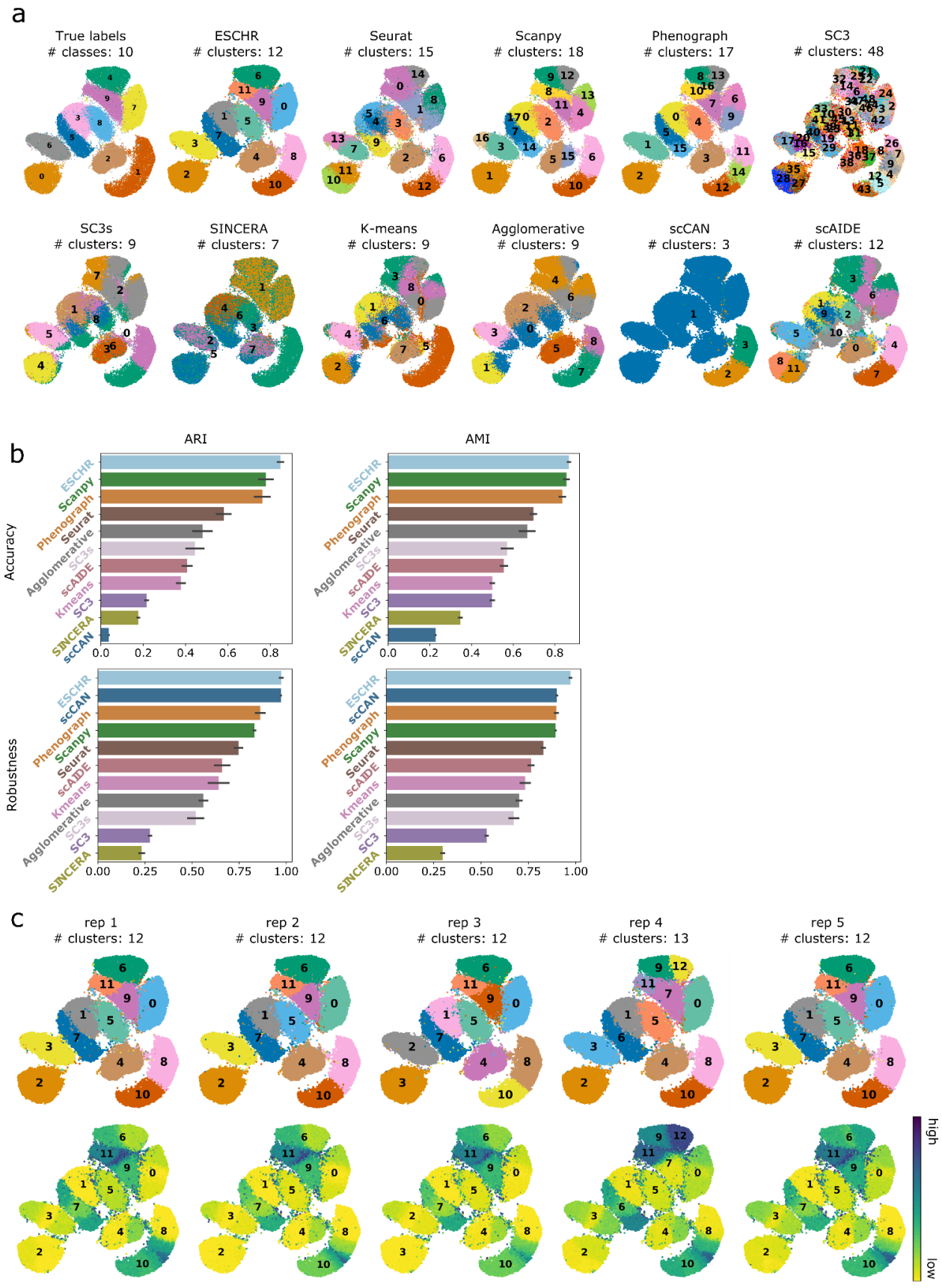



**Supplementary Figure 7: PCA embeddings for example simulated dataset.** Visualization of data points projected onto the first two principal components, with points colored by: (A) ground truth labels, (B) ESCHR hard cluster labels, (C) cluster labels from the method indicated in the respective title.

