scTree: Discovering Cellular Hierarchies in the Presence of Batch Effects in scRNA-seq Data

Moritz Vandenhirtz [*1] **Florian Barkmann** [*1] **Laura Manduchi** [1] **Julia E. Vogt** [†1] **Valentina Boeva** [†1]

Abstract

We propose a novel method, scTree, for single-cell Tree Variational Autoencoders, extending a hierarchical clustering approach to single-cell RNA sequencing data. scTree corrects for batch effects while simultaneously learning a tree-structured data representation. This VAE-based method allows for a more in-depth understanding of complex cellular landscapes independently of the biasing effects of batches. We show empirically on seven datasets that scTree discovers the underlying clusters of the data and the hierarchical relations between them, as well as outperforms established baseline methods across these datasets. Additionally, we visualize the learned trees to better understand the hierarchy and their biological relevance, thus underpinning the importance of integrating batch correction directly into the clustering procedure.

1. Introduction

Recent progress in high-throughput sequencing technologies has enabled single-cell RNA sequencing (scRNA-seq) to emerge as a powerful approach for investigating cellular diversity in various tissues and organisms (Sikkema et al., 2022; Eraslan et al., 2022). This technique offers a comprehensive overview of gene expression variation across multiple individual cells. Clustering analysis is a critical tool in understanding scRNA-seq data, as it enables the identification of homogeneous sub-populations of cells (Kiselev et al., 2019). By analyzing gene expression patterns, clustering analysis can reveal previously unknown cell identities and functions and detect both common and rare cell types (Osumi-Sutherland et al., 2021). Among various clustering techniques, hierarchical clustering is a popular tool as it provides an unsupervised path to find cell sub-populations and their hierarchical relationships at different granularity (Žurauskienė & Yau, 2016; Jiang et al., 2018; Zou et al., 2021).

Despite its potential, traditional hierarchical clustering techniques might lead to sub-optimal results when applied to scRNA-seq data due to factors like amplification biases, and high-dimensional input spaces. The presence of batch effects poses an additional challenge, confounding the accurate identification of cell populations (Lähnemann et al., 2020). Batch effects refer to the inherent technical variations across different experimental batches, such as variations in sample preparation, sequencing platforms, or environmental conditions.

To overcome this issue, batch integration methods have been developed to harmonize scRNA-seq data from multiple experimental batches, thereby reducing or eliminating the effects of technical variability while preserving biological signals (Luecken et al., 2022). By aligning and integrating data from different batches, these methods enable more robust downstream analyses, such as clustering, differential expression analysis, and trajectory inference. Current best practices for analyzing scRNA-seq data involve a two-step procedure. The first step is dimensionality reduction with batch integration to compress the data. Then, clustering at different resolutions is performed on the lower dimensional data representation (Luecken & Theis, 2019; Hua & Zhang, 2019).

Variational autoencoders (VAEs) are widely utilized in the realm of scRNA-seq data analysis to leverage large amounts of available data and learn compressed latent representations with batch integration (Lopez et al., 2018; Svensson et al., 2020b; Grønbech et al., 2020; Lotfollahi et al., 2022). Although clustering on the latent space of VAEs improves upon clustering on the raw data (Luecken et al., 2022), the combined optimization of clustering algorithms and VAE's representations has demonstrated substantial improvements in clustering performance (Shin et al., 2019). Among these works, Tree Variational Autoencoders (TreeVAE) (Manduchi et al., 2023) is an end-to-end VAE-based method that discovers the inherent hierarchical structure of the data by

[*]Equal contribution [†]Shared last authors [1]Department of Computer Science, ETH Zurich, Switzerland. Correspondence to: Moritz Vandenhirtz <moritz.vandenhirtz@inf.ethz.ch>, Florian Barkmann <florian.barkmann@inf.ethz.ch>.

Accepted by the Structured Probabilistic Inference & Generative Modeling workshop of ICML 2024, Vienna, Austria.

learning a tree-based posterior probability of latent variables.

Building upon the work of TreeVAE, we extend this framework to address the challenges presented by scRNA-seq data. We propose scTree, a method that integrates hierarchical clustering with batch correction techniques to enhance the clustering of scRNA-seq data. Additionally, we introduce a splitting rule that is able to capture the imbalanced clusters in the data. Our approach identifies the inherent hierarchical structure of cellular populations while simultaneously mitigating batch effects, thereby enabling a more precise understanding of cell types and states. By jointly optimizing hierarchical clustering and batch-integrated representation learning within the VAE framework, we offer a powerful tool for dissecting complex cellular landscapes and unraveling the intricacies of biological systems at a single-cell resolution. To the best of our knowledge, this is the first work that explores hierarchical clustering with VAEs trained jointly with batch integration for scRNA-seq data.

Our main contributions are as follows: i) We propose an extension of TreeVAE to scRNA-seq data that simultaneously corrects for batch effects and learns a binary tree to mimic the hierarchies present in the data. Additionally, we propose a novel splitting rule, removing the assumption of balanced clusters. ii) We evaluate our method on seven different datasets and compare it to three baselines to demonstrate its effectiveness. iii) We qualitatively assess the learned hierarchy and show the correspondence to the underlying biological systems.

