Variational Inference for Single-Cell Data

Valentin Badea[1,2] and Etienne Lefranc[1,3]

[1]Ecole Normale Supérieure Paris-Saclay
[2]Université Paris-Cité
[3]Centrale Supelec

Abstract

Single-cell RNA sequencing (scRNA-seq) yields high-dimensional, sparse count data affected by strong technical confounders (library size, batch effects, dropouts). Variational autoencoders provide a principled way to learn low-dimensional, biologically meaningful cell representations while accounting for such noise. In this project, we re-derive and re-implement the *scVI* model from Lopez et al. 2018 and validate it on the CORTEX dataset, where a t-SNE of the learned latent space recovers annotated neuronal and glial populations. We then study a CITE-seq PBMC dataset in two settings: RNA-only, and an RNA+protein extension that learns a joint RNA–protein representation jointly with the scVI backbone. The resulting multimodal embedding shows some improved alignment with cell membrane markers compared to RNA-only, illustrating the benefit of integrating surface proteins within this variational framework.

Keywords: Single-cell analysis, Transcriptomics, Proteomics, Variational Inference

Introduction

Single-cell RNA sequencing (scRNA-seq) enables the study of heterogeneous tissues at cellular resolution by measuring gene expression profiles for thousands of genes across large populations of cells. From a statistical standpoint, scRNA-seq data are challenging: they are high-dimensional and sparse, and they are heavily affected by technical variability such as library size differences, batch effects, and dropouts. For many downstream tasks (cell type discovery, trajectory inference, differential expression), it is therefore crucial to learn low-dimensional representations that capture biological variation while controlling for these confounders.

Variational autoencoders (VAEs) provide a natural probabilistic framework for representation learning in this setting. In particular, *scVI* (Lopez et al. 2018) proposes a hierarchical generative model for raw UMI counts, combining a latent cell state with explicit modelling of library size and batch information, and using amortized variational inference with neural networks. This approach yields embeddings that are both scalable and interpretable, and has become a reference method for scRNA-seq analysis.

Beyond transcriptomics, multimodal single-cell assays such as CITE-seq (Stoeckius et al. 2017) measure, for each cell, both RNA counts and surface protein abundances (ADT) through DNA-

barcoded antibodies. Protein markers often provide complementary and sometimes cleaner signals of cell identity than mRNA counts alone. This motivates extending scVI-style models to integrate RNA and proteins within a unified latent representation, while preserving the interpretability and robustness of the scVI generative assumptions.

In this work, we proceed in two steps. First, we re-derive and re-implement scVI from scratch in PyTorch and validate the learned RNA latent space on the CORTEX dataset (Zeisel et al. 2015). Second, we consider a CITE-seq PBMC dataset (Stuart et al. 2019, Chervov 2025) and compare RNA-only representations to an RNA+ADT extension that learns a joint RNA–protein representation trained end-to-end, with embeddings learned jointly.

Contributions.

- We re-derive and implement the scVI model (Lopez et al. 2018) in PyTorch, including batch and library-size modelling for count data.
- We validate the RNA latent space on the CORTEX dataset (Zeisel et al. 2015) and report a t-SNE visualization recovering known neuronal and glial populations.
- On CITE-seq PBMC data (Stoeckius et al. 2017), we compare RNA-only embeddings to an RNA+ADT extension and show some degree of improvement in cell state representation when integrating protein information.

Methods

Problem notations

In this section, we define a few useful mathematical notations. We let G be a set of genes whose transcripts are studied in a single-cell transcriptomics dataset of N cells. In particular, we are able to measure for every cell $c_n, n \in \{1, ..., N\}$ in the experiment, the counts x_{ng} of RNA strands that were detected for every gene $g \in G$ in the assay. Note that observations, parameters or random variables relative to one specific pair "cell-gene" will be written with the subscript ng, whereas values only depending on the gene g or cell n will only have either one of them as subscript (namely g or n).

In the following, for every cell c_n, we denote z_n, its latent representation as a $p-$dimensional vector, s_n its batch and l_n its library size (i.e. how many RNAs were counted in each cell). Batches are assumed to belong to a set B of technical batches. This information is particularly important in single-cell analysis, since we expect the technical errors on measurements to correlate across processing batches. If no batch information is available in the dataset, we consider that $\#B = 1$.

Biologically-informed Mathematical formulation

In this section, we take a closer look at the mathematical foundation behind the **scVI** paper (Lopez et al. 2018). This hierarchical model ultimately aims at predicting the number of detected RNA counts for every gene and every cell measured in a dataset. These measures are expected to depend

on the genes and cells we are looking at (and their latent representations), but also on the batch information and the library size. **Batch effect** can be introduced in the analysis when single-cell measurements are acquired in several rounds or by different experimenters. This can lead to single-cell variability in the latent representations that is not explained by cell states. For every cell in the dataset, **library size** denotes the total amount of RNA transcripts that were processed in the experiment. For instance, if the measurement procedure was faulty in one particular cell or batch, we'd expect a low number of detected RNAs to correlate across genes. As a result, library size is used in (Lopez et al. 2018) as a measure of detection efficiency and sequencing depth across cells and batches. The library size l_n is assumed to be drawn under a log-normal distribution ($\mathcal{LN}$) with mean and variance $(l_\mu^{(b)}, {l_\sigma^{(b)}}^2)$ where cell c_n belongs to batch $b \in B$. The remaining variability should be explained by cell transcription states alone and is modeled by the latent variable z_n which is drawn from a low-dimensional centered Gaussian $\mathcal{N}(0, I)$, using the reparameterization trick.