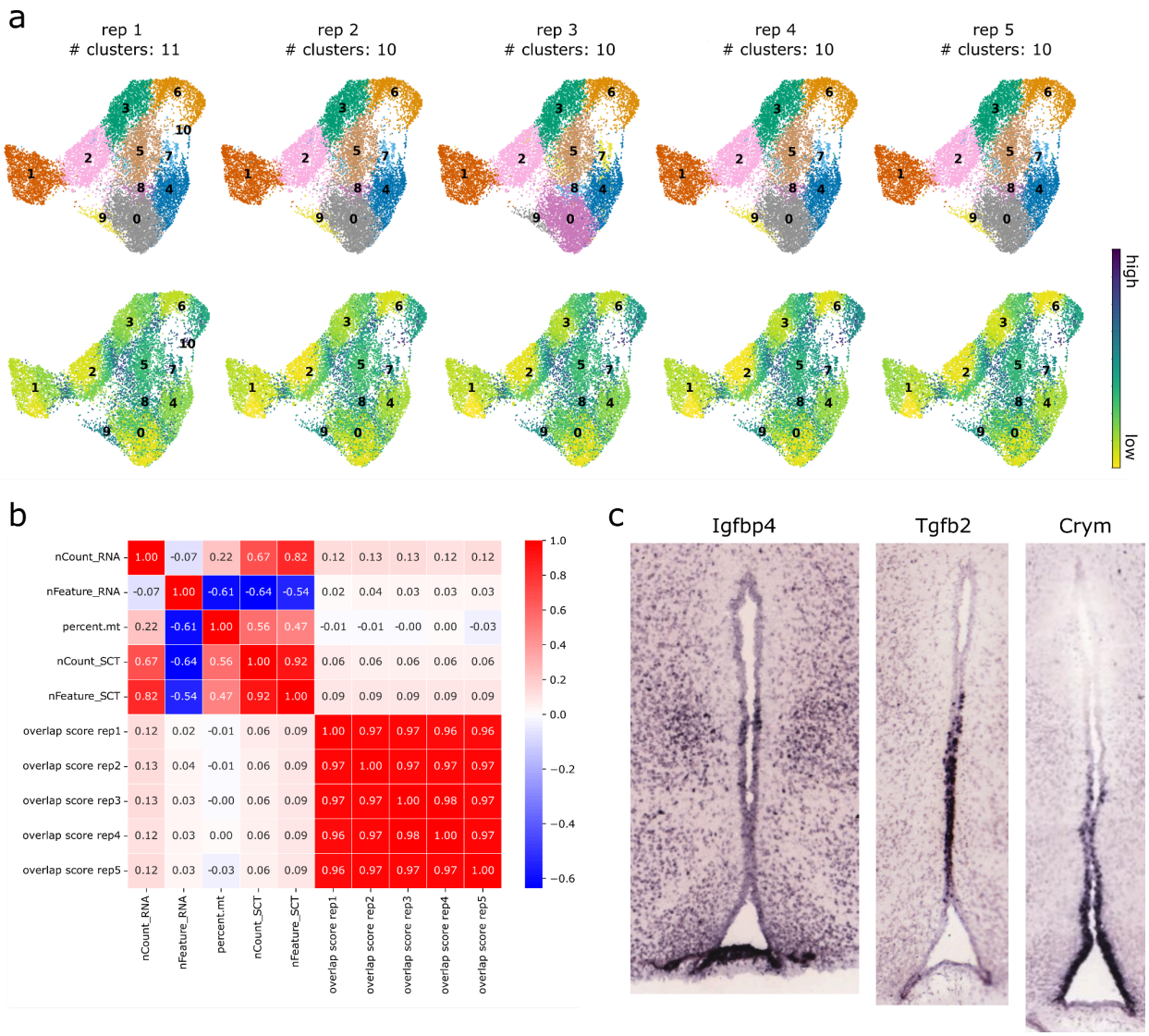



**Supplementary Figure 8: Individual scores for each dataset and each method used in systematic benchmarking.** (A) Mean rank across all scores; (B) accuracy measured by Adjusted Rand Index (ARI) between cluster labels and true labels; (C) accuracy measured by Adjusted Mutual Information (AMI) between cluster labels and true labels; (D) robustness measured by ARI between cluster labels generated for 5 independent random 90% subsamples of each dataset; (E) robustness measured by AMI between cluster labels generated for 5 independent random 90% subsamples of each dataset for each dataset per each method. Blank

spaces with no filled circle indicate that results were not able to be generated for the given dataset by the given method.


a
True labels
# classes: 10
ESCHR
# clusters: 12
Seurat
# clusters: 15
Scanpy
# clusters: 18
Phenograph
# clusters: 17
SC3
# clusters: 48
SC3s
# clusters: 9
SINCERA
# clusters: 7
K-means
# clusters: 9
Agglomerative
# clusters: 9
scCAN
# clusters: 3
scAIDE
# clusters: 12
b
ARI
AMI
Accuracy
Robustness
ESCHR
Scanpy
Phenograph
Seurat
Agglomerative
SC3s
scAIDE
Kmeans
SC3
SINCERA
scCAN
0.0
0.2
0.4
0.6
0.8
0.00
0.25
0.50
0.75
1.00
c
rep 1
# clusters: 12
rep 2
# clusters: 12
rep 3
# clusters: 12
rep 4
# clusters: 13
rep 5
# clusters: 12
high
low