2. Related Work

Hierarchical clustering algorithms are a frequently used technique for unraveling the intricate hierarchical structures inherent in biological data. Agglomerative hierarchical clustering algorithms (Sneath, 1957; Ward, 1963; Murtagh & Contreras, 2012) treat each data point as a separate cluster and progressively merge these clusters based on their proximity, as defined by a specific distance metric. Diverging from traditional agglomerative techniques, Bayesian Hierarchical Clustering (Heller & Ghahramani, 2005) introduces a probabilistic framework that utilizes hypothesis testing for cluster merging decisions. Divisive hierarchical clustering algorithms (Kaufman & Rousseeuw, 2009), the category which TreeVAE falls into, offer an alternative strategy, starting with a single cluster that encompasses all data points and iteratively dividing it into smaller clusters. The Bisecting-K-means algorithm (Steinbach et al., 2000; Nistér & Stewénius, 2006) repeatedly applies k-means clustering to divide data into two parts. Relatedly, Williams (1999) learn a hierarchical probabilistic Gaussian mixture model. Further hierarchical probabilistic clustering methods include VAE-nCRP (Goyal et al., 2017; Shin et al., 2019) and the TMC-VAE (Vikram et al., 2018). that use Bayesian nonparametric hierarchical clustering based on the nested Chinese restaurant process (nCRP) prior (Blei et al., 2003) or the time-marginalized coalescent (TMC).

Various hierarchical clustering algorithms have emerged to address the unique challenges encountered in single-cell RNA sequencing (scRNA-seq) data analysis. (Lin et al., 2017) proposed Clustering through Imputation and Dimensionality Reduction (CIDR), leveraging imputation techniques within a hierarchical framework to mitigate the impact of dropouts inherent in scRNA-seq data. (Morelli et al., 2021) presented Nested Stochastic Block Models (NSBM) and (Zou et al., 2021) proposed Hierarchical Graph-based clustering (HGC), both offering methods for hierarchical clustering directly on the k-nearest neighbor graph of cells, bypassing the count matrix. Additionally, scDEF, a method introduced by (Ferreira et al., 2022), employs a two-level Bayesian matrix factorization model to jointly generate hierarchical clustering and infer gene signatures for each cluster. Notably, among these methods, only scDEF has the capability to handle batch effects. However, it generates a two-level hierarchy rather than a binary tree, posing challenges for comparisons with methods such as scTree.

3. Methodology

We propose scTree, a VAE-based method that uncovers hierarchical structures in single-cell RNA sequencing data. We build upon the recently proposed TreeVAE (Manduchi et al., 2023) and extend it to learn a structured latent space corrected for batch effects, thereby enabling the discovery of cell types (and subtypes) in an unsupervised way. Figure 1 provides a schematic overview of scTree. In Section 3.1, we summarize TreeVAE, a method designed to perform hierarchical clustering with VAEs. In Section 3.2, we then propose an extension that allows for the discovery of hierarchies in scRNA-seq data.

3.1. Tree Variational Autoencoders

This section provides an overview on the Tree Variational Autoencoder (TreeVAE Manduchi et al., 2023). TreeVAE is a hierarchical VAE composed of a tree structure of latent variables, whose structure is learned during training. It thus learns (i) a hierarchical generative model that permits the generation of new samples and (ii) a hierarchical clustering of data points, thus uncovering meaningful patterns in the data, and a hierarchical categorization of samples.

TreeVAE defines a probabilistic binary tree $\mathcal{T}$, where each node i is characterized by a sample-specific embedding $\mathbf{z}_i$. The generative path of a sample is as follows: First, the root node's latent embedding $\mathbf{z}_0$ is sampled from a standard Gaussian. From this embedding, the probabilities of

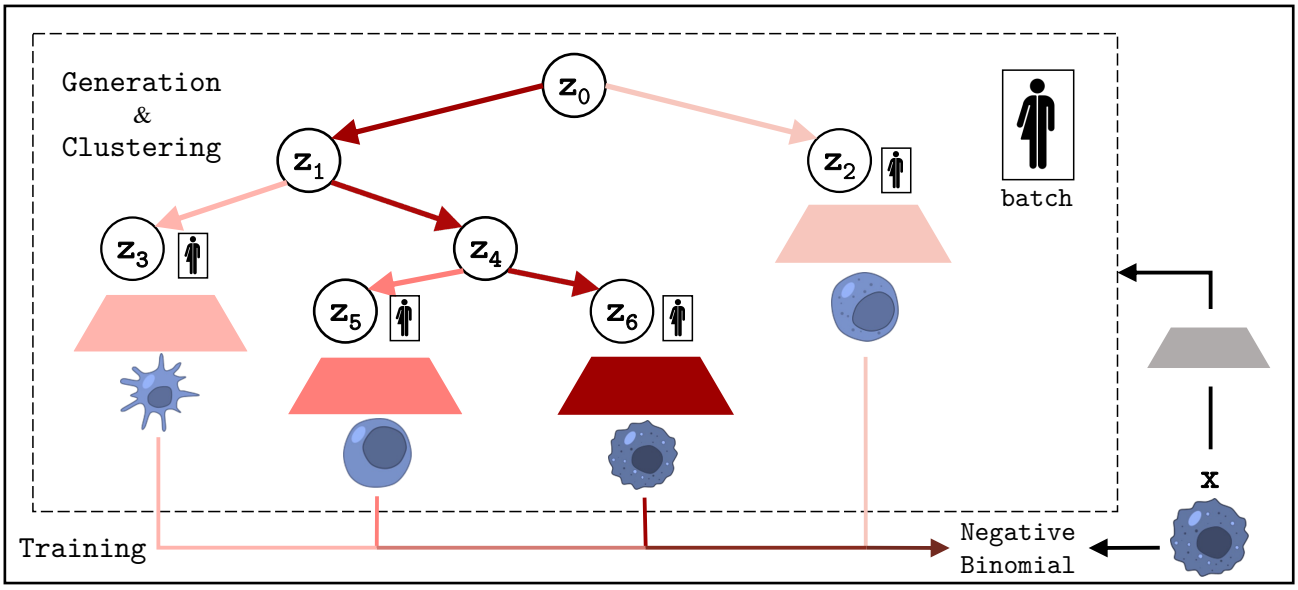

Figure 1. Schematic overview of the proposed method. The input $\boldsymbol{x}$ is passed through an encoder to be consequently reconstructed through a tree-shaped process. The process consists of probabilistically going left or right in each node, followed by a nonlinear transformation on the embedding $\boldsymbol{z}_i$. The cluster-specific decoders take as input their leaf-embedding and batch information and reconstruct the gene count parameters of the negative binomial distribution.

