Extending the scVAE Framework: A Hierarchical Mixture-of-Mixtures Approach for Single-Cell Gene Expression Data

Raphaël RUBRICE, Adam KEDDIS, Tiffney AINA

Probabilistic Graphical Models and Deep Generative Models, MVA

November 2025

1. Motivation

Single-cell RNA sequencing (scRNA-seq) enables detailed study of cellular heterogeneity, yet analysis remains difficult because gene-expression data are high-dimensional, sparse, and overdispersed. Biological organization is inherently hierarchical,broad immune lineages subdivide into finer subtypes and activation states,whereas most standard pipelines rely on sequential, hand-crafted preprocessing and clustering steps. A probabilistic model that can learn these multi-scale structures directly from raw counts is therefore desirable. Variational auto-encoders (VAEs) [4] provide such a framework by combining a generative decoder with an amortized inference network. In single-cell applications, models like scVI [7] and scVAE [2] use Negative Binomial likelihoods to capture count overdispersion, enabling end-to-end modeling of expression data. However, these architectures employ a single flat mixture prior, assuming that all heterogeneity can be captured by one discrete latent variable,a limitation for modeling biological hierarchies.

To address this, we extend scVAE in two complementary directions. The Independent Mixture-of-Mixtures VAE (IndMoMVAE) introduces multiple unlinked mixture branches, each learning a distinct data partition, while the Hierarchical Mixture-of-Mixtures VAE (MoMixVAE) adds explicit dependencies between mixture levels to represent coarse-to-fine organization. Trained on Peripheral Blood Mononuclear Cell (PBMC) data these models leverage initialization via PCA-KMeans, KL warm-up, and marginal-usage regularization to ensure stability. Together, they test whether hierarchical coupling in latent space improves clustering coherence and interpretability beyond standard mixture VAEs.

2. Introduction

A. Variational Auto-Encoders. Variational auto-encoders (VAEs) [4] provide a probabilistic framework for learning low-dimensional representations of high-dimensional data. A latent variable $z \in \mathbb{R}^d$ is drawn from a prior $p(z) = \mathcal{N}(0, I)$, and a decoder network parameterizes the conditional likelihood $p_\theta(x|z)$. The joint distribution is therefore

$$p_\theta(x, z) = p_\theta(x|z)p(z).$$

Since the posterior $p_\theta(z|x)$ is intractable, VAEs introduce an inference network $q_\phi(z|x)$ (the encoder) to approximate it, typically as a Gaussian with mean and variance predicted from x.

Training maximizes the evidence lower bound (ELBO) on the log-likelihood,

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \mathrm{KL}(q_\phi(z|x)\|p(z)),$$

balancing accurate reconstruction of the data with regularization toward the prior through the KL divergence term. Posterior collapse can occur when the decoder ignores the latent code ($q_\phi(z|x) \approx p(z)$), causing the model to behave like a deterministic autoencoder. To prevent this, a *KL warm-up* schedule gradually increases the KL weight β_t from 0 to 1 during early training, delaying strong regularization until the reconstruction term stabilizes. This foundational formulation underlies scVAE and the hierarchical extensions developed in this work.

B. Mixture Models and Structured Latents. While standard VAEs assume a single unimodal prior $p(z) = \mathcal{N}(0, I)$, many datasets contain distinct subpopulations requiring a multimodal latent structure. Gaussian Mixture VAEs (GMVAEs) [1] address this by introducing a discrete latent variable $y \in \{1, \dots, K\}$ and defining a mixture prior

$$p_\theta(x, y, z) = p_\theta(x|z)\, p_\theta(z|y)\, p_\theta(y),$$
$$p_\theta(y) = \mathrm{Cat}(\pi), \qquad p_\theta(z|y = k) = \mathcal{N}(\mu_k, \sigma_k^2 I).$$

where each component k represents a cluster in the latent space. This structure allows the model to perform unsupervised clustering through generative modeling and provides interpretable discrete representations of data.

The scVAE model [2] extends this idea to single-cell gene-expression counts by coupling the GMVAE latent formulation with a Negative Binomial likelihood, thereby enabling joint modeling of raw counts and cell-type clusters. Subsequent hierarchical VAE frameworks [9, 5] stack multiple latent layers to capture increasingly abstract factors of variation. These ideas collectively motivate our hierarchical Mixture-of-Mixtures (MoMixVAE) formulation, which extends the flat mixture prior of scVAE into a multi-level hierarchy capable of representing coarse-to-fine cellular structure.

C. Modeling gene count data. Single-cell RNA-sequencing (scRNA-seq) yields non-negative integer counts that are sparse and highly overdispersed, with variance substantially exceeding the mean. Early

VAE models used Gaussian decoders, which are inappropriate for discrete count data. Both scVI [7] and scVAE [2] replace this assumption with count-based likelihoods,primarily the *Negative Binomial* (NB) or its zero-inflated variant (ZINB),to better capture gene-expression variability.