**Supplementary Figure 9: Benchmarking and robustness analysis of ESCHR clustering of the MNIST dataset.** (A) UMAP visualizations of ground truth cluster labels and hard cluster assignments from the first robustness replicate of ESCHR and each of the comparison methods used in benchmarking analysis. Points are colored by cluster ID. (B) Barplots showing accuracy by ARI (top left), accuracy by AMI (top right), robustness by ARI (bottom left), and robustness by AMI (bottom right) for 5 runs of each method on a randomly sampled 90% of the MNIST dataset. Error bars represent 1 standard deviation. Bars are plotted in rank order with highest mean score at the top. (C) UMAP visualizations of ESCHR clustering replicates of the MNIST dataset, where each replicate was generated on a randomly subsampled 90% of data points. First row points are colored by hard cluster label, second row points are colored by uncertainty score.




**Supplementary Figure 10: Tanycyte clustering and expression profiles.** (A) UMAP visualizations of ESCHR clustering replicates of Tany-seq data, where each replicate was generated on a randomly subsampled 90% of data points. First row points are colored by hard cluster label, second row points are colored by per-cell uncertainty score. (B) Correlation of metadata features and per-cell uncertainty scores from each of the 5 robustness replicates visualized in A. (C) Allen Mouse Brain Atlas in situ hybridization images of the full tancyte-containing region.

# Supplementary Tables

(not sure of the appropriate way to include the large tables? I did see some theses with additional files, so presumably I could just as them as .xlsx?)

| Name | Reference | Software | Software language | Software Link | Base clustering method | Scalability | Requires hyperparameter tuning? | Confidence/stability? | Consensus/ensemble? | Soft/fuzzy? | Modality-specific? |
|---|---|---|---|---|---|---|---|---|---|---|---|
| K-means | Pedregosa et al. 2011 | yes (many) | many | https://scikit-learn.org/stable/modules/generated/sklearn.cluster | - | high cell number, high feature number | yes (k number of clusters) | no | no | no | no |
| Agglomerative | Pedregosa et al. 2011 | yes (many) | many | https://scikit-learn.org/stable/modules/generated/sklearn.cluster | - | moderate cell number, high feature number | yes (number of clusters or distance threshold) | no | no | no | no |
| Phenograph | Levine et al. 2015 | yes | python, R | https://github.com/dpeerlab/phenograph | graph-based community detection | high cell number, moderate feature number | yes (at minimum: k nearest neighbors, resolution for community detection) | no | no | no | no |
| SINCERA | Guo et al. 2015 | yes | R | https://research.cchmc.org/pbge/sincera.html | hierarchical | moderate cell number, high feature number | yes (number of clusters or distance threshold) | yes - cluster level | no | no | somewhat - hyperparameters and recommended workflow |
| SC3 | Kiselev et al. 2017 | yes | R | https://bioconductor.org/packages/release/bioc/html/SC3.html | k-means, hierarchical | low cell number, high feature number | yes (number of eigenvectors) | yes - cluster level | yes | no | somewhat - hyperparameters and recommended workflow |
| SC3s | Quah et al. 2022 | yes | python, R | https://github.com/hemberg-lab/sc3s | k-means | high cell number, high feature number | yes (k number of clusters) | no | yes | no | somewhat - hyperparameters and recommended workflow |
| Seurat | Butler et al. 2018 | yes | R | https://satijalab.org/seurat/ | graph-based community detection | moderate/high cell number, high feature number | yes (at minimum: k nearest neighbors, resolution for community detection) | no | no | no | somewhat - hyperparameters and recommended workflow |
| Scanpy | Wolf et al. 2018 | yes | python | https://scanpy.readthedocs.io/en/stable/ | graph-based community detection | high cell number, high feature number | yes (k nearest neighbors, resolution for community detection) | no | no | no | somewhat - hyperparameters and recommended workflow |
| scCAN | Tran et al. 2022 | yes | R | https://github.com/bangtran365/scCAN/ | spectral clustering | high cell number, high feature number | yes(number of network fusion iterations and k number of nearest neighbors) | no | yes | no | no |
| scAIDE | Xie et al. 2020 | yes (code modifications were required to run without error) | python | https://github.com/tinglabs/scAIDE/ | k-means | moderate cell number, high feature number | yes (max_point, kmax) | no | no | no | somewhat - hyperparameters and recommended workflow |

