Ensembling in Variational Autoencoder Architectures for Effective Posterior Distribution of Cell States Estimation in Single-Cell Data

Tamjeed Azad, Elham Azizi, Max Gupta, Achille Nazaret
Department of Computer Science and Biomedical Engineering, Columbia University
New York, NY 10027
Azizi Lab

Abstract

We aim to develop a method combining variational autoencoders (VAEs) and ensembling techniques to model single-cell RNA sequencing datasets using generative modeling. The approximated posteriors of existing VAE-based methods with no ensembling may poorly estimate cells' gene expression, especially with regards to its uncertainty; the hope is that some ensembling structure via multiple encoders, decoders, or entire VAEs will improve the confidence and accuracy of the model's posterior inference. Such a model would potentially be useful for Bayesian decision-making problems such as detecting differential gene expression.

1 Introduction

Generative modeling of single-cell data is a promising method of analysis of cell states and gene expression, particularly in the field of decision-making tasks such as differential gene expression across cell states. We combine the use of variational inference, which has been shown to be very powerful in Bayesian decision making systems [1]. Methods such as [2] and [3] can be used to exploit generative modeling of single-cell data to complete Bayesian hypothesis testing for differential gene expression. A Variational Autoencoder framework lies at the core of both of these methods. An excellent review and tutorial of these kinds of methods is shown in [4].

2 Synthetic Dataset Generation

Going off of the approach in [3], we generate a synthetic dataset for the detection of differentially expressed genes in the model. N and G are the number of cells and genes in X, respectively; there are two states a, b, and each follow a Poisson-Log Normal distribution with means μ_{ag} and μ_{bg} for $g \leq G$, with shared covariance Σ. We assume that the cell state c_n of each cell n is modeled using a categorical distribution with parameter p, and these have means $h_{ng} \sim \text{LogNormal}(\mu_{c_n}, \Sigma)$; we have that x_{ng} are assumed to be distributed by $x_{ng} \sim \text{Poisson}(h_{ng})$. The covariance structure is given by

$$\Sigma = (0.5 + u)I_g + 2aa^T$$

where $a \sim \mathcal{U}((-1, 1)^g)$ and $u \sim \mathcal{U}((-0.25, 0.25)^g)$. We have that the ground-truth log-fold-change values that signify differential expression between the two cell states a, b is denoted by Δ_g; we randomly assign a DE status to each gene, and for those that are DE, we have up-regulation or down-regulation between the two states for the genes. With this in mind, we simulate with a varying scale factor S of DE level, so we have that when the gene is down-regulated, Δ_g is sampled from a Gaussian distribution with mean $-S/2$, and when upregulated, Δ_g is sampled from a Gaussian distribution with mean $S - S/2$. Both distributions have standard deviation 0.16.

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

Gene expression means for cells with state a were sampled uniformly from the interval (10,100); the means for population b were given by

$$\mu_b = 2^{\Delta_g} \mu_a$$

3 Model Description

We inspire our model from [2] and [3]. We have a data matrix X representing scRNA-seq data, representing cells by genes, where X_{ij} for some cell i and some gene j is the number of transcripts for gene j that were observed in cell i. For each of these data points, we construct a latent variable h_{ij} to more accurately model the state of the cell with respect to expression of that specific gene, in order to account for technical noise variations. We assume that this dataset X come as paired with cell type labels, and we can detect differential gene expression using log-fold changes based on the latent variables h across the cell types. We assume that there are two cell types a, b. We use this architecture to do posterior estimation of the probability that one gene is differentially expressed in one cell type over another, via the latent variables h.

For each cell n, we have that z_n is a normally distributed latent variable with mean 0 and variance I_d, and it represents the overall state of cell n. We have that $l_n \sim \text{LogNormal}(\mu_l, \sigma_l^2)$, and this represents the library size accounting for sampling noise in the dataset. We have that each expression count x_{ng} follows a zero-inflated negative binomial distribution; the mean of the distribution is the product of l_n and $h_{ng} = f_w(z_n)$, where f_w is a neural network. We thus have that $p_\theta(h_{ng}|x_n)$ is the push-forward of $p_\theta(z_n|x_n)$, through the gth output of f_w. Thus, the distribution $p_\theta(x)$ is modeled.