In scVAE, each observation x_n (gene n in a given cell) is modeled as

$$x_n \sim \mathrm{NB}(r_n, p_n)\,,$$
$$\mathbb{E}[x_n] = r_n \,\frac{1-p_n}{p_n},$$
$$\mathrm{Var}[x_n] = r_n \,\frac{1-p_n}{p_n^2}.$$

where r_n controls dispersion and p_n is the success probability. The decoder network $f_\theta(z)$ predicts gene-wise mean and dispersion parameters through neural outputs $\mu_\theta(z)$ and $r_\theta(z)$, allowing the expectation of each gene's expression to depend smoothly on the latent variable z. Optionally, a dropout probability $\pi_\theta(z)$ extends the model to a zero-inflated NB:

$$p_\theta(x|z) = \pi_\theta(z)\,\delta_0(x) + (1-\pi_\theta(z))\,\mathrm{NB}(x;\, r_\theta(z), p_\theta(z)).$$

This parameterization enables the network to explain both biological overdispersion and technical variability while remaining fully differentiable. Our models adopt the same NB likelihood formulation to maintain methodological consistency with scVAE.

3. Reference Method

A. scVAE: Architecture and Likelihood Modeling. The *scVAE* framework [2] applies a VAE to model scRNA-seq gene expression directly from raw count data. Unlike traditional methods that rely on normalization or log-transformation, scVAE treats the observed expression counts as samples from discrete probability distributions that naturally capture the stochastic nature of transcriptional activity.

Architecture. The encoder network, parameterized by ϕ, maps each cell's high-dimensional count vector $\mathbf{x} \in \mathbb{N}^G$ to the parameters of a latent Gaussian distribution, $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_\phi(\mathbf{x}), \boldsymbol{\sigma}^2_\phi(\mathbf{x})I)$, where $\mathbf{z}$ represents a low-dimensional latent embedding capturing biological variability across cells. The decoder, parameterized by θ, maps latent variables back to the expected gene expression levels through a generative model $p_\theta(\mathbf{x}|\mathbf{z})$. This probabilistic mapping allows scVAE to reconstruct the input count profiles while learning structured latent representations.

Likelihood modeling. The choice of likelihood distribution is critical for modeling overdispersed and zero-inflated count data common in scRNA-seq experiments. scVAE explores several likelihood functions, including the Poisson, the Negative Binomial (NB), and the Zero-Inflated Negative Binomial (ZINB) distributions. The NB distribution introduces a dispersion parameter that decouples the variance from the mean, enabling the model to account for cell-to-cell variability beyond Poisson noise. The ZINB variant further incorporates an additional zero-inflation parameter to handle the high frequency of zero counts in sparse expression matrices.

Variational objective. Model training maximizes the evidence lower bound (ELBO):

$$\mathcal{L}(\theta, \phi; \mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z})), \quad [1]$$

where $p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, I)$ is the prior over latent variables, and D_{KL} denotes the Kullback–Leibler divergence. Maximizing $\mathcal{L}$ corresponds to jointly learning expressive latent encodings and likelihood parameters that best explain the observed count profiles. By leveraging the NB-based likelihoods, scVAE provides a biologically grounded generative model that captures the overdispersion and sparsity inherent in scRNA-seq data.

4. Proposed Methods

In this section we present the methods we developed as extensions to the scVAE framework.

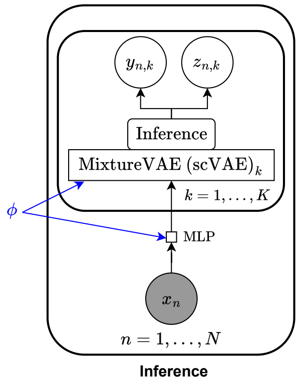

Fig. 1. Probabilistic graphical model of the Independent Mixture of Mixture (IndMoMVAE). In the figure, the MixtureVAE (scVAE) node denotes the scVAE model.

A. IndMoMVAE (Independent Mixture-of-Mixture VAE). Conceptually, this model can be interpreted as multiple independent mixture views of the data, each representing a distinct clustering of the observations. This formulation serves as an ablation study for hierarchical designs, allowing us to isolate the effect of independent clustering branches without inter-level dependency.

For the IndMoMVAE model, we formulate the generative process as:

$$p(x) = \sum_k \left[\int_z \left(\sum_y p(x|z)\, p(z|y)\, p(y|k)\, p(k) \right) dz \right] \quad [2]$$

This formulation allows us to model a varying number of clusters at each hierarchical level and naturally supports hierarchical clustering structures. Each clustering level k can represent a different granularity or perspective on the data distribution.

Importantly, in this framework, each clustering branch is independent and does not incorporate information from previous levels. Consequently, this independence implies that clustering at a given level is uninformed by the structure discovered at earlier levels, which provides a baseline comparison. We anticipate that this will result in less informed clustering relative to models like MoMixVAE, where hierarchical dependencies explicitly guide the clustering process and allow richer representations.

This independent multi-branch design highlights the importance of inter-level communication in hierarchical mixture models, as dependencies can enhance clustering coherence across levels.