**Supplementary Table 1: Overview of methods used in benchmarking analysis.**

| Name | Package | # clusters | # samples | # features | # samples per cluster | Standard deviation per cluster | Additional information |
|---|---|---|---|---|---|---|---|
| 5k_3c_20f_1 | Scikit-learn | 3 | 5000 | 20 | 1500, 500, 3000 | 7.0, 8.0, 8.0 | |
| 5k_3c_20f_2 | Scikit-learn | 3 | 5000 | 20 | 1500, 500, 3000 | 7.0, 8.0, 8.0 | |
| 5k_3c_20f_3 | Scikit-learn | 3 | 5000 | 20 | 1500, 500, 3000 | 7.0, 8.0, 9.0 | |
| 5k_3c_20f_4 | Scikit-learn | 3 | 5000 | 20 | 1500, 500, 3000 | 7.0, 8.0, 9.0 | |
| 5k_8c_20f_1 | Scikit-learn | 8 | 5000 | 20 | 750, 500, 2000, 100, 450, 400, 250, 550 | 5.0, 5.0, 6.0, 4.0, 6.0, 7.0, 6.0, 4.0 | |
| 5k_8c_20f_2 | Scikit-learn | 8 | 5000 | 20 | 750, 500, 2000, 100, 450, 400, 250, 551 | 7.0, 5.0, 6.0, 4.0, 6.0, 7.0, 6.0, 4.0 | |
| 5k_8c_20f_3 | Scikit-learn | 8 | 5000 | 20 | 750, 500, 2000, 100, 450, 400, 250, 552 | 7.0, 5.0, 7.0, 4.0, 6.0, 7.0, 6.0, 6.0 | |
| 5k_8c_20f_4 | Scikit-learn | 8 | 5000 | 20 | 750, 500, 2000, 100, 450, 400, 250, 553 | 7.0, 7.0, 7.0, 5.0, 6.0, 7.0, 6.0, 6.0 | |
| 10k_15c_20f_1 | Scikit-learn | 15 | 10000 | 20 | 750, 500, 2000, 100, 450, 400, 250, 550, 750, 500, 2000, 500, 400, 250, 600 | 7.0, 7.0, 7.0, 5.0, 6.0, 7.0, 6.0, 6.0, 8.0, 7.0, 7.0, 5.0, 6.0, 7.0, 6.0, 6.0, 8.0, 7.0, 7.0, 5.0, 6.0, 7.0, 6.0, 7.0, 6.0 | |
| 10k_15c_20f_2 | Scikit-learn | 15 | 10000 | 20 | 750, 500, 2000, 100, 450, 400, 250, 550, 750, 500, 2000, 500, 400, 250, 601 | 10.0, 9.0, 11.0, 10.0, 8.0, 10.0, 10.0, 9.0, 10.0, 11.0, 9.0, 11.0, 10.0, 9.0, 8.0 | |
| 10k_15c_20f_3 | Scikit-learn | 15 | 10000 | 20 | 750, 500, 2000, 100, 450, 400, 250, 550, 750, 500, 2000, 500, 400, 250, 602 | 10.0, 9.0, 11.0, 10.0, 8.0, 10.0, 10.0, 9.0, 10.0, 11.0, 9.0, 11.0, 10.0, 9.0, 8.0 | |
| 10k_15c_40f_4 | Scikit-learn | 15 | 10000 | 40 | 750, 500, 1500, 100, 450, 400, 250, 550, 750, 500, 200, 500, 400, 250, 600, 500, 300, 1000, 250, 250, 2000, 500, 400, 250, 603 | 10.0, 9.0, 11.0, 10.0, 8.0, 10.0, 10.0, 9.0, 10.0, 11.0, 9.0, 11.0, 10.0, 9.0, 8.0, 11.0, 10.0, 9.0, 8.0 | |
| 10k_20c_20f_1 | Scikit-learn | 20 | 10000 | 20 | 750, 500, 1500, 100, 450, 400, 250, 550, 750, 500, 200, 500, 400, 250, 600, 500, 300, 1000, 250, 250 | 10.0, 9.0, 11.0, 10.0, 8.0, 10.0, 10.0, 9.0, 10.0, 11.0, 9.0, 11.0, 10.0, 9.0, 8.0, 11.0, 10.0, 9.0, 8.0 | 6 features anisotropically transformed post-hoc |
| 10k_20c_40f_2 | Scikit-learn | 20 | 10000 | 40 | 750, 500, 1500, 100, 450, 400, 250, 550, 750, 500, 200, 500, 400, 250, 600, 500, 300, 1000, 250, 251 | 10.0, 9.0, 11.0, 10.0, 8.0, 10.0, 10.0, 9.0, 10.0, 11.0, 9.0, 11.0, 10.0, 9.0, 8.0, 11.0, 10.0, 9.0, 8.0 | 10 features anisotropically transformed post-hoc |
| 10k_20c_50f_3 | Scikit-learn | 20 | 10000 | 50 | 750, 500, 1500, 100, 450, 400, 250, 550, 750, 500, 200, 500, 400, 250, 600, 500, 300, 1000, 250, 252 | 10.0, 9.0, 11.0, 10.0, 8.0, 10.0, 10.0, 9.0, 10.0, 11.0, 9.0, 11.0, 10.0, 9.0, 8.0, 11.0, 10.0, 9.0, 8.0 | 14 features anisotropically transformed post-hoc |
| 10k_20c_60f_4 | Scikit-learn | 20 | 10000 | 60 | 750, 500, 1500, 100, 450, 400, 250, 550, 750, 500, 200, 500, 400, 250, 600, 500, 300, 1000, 250, 253 | 10.0, 9.0, 11.0, 10.0, 8.0, 10.0, 10.0, 9.0, 10.0, 11.0, 9.0, 11.0, 10.0, 9.0, 8.0, 11.0, 10.0, 9.0, 8.0 | 18 features anisotropically transformed post-hoc |