3.1 Ensembling Directions

Multiple directions exist in the implementation of ensembling. We could use multiple encoders to encode latent variables h_{aij}, where a represents the encoder being used, i representing the cell, and j representing the gene. From these latent variables we can either sum, average, take some activation of them, and use a single decoder framework to reconstruct x.

We could also use multiple decoders instead; we'd have multiple encoders decoding each h_{ij}.

We could also have simply a set of VAEs with a single encoder, decoder each, and combine the outputs of each of them to get a single unified output; we could combine using linear methods or nonlinear activations. Currently this is the approach, we average the reconstructions of the datasets to get a unified reconstruction.

4 Testing, Benchmarking

We successfully reproduced the results from the FDR approximation experiment from [3], showing that the FDR estimation of a single VAE architecture poorly estimates FDR. We use FDR to test for the accuracy of estimation of the underlying distributions of cell states. We know the underlying distributions of the synthetic dataset a priori, and we thus compute the ground truth FDR under a series of thresholds for these synthetic datasets and compare them to the expected FDR under a single VAE architecture with the expected FDR under an ensembled VAE architecture.

We consider FDR in the context of detecting differentially expressed genes. We have that function f returns whether a gene is differentially expressed across clusters a, b. Specifically, considering two cells a', b' with underlying cell state values $h_{a'g}$ and $h_{b'g}$ for some gene g, we consider that for some threshold δ, we have that the function f is given by

$$f(a', b', g) = \mathbf{1}\left[\left|\log \frac{h_{a'g}}{h_{b'g}}\right| \geq \delta\right]$$

We estimate the general differential expression across clusters a, b with the following method. We sample t cells from both clusters a and b, and we calculate for t distinct cell pairs a_i and b_i, one cell from each cluster, $f(a_i, b_i, g)$ for some gene g. We then have that $\tilde{f}$ gives a predicted differential expression based on all the cells, and we take

$$\tilde{f}(g) = \mathbf{1}\left[\frac{\sum_{i=1}^{t} f(a_i, b_i, g)}{t} \geq 0.5\right]$$

Using the known differentially expressed genes in the data generation process, we compare the estimation of differential expression with the known value of differential expression. Suppose that

$$t(g) = \mathbf{1}[\text{g is differentially expressed}]$$

and we know that the number of False Positives FP is

$$\text{FP} = \sum_{g=1}^{k} \mathbf{1}\left[\mathbf{1}[\tilde{f}(g) = 1] \text{ and } \mathbf{1}[t(g) = 0]\right]$$

the number of True Positives TP is

$$\text{TP} = \sum_{g=1}^{k} \mathbf{1}\left[\mathbf{1}[\tilde{f}(g) = 1] \text{ and } \mathbf{1}[t(g) = 1]\right]$$

the number of False Negatives FN is

$$\text{FN} = \sum_{g=1}^{k} \mathbf{1}\left[\mathbf{1}[\tilde{f}(g) = 0] \text{ and } \mathbf{1}[t(g) = 1]\right]$$

and the number of True Negatives TN is

$$\text{TN} = \sum_{g=1}^{k} \mathbf{1}\left[\mathbf{1}[\tilde{f}(g) = 0] \text{ and } \mathbf{1}[t(g) = 0]\right]$$

and we thus have the False Discovery Rate FDR is

$$\text{FDR} = \frac{\text{FP}}{\text{FP + TP}}$$

that the Precision Rate is

$$\text{Precision} = \frac{\text{TP}}{\text{FP + TP}}$$

the False Positive Rate FPR is

$$\text{FPR} = \frac{\text{FP}}{\text{FP + TN}}$$

and the True Positive Rate TPR / Recall is

$$\text{TPR} = \frac{\text{TP}}{\text{TP + FN}}$$

Using these metrics, across ground truth predictions, single VAE architecture predictions, and ensembled VAE predictions (for multiple numbers of VAEs), we compute