going to the left or right child in the tree are computed by a multilayer perceptron. The latent embedding of the selected child $\boldsymbol{z}_i$ is sampled from a Gaussian distribution $p_\theta(\boldsymbol{z}_i \mid \boldsymbol{z}_0) = \mathcal{N}\left(\mathbf{z}_i \mid \mu_{p,i}\left(\boldsymbol{z}_0\right), \sigma^2_{p,i}\left(\boldsymbol{z}_0\right)\right)$ conditioned on its parent. This routing–transformation process is repeated until a leaf node is reached. Each leaf corresponds to one cluster and includes a decoder through which the observed sample is generated, conditioned on the sample-specific leaf embedding. To recover the assumed generative model, the inference model of TreeVAE matches the tree structure. To avoid a posterior collapse of the root, they utilize the trick of LadderVAE (Sønderby et al., 2016) to learn a bottom-up chain from the sample $\boldsymbol{x}$ to the root $\boldsymbol{z}_0$ with which the generative model can be guided.

To optimize the parameters of the generative and inference model and to learn the tree structure, TreeVAE iterates two training steps sequentially: model refinement and tree growing. During the model refinement, it assumes a fixed tree (starting from a root and two children) and optimizes the Evidence Lower Bound (ELBO): $\mathcal{L}(\boldsymbol{x} \mid \mathcal{T}) := \mathbb{E}_{q(\boldsymbol{z}_{\mathcal{P}_l}, \mathcal{P}_l \mid \boldsymbol{x})}[\log p(\boldsymbol{x} \mid \boldsymbol{z}_{\mathcal{P}_l}, \mathcal{P}_l)] - \mathrm{KL}\left(q\left(\boldsymbol{z}_{\mathcal{P}_l}, \mathcal{P}_l \mid \boldsymbol{x}\right) \| p\left(\boldsymbol{z}_{\mathcal{P}_l}, \mathcal{P}_l\right)\right)$, where $\mathcal{P}_l$ denotes the path in the tree from the root to leaf l which has been followed. A more detailed analysis of the ELBO is omitted, but intuitively, the loss consists of two parts: The first term represents the reconstruction loss, which is characterized by a weighted sum over the reconstruction loss of each leaf, where each weight is the probability that the sample reaches this leaf. For each sample encourages that the leaf with the highest probability has the lowest reconstruction loss, which, combined with the cluster-specific decoders, guides the learning of the clusters. The second part is the Kullback–Leibler divergence (KL), which regularizes the learned embeddings, as well as the routing probabilities.

In the growing step, the leaf with the highest number of assigned samples is split by attaching two new leaves. The new tree is then updated via the model's refinement step. This scheme is repeated until the tree is fully grown. This imposes an inductive bias towards balanced clusters, which is unsuitable for scRNA-seq data where important cell types might be underrepresented. For a more detailed description of TreeVAE, we refer to their work.

3.2. Tree Variational Autoencoders for Single-Cell Data

In this work, we investigate whether TreeVAE can be employed to discover cell subtypes in scRNA-seq data. To do so, we propose scTree, which extends TreeVAE by (a) defining a new reconstruction loss for the new data type, (b) integrating batch information into the architecture, and (c) defining a new splitting criterion.

First, to accommodate for the discrete nature of the data representing read counts, we redefine the loss function. Instead of assuming Bernoulli or Gaussian data, we now assume a Negative Binomial distribution $\boldsymbol{x} \mid \boldsymbol{z}_{\mathcal{P}_l}, \mathcal{P}_l \sim \mathrm{NB}\left(\mu_l\left(\boldsymbol{z}_l\right), \boldsymbol{\theta}\right)$, where the mean is predicted by the leaf-

specific decoder μ_l from the sample-wise latent leaf embeddings $\boldsymbol{z}_l$, while the dispersion parameters are learned per gene and remain the same across leaves. This parameterization encourages that samples in each leaf are supposed to have unique characteristics and, as such, supports meaningful clustering.

A frequent issue in scRNA-seq is the handling of batch effects. While previous hierarchical clustering methods perform batch integration either ante-hoc (Li et al., 2020) or cluster the data at different resolutions (Luecken & Theis, 2019; Hua & Zhang, 2019), the gradient-based nature of the clustering in TreeVAE allows for an end-to-end integration of batch effects into the learning process. As such, we hand the leaf-specific decoders μ_l the batch information as additional input information. Therefore, the learned embeddings $\boldsymbol{z}_l$ do not have to contain this unwanted information, leading to batch-corrected representations and clustering.

An important question for every divisive clustering algorithm is the finding of an adequate criterion that determines which leaf to split. For TreeVAE, the split is performed by splitting the leaf with the highest number of samples falling into it, which encodes an implicit bias towards balanced clusters. While this works for balanced imaging benchmarks, in scRNA-seq data, oftentimes, the target cell types are distributed unevenly. For this reason, we introduce a novel splitting rule based on the reconstruction loss. For each leaf, we grow a proposal subtree with leaves l_1, l_2 for 10 epochs and compute the average difference in reconstruction loss $\|\log p(\boldsymbol{x} \mid \boldsymbol{z}_{\mathcal{P}_{l_1}}, \mathcal{P}_{l_1}) - \log p(\boldsymbol{x} \mid \boldsymbol{z}_{\mathcal{P}_{l_2}}, \mathcal{P}_{l_2})\|_1$ over the dataset it was trained with. The bigger the difference, the more specialized the leaves have become, indicating that a meaningful split has been found. Thus, the proposal subtree with the highest average difference in reconstruction loss is selected and trained for a longer time.