**Supplementary Table 2: Overview of synthetic datasets used in benchmarking analysis.**

| Name | Package | # clusters | # samples | # features | Trajectory model |
|---|---|---|---|---|---|
| dyntoy_multi_1 | DynToy | 8 | 1000 | 1000 | multifurcated |
| dyntoy_multi_2 | DynToy | 8 | 1000 | 1000 | multifurcated |
| dyntoy_multi_3 | DynToy | 13 | 1000 | 1000 | multifurcated |
| dyntoy_discon_1 | DynToy | 10 | 1000 | 1000 | disconnected |
| dyntoy_discon_2 | DynToy | 18 | 1000 | 1000 | disconnected |

**Supplementary Table 3: Overview of simulated scRNA-seq datasets used in benchmarking analysis.**

| Data set | Train/Test | Category | # Cells/ instances | # Features | Published # Classes | Annotation Source | Tissue | Protocol | Reference | Data download link |
|---|---|---|---|---|---|---|---|---|---|---|
| Deng | train | scRNA-seq | 268 | 22,431 | 6 | experimental condition - developmental stage | Mouse embryo | Smart-Seq2 | Deng et al., 2014 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/edev/#deng |
| Usoskin | train | scRNA-seq | 622 | 25,334 | 4, 8, and 11 | clustering | Mouse brain | STRT-Seq | Usoskin et al., 2015 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/brain/#usoskin |
| Zeisel | train | scRNA-seq | 3,005 | 19,972 | 9 and 47 | clustering | Mouse brain | STRT-Seq | Zeisel et al., 2015 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/brain/#zeisel |
| Kolodziejczyk | train | scRNA-seq | 704 | 38,653 | 3 | experimental condition - culture media | Mouse embryo stem cells | SMARTer | Kolodziejczyk et al., 2015 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/esc/#kolodziejczyk |
| Campbell | train | scRNA-seq | 21,086 | 26,774 | 21 | clustering | Mouse arcuate nucleus and median eminence | Drop-seq | Campbell et al., 2017 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/brain/ |
| Tasic | train | scRNA-seq | 12,832 | 42,625 | 4, 22, and 110 | clustering | Primary mouse visual cortex | SMART-Seq v4 | Tasic et al., 2018 | https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE115746 |
| Gokce | train | scRNA-seq | 1,208 | 16,673 | 10 | clustering | Mouse striatum | Smart-Seq2 | Gokce et al., 2016 | https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE82187 |
| Samusik | train | Other single cell (mass cytometry) | 86,864 | 39 | 24 | expert manual gating (not all cells were assigned) | Mouse bone marrow | CyTOF | Samusik et al., 2016 | https://github.com/lmweber/HDCytoData |
| Levine (32dim) | train | Other single cell (mass cytometry) | 265,627 | 32 | 14 | expert manual gating (not all cells were assigned) | Human bone marrow healthy donors | CyTOF | Levine et al., 2015 | https://github.com/lmweber/HDCytoData |
| Levine (13dim) | train | Other single cell (mass cytometry) | 167,044 | 13 | 24 | expert manual gating (not all cells were assigned) | Human bone marrow healthy donors | CyTOF | Levine et al., 2015 | https://github.com/lmweber/HDCytoData |
| Keeler (10k) | train | Other single cell (mass cytometry) | 10,000 | 37 | 41 | clustering | Mouse dorsal root ganglion | CyTOF | Keeler et al., 2022 | https://community.cytobank.org/cytobank/experiments/102249 |
| Keeler (100k) | train | Other single cell (mass cytometry) | 100,000 | 37 | 41 | clustering | Mouse dorsal root ganglion | CyTOF | Keeler et al., 2022 | https://community.cytobank.org/cytobank/experiments/102249 |
| Keeler (full) | train | Other single cell (mass cytometry) | 532,926 | 37 | 41 | clustering | Mouse dorsal root ganglion | CyTOF | Keeler et al., 2022 | https://community.cytobank.org/cytobank/experiments/102249 |
| Yan | test | scRNA-seq | 90 | 20,214 | 6 | experimental condition - developmental stage | Human embryo | Tang | Yan et al., 2013 | https://hemberg-lab.github.io/scRNA.seq.datasets/human/edev/ |
| Goolam | test | scRNA-seq | 124 | 41,480 | 5 | experimental condition - developmental stage | Mouse embryo | Smart-Seq2 | Goolam et al., 2016 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/edev/#goolam |
| Pollen | test | scRNA-seq | 301 | 23,730 | 4 and 11 | experimental condition - cell line | Human tissues | SMARTer | Pollen et al., 2014 | https://hemberg-lab.github.io/scRNA.seq.datasets/human/tissues/#pollen |
| Patel | test | scRNA-seq | 430 | 5,948 | 5 | experimental condition - different tumors | Human tissues | Smart-Seq | Patel et al., 2014 | https://hemberg-lab.github.io/scRNA.seq.datasets/human/tissues/#patel |
| Darmanis | test | scRNA-seq | 466 | 20,214 | 9 | clustering (supervised and unsupervised with high concordance) | Human brain | SMARTer | Darmanis et al., 2015 | https://hemberg-lab.github.io/scRNA.seq.datasets/human/brain/#darmanis |
| Camp (Liver) | test | scRNA-seq | 777 | 19,020 | 7 | clustering | Human liver | SMARTer | Camp et al., 2017 | https://hemberg-lab.github.io/scRNA.seq.datasets/human/liver/ |
| Xin | test | scRNA-seq | 1,600 | 39,851 | 8 | single marker gene per established cell type | Human pancreas | SMARTer | Xin et al., 2016 | https://hemberg-lab.github.io/scRNA.seq.datasets/human/pancreas/#xin |