A deep neural network f_w, taking as inputs (z_n, s_n) computes $\rho_n \in \mathbb{R}^G$ as a proportion vector that represents the normalized contribution of each gene to the cell library size. An external dispersion parameter $\theta \in \mathbb{R}^G$ is learned throughout training to model $w_{ng} \sim \Gamma(\rho_{ng}, \theta_g)$ as the normalized latent abundance of each gene transcripts in the dataset. This abundance is then scaled by the library size l_n to produce a robust estimate of the number y_{ng} of RNA transcripts of any given gene g, actually available in the sample through a Poisson distribution with mean $l_n w_{ng}$. Finally, the number x_{ng} of gene g transcripts detected in cell c_n is set to 0 with some probability π_{ng} and y_{ng} otherwise, to simulate some technical dropout. The parameters π_n of this set of Bernoulli distributions are computed by a deep neural network f_h from inputs (z_n, s_n). Following the supplementary materials, we define the following hierarchical graphical model for every cell data at row n in our single-cell omics dataset:

$$
\begin{aligned}
z_n &\sim \mathcal{N}(0, I) \\
l_n &\sim \mathcal{LN}\left(l_\mu^{(b)}, {l_\sigma^{(b)}}^2\right) \\
\rho_n &= f_w(z_n, s_n) \\
w_{ng} &\sim \Gamma(\rho_{ng}, \theta_g), \forall g \in G \\
y_{ng} &\sim \mathcal{P}(l_n w_{ng}), \forall g \in G \\
\pi_n &= f_h(z_n, s_n) \\
h_{ng} &= \mathcal{B}(\pi_{ng}), \forall g \in G \\
x_{ng} &= \begin{cases} y_{ng} & \text{if } h_{ng} = 0 \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
\tag{1}
$$

The corresponding probabilistic graphical model is sketched in Fig. 1 using the notations seen in the lectures.

Conditional distribution of x_n

The previous graphical model enables us to compute a closed form distribution for x_n conditionally to z_n, l_n, s_n (see **Annex 1**). First, we found that $y_{ng}|z_n, l_n, s_n$ follows a negative binomial

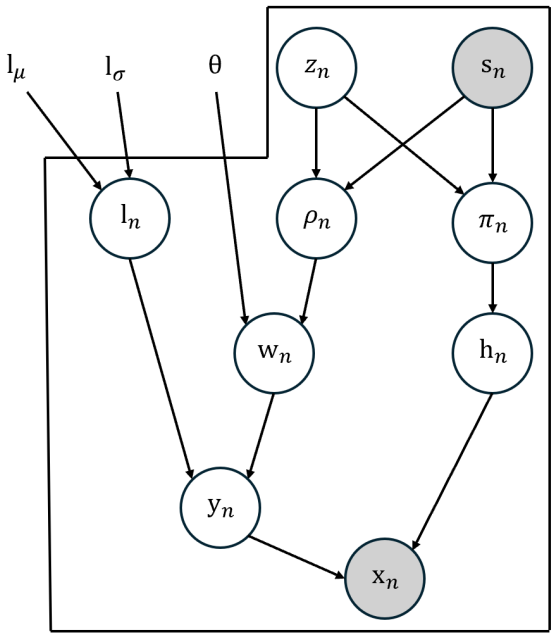

Figure 1: Probabilistic Graphical Model associated with scVI

distribution with number of successes $r = \rho_{ng}$ and probability of success $p = \frac{1/\theta_g}{l_n+1/\theta_g} = \frac{1}{1+l_n\theta_{ng}}$:

$$p(y_{ng} = k|z_n, l_n, s_n) = \frac{\Gamma(\rho_{ng}+k)}{k!\Gamma(\rho_{ng})}\left(1 - \frac{1/\theta_g}{l_n+1/\theta_g}\right)^k \left(\frac{1/\theta_g}{l_n+1/\theta_g}\right)^{\rho_{ng}} \tag{2}$$