B. MoMixVAE (Hierarchical Mixture-of-Mixtures VAE). We introduce MoMixVAE, a hierarchical variational autoencoder that, in contrast to IndMoMVAE, explicitly models the dependencies between clusterings at different granularities. For this modeling approach we use categorical latent variables $y^{(1)} \sim Cat(\pi^{(1)})$ at level 1 (highest level of the hierarchy) and $y^{(k)} \sim p(y^{(l)}|y^{(l-1)}, \dots, y^{(1)})$ for level $l > 1$. These variables allow the definition of a mixture of mixture in the sense that the latent variable z is such that:

$$z \sim \sum_{k_1=1}^{K_1} \pi_{k_1} \left(\dots \left(\sum_{k_{L-1}=1}^{K_{L-1}} \pi_{k_{L-1}} \left(\sum_{k_L=1}^{K_L} \pi_{k_L} Law(\theta) \right) \right) \dots \right) \quad [3]$$

Using this formalism we define smooth hierarchies where there is no notion of strict, rigid ascendance between levels like in classical hierarchical clustering, instead, the joint probability allows the definition of probabilities of ascendance. The rationale for this extension is that the dependency structure should allow consistent and coherent clustering across hierarchical levels, resulting in nested cluster assignments that reflect a coarse-to-fine organization of the data.

We present the associated graphical model, focusing on the inference procedure since the generative model follows the original framework by the authors. The joint distribution over observed data x, latent representation z, and hierarchical cluster labels $\{y_l\}_{l=1}^{L}$ is factorized as:

$$\begin{aligned} p(x, z, y_1, \dots, y_L) = {} & p(x \mid z)\, p(z \mid y_L, \dots, y_1) \\ & \times p(y_L \mid y_{L-1}, \dots, y_1) \\ & \cdots \\ & \times p(y_2 \mid y_1)\, p(y_1) \end{aligned} \quad [4]$$

This factorization captures the hierarchical dependencies of cluster assignments, where cluster labels at each level condition on those from all preceding, coarser levels.

To approximate the intractable true posterior, we employ a structured variational posterior that respects the hierarchical dependencies:

$$p(z|y_L, ..., y_1) \approx q_\theta(z|x, y_L, ..., y_1) \quad [5]$$

$$p(y_L|y_{L-1}, ..., y_1) \approx q_\theta(y_L|x, y_{L-1}, ..., y_1) \quad [6]$$

$$p(y_1) \approx q_\theta(y_1|x) \quad [7]$$

By explicitly modeling these conditional distributions, the posterior captures nested clustering structures from coarse to fine granularities.

We optimize a full hierarchical evidence lower bound (ELBO) that sums KL divergences across all levels, thereby jointly learning the latent variables and cluster assignments. To stabilize training across multiple KL terms, each potentially operating on different scales, we incorporate β-scaling coefficients and regularization strategies that mitigate posterior collapse and encourage meaningful information flow through the hierarchy.

Due to the computational complexity inherent in jointly inferring multiple dependent clusterings, we restrict our experiments to four hierarchical levels, which align naturally with the biological hierarchy present in our data.

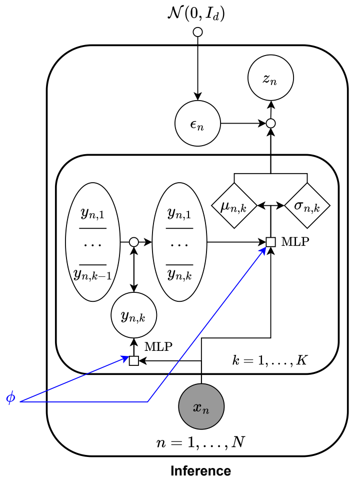

Fig. 2. Probabilistic graphical model of the Hierarchical Mixture of Mixture (MoMixVAE). The ellipsoid-shaped variables correspond to the concatenated categorical latent variables

C. Training and optimization details. To ensure stable optimization, we employ a KL warm-up strategy, as done by the authors, in which the weight of the KL divergence term, β_{KL}. This allows to balance the scale of the KL term with that of the reconstruction to ensure that the strongest signal remains reconstruction. The initial value is set to $\beta_{\text{KL}} = 1$. The authors used a batch size of 100, 200 warmup out of 500 epochs, which, on the full PBMC dataset, corresponds to a warmup to total batch steps ratio of 0.0007. By applying this ratio on our settings led us to use 5 epoch for warmup.

In addition, a marginal regularization term weighted by λ_{marg} is introduced. The coefficient λ_{marg} is kept constant during the first half of training and subsequently decays down to 10% of its initial value following a cosine annealing schedule. This strategy prevents early component collapse while allowing greater flexibility in later training stages. Optimization is performed using the Adam optimizer with a learning rate of 10^{-3}. Training is carried out for 100 epochs with a batch size of 128. To select the model achieving the best generalization performance, early stopping based on the validation loss is used, with a fixed patience of 20 epochs.

At each epoch, the total loss and its individual components,reconstruction loss, latent KL divergence, cluster KL divergence, and marginal KL divergence,are averaged across all mini-batches and recorded for comparison across runs.

For Mixture-of-Mixtures models, models are instantiated to assume $2, 4, 5, 9$ mixture components respectively and uses a latent space of dimension 20. The encoder and decoder are implemented as multilayer perceptrons. For the IndMoMVAE, an additional multilayer perceptron is applied to the input data.

Multilayer perceptrons used in our models are 2 layers deep and 100 hidden units wide with ReLU activation function. To prevent all components from being projected on the same region in latent space, the posterior of latent variables is initialized using PCA and KMeans. Cluster centroids are defined as means and in-cluster variance used as variance.