| Data set | Train/Test | Category | # Cells/ instances | # Features | Published # Classes | Annotation Source | Tissue | Protocol | Reference | Data download link |
|---|---|---|---|---|---|---|---|---|---|---|
| Baron (Mouse) | test | scRNA-seq | 1,886 | 14,878 | 13 | clustering, with biological validation | Mouse pancreas | inDrop | Baron et al., 2016 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/pancreas/ |
| Muraro | test | scRNA-seq | 2,126 | 19,140 | 10 | clustering | Human pancreas | CEL-Seq2 | Muraro et al., 2016 | https://hemberg-lab.github.io/scRNA.seq.datasets/human/pancreas/#muraro |
| Klein | test | scRNA-seq | 2,717 | 24,175 | 4 | experimental condition - collection day | Mouse embryo stem cells | inDrop | Klein et al., 2015 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/esc/ |
| Romanov | test | scRNA-seq | 2,881 | 24,341 | 7 | clustering | Mouse brain | SMARTer | Romanov et al., 2017 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/brain/#romanov |
| Lake | test | scRNA-seq | 3,042 | 25,123 | 16 | clustering | Human brain | Fluidigm C1 | Lake et al., 2016 | https://hemberg-lab.github.io/scRNA.seq.datasets/human/brain/#lake |
| Puram | test | scRNA-seq | 5,902 | 23,686 | 10 | clustering | Human tissues | Smart-Seq2 | Puram et al., 2017 | https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE103322 |
| Montoro | test | scRNA-seq | 7,193 | 27,716 | 7 | clustering | Human pancreas | Smart-Seq2 | Montoro et al., 2018 | https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE103354 |
| Baron (Human) | test | scRNA-seq | 8,569 | 20,125 | 14 | clustering | Human pancreas | inDrop | Baron et al., 2016 | https://hemberg-lab.github.io/scRNA.seq.datasets/human/pancreas/ |
| Chen | test | scRNA-seq | 12,089 | 23,284 | 46 | clustering | Mouse brain | Drop-seq | Chen et al., 2017 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/brain/#chen |
| Hochgerner | test | scRNA-seq | 18,213 | 27,998 | 13 | clustering | Postnatal developing dentate gyrus (P1 and P5) | 10x Genomics | Hochgerner et al., 2018 | https://linnarssonlab.org/dentate/ |
| Zilionis | test | scRNA-seq | 34,558 | 41,861 | 9 | classification based on protein markers | Human lung | inDrop | Zilionis et al., 2019 | https://singlecell.broadinstitute.org/single_cell/study/SCP739/ |
| Macosko | test | scRNA-seq | 44,808 | 23,288 | 12 | clustering | Mouse retina | Drop-seq | Macosko et al., 2015 | https://hemberg-lab.github.io/scRNA.seq.datasets/mouse/retina/ |
| Hrvatin | test | scRNA-seq | 48,266 | 25,187 | 8 | clustering | Mouse visual cortex | inDrop | Hrvatin et al., 2018 | https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE102827 |
| Tabula Muris | test | scRNA-seq | 54,439 | 23,337 | 40 | clustering | Mouse tissues | 10X Genomics | Schaum et al., 2018 | https://doi.org/10.6084/m9.figshare.5968960.v3 |
| Karagiannis | test | scRNA-seq | 72,914 | 19,012 | 12 | clustering | Human blood | 10X Genomics | Karagiannis et al., 2020 | https://singlecell.broadinstitute.org/single_cell/study/SCP587/ |
| Darrah | test | scRNA-seq | 162,490 | 32,437 | 14 | clustering | Human blood | Drop-seq | Darrah et al., 2020 | https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE139598 |
| Sullivan | test | scRNA-seq | 14,402 | 3,000 | 4 | clustering | Integrated dataset of murine tanycytes | mixed | Sullivan et al., 2022 | https://data.mendeley.com/datasets/w8yw2c92jg/1 |
| 10x PMBC - ATAC | test | multiome | 12,016 | 63,751 | 16 | clustering | Human blood | 10x Chromium | 10x Genomics, 2020 | https://www.10xgenomics.com/datasets/pbmc-from-a- |
| 10x PMBC - RNA | test | multiome | 12,016 | 30,415 | 16 | clustering | Human blood | 10x Chromium | 10x Genomics, 2020 | https://www.10xgenomics.com/datasets/pbmc-from-a- |
| Chen multiome ATAC | test | multiome | 5,081 | 229,429 | 19 | clustering | Neonatal mouse brain | SNARE-seq | Chen et al., 2019 | https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE126074 |
| Chen muliome RNA | test | multiome | 5,081 | 19,322 | 19 | clustering | Neonatal mouse brain | SNARE-seq | Chen et al., 2019 | https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE126074 |
| Bodenmiller | test | Other single cell (mass cytometry) | 172,791 | 24 | 14 | clustering | Human PBMC | CyTOF | Bodenmiller et al., 2012 | http://bioconductor.org/packages/HDCytoData. |