The average number of genes y_{ng} in each cell is therefore:

$$\mathbb{E}(y_{ng}|z_n, l_n, s_n) = \frac{r(1-p)}{p} = \rho_{ng}\frac{\frac{l_n\theta_g}{1+l_n\theta_g}}{\frac{1}{1+l_n\theta_g}} = \rho_{ng}l_n\theta_g \tag{3}$$

which highlights the contributions of both scaling parameters θ_g and l_n to the number of transcripts y_{ng} of gene g in cell c_n. Finally (see **Annex 1**), we find that the conditional distribution of $x_n|z_n, l_n, s_n$ is a Zero-Inflated Negative Binomial (ZINB):

$$\forall k \in \mathbb{N}, p(x_{ng} = k|z_n, l_n, s_n) = \begin{cases} \pi_{ng} + (1-\pi_{ng})\left(\frac{1}{1+l_n\theta_g}\right)^{\rho_{ng}} & \text{if } k = 0 \\ (1-\pi_{ng})\frac{\Gamma(\rho_{ng}+k)}{k!\Gamma(\rho_{ng})}\left(1 - \frac{1}{1+l_n\theta_g}\right)^k \left(\frac{1}{1+l_n\theta_g}\right)^{\rho_{ng}} & \text{otherwise} \end{cases} \tag{4}$$

Variational Inference

In practice, the posterior $p(z_n, l_n|x_n, s_n)$ is intractable, as it requires marginalizing over all latent variables, which involves integrals with no closed form (due to the ZINB likelihood). We're therefore looking for a parametric distribution $q_\phi(z_n, l_n|x_n, s_n)$ to approximate this distribution with a

set of deep neural networks. To enable efficient inference and learning, we resort to amortized variational inference. Following a mean-field approximation, as in (Lopez et al. 2018), we enforce conditional independence of latents z_n, l_n:

$$q_\phi(z_n, l_n|x_n, s_n) = q_\phi(z_n|x_n, s_n)q_\phi(l_n|x_n, s_n) \tag{5}$$

These two distributions are parametrized with two encoders $f_{z,\phi} : (x, s) \mapsto (\mu_z(x, s), \sigma_z(x, s)) \in \mathbb{R}^p \times \mathbb{R}^p$ and $f_{l,\phi} : (x, s) \mapsto (\mu_l(x, s), \sigma_l(x, s)) \in \mathbb{R} \times \mathbb{R}$, such that:

$$\begin{aligned} z_n|x_n, s_n &= \mu_z(x_n, s_n) + \sigma_z(x_n, s_n) \odot \epsilon_z, \quad \epsilon_z \sim \mathcal{N}(0, I) \\ \log l_n|x_n, s_n &= \mu_l(x_n, s_n) + \sigma_l(x_n, s_n)\epsilon_l, \quad \epsilon_l \sim \mathcal{N}(0, 1) \end{aligned} \tag{6}$$

where the reparameterization trick ensures that the gradients flow properly through the model. Next, we derive the variational formulation of the model. By marginalizing over the latents (see **Annex 2**), we find the following Evidence-Based Lower Bound on the log-likelihood:

$$\log p(x|s) \geq \mathcal{L}(x) = \mathbb{E}_{q(z,l|x,s)}(\log p(x|z, l, s)) - D_{KL}(q(z|x, s)||p(z|s)) - D_{KL}(q(l|x, s)||p(l|s)) \tag{7}$$

where $p(z|s) = p(z) = \mathcal{N}(z; 0, I)$ and $q(z|x, s) = \mathcal{N}(z; \mu_z(x, s), \text{Diag}(\sigma_z(x, s)^2))$, such that:

$$D_{KL}(q(z|x, s)||p(z|s)) = \frac{1}{2}\sum_{i=1}^{p}(\sigma_{z,i}(x, s)^2 + \mu_{z,i}(x, s)^2 - 1 - \log(\sigma_{z,i}(x, s)^2)) \tag{8}$$

Moreover, we have $D_{KL}(q(l|x, s)||p(l|s)) = D_{KL}(q(\log l|x, s)||p(\log l|s))$ such that:

$q(\log l|x, s) = \mathcal{N}(\log l; \mu_l(x, s), \text{Diag}(\sigma_l(x, s)^2))$ and $p(\log l|x, s) = \mathcal{N}(\log l; l_\mu^{(s)}, (l_\sigma^{(s)})^2)$. Then:

$$D_{KL}(q(l|x, s)||p(l|s)) = \frac{1}{2}\left(\log \frac{(l_\sigma^{(s)})^2}{\sigma_l(x, s)^2} + \frac{\sigma_l(x, s)^2 + (\mu_l(x, s) - l_\mu^{(s)})^2}{(l_\mu^{(s)})^2} - 1\right) \tag{9}$$

We train the model by jointly optimizing the neural nets' parameters to maximize the ELBO.