D. Evaluation Metrics. To quantitatively assess model performance, we rely on several complementary metrics.

First, we report the importance-weighted autoencoder (IWAE) log-likelihood estimate, which provides a tighter lower bound on the marginal log-likelihood than the standard ELBO and serves as an indicator of generative and inference quality (higher IWAE values correspond to a better approximation of the true data distribution).

Second, As we have labels, we can also use the F1 score to assess clustering quality after hungarian alignment of the model's clusters with true labels. This allows us to investigate whether the latent representations and derived clusters are discriminative in a way that matches true labels.

Finally, we also evaluate clustering performance using the Adjusted Rand Index (R_{adj} or ARI). ARI measures the agreement between predicted cluster and true labels with a correction for chance, making it reliable for comparing clustering across different model. Values close to 1 indicate perfect alignment, while values close to 0 correspond to random assignments and below 0 indicate discordance.

Details about metrics are given in Appendix A.

In addition to these metrics, we use low-dimensional projections (UMAP or t-SNE) of the latent space to assess the quality.

To ensure robust model comparison, each metric is computed multiple times for every run. This repeated evaluation allows us to estimate confidence intervals for all scores. These confidence intervals are subsequently used to guide model selection.

5. Data

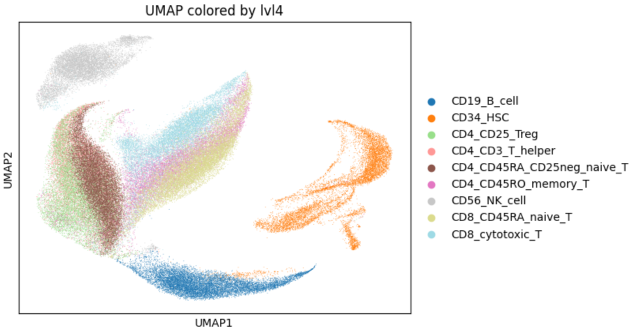

Fig. 3. UMAP of annotated PBMC cells

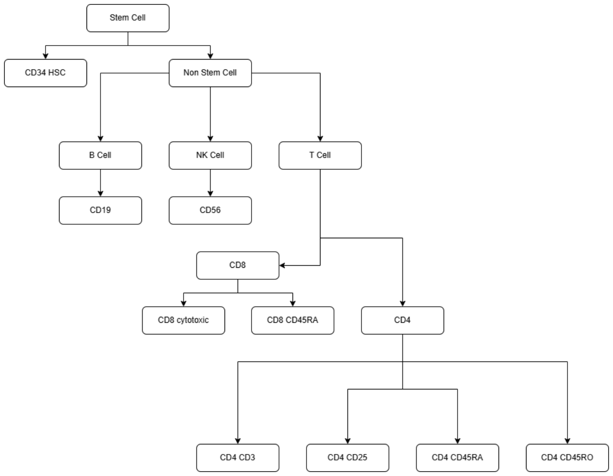

Fig. 4. PBMC Cell Hierarchy. The hierarchical relationships between major lineages and representative subtypes are shown from left to right

A. PBMC Data Processing and Hierarchical Label Creation. Gene expression profiles are intrinsically hierarchical: broad immune lineages subdivide into finer subtypes, each with distinct transcriptional programs. Because the original scVAE paper includes only flat cell-type labels, we developed a custom hierarchical labeling scheme informed by immunological literature and implemented it using `Scanpy`. It comprised four annotation levels corresponding to biological lineage depth: (Level 1) developmental origin distinguishing stem versus non-stem cells; (Level 2) major immune lineage (*e.g.*, T, B, NK cells); (Level 3) intermediate sub-lineage (*e.g.*, CD4$^+$ T or CD8$^+$ T); and (Level 4) fine subtype or activation state (*e.g.*, regulatory T, naive T, cytotoxic T). Each annotation was stored as both a categorical and integer-coded column within the `AnnData` object, ensuring compatibility with the hierarchical mixture models that require fixed-depth label tensors. The modular PBMC pipeline: downloading, loading, combination, and PyTorch adaptation scriptss, handles nine raw datasets and produces harmonized sharded files and `DataLoader` instances reproducible in both local and Colab environments.

Due to time and computational constraints, only 10% of the original dataset is used for training, by subsampling cells and the 5,000 selected most highly variable genes leading to a dataset of 500 genes across 9, 200 cells.

6. Results

A. Reimplementing scVAE and Comparative Evaluation.

Figure 5 compares our re-implementation of scVAE with the proposed IndMoMVAE and MoMixVAE models across Adjusted Rand Index (ARI), Weighted F1-score, and IWAE.

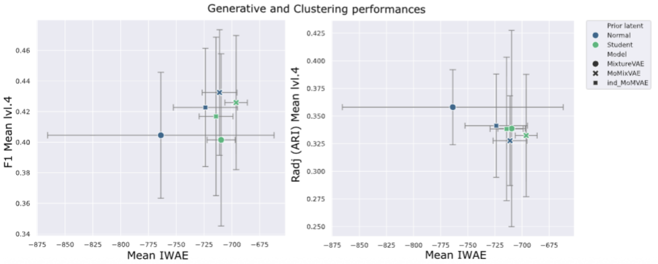

Fig. 5. Latent space projection at low dimensionality for IndMoMVAE. The model uncovers complementary partitions, though some biologically similar cell types remain less distinctly separated compared to MoMixVAE.