4. Experiments

Datasets and Metrics: We assess the clustering and batch integration capabilities of scTree using seven distinct scRNA-seq datasets. The first dataset from (Ding et al., 2019) consists of peripheral blood mononuclear cells (PBMC) sourced from two healthy donors, which were sequenced on seven different sequencing technologies. In this dataset, the primary challenge lies in harmonizing batches originating from different donors and sequencing technologies. The second dataset is the mouse retinal bipolar neuron dataset (Retina) from (Shekhar et al., 2016) which was also used in (Lopez et al., 2018). Further three dataset, immune human cell dataset (IHC), Pancreas and Lung Atlas, are taken from (Luecken et al., 2022). In the IHC dataset, five cell types are only present in three out of ten batches. On this dataset, we evaluate the method's capability to merge cell types consistently found in all batches without excessively merging cell types that are only present in a few batches. The remaining two datasets consist of malignant cells from cancer patients. Specifically, we utilize a glioblastoma dataset (GBM) from (Neftel et al., 2019) and a squamous cell carcinoma dataset (SCC) from (Ji et al., 2020). In datasets derived from malignant cells, strong patient-specific effects due to genetic differences between cells pose significant challenges for data integration. For a detailed description of the datasets, see Appendix B.

To evaluate the biological meaningfulness of the hierarchical clustering, we compute the Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI), as well as Dendrogram Purity (DP) and Leaf Purity (LP), as defined by (Kobren et al., 2017), using the cell type labels as ground truth. Furthermore, we calculate the NMI of the batch labels and the clustering ($\text{NMI}_{\text{batch}}$) to assess the batch integration performance. We report all metrics for the number of clusters set equal to the number of cell types in each dataset.

Baselines: We compare scTree to three baselines. Firstly, we employ Principal Component Analysis (PCA) (Pearson, 1901) coupled with Ward's Agglomerative clustering (Agg) (Ward, 1963) applied to the first 50 Principal Components (PCs). Additionally, we utilize Agg on the latent representations learned by two commonly used batch integration method for scRNA-seq data: scVI by Lopez et al. (2018) + Agg and LDVAE by Svensson et al. (2020a) + Agg. For all baselines, we used the scikit-learn's (Pedregosa et al., 2011) AgglomerativeClustering implementation with default parameters.

Implementation Details The model architecture of scTree was determined via datasets that are not included in this work to prevent biased results. For consistency, we set the latent dimensions for all VAE-based methods to 10 and report the results for scTree using both the proposed reconstruction-loss-based and the previous sample-count-based splitting rules. A full list of scTree's architecture can be found in Appendix A.

5. Results & Discussion

Hierarchical Clustering Results As evidenced by Table 1, on most datasets scTree performs hierarchical clustering on par or better than the baseline methods. Especially for the tumor datasets, scTree shows promising performance, as evidenced by NMI, as well as DP, where DP also takes the learned hierarchy into account. As the tumor datasets are the ones that pose the most significant challenge regarding batch integration, we interpret these results that scTree performs best when there is a strong need to correct for these patient-specific effects. Incidentally, the clustering of scTree compared to the baselines has a worse batch NMI. This worse numerical performance should be interpreted

Table 1. Hierarchical clustering performances of scTree compared with baselines. Means and standard deviations are computed across 10 runs with different random model initializations. The best-performing methods are bolded.