An extension of the scVI model to proteomics data

In a CITE-seq dataset, each cell is characterized not only by its RNA count vector $x_n \in \mathbb{N}^G$ but also by a vector of $P = 25$ surface protein measurements $a_n \in \mathbb{R}^P$ (ADT). The scVI model described above only defines a generative distribution for RNA counts, so we had to implement a new architecture inspired from scVI, which we call citeVI. We extend the graphical model described in **(1)**, by introducing a deep neural network function f_a such that:

$$\begin{aligned} \mu_n^a &= f_a(z_n, s_n) \\ a_{np} &\sim \text{NB}(\mu_{np}^a, \phi_p) \end{aligned} \tag{10}$$

where the protein counts are modeled with a negative binomial distribution with protein-cell-specific parameters μ_{np}^a and protein-specific dispersion ϕ_p. The new model factorization writes:

$$p(x_n, a_n, z_n, l_n|s_n) = p(z_n)p(l_n|s_n)p(x_n|z_n, l_n, s_n)p(a_n|z_n, s_n) \tag{11}$$

whereas the updated variational posterior q is such that:

$$q(z_n, l_n|x_n, a_n, s_n) = q(z_n|x_n, a_n, s_n)q(l_n|x_n, s_n) \tag{12}$$

with $q(z_n|x_n, a_n, s_n) = \mathcal{N}(z_n; \mu_z(x_n, a_n, s_n), \text{Diag}(\sigma_z(x_n, a_n, s_n)^2))$. Finally, the updated ELBO is:

$$\mathcal{L}(x, a) = \mathbb{E}_{q(z,l|x,a,s)}(\log p(x|z, l, s) + \log p(a|z, s)) - D_{KL}(q(z|x, a, s)||p(z)) - D_{KL}(q(l|x, s)||p(l|s)) \tag{13}$$

Implementation details

In the end, we trained three models adapted from our analysis of the scVI paper (Lopez et al. 2018):

1. A scVI model trained on the CORTEX dataset to check that our implementation reached results comparable with the scVI paper.

2. A scVI model trained on CITE-seq single-cell transcriptomics data.

3. A citeVI model trained on CITE-seq single-cell transcriptomics and proteomics data.

All models were trained for a maximum of 20 epochs on $L4$ GPUs provided through our personal Google accounts. Models had on average between 8 and 10 million learnable parameters, and one epoch would take on average 15-20 seconds. Learning was performed with the Adam optimizer (learning rate $l_r \in \{10^{-4}, 10^{-3}\}$). While training our models, we encountered some very problematic stability issues, as the Normal KL-divergence term would often diverge to infinity, resulting to *NaN* errors in the following iterations. Clamping the decoder outputs to reasonable ranges $[10^{-6}, 10^{6}]$ helped solve this issue, although this forced us to derive slightly from the technical details in the original scVI paper (Lopez et al. 2018).

Results

Training behavior of the RNA scVI model

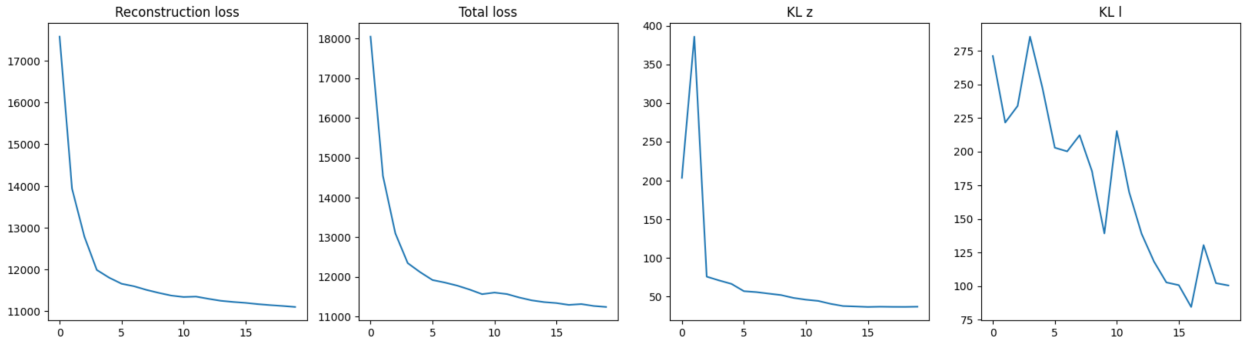

Figure 2: Training History of scVI model trained on the CORTEX dataset used in Lopez et al. 2018

In Fig. 2, we reported the evolutions of the training losses while training the scVI model on the CORTEX dataset. Overall, the ELBO objective seems to decrease smoothly in terms of reconstruction loss and KL divergences. The latter decreases rapidly during the first few epochs before plateauing, indicating that the decoder quickly learns to capture the main structure of the count distribution. Overall, the total loss shows a clear downward trend and stabilizes after about ten epochs, which we use as a criterion for early stopping. We produce the same plots in **Annex 3** for the scVI model trained on the single-cell RNA data in the CITE-seq dataset and make the same overall comments on the corresponding training history.

Latent representation from RNA counts only

After training, we extracted the approximate posterior means $\mu_z(x_n, s_n)$ for all entries in the CORTEX dataset and visualized the resulting 10-dimensional RNA latent space in two dimensions using the TSNE algorithm (Maaten and Hinton 2008). Figure 3 shows that this embedding correlates with known neurological cell types. It is important to note that this annotation was never given to the model directly (as a batch index for instance). As a result, we see that our latent representation of cell states closely aligns with known cellular types.