Latent prior effect. Models using a Student's-t prior (green) consistently achieve higher, less-negative IWAE values, confirming that heavy-tailed priors improve generative modeling of overdispersed single-cell counts.

Clustering stability. ARI varies widely across runs, especially for MixtureVAE and MoMixVAE, indicating metric sensitivity to manifold geometry. Weighted F1 scores are far more stable, showing that cell-type identities are learned reliably even when cluster geometry shifts.

Architecture trends. MoMixVAE with a Student's-t prior achieves the best balance of generative quality (high IWAE) and accurate clustering (high, low-variance F1). IndMoMVAE underperforms slightly, implying that hierarchical coupling aids representation learning.

Overall, the extended models reproduce the expected trends of scVAE while demonstrating that Student's-t latents yield more robust, biologically meaningful structure.

B. MoMix Cluster Alignment.

Figure 6 demonstrates that MoMixVAE preserves hierarchical consistency between true biological labels and model-predicted clusters across all four levels. At Level 1, the model cleanly separates Stem and Non-stem lineages. Levels 2 and 3 progressively refine this structure, resolving major immune groups (B, NK, and T cells) and further dividing T cells into CD4 and CD8 subsets. At Level 4, the latent space captures nine terminal subtypes with high spatial correspondence, showing that MoMixVAE successfully propagates biological identity from broad lineages to fine-grained cellular states.

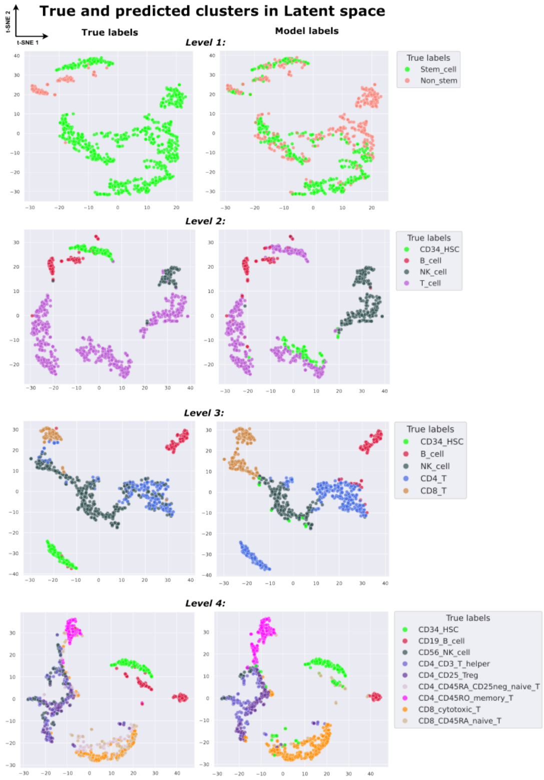

Fig. 6. Hierarchical alignment between true and predicted labels. Each column pair compares the true biological identities (left) with MoMixVAE-predicted clusters after hungarian alignment (right) across the four hierarchy levels.

C. Hierarchical Latent Organization and Nested Clustering.

Figure 7 illustrates how MoMixVAE captures biological hierarchy through a nested, coarse-to-fine latent structure. Each row corresponds to a hierarchy level, showing a progressive refinement of cell-type identity. At Level 1, the model separates broad lineages such as Stem and Non-stem cells; Levels 2 and 3 resolve major immune lineages (B, NK, and T cells) and further distinguish CD4 and CD8 T-cell subtypes. Level 4 captures fine-grained cellular states, including memory and naive T-cells, revealing a structured, biologically coherent organization.

Across levels, clusters remain nested within their parent lineages, ensuring hierarchical consistency and explaining MoMixVAE's stable F1-scores despite ARI variability.

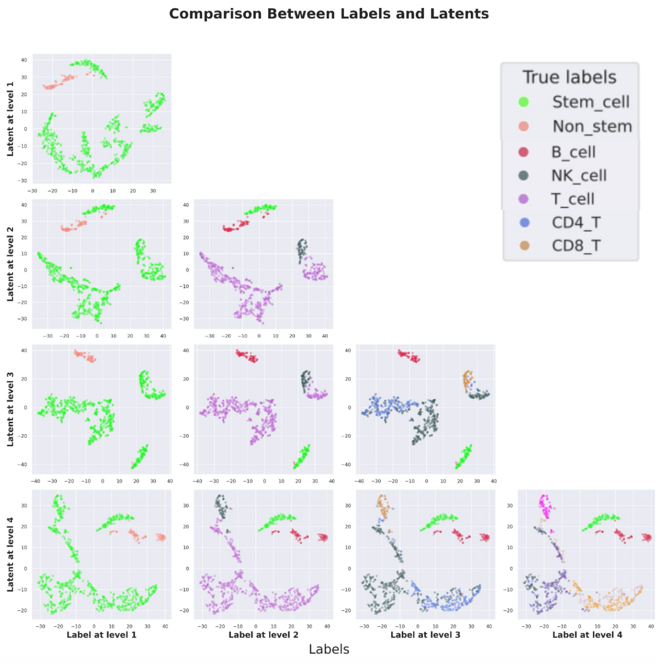

Fig. 7. Hierarchical organization of the learned latent space. Each row shows the latent representation at a given level compared with cell-type labels from lower levels. The latent space progressively refines from broad lineages to fine-grained subtypes, revealing a nested, biologically coherent hierarchy.