Dataset	Method	NMI (↑)	ARI (↑)	DP (↑)	LP (↑)	$\text{NMI}_{\text{batch}}$ (↓)
Pancreas	Agg	0.75±0.00	0.52±0.00	0.89±0.00	0.90±0.00	0.36±0.00
	scVI+Agg	0.76±0.01	0.57±0.04	**0.94**±0.01	0.94±0.00	**0.10**±0.01
	ldVAE+Agg	0.77±0.01	0.55±0.03	**0.94**±0.01	**0.95**±0.00	0.15±0.01
	$\text{scTree}_{\#\text{sample}}$	0.75±0.03	0.56±0.08	0.90±0.08	0.91±0.03	0.21±0.03
	$\text{scTree}_{\text{reconstruction}}$	**0.84**±0.04	**0.83**±0.10	0.92±0.07	0.94±0.02	0.13±0.03
Lung Atlas	Agg	0.69±0.00	0.46±0.00	0.54±0.00	0.71±0.00	0.47±0.00
	scVI+Agg	0.69±0.01	0.52±0.03	0.65±0.02	**0.75**±0.01	**0.27**±0.01
	ldVAE+Agg	0.68±0.01	0.50±0.04	0.63±0.03	0.73±0.01	0.30±0.01
	$\text{scTree}_{\#\text{sample}}$	0.70±0.01	0.48±0.02	0.68±0.03	0.74±0.01	0.35±0.01
	$\text{scTree}_{\text{reconstruction}}$	**0.76**±0.01	**0.58**±0.02	**0.69**±0.03	**0.75**±0.01	0.32±0.01
PBMC	Agg	0.55±0.00	0.40±0.00	0.59±0.00	0.74±0.00	0.34±0.00
	scVI+Agg	0.67±0.02	0.56±0.04	0.72±0.02	0.82±0.02	**0.05**±0.00
	ldVAE+Agg	0.69±0.02	0.57±0.04	0.72±0.03	0.82±0.02	**0.05**±0.00
	$\text{scTree}_{\#\text{sample}}$	0.69±0.03	**0.61**±0.06	**0.75**±0.04	**0.83**±0.02	0.06±0.01
	$\text{scTree}_{\text{reconstruction}}$	**0.71**±0.02	0.56±0.08	0.63±0.08	0.73±0.08	0.06±0.01
Retina	Agg	0.79±0.01	0.56±0.05	0.95±0.01	0.89±0.00	0.03±0.00
	scVI+Agg	0.83±0.01	0.55±0.02	0.96±0.01	0.94±0.01	**0.02**±0.00
	ldVAE+Agg	**0.88**±0.01	0.74±0.08	**0.97**±0.01	**0.95**±0.00	**0.02**±0.00
	$\text{scTree}_{\#\text{sample}}$	0.87±0.03	0.86±0.10	**0.97**±0.01	0.91±0.02	0.03±0.00
	$\text{scTree}_{\text{reconstruction}}$	0.86±0.13	**0.87**±0.18	0.96±0.04	0.88±0.14	0.03±0.00
IHC	Agg	0.66±0.00	0.44±0.00	0.68±0.00	0.74±0.00	0.45±0.00
	scVI+Agg	**0.75**±0.02	**0.59**±0.05	**0.81**±0.01	**0.85**±0.02	**0.14**±0.00
	ldVAE+Agg	**0.75**±0.01	0.54±0.03	0.78±0.02	0.83±0.01	**0.14**±0.00
	$\text{scTree}_{\#\text{sample}}$	0.73±0.02	0.53±0.04	**0.81**±0.01	0.83±0.02	0.18±0.02
	$\text{scTree}_{\text{reconstruction}}$	0.68±0.07	0.48±0.12	0.60±0.08	0.69±0.07	0.15±0.01
SCC	Agg	0.36±0.00	0.47±0.00	0.70±0.00	0.74±0.00	0.25±0.00
	scVI+Agg	0.34±0.07	0.42±0.10	0.66±0.08	0.71±0.06	**0.07**±0.02
	ldVAE+Agg	0.46±0.06	0.56±0.10	0.75±0.05	0.77±0.04	0.08±0.03
	$\text{scTree}_{\#\text{sample}}$	0.50±0.09	0.51±0.10	0.80±0.06	0.80±0.05	0.14±0.05
	$\text{scTree}_{\text{reconstruction}}$	**0.56**±0.08	**0.63**±0.11	**0.81**±0.05	**0.81**±0.04	0.11±0.05
GBM	Agg	0.42±0.00	0.42±0.00	0.58±0.00	0.66±0.00	0.37±0.00
	scVI+Agg	0.28±0.03	0.23±0.03	0.43±0.01	0.56±0.03	**0.07**±0.00
	ldVAE+Agg	0.48±0.03	0.44±0.07	0.52±0.05	0.66±0.04	0.15±0.01
	$\text{scTree}_{\#\text{sample}}$	**0.53**±0.04	**0.51**±0.07	**0.66**±0.06	**0.77**±0.06	0.23±0.02
	$\text{scTree}_{\text{reconstruction}}$	0.52±0.04	0.47±0.05	0.60±0.03	0.69±0.05	0.22±0.02

with care, as the ground-truth clustering is not independent of the batch labels as exhibited by non-zero $\text{NMI}_{\text{batch}}$ of celltype labels and batch labels (Pancreas: 0.08, Lung Atlas: 0.37, PBMC: 0.05, Retina: 0.02, IHC: 0.16, SCC: 0.08, GBM: 0.23). This suggests that scVI and LDVAE might over-integrate in some cases. Contrarily, scTree's end-to-end batch integration can separate irrelevant batch effects from the ones correlated with clusters, as scTree's clustering performance remains high despite worse batch integration.

Discovery of Hierarchies Figure 2 (left) shows that early in the hierarchy scTree correctly separates Lymphoid and Myeloid cell types present in both PBMCs and bone marrow datasets. At first, it inaccurately allocates many Myeloid cell types exclusive to bone marrow datasets to the Lymphoid branch but then rectifies this at a lower hierarchy level. Notably, scTree segregates bone marrow exclusive cell types without integrating them with other cell types, as seen in the leftmost subtree of Figure 2 (right). scTree achieves pure leaves for most cell types and only struggles with accurate separation of CD4+ and CD8+ T cells, as well as the small bone-marrow-specific cell types. This is also evident by the root node embedding as shown in Figure 3. This suggests that the encoder is well-suited to accurately split cell types into distinct clusters while not over-integrating batch effects.