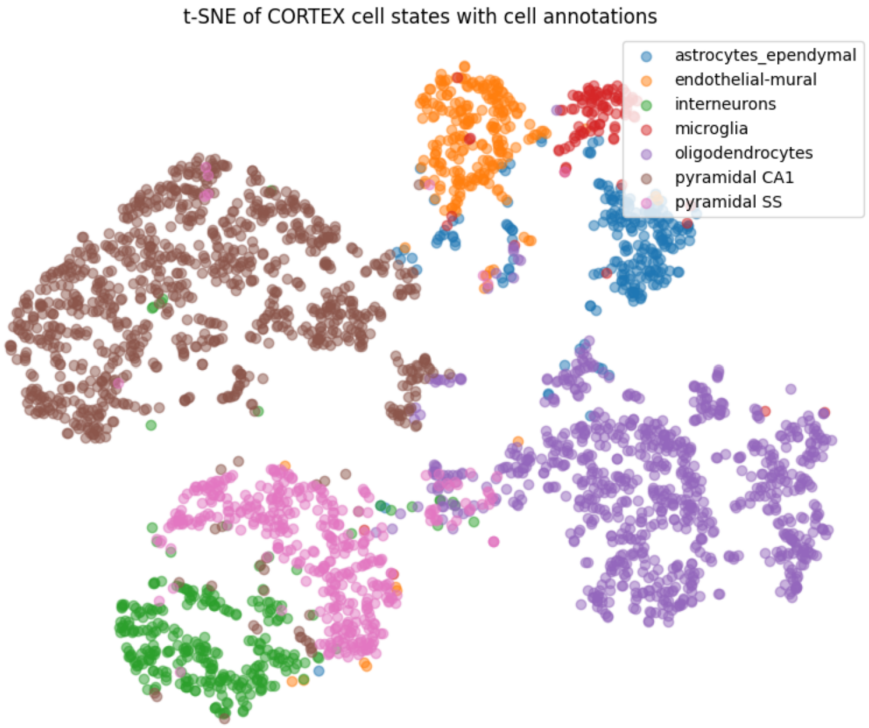

Figure 3: 2D-TSNE embeddings of the RNA latent space learned by scVI in the CORTEX dataset, colored by cell types.

We followed a similar methodology and plotted the 2D-TSNE embeddings from the latent space of scVI trained on RNA CITE-seq data only. We did not have access to a proper single-cell annotation of the CITE-seq dataset, but knowing that this dataset was mostly made of peripheral mononuclear cells, we applied a simple heuristic to isolate a few cell types from the protein expression values.

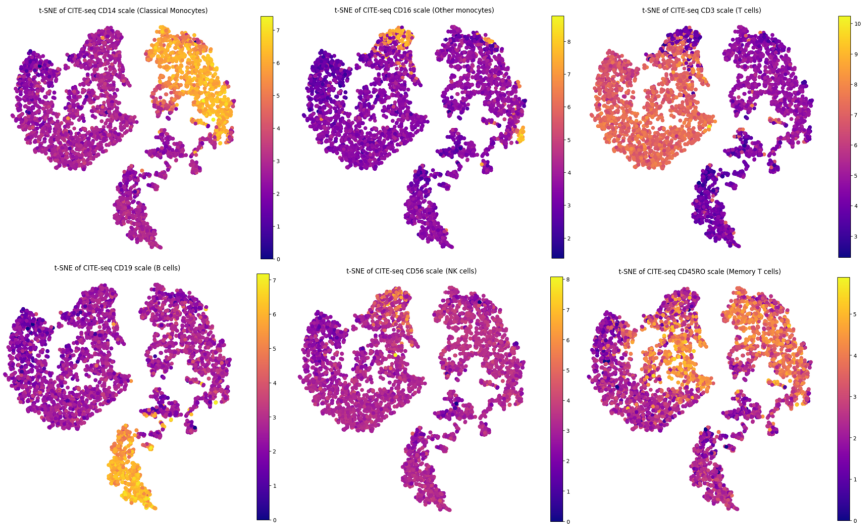

Figure 4: 2D-TSNE embeddings of scVI latent space, trained on single-cell RNA data, overlayed with membrane marker expression levels (in log scale).

Fig. 4 shows a few broad cellular classes that seem to cluster in this 2D representation of scVI latent space:

Joint RNA–protein latent space

We then trained the protein VAE on log-transformed ADT measurements and extracted the posterior means $\mu_{\phi_{\text{prot}}}(x_n, a_n, s_n)$ for each cell. A summary plot of this model training history is reported in **Annex 3.** Then, we embedded these new latent representations in two dimensions using the same TSNE algorithm. Figure 5 displays the same TSNE joint embedding scatter plots, colored by various membrane markers.