Adding hierarchical depth increases the representational granularity of the latent space. In IndMoMVAE, independent branches learn complementary but uncoordinated partitions, whereas MoMixVAE enforces parent–child dependencies, producing more interpretable and coherent hierarchies. For example, cell types such as CD34-HSC and CD19-B-cells remain challenging to separate due to biological similarity, yet the hierarchical prior preserves their relative positioning across levels.

Overall, these results show that hierarchical latent modeling—particularly in MoMixVAE—improves the interpretability, stability, and biological relevance of single-cell representations.

7. Discussion

In this work, we extended the scVAE framework to model hierarchical structure within single-cell transcriptomic data. Through our independent and hierarchical mixture extensions, we explored whether explicit coupling across latent levels improves representation quality and biological interpretability. Our experiments show that hierarchical coupling and heavy-tailed priors substantially enhance generative quality and clustering stability compared to the flat scVAE baseline. In particular, the MoMixVAE architecture demonstrated the most balanced trade-off between reconstruction fidelity and biological coherence across cell-type hierarchies.

Limitations MoMixVAE's ELBO requires marginalizing over all joint cluster assignments $(y^{(1)}, \dots, y^{(L)})$, resulting in $\mathcal{O}(K^L)$ complexity per data point. While tractable for shallow hierarchies ($L = 2$ or 3), this becomes prohibitive for deep taxonomies. Approximate inference strategies such as Gumbel-Softmax relaxation [3] or structured variational families could reduce this burden. Fine-level clusters often collapsed to a subset of components despite balanced coarse-level mixtures, as the hierarchical KL penalty does not directly constrain marginal usage $\bar{q}(y^{(\ell)}) = \mathbb{E}_x[q(y^{(\ell)}|x)]$ at each level. We addressed this via marginal-usage regularization:

$$\mathcal{L}_{\text{marginal}} = \sum_{\ell=1}^{L} \lambda_\ell \, \mathrm{KL}\big(\bar{q}(y^{(\ell)}) \parallel p(y^{(\ell)})\big), \qquad [8]$$

with annealing schedules that start λ_ℓ high to prevent early collapse, then gradually reduce. This remains hyperparameter-sensitive; alternatives like entropy regularization warrant investigation.

Our decoder conditions only on $z^{(L)}$: $p_\theta(x|z^{(L)})$, discarding discrete cluster information $\{y^{(\ell)}\}_{\ell=1}^{L}$. Conditioning on both,$p_\theta(x|y^{(1)}, \dots, y^{(L)}, z^{(L)})$,could improve reconstruction through cluster-specific parameters and enable controlled generation via mixture-of-experts architectures, where each $y^{(\ell)}$ gates specialized decoder components for lineage-specific regulatory networks.

Engineering improvements. Key refinements included PCA + KMeans cluster initialization to avoid component collapse, flexible per-level priors for non-identical mixtures, and a dedicated `CategoricalDistribution` for reliable initialization of cluster probabilities π. The IWAE computation was corrected, and KL terms were normalized by the number of mixture levels, producing smoother loss curves. An additional marginal KL regularizer $\mathrm{KL}(\bar{q}(y)\|p(y))$ mitigated marginal collapse and improved training stability and reproducibility. See Appendix D for extended discussion on the matter.

Biological Interpretation of Misclassifications Our models consistently confused CD19+ B cells (red) and CD34+ hematopoietic stem cells (green). This reflects genuine biological similarity: during B lymphopoiesis, pro-B and pre-B progenitors transiently co-express CD34 and CD19 [6], creating a transcriptional continuum between populations. Both cell types also share expression of lymphoid-specification factors (e.g., *PAX5*, *EBF1*) and proliferation-associated genes [8]. Additionally, HSCs comprise only ~0.01–0.1% of PBMCs, resulting in sparse, noisy profiles that complicate discrete clustering.

Future directions. Future work should condition generation jointly on (z, y) for controllable decoding, explore adaptive entropy-based regularization, and develop scalable inference (e.g. hierarchical sampling or variational pruning) to reduce the computational cost of deep hierarchies. Hierarchical mixture models thus remain a promising framework for probabilistic single-cell representation learning, pending further advances in efficiency and regularization.

8. Conclusion

MoMixVAE extends scVAE by introducing hierarchical latent dependencies and a heavy-tailed Student's t prior, enabling a coarse-to-fine representation of single-cell identity. Across all architectures, the Student's t prior consistently improves generative quality (higher IWAE) and better models the overdispersed nature of gene expression. MoMixVAE maintains stable F1-scores despite ARI variability, indicating accurate biological classification within complex, non-linear manifolds. The latent spaces unfold hierarchically from broad lineages to terminal subtypes, preserving structural consistency between predicted and true cell labels. Overall, MoMixVAE with a Student's t prior provides a robust and interpretable framework for modeling hierarchical organization in single-cell data.

Code availability statement

All training scripts are publicly available on GitHub at https://github.com/raphaelrubrice/scVAE_mva2025/tree/main. You can reproduce the full experimental pipeline using Google Colab, from data downloading and model training to result visualization. Additionally, we provide a `requirements.txt` file as well as `pyproject.toml` and `uv.lock` files to guarantee a consistent software environment.