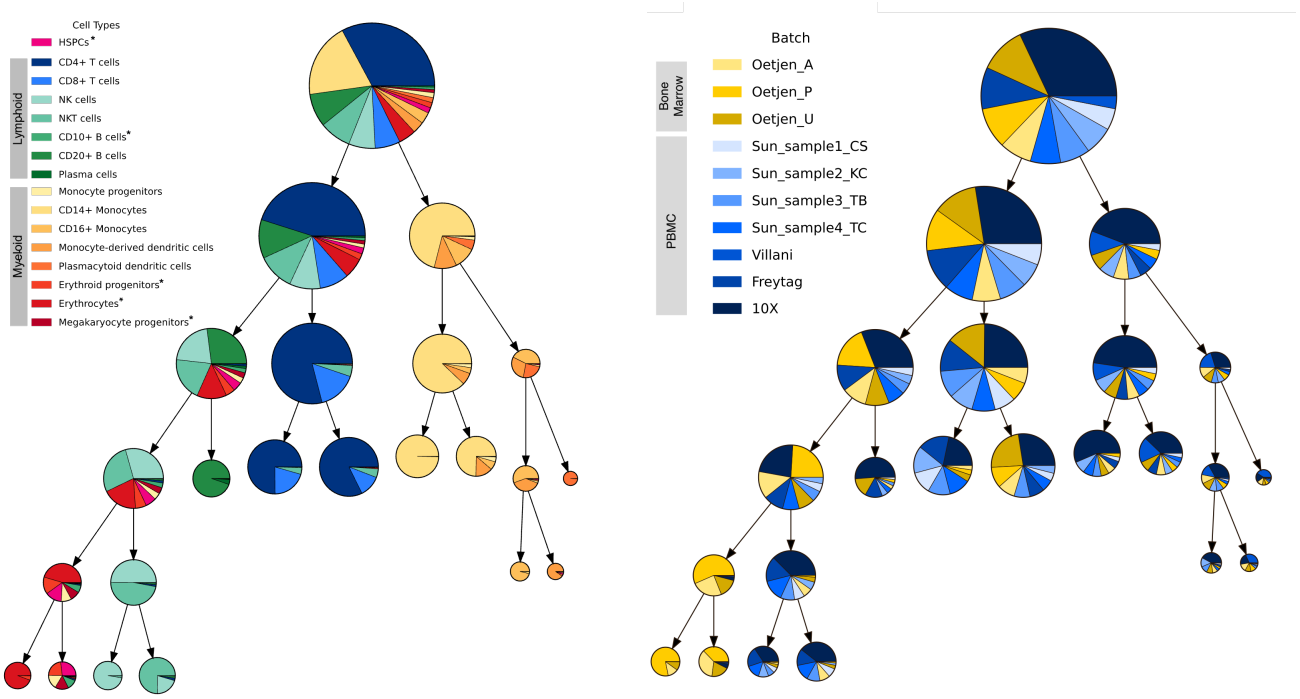

Figure 2. Visualization of hierarchy discovered by scTree on IHC. The size of each node represents the number of cells assigned to it. We excluded empty leaves from the tree. Left: Hierarchy of cell types. Lymphoids, Myeloids and HSPCs have distinct colors. The "*" indicates cell types exclusive to the bone marrow samples. Right: Hierarchy of batches.

6. Conclusion

In this paper, we introduced scTree, a new hierarchical clustering method that directly integrates batch correction techniques into the training to enhance the clustering of scRNA-seq data. We have proposed a novel splitting rule based on the reconstruction loss that detects small clusters. We have shown that scTree performs as well as or better than state-of-the-art methods, especially on data with strong batch effects, such as cancer datasets. We presented qualitatively that scTree learns a biologically plausible hierarchical structure, thereby facilitating the exploration and analysis of scRNA-seq data. To the best of our knowledge, scTree is the first work to explore hierarchical clustering with VAEs jointly with batch integration, where we have highlighted the significant potential of such an approach for exciting advancements in the field.

Limitations & Future Work Having shown that scTree is equipped to discover hierarchical structures, there are still many interesting avenues to explore. Finding a stopping criterion is an exciting question, as the ground-truth number of clusters is usually unknown. Regarding the model architecture, finding a way to reduce the number of hyperparameters to tune would be beneficial. We believe our proposed configuration in Appendix A can serve as a good starting point for this. Similarly, the method is currently restricted to a binary tree, which could be generalized to better represent cell type hierarchies. Furthermore, as each leaf has a separate embedding, there is only the root embedding representing all samples, which hinders simple interpretations of the latent space(s). Lastly, $\text{NMI}_{\text{batch}}$ shows that scTree sometimes does not full correct for batch effects, and regularizing the learned representations more explicitly to prevent this might increase clustering performance even more.

Code Availablility

An implementation of scTree is available at `https://github.com/mvandenhi/sctree-public`. To reproduce all results, we provide `https://github.com/mvandenhi/sctree-supplementary-public`.

Acknowledgements

MV is supported by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract number MB22.00047. LM is supported by the SDSC PhD Fellowship #1-001568-037. FB is supported by the Swiss National Science Foundation (SNSF) (grant number 205321_207931).

References

Blei, D. M., Jordan, M. I., Griffiths, T. L., and Tenenbaum, J. B. Hierarchical topic models and the nested chinese restaurant process. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, NIPS'03, pp. 17–24, Cambridge, MA, USA, 2003. MIT Press.

Ding, J., Adiconis, X., Simmons, S. K., Kowalczyk, M. S., Hession, C. C., Marjanovic, N. D., Hughes, T. K., Wadsworth, M. H., Burks, T., Nguyen, L. T., et al. Systematic comparative analysis of single cell rna-sequencing methods. *BioRxiv*, pp. 632216, 2019.

Eraslan, G., Drokhlyansky, E., Anand, S., Fiskin, E., Subramanian, A., Slyper, M., Wang, J., Van Wittenberghe, N., Rouhana, J. M., Waldman, J., et al. Single-nucleus cross-tissue molecular reference maps toward understanding disease gene function. *Science*, 376(6594):eabl4290, 2022.

Ferreira, P. F., Kuipers, J., and Beerenwinkel, N. Deep exponential families for single-cell data analysis. *bioRxiv*, 2022. 10.1101/2022.10.15.512383.

Goyal, P., Hu, Z., Liang, X., Wang, C., Xing, E. P., and Mellon, C. Nonparametric variational auto-encoders for hierarchical representation learning. *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5104–5112, 2017.

Grønbech, C. H., Vording, M. F., Timshel, P. N., Sønderby, C. K., Pers, T. H., and Winther, O. scvae: variational auto-encoders for single-cell gene expression data. *Bioinformatics*, 36(16):4415–4422, 2020.

Heller, K. A. and Ghahramani, Z. Bayesian hierarchical clustering. In *Proceedings of the 22nd international conference on Machine learning*, pp. 297–304, 2005.

Hua, K. and Zhang, X. A case study on the detailed reproducibility of a human cell atlas project. *Quantitative Biology*, 7:162–169, 2019.

Ji, A. L., Rubin, A. J., Thrane, K., Jiang, S., Reynolds, D. L., Meyers, R. M., Guo, M. G., George, B. M., Mollbrink, A., Bergenstråhle, J., et al. Multimodal analysis of composition and spatial architecture in human squamous cell carcinoma. *Cell*, 182(2):497–514, 2020.