It is difficult to make a precise statement on the quality of these latent representations without domain-specific expertise or by applying them to some precise clinical context. For example, we could use our scVI model trained on the full CITE-seq data and try to see if it can detect more accurately abnormal immune cells (from patients with leukemia for example). Here, one should not attempt to compare the color maps from Figure 4 to 5, since we computed the TSNE plots on two subsets of the CITE-seq dataset of 5000 cells each, that do not necessarily overlap (in an attempt to reduce the computational burden of fitting a TSNE model on 33k cells). In the end, it does look like some clusters of cells are slightly more clearly separated from the others in the joint latent space: for instance, it feels that way when we look at the boundary T-cells/other monocytes or B-cells from the rest in Figure 5. It is still very interesting to see how good the RNA-only model was in the beginning at separating immune cell types described by membrane protein expression levels.

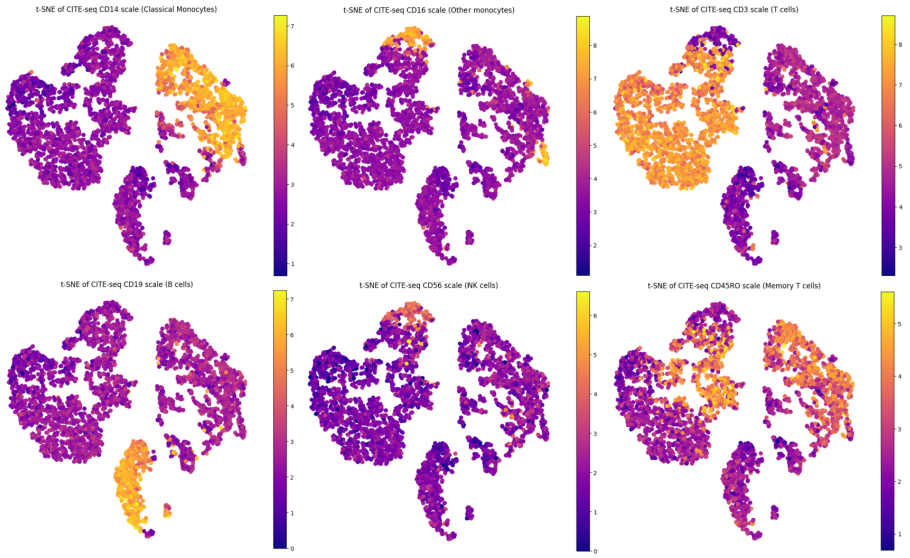

Figure 5: 2D-TSNE embeddings of citeVI latent space, trained on single-cell RNA and proteomics data, overlayed with membrane marker expression levels (in log scale).

Conclusion

In this work, we have re-implemented from scratch a variational inference model for cell state extraction of complex single-cell transcriptomics data, as described in the paper *Deep generative modeling for single-cell transcriptomics* (Lopez et al. 2018). We have shown that our implementation closely aligns with the results obtained on a single-cell transcriptomics dataset of mouse cortical cells used in this paper. Then, we proceeded to apply this method to an entirely different dataset of peripheral blood mononuclear cells on the RNA modality alone, and extended it to process both RNA and protein inputs. This led us to extend the work of Lopez et al. 2018 and derive our own variational inference framework that learns a joint representation of single-cell states with respect to their transcriptomics and proteomics signatures.

Contribution statement

- **Valentin Badea:** design of scVI architecture from the paper, graphical model of scVI, design of citeVI architecture, scVI implementation, more than half of citeVI implementation, deriving ZINB/ELBO equations, model training and tuning, Methods section in the report.

- **Etienne Lefranc:** research and identification of the paired RNA+protein dataset, some contribution in the design of citeVI, implementation of less than half of citeVI, Abstract, Introduction in the report, Final Poster.

References

Chervov, Alexander. 2025. *CITE-seq = scRNA-seq + Proteins: Human PBMCs 2019*. Https://www.kaggle.com/datasets/alexandervc/citeseq-scrnaseq-proteins-human-pbmcs-2019 [Accessed: 2025-12-17].

Lopez, Romain, Jeffrey Regier, Michael B.Cole, Michael I. Jordan, and Nir Yosef. 2018. "Deep generative modeling for single-cell transcriptomics." *Nature Methods* 15, no. 5 (December): 1053. https://doi.org/10.1038/s41592-018-0229-2.

Maaten, L.J.P. van der, and G.E. Hinton. 2008. "Visualizing High-Dimensional Data Using t-SNE." *Journal of Machine Learning Research* 9.

Stoeckius, Marlon, Christoph Hafemeister, William Stephenson, Brian Houck-Loomis, Pratip Chattopadhyay, Harold Swerdlow, Rahul Satija, and Peter Smibert. 2017. "Simultaneous epitope and transcriptome measurement in single cells." *Nature Methods* 14 (4): 865. https://doi.org/10.1038/nmeth.4380.

Stuart, Tim, Andrew Butler, Paul Hoffman, and Christoph Hafemeister. 2019. "Comprehensive Integration of Single-Cell Data." PMID: 31178118, *Cell* 177:1888.