Contributions

All authors jointly contributed to the conceptual design of the project and the discussion of experimental directions. **Raphaël Rubrice** formulated and implemented the hierarchical MoMixVAE model, extending the base scVAE architecture and managing probabilistic layer design, optimization, and formulation of the prior distributions. He implemented the Distributions and the final MixtureVAE model. **Adam Keddis** implemented the very first model of the MixtureVAE and the formulation and implementation of the independent multi-mixture (IndMoMVAE) variant extending the base scVAE architecture, developed the pipelines for latent-space visualization and figure generation. **Tiffney Aina** proposed the integration of hierarchical clustering into the latent-variable framework, motivating the hierarchical Mixture-of-Mixtures (MoMix-VAE) architecture. She developed the data-processing pipeline for PBMC datasets, implemented the hierarchical label manipulations, and led the report documentation. All three authors collaborated on experimental design, result interpretation, and manuscript preparation.

References

[1] Dilokthanakul, N., Mediano, P. A. M., Garnelo, M., Lee, M. C. H., Salimbeni, H., Arulkumaran, K., and Shanahan, M. (2017). Deep Unsupervised Clustering with Gaussian Mixture Variational Autoencoders. arXiv:1611.02648.

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

[3] Jang, E., Gu, S., and Poole, B. (2016). Categorical reparameterization with gumbel-softmax.

[4] Kingma, D. P. and Welling, M. (2022). Auto-Encoding Variational Bayes. arXiv:1312.6114.

[5] Klushyn, A., Chen, N., Kurle, R., Cseke, B., and van der Smagt, P. (2019). Learning hierarchical priors in vaes. *arXiv preprint arXiv:1905.04982*. Published at NeurIPS 2019 (spotlight).

[6] LeBien, T. W. and Tedder, T. F. (2008). B lymphocytes: how they develop and function. *Blood*, 112(5):1570–1580.

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

[8] Nutt, S. L. and Kee, B. L. (1999). Commitment to the b-lymphoid lineage depends on the transcription factor pax5. *Nature*, 401(6753):556–562.

[9] Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K., and Winther, O. (2016). Ladder variational autoencoders. arXiv:1602.02282 [stat.ML].

Appendix

A. Additional Evaluation Metrics

This appendix provides formal definitions of the metrics reported in **??**: Adjusted Rand Index (ARI), F1 score, and the importance-weighted autoencoder (IWAE) bound. Throughout, we denote ground-truth labels by $y \in \{1, \dots, C\}^N$ and predicted cluster assignments by $\hat{y} \in \{1, \dots, K\}^N$ for N cells.

A. Adjusted Rand Index (ARI). The Rand Index measures agreement between two partitions by counting pairwise decisions. Let n_{ij} be the contingency table between y and $\hat{y}$ (number of samples assigned to class i and cluster j), with row sums $a_i = \sum_j n_{ij}$ and column sums $b_j = \sum_i n_{ij}$. The ARI corrects the Rand Index for chance:

$$\mathrm{ARI}(y, \hat{y}) = \frac{\sum_{i,j} \binom{n_{ij}}{2} - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{N}{2}}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{N}{2}}}. \qquad [9]$$

ARI takes values in $[-1, 1]$, with 1 indicating perfect agreement and values near 0 corresponding to chance-level agreement.

B. F1 score with Hungarian alignment. Since cluster indices are identifiable only up to permutation, we align predicted clusters $\hat{y}$ with ground-truth classes y via Hungarian matching on the confusion matrix. Let π be the optimal permutation (assignment) maximizing total correct matches:

$$\pi^\star = \arg\max_{\pi \in \mathcal{S}_K} \sum_{k=1}^{K} \#\{n : y_n = \pi(k),\ \hat{y}_n = k\}. \qquad [10]$$

After alignment, we compute per-class precision and recall:

$$\mathrm{Precision}_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FP}_c}, \qquad \mathrm{Recall}_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FN}_c}, \qquad [11]$$

and the per-class F1 score:

$$\mathrm{F1}_c = \frac{2\,\mathrm{Precision}_c\,\mathrm{Recall}_c}{\mathrm{Precision}_c + \mathrm{Recall}_c}. \qquad [12]$$

We report the *weighted* F1 score to account for class imbalance:

$$\mathrm{F1}_{\mathrm{weighted}} = \sum_{c=1}^{C} \frac{N_c}{N}\,\mathrm{F1}_c, \qquad [13]$$

where N_c is the number of samples in class c.

C. Importance-Weighted Autoencoder (IWAE) bound. For a latent-variable model $p_\theta(x, z) = p_\theta(x \mid z)p(z)$ with variational posterior $q_\phi(z \mid x)$, the standard ELBO is

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p_\theta(x, z)}{q_\phi(z \mid x)}\right]. \qquad [14]$$

The IWAE objective tightens this bound by using K importance samples $z_{1:K} \sim q_\phi(z \mid x)$:

$$\log p_\theta(x) \geq \mathcal{L}^{(K)}_{\mathrm{IWAE}}(x) = \mathbb{E}_{z_{1:K} \sim q_\phi(\cdot|x)}\left[\log\left(\frac{1}{K}\sum_{k=1}^{K} w_k\right)\right], \qquad [15]$$

$$w_k = \frac{p_\theta(x, z_k)}{q_\phi(z_k \mid x)} = \frac{p_\theta(x \mid z_k)p(z_k)}{q_\phi(z_k \mid x)}. \qquad [16]$$

As K increases, $\mathcal{L}^{(K)}_{\mathrm{IWAE}}(x)$ becomes a tighter lower bound on $\log p_\theta(x)$ (monotonically non-decreasing in K under mild conditions), providing a more sensitive estimate of generative and inference quality than the single-sample ELBO.