1. FDR curves for a variety of thresholds
2. ROC curves for a variety of thresholds
3. Precision-Recall curves for a variety of thresholds

For two cells, we say that they differentially express gene g at level δ if their log-fold change in gene expression of g exceeds a threshold δ. Since the noiseless estimate of the expression of gene g in cell i is given by $h_{i,g}$, we obtain the binary function: $DE(i, j, g, \delta) = 1(|logh_{i,g}/h_{j,g}| \geq \delta)$. We extend this definition to the level of clusters of cells.

For two clusters of cells, cluster a and cluster b, we define the DE-score of g at level δ as $DE - score(a, b, g, \delta) = E[I \in a; j \in b]DE(i, j, g, \delta)$. We decide to binarize this score using a majority vote, so $DE(a, b, g, \delta) = 1 if DE - score(a, b, g, \delta) > 0.5 else 0$.

Because h itself is a noisy representation of the true gene DE status, we expect a practitioner having access to the hidden variables h to till falsely classify some genes as DE. The incurred FDR is called the ground truth FDR. We expect this phenomena to be even stronger for a model only observing noisy samples of h. The posterior FDR summarizes the error rate. Both FDR are computed with respect to the true DE-gene status from the generative process.

Experiments and Results as of May 2022

From these architectures, we created ensembled VAE architectures (where the reconstructions of separately trained VAEs are averaged when reconstructing a dataset) of size 1 VAE, 3 VAEs, 5 VAEs, and 10 VAEs; we compared the performance of these along with 1 VAE architectures with larger hidden unit sizes. The default encoder has 256 hidden units and the default decoder has 128 hidden units; two other architectures have twice as many and four times as many hidden units. We explored performance of DE gene detection on datasets with 100, 200, 500, 1000, and 2000 genes, with 1000 cells (with some experiments later on with fewer cells, 300); each dataset had a mean difference of one of 2, 3, 4, or 5, with a standard deviation of 0.08, 0.16, 0.32, 0.64, and 1.28. We observe that generally all VAEs model the ground truth very well, especially when a training loop with 10 epochs is used to train, with validation on a separate dataset generated with the same parameters and early stopping when validation loss does not consistently decrease. We model FDR curves, ROC curves, and Precision-Recall curves. There are some specific instances when performance of the ensembles as measured by ROC curves exceeds the large VAE model performances, such as a mean difference (scale factor) of 5 and a standard deviation of 0.32. Results are apparently consistent across different gene numbers. A lot of differential performance of different architectures on these datasets can be attributed to run-to-run variations, as evidenced by intersecting error bars shown on multi-run combined AUROC plots comparing performance for constant standard deviations and varying scale factors. More investigations are necessary in different ranges of the parameters, such as scale factors and standard deviations, as well as different variations of larger single VAE hidden unit sizes. Additionally, a more complex dataset structure would be worth exploring as these datasets are more biologically realistic, and more consistent improvements of ensembled architectures vs single VAEs could be possible.

All the code is included in the lab GitHub under the bayesianuncertainty repo. Plots of FDR, ROC, and Precision-Recall are in vae-final-edits.ipynb, and the most up-to-date architectures and combined AUROC plots are included in vae-final-less-cell-exp.ipynb.

References

[1] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[2] Romain Lopez, Jeffrey Regier, Michael B. Cole, Michael I. Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15(12):1053–1058, Dec 2018.

[3] Romain Lopez, Pierre Boyeau, Nir Yosef, Michael Jordan, and Jeffrey Regier. Decision-making with auto-encoding variational bayes. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 5081–5092. Curran Associates, Inc., 2020.

[4] Carl Doersch. Tutorial on variational autoencoders, 2021.