Zeisel, Amit, Ana B. Muñoz-Manchado, Simone Codeluppi, Peter Lönnerberg, Gioele La Manno, Anna Juréus, Sueli Marques, et al. 2015. "Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq." *Science* 347 (6226): 1138–1142. https://doi.org/10.1126/science.aaa1934. eprint: https://www.science.org/doi/pdf/10.1126/science.aaa1934. https://www.science.org/doi/abs/10.1126/science.aaa1934.

Annex

1) Deriving the conditional distribution of x_n

Let us have $k \in \mathbb{N}$. As we marginalize over w_n:

$$\begin{aligned}
p(y_{ng} = k|z_n, l_n, s_n) &= \int_0^{+\infty} p(y_{ng} = k|z_n, l_n, s_n, w)p(w|z_n, s_n, l_n)dw \\
&= \int_0^{+\infty} \mathcal{P}(k; l_n w)\Gamma(w; \rho_{ng}, \theta_g)dw \\
&= \int_0^{+\infty} \frac{(l_n w)^k}{k!}e^{-l_n w}\frac{1}{\Gamma(\rho_{ng})}\frac{w^{\rho_{ng}-1}}{\theta_g^{\rho_{ng}}}e^{-w/\theta_g}dw \\
&= \frac{l_n^k}{\Gamma(\rho_{ng})k!\theta_g^{\rho_{ng}}}\int_0^{+\infty} w^{\rho_{ng}+k-1}e^{-(l_n+1/\theta^g)w}dw, \quad u = (l_n + 1/\theta_g)w \\
&= \frac{l_n^k}{\Gamma(\rho_{ng})k!\theta_g^{\rho_{ng}}}\left(\frac{1}{(l_n + 1/\theta_g)}\right)^{\rho_{ng}+k}\int_0^{+\infty} u^{\rho_{ng}+k-1}e^{-u}du \\
&= \frac{l_n^k}{\Gamma(\rho_{ng})k!\theta_g^{\rho_{ng}}}\left(\frac{1}{l_n + 1/\theta_g}\right)^{\rho_{ng}+k} \times \Gamma(\rho_{ng} + k) \\
&= \frac{\Gamma(\rho_{ng} + k)}{k!\Gamma(\rho_{ng})}\left(1 - \frac{1/\theta_g}{l_n + 1/\theta_g}\right)^k\left(\frac{1/\theta_g}{l_n + 1/\theta_g}\right)^{\rho_{ng}}
\end{aligned} \tag{14}$$

Then, for any $k > 0$:

$$\begin{aligned}
p(x_{ng} = k|z_n, l_n, s_n) &= p(y_{ng} = k, h_{ng} = 0|z_n, l_n, s_n) \\
&= p(y_{ng} = k|z_n, l_n, s_n)p(h_{ng} = 0|z_n, l_n, s_n) \quad \text{(cond. indep. of } y \text{ and } h\text{)} \\
&= (1 - \pi_{ng})\frac{\Gamma(\rho_{ng} + k)}{k!\Gamma(\rho_{ng})}\left(1 - \frac{1/\theta_g}{l_n + 1/\theta_g}\right)^k\left(\frac{1/\theta_g}{l_n + 1/\theta_g}\right)^{\rho_{ng}}
\end{aligned} \tag{15}$$

On the other hand, if $k = 0$:

$$\begin{aligned}
p(x_{ng} = 0|z_n, l_n, s_n) &= p(h_{ng} = 1|z_n, l_n, s_n) + p(h_{ng} = 0, y_{ng} = 0|z_n, l_n, z_n) \\
&= p(h_{ng} = 1|z_n, l_n, s_n) + p(h_{ng} = 0|z_n, l_n, z_n)p(y_{ng} = 0|z_n, l_n, z_n) \\
&= \pi_{ng} + (1 - \pi_{ng})\left(\frac{1}{1 + l_n\theta_g}\right)^{\rho_{ng}}
\end{aligned} \tag{16}$$

Finally, we have:

$$\forall k \in \mathbb{N}, p(x_{ng} = k|z_n, l_n, s_n) = \begin{cases} \pi_{ng} + (1 - \pi_{ng})\left(\frac{1}{1+l_n\theta_g}\right)^{\rho_{ng}} & \text{if } k = 0 \\ (1 - \pi_{ng})\frac{\Gamma(\rho_{ng}+k)}{k!\Gamma(\rho_{ng})}\left(1 - \frac{1}{1+l_n\theta_g}\right)^k\left(\frac{1}{1+l_n\theta_g}\right)^{\rho_{ng}} & \text{otherwise} \end{cases} \tag{17}$$