B. KL Regularization and Marginal Collapse in Mixture VAEs

Mixture VAEs introduce categorical latents $y \in \{1, \dots, K\}$ with a prior $p(y) = \mathrm{Cat}(\pi)$ and an amortized posterior $q_\phi(y \mid x)$. In practice, optimization can exhibit *component collapse*: most datapoints are assigned to a small subset of components, while the remaining components are unused. This is often visible in the *aggregated* (marginal) posterior

$$\bar{q}(y) \;=\; \mathbb{E}_{p_{\mathrm{data}}(x)}\big[q_\phi(y \mid x)\big] \;\approx\; \frac{1}{N}\sum_{n=1}^{N} q_\phi(y \mid x_n), \qquad [17]$$

which becomes highly peaked.

A. Why standard KL terms may not prevent marginal collapse. In mixture VAEs, the ELBO typically contains a per-sample categorical KL term

$$\mathbb{E}_{p_{\mathrm{data}}(x)}\big[\mathrm{KL}(q_\phi(y \mid x) \,\|\, p(y))\big] = \mathbb{E}_{p_{\mathrm{data}}(x)}\Big[Q\Big]. \qquad [18]$$

$$Q = \sum_{k=1}^{K} q_\phi(y{=}k \mid x)\, \log \frac{q_\phi(y{=}k \mid x)}{\pi_k}$$

While this encourages each *individual* posterior $q_\phi(y \mid x)$ to be close to the prior, it does not directly constrain the *dataset-level* usage $\bar{q}(y)$. When the decoder can explain the data well using only a few components (or when early training dynamics favor a subset of components), the model may converge to a solution where many components receive negligible mass across the dataset.

B. Marginal-usage regularization. To explicitly control component utilization, we add a marginal-usage penalty that matches the aggregated posterior $\bar{q}(y)$ to the prior $p(y)$:

$$\begin{aligned} \mathcal{L}_{\mathrm{marg}} \;&=\; \lambda_{\mathrm{marg}}\,\mathrm{KL}\big(\bar{q}(y) \,\|\, p(y)\big) \qquad [19] \\ &=\; \lambda_{\mathrm{marg}} \sum_{k=1}^{K} \bar{q}(y{=}k)\, \log \frac{\bar{q}(y{=}k)}{\pi_k}. \end{aligned}$$

This term is *global* in the sense that it depends on the batch-averaged posterior probabilities and therefore directly discourages collapse of unused components. In practice, we estimate $\bar{q}(y)$ on each mini-batch of size B:

$$\bar{q}_{\mathcal{B}}(y{=}k) = \frac{1}{B}\sum_{b=1}^{B} q_\phi(y{=}k \mid x_b). \qquad [20]$$

C. Implementation used in this work. The following implementation computes $\mathrm{KL}(\bar{q}(y)\|p(y))$ on a mini-batch, matching the definition above. Let `level_probas` be the matrix of categorical probabilities $q_\phi(y \mid x)$ for the batch, and `ref_probas` be the prior probabilities π:

$$\mathrm{KL}(\bar{q}\|\pi) = \sum_{k=1}^{K} \bar{q}_k \left(\log \bar{q}_k - \log \pi_k\right), \qquad \bar{q}_k = \frac{1}{B}\sum_{b=1}^{B} q_\phi(y{=}k \mid x_b). \qquad [21]$$

A small constant ε is used for numerical stability through clamping:

$$\bar{q}_k \leftarrow \max(\bar{q}_k, \varepsilon), \qquad \pi_k \leftarrow \max(\pi_k, \varepsilon). \qquad [22]$$

D. Relation to "marginal collapse" terminology. We refer to this failure mode as *marginal collapse* because the collapse is evident at the level of the marginal/aggregated posterior $\bar{q}(y)$ rather than necessarily in each local posterior $q_\phi(y \mid x)$. The marginal-usage KL penalty is therefore a targeted regularizer: it enforces *population-level* component coverage while leaving the model free to assign confident, sparse posteriors to individual datapoints when justified by the data.

In hierarchical mixture settings with levels $\ell = 1, \ldots, L$, the same idea extends by regularizing each level's aggregated posterior $\bar{q}(y^{(\ell)})$ toward its prior $p(y^{(\ell)})$, potentially with level-specific weights λ_ℓ:

$$\mathcal{L}_{\mathrm{marginal}} = \sum_{\ell=1}^{L} \lambda_\ell \, \mathrm{KL}\big(\bar{q}(y^{(\ell)}) \,\|\, p(y^{(\ell)})\big). \qquad [23]$$

This is particularly important because deeper levels can collapse even when coarse levels remain well-utilized, due to conditional dependencies and uneven gradient allocation across multiple KL terms.