Jiang, H., Sohn, L. L., Huang, H., and Chen, L. Single cell clustering based on cell-pair differentiability correlation and variance analysis. *Bioinformatics*, 34(21):3684–3694, 2018.

Kaufman, L. and Rousseeuw, P. J. *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009.

Kiselev, V. Y., Andrews, T. S., and Hemberg, M. Challenges in unsupervised clustering of single-cell rna-seq data. *Nature Reviews Genetics*, 20(5):273–282, 2019.

Kobren, A., Monath, N., Krishnamurthy, A., and McCallum, A. A hierarchical algorithm for extreme clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pp. 255–264. ACM, 2017. 10.1145/3097983.3098079. URL https://doi.org/10.1145/3097983.3098079.

Lähnemann, D., Köster, J., Szczurek, E., McCarthy, D. J., Hicks, S. C., Robinson, M. D., Vallejos, C. A., Campbell, K. R., Beerenwinkel, N., Mahfouz, A., et al. Eleven grand challenges in single-cell data science. *Genome biology*, 21(1):1–35, 2020.

Li, X., Wang, K., Lyu, Y., Pan, H., Zhang, J., Stambolian, D., Susztak, K., Reilly, M. P., Hu, G., and Li, M. Deep learning enables accurate clustering with batch effect removal in single-cell rna-seq analysis. *Nature communications*, 11(1):2338, 2020.

Lin, P., Troup, M., and Ho, J. W. Cidr: Ultrafast and accurate clustering through imputation for single-cell rna-seq data. *Genome biology*, 18(1):1–11, 2017.

Lopez, R., Regier, J., Cole, M. B., Jordan, M. I., and Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018.

Lotfollahi, M., Naghipourfar, M., Luecken, M. D., Khajavi, M., Büttner, M., Wagenstetter, M., Avsec, Ž., Gayoso, A., Yosef, N., Interlandi, M., et al. Mapping single-cell data to reference atlases by transfer learning. *Nature biotechnology*, 40(1):121–130, 2022.

Luecken, M. D. and Theis, F. J. Current best practices in single-cell rna-seq analysis: a tutorial. *Molecular systems biology*, 15(6):e8746, 2019.

Luecken, M. D., Büttner, M., Chaichoompu, K., Danese, A., Interlandi, M., Müller, M. F., Strobl, D. C., Zappia, L., Dugas, M., Colomé-Tatché, M., et al. Benchmarking atlas-level data integration in single-cell genomics. *Nature methods*, 19(1):41–50, 2022.

Manduchi, L., Vandenhirtz, M., Ryser, A., and Vogt, J. E. Tree variational autoencoders. In *Advances in Neural Information Processing Systems*, 2023.

Morelli, L., Giansanti, V., and Cittaro, D. Nested stochastic block models applied to the analysis of single cell data. *BMC bioinformatics*, 22(1):1–19, 2021.

Murtagh, F. and Contreras, P. Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1): 86–97, 2012.

Neftel, C., Laffy, J., Filbin, M. G., Hara, T., Shore, M. E., Rahme, G. J., Richman, A. R., Silverbush, D., Shaw, M. L., Hebert, C. M., et al. An integrative model of cellular states, plasticity, and genetics for glioblastoma. *Cell*, 178(4):835–849, 2019.

Nistér, D. and Stewénius, H. Scalable recognition with a vocabulary tree. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2:2161–2168, 2006.

Osumi-Sutherland, D., Xu, C., Keays, M., Levine, A. P., Kharchenko, P. V., Regev, A., Lein, E., and Teichmann, S. A. Cell type ontologies of the human cell atlas. *Nature cell biology*, 23(11):1129–1135, 2021.

Pearson, K. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11): 559–572, 1901.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Shekhar, K., Lapan, S. W., Whitney, I. E., Tran, N. M., Macosko, E. Z., Kowalczyk, M., Adiconis, X., Levin, J. Z., Nemesh, J., Goldman, M., et al. Comprehensive classification of retinal bipolar neurons by single-cell transcriptomics. *Cell*, 166(5):1308–1323, 2016.

Shin, S.-J., Song, K., and Moon, I.-C. Hierarchically clustered representation learning. In *AAAI Conference on Artificial Intelligence*, 2019.

Sikkema, L., Strobl, D. C., Zappia, L., Madissoon, E., Markov, N. S., Zaragosi, L.-E., Ansari, M., Arguel, M.-J., Apperloo, L., Becavin, C., et al. An integrated cell atlas of the human lung in health and disease. *bioRxiv*, pp. 2022–03, 2022.

Sneath, P. H. The application of computers to taxonomy. *Microbiology*, 17(1):201–226, 1957.

Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K., and Winther, O. Ladder variational autoencoders. *Advances in neural information processing systems*, 29, 2016.

Steinbach, M. S., Karypis, G., and Kumar, V. A comparison of document clustering techniques. *Department of Computer Science and Engineering, University of Minnesota*, 2000.

Svensson, V., Gayoso, A., Yosef, N., and Pachter, L. Interpretable factor models of single-cell rna-seq via variational autoencoders. *Bioinformatics*, 36(11):3418–3421, 2020a.

Svensson, V., Gayoso, A., Yosef, N., and Pachter, L. Interpretable factor models of single-cell RNA-seq via variational autoencoders. *Bioinformatics*, 36(11):3418–3421, 03 2020b. ISSN 1367-4803.

Vikram, S., Hoffman, M. D., and Johnson, M. J. The loracs prior for vaes: Letting the trees speak for the data. *ArXiv*, abs/1810.06891, 2018.

Ward, J. H. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58:236–244, 1963.

Williams, C. A mcmc approach to hierarchical mixture modelling. *Advances in Neural Information Processing Systems*, 12, 1999.

Wolf, F. A., Angerer, P., and Theis, F. J. Scanpy: large-scale single-cell gene expression data analysis. *Genome biology*, 19:1–5, 2018.