2) Deriving the scVI ELBO

By marginalizing over latents:

$$
\begin{aligned}
\log p(x|s) &= \log \iint p(x,z,l|s)dzdl = \log \iint q(z,l|x,s)\frac{p(x,z,l|s)}{q(z,l|x,s)}dzdl \\
&= \log \mathbb{E}_{q(z,l|x,s)}\left(\frac{p(x,z,l|s)}{q(z,l|x,s)}\right) \\
&\geq \mathbb{E}_{q(z,l|x,s)}\left(\log \frac{p(x,z,l|s)}{q(z,l|x,s)}\right) \quad \text{(Jensen)} \\
&= \mathbb{E}_{q(z,l|x,s)}\left(\log \frac{p(x|z,l,s)p(z,l|s)}{q(z,l|x,s)}\right) \\
&= \mathbb{E}_{q(z,l|x,s)}\left(\log \frac{p(x|z,l,s)p(z|s)p(l|s)}{q(z|x,s)q(l|x,s)}\right) \\
&= \mathbb{E}_{q(z,l|x,s)}(\log p(x|z,l,s)) - \mathbb{E}_{q(z,l|x,s)}\left(\log \frac{q(z|x,s)}{p(z|s)}\right) - \mathbb{E}_{q(z,l|x,s)}\left(\log \frac{q(l|x,s)}{p(l|s)}\right) \\
&= \mathbb{E}_{q(z,l|x,s)}(\log p(x|z,l,s)) - \mathbb{E}_{q(z|x,s)}\left(\log \frac{q(z|x,s)}{p(z|s)}\right) - \mathbb{E}_{q(l|x,s)}\left(\log \frac{q(l|x,s)}{p(l|s)}\right) \\
&= \mathbb{E}_{q(z,l|x,s)}(\log p(x|z,l,s)) - D_{KL}(q(z|x,s)||p(z|s)) - D_{KL}(q(l|x,s)||p(l|s))
\end{aligned}
\tag{18}
$$

3) Training history

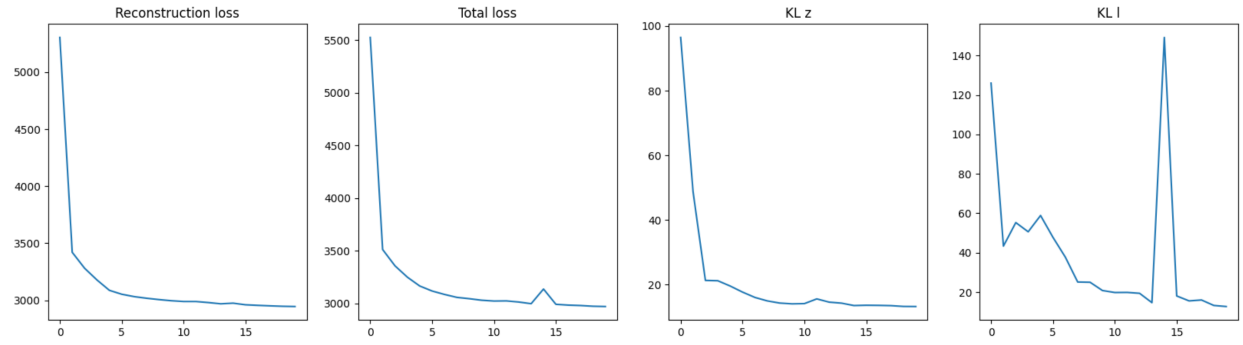

Figure 6: Evolution of the different terms of the ELBO during training of the scVI model on CITE-seq PBMC data (RNA data only).

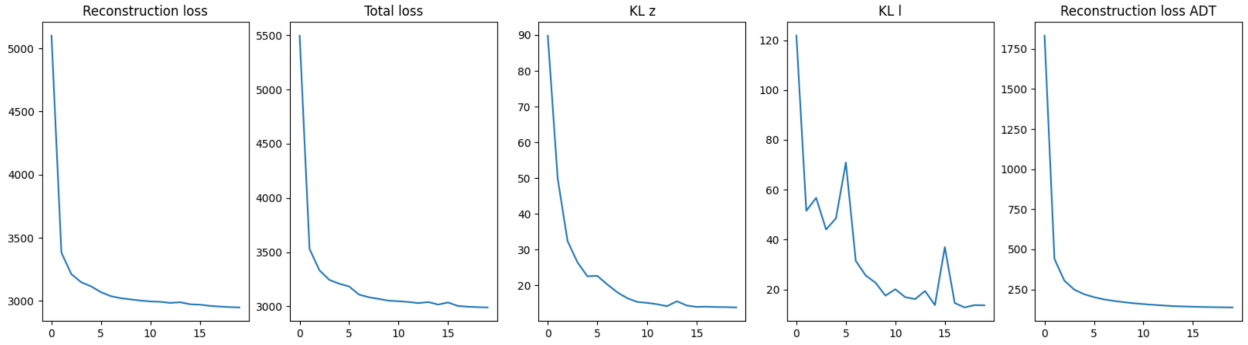

Figure 7: Evolution of the different terms of the ELBO during training of the citeVI model on CITE-seq PBMC data (RNA and protein data).