| Data set | Train/Test | Category | # Cells/ instances | # Features | Published # Classes | Annotation Source | Tissue | Protocol | Reference | Data download link |
|---|---|---|---|---|---|---|---|---|---|---|
| Krieg (antiPD1) | test | Other single cell (mass cytometry) | 85,715 | 24 | 7 | clustering | Human PBMC, melanoma, anti-PD1 therapy | CyTOF | Krieg et al., 2018 | https://github.com/lmweber/HDCytoData |
| MNIST | test | Not single cell | 70,000 | 784 | 10 | true intended digit | Images of handwritten digits | NA | Lecun et al., 1998 | https://keras.io/api/datasets/mnist/ |
| Sattelite Images | test | Not single cell | 6,435 | 36 | 6 | true class | Landsat satellite images of U.S. federal land | NA | https://archive.ics.uci.edu/dataset/146/statlog+landsat+satellite | https://archive.ics.uci.edu/dataset/146/statlog+landsat+satellite |

**Supplementary Table 4: Overview of real datasets used in benchmarking analysis.**

| Method | Evaluation | Metric | Statistic | Corrected p-value |
|---|---|---|---|---|
| Seurat | robustness | ARI | 26 | 9.96E-06 |
| Scanpy | robustness | ARI | 24 | 1.86E-08 |
| Phenograph | robustness | ARI | 15 | 1.19E-04 |
| SC3 | robustness | ARI | 0 | 7.10E-06 |
| SC3s | robustness | ARI | 15 | 9.31E-09 |
| SINCERA | robustness | ARI | 17 | 4.08E-05 |
| Kmeans | robustness | ARI | 26 | 9.31E-09 |
| Agglomerative | robustness | ARI | 0 | 5.96E-07 |
| scCAN | robustness | ARI | 0 | 1.28E-06 |
| scAIDE | robustness | ARI | 6 | 7.45E-08 |
| Seurat | robustness | AMI | 1 | 1.93E-06 |
| Scanpy | robustness | AMI | 43 | 1.86E-08 |
| Phenograph | robustness | AMI | 0 | 9.96E-06 |
| SC3 | robustness | AMI | 1 | 1.57E-06 |
| SC3s | robustness | AMI | 1 | 9.31E-09 |
| SINCERA | robustness | AMI | 1 | 4.17E-06 |
| Kmeans | robustness | AMI | 16 | 9.31E-09 |
| Agglomerative | robustness | AMI | 6 | 2.98E-07 |
| scCAN | robustness | AMI | 0 | 1.30E-07 |
| scAIDE | robustness | AMI | 1 | 7.45E-08 |
| Seurat | accuracy | ARI | 54 | 4.54E-04 |
| Scanpy | accuracy | ARI | 2 | 9.31E-08 |
| Phenograph | accuracy | ARI | 0 | 5.13E-03 |
| SC3 | accuracy | ARI | 0 | 2.79E-08 |
| SC3s | accuracy | ARI | 7 | 9.31E-09 |
| SINCERA | accuracy | ARI | 21 | 2.98E-07 |
| Kmeans | accuracy | ARI | 41 | 9.31E-09 |
| Agglomerative | accuracy | ARI | 1 | 5.96E-07 |
| scCAN | accuracy | ARI | 0 | 1.77E-07 |
| scAIDE | accuracy | ARI | 3 | 3.73E-08 |
| Seurat | accuracy | AMI | 5 | 4.16E-06 |
| Scanpy | accuracy | AMI | 78 | 9.31E-09 |
| Phenograph | accuracy | AMI | 0 | 9.19E-05 |
| SC3 | accuracy | AMI | 1 | 4.66E-08 |
| SC3s | accuracy | AMI | 0 | 1.86E-08 |
| SINCERA | accuracy | AMI | 0 | 2.98E-07 |
| Kmeans | accuracy | AMI | 3 | 9.31E-09 |
| Agglomerative | accuracy | AMI | 0 | 5.96E-07 |
| scCAN | accuracy | AMI | 1 | 4.66E-08 |
| scAIDE | accuracy | AMI | 0 | 3.73E-08 |