Zou, Z., Hua, K., and Zhang, X. Hgc: fast hierarchical clustering for large-scale single-cell data. *Bioinformatics*, 37(21):3964–3965, 2021.

Žurauskienė, J. and Yau, C. pcareduce: hierarchical clustering of single cell transcriptional profiles. *BMC bioinformatics*, 17:1–11, 2016.

A. Model Architecture

As TreeVAE, and therefore also scTree, has a large number of important hyperparameters that need to be tuned, we use a suitable configuration, which is adjusted for the simpler nature of the input data such that it clusters scRNA-seq data in a meaningful way. All experiments were performed on datasets that are not presented in Section 4 to not bias the results. The hyperparameters and their determined values are presented in Table 2.

Latent dimensions	10
Bottom-up latent dimensions	128
Encoder	Linear layer + Batchnorm + LeakyReLU
Decoder	Linear layer
Transformations	1 Hidden Layer
Routers	1 Hidden Layer
Kl-annealing	Linear from 0.001 to 1
Subtree training epochs	100
Intermediate finetuning epochs	50
Final finetuning epochs	50
Batch size	128
Optimizer	Adam
Learning rate	0.001
Weight decay	0.00001

Table 2. Hyperparameter configuration of scTree

B. Datasets

All datasets used in this paper are publicly available. For all datasets and methods we used the 4000 most highly variable genes computed with scanpy's highly_variable_genes (Wolf et al., 2018) function with default parameters. We used the same preprocessing as proposed by the authors for each dataset and removed all cells not assigned to a celltype.

IHC: The IHC dataset contains 33,506 cells from 10 different batch from five different studies with 16 unique cell types. The whole dataset is available under https://doi.org/10.6084/m9.figshare.12420968.v8.

PBMC: The PBMC datasets consists of 30,449 cells from two healthy donors sequenced with six different sequencing technologies (10x Chromium (v2), 10x Chromium (v3), CEL-Seq2, Drop-seq, Seq-Well, and inDrops). Since LDVAE cannot take additional categorical covariates as input, we generated a new batch column from the donor ID and the sequencing technology and used it for all methods. The dataset is available at https://singlecell.broadinstitute.org/single_cell/study/SCP424/single-cell-comparison-pbmc-data.

Pancreas: The Pancreas dataset consists of 16,382 cells, 9 batches and 14 cell types. The dataset is available at https://figshare.com/ndownloader/files/24539828.

Lung Atlas: The Lung Atlas dataset consists of 32,426 cells, 16 batches and 16 cell types. The dataset is available at https://figshare.com/ndownloader/files/24539942.

Retina: The Retina dataset consists of 19,829 cells, 2 batches and 15 cell types. The dataset is available at https://github.com/broadinstitute/BipolarCell2016.

GBM: The GBM dataset contains 6,855 malignant cells from 27 different patient. The authors annotated four cellular stats (AC-like, MES-like, NPC-like and OPC-like) describing intra-patient heterogeneity. We removed all cycling cells from the dataset to ensure that we only retain cells assigned to one of the stats. All samples were sequenced using Smart-seq 2. The dataset is available at https://singlecell.broadinstitute.org/single_cell/study/SCP393/single-cell-rna-seq-of-adult-and-pediatric-glioblastoma.

SCC: The SCC dataset encompasses 10,529 malignant cells from 8 patients. The authors assigned cells to three cellular stats (Basel, Differentiated and TSK). We again removed all cycling cells. The dataset is available at https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE144240.

C. Visualization of the IHC dataset

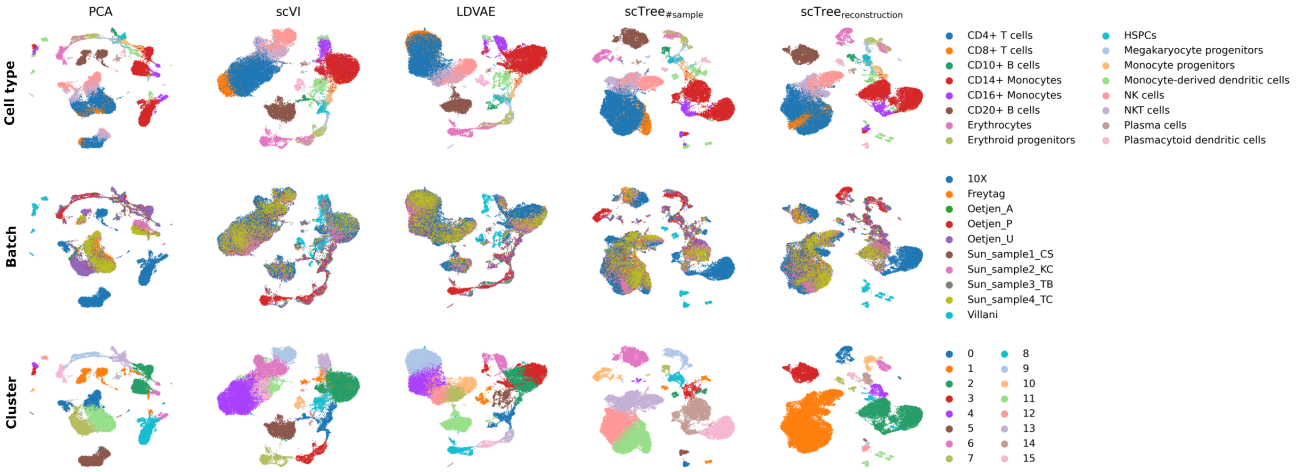

Figure 3. The plots show uniform manifold approximation and projections based on the first 50 PCs computed on the log-transformed normalized gene expression, the latent representations of scVI and LDVAE, and the Root node representation of scTree with both splitting rules. The plots are colored by cell type (top), batch (middle), and cluster (bottom).