**Supplementary Table 5: Statistics for benchmarking analysis with synthetic datasets.**
Two-sided Wilcoxon signed-rank test with Bonferroni correction was used for statistical analysis comparing ESCHR to each method. N = 20 for comparisons using synthetic datasets.

| Method | Evaluation | Metric | Statistic | Corrected p-value |
|---|---|---|---|---|
| Seurat | robustness | ARI | 103 | 7.08E-06 |
| Scanpy | robustness | ARI | 109 | 1.14E-12 |
| Phenograph | robustness | ARI | 49 | 1.90E-05 |
| SC3 | robustness | ARI | 8 | 4.47E-05 |
| SC3s | robustness | ARI | 288 | 1.14E-12 |
| SINCERA | robustness | ARI | 152 | 3.21E-05 |
| Kmeans | robustness | ARI | 59 | 2.84E-11 |
| Agglomerative | robustness | ARI | 0 | 8.15E-09 |
| scCAN | robustness | ARI | 8 | 1.76E-01 |
| scAIDE | robustness | ARI | 215 | 9.09E-11 |
| Seurat | robustness | AMI | 0 | 2.53E-04 |
| Scanpy | robustness | AMI | 75 | 1.14E-11 |
| Phenograph | robustness | AMI | 0 | 7.23E-06 |
| SC3 | robustness | AMI | 4 | 1.04E-04 |
| SC3s | robustness | AMI | 3 | 1.14E-12 |
| SINCERA | robustness | AMI | 5 | 1.73E-06 |
| Kmeans | robustness | AMI | 120 | 2.84E-11 |
| Agglomerative | robustness | AMI | 28 | 3.49E-09 |
| scCAN | robustness | AMI | 2 | 8.53E-03 |
| scAIDE | robustness | AMI | 9 | 6.00E-10 |
| Seurat | accuracy | ARI | 45 | 7.44E-06 |
| Scanpy | accuracy | ARI | 36 | 4.37E-04 |
| Phenograph | accuracy | ARI | 97 | 4.15E-04 |
| SC3 | accuracy | ARI | 190 | 9.96E-06 |
| SC3s | accuracy | ARI | 420 | 1.31E-02 |
| SINCERA | accuracy | ARI | 120 | 1.22E-02 |
| Kmeans | accuracy | ARI | 112 | 2.15E-02 |
| Agglomerative | accuracy | ARI | 56 | 1.00E+00 |
| scCAN | accuracy | ARI | 75 | 1.00E+00 |
| scAIDE | accuracy | ARI | 360 | 7.87E-03 |
| Seurat | accuracy | AMI | 112 | 1.04E-04 |
| Scanpy | accuracy | AMI | 102 | 4.47E-05 |
| Phenograph | accuracy | AMI | 179 | 4.37E-04 |
| SC3 | accuracy | AMI | 198 | 3.51E-06 |
| SC3s | accuracy | AMI | 136 | 2.92E-07 |
| SINCERA | accuracy | AMI | 109 | 3.22E-06 |
| Kmeans | accuracy | AMI | 66 | 2.22E-06 |
| Agglomerative | accuracy | AMI | 32 | 1.55E-03 |
| scCAN | accuracy | AMI | 87 | 1.00E+00 |
| scAIDE | accuracy | AMI | 82 | 3.39E-05 |

**Supplementary Table 6: Statistics for benchmarking analysis with real datasets.**
Two-sided Wilcoxon signed-rank test with Bonferroni correction was used for statistical analysis comparing ESCHR to each method. N = 42 for comparisons using real datasets.