Variational Inference Methods for Single-Cell Genomics

Anonymous authors

Abstract

The analysis of single-cell genomics data aims to characterize cellular heterogeneity from high-dimensional, sparse, and noisy count measurements. Many statistical approaches in this regard can be expressed as hierarchical latent-variable models to capture the underlying biological structure. However, conducting statistical inference with these methods involves estimating posterior distributions over latent variables, a process that can be computationally challenging due to the scale of single-cell datasets and the use of non-Gaussian likelihoods. Variational inference (VI) addresses this challenge by approximating the posterior with a tractable distribution. We here present variational inference from a modern machine learning perspective and demonstrate its application to single-cell genomics. We first connect it to the EM algorithm and develop a unified view of scalable approaches, including stochastic, black-box, and amortized variational inference. To bridge theory and practice, we demonstrate how these methods can be implemented within the probabilistic programming framework of Pyro and apply these methods to real data for dimension reduction and temporal trajectory inference. We also discuss how modern coding agents can assist in implementation while leaving model formulation and approximation diagnosis to the researcher. Thus, we provide a practical guide to scalable variational inference for single-cell data analysis. The code for reproducing the simulation and real data analysis results is available at https://anonymous.4open.science/r/Variational_Inference_Tutorial-F5AC.

Keywords: Variational inference, single-cell genomics, latent variable models, computational statistics, probabilistic programming

Mathematics Subject Classification (2020): 62-01, 62F15, 62R07, 62P10

1 Introduction

Single-cell RNA sequencing (scRNA-seq) technologies have transformed the study of complex tissues by enabling molecular measurements at cellular resolution (Gawad et al., 2016). In contrast to bulk assays, which average signals across heterogeneous cell populations within a single sample, single-cell genomics enables the identification of rare cell populations (Xu et al., 2024), the detection of disease-associated cellular processes (Jagadeesh et al., 2022), and the prediction of drug sensitivity within highly heterogeneous tumor subpopulations (Chawla et al., 2022). This unprecedented resolution also leads to significant statistical challenges: the data are large-scale, high-dimensional, sparse, noisy, and strongly confounded by technical effects such as platform effects and batch structure (Korsunsky et al., 2019; Agarwal et al., 2020).

A unifying perspective is that many statistical models used in single-cell genomics data analysis can be formulated as probabilistic latent variable models (Blei, 2014). Typically, these methods seek a low-dimensional latent representation that captures biologically meaningful variation,

while additional components represent nuisance variation such as batch effects, multimodal background, or spatial dependence (Lin et al., 2019). Under this broad framework, classical factor models (Bartholomew et al., 2011), generalized mixed models (Huber et al., 2004), and modern deep generative models (Maaløe et al., 2019) can all be viewed as different realizations of probabilistic modeling.

Posterior inference for latent variable models can uncover hidden structures of interest, forming the basis for numerous downstream tasks. These tasks include, but are not limited to, dimension reduction, clustering analysis, batch correction, and multimodal integration. In simple latent variable models (e.g., Gaussian mixtures (Bishop, 2006)), the posterior distribution can be computed exactly by the expectation-maximization (EM) algorithm (Dempster et al., 1977). However, unlike continuous measurements, single-cell genomics datasets primarily consist of discrete and even overdispersed counts. Such models for single-cell genomics rarely enjoy such tractability. For count-based likelihoods, hierarchical structure, or nonlinear decoders, the posterior distribution typically involves high-dimensional integrals that are analytically intractable.

Markov chain Monte Carlo (MCMC) methods directly target the posterior distribution by constructing Markov chains with the desired stationary distribution (Neal, 1993; Brooks, 1998), but are often too computationally expensive for large-scale datasets. Thus, the central challenge is to retain the flexibility of probabilistic models while developing inference procedures that remain computationally feasible.

As an efficient alternative, variational inference (VI) addresses this challenge by converting posterior approximation into an optimization problem (Blei et al., 2017). Rather than computing the exact posterior directly, VI introduces a tractable family of approximating distributions for the posterior distribution. For single-cell genomics data analysis, this framework has motivated a variety of methods. For instance, scVI (Lopez et al., 2018) establishes the foundational framework for scalable and generative modeling of sparse transcriptomic count data. totalVI (Gayoso et al., 2021) integrates multimodal transcriptomic datasets by fusing RNA and protein measurements while explicitly accounting for antibody background noise. scArches (Lotfollahi et al., 2022) addresses decentralized data integration by transfer learning to map new query datasets onto established reference atlases. STAMP (Zhong et al., 2024) extends probabilistic modeling into spatial transcriptomics by performing non-negative matrix factorization to discover spatially aware gene modules. MrVI (Boyeau et al., 2025) models sample-level heterogeneity through multi-resolution latent representations to stratify samples and identify cell-state-specific differential expression. As a result, VI has become one of the core computational tools behind modern probabilistic analysis of large-scale single-cell data.

The goal of this paper is to provide a methodological introduction to variational inference with its application to single-cell genomics. Section 2 establishes the variational inference fundamentals in latent variable models. Section 3 introduces AI-based variational inference in modern machine learning to address computational challenges. Section 4 uses three simulation studies to illustrate the performance of variational inference as introduced in the previous sections. Section 5 further provides two real data examples in scRNA-seq data analysis for dimension reduction and temporal trajectory inference. The paper concludes with a discussion in Section 6. Technical details are provided in the Appendix.

2 Variational Inference Fundamentals

2.1 Latent Variable Models and the EM Algorithm

Many probabilistic models can be formulated as latent variable models, in which the observed data are modeled jointly with unobserved variables that encode hidden structure (Bishop, 2006). Let $X = \{x_1, \cdots, x_N\}$ denote the observed data, let Z denote the latent variables, and let θ denote unknown model parameters. We should remark that, for full Bayesian models, the model parameter θ is also treated as latent variables with a given prior, and can be inferred together with the latent variable Z. In contrast, we mainly adopt an empirical Bayes perspective throughout this paper, which remains the latent variables Z random under flexible hierarchical model, while θ is treated as unknown and fixed that is to be estimated from the data (Efron, 2012). Then, the complete-data joint distribution is given by

$$p(X, Z|\theta) = p(Z|\theta)p(X|Z, \theta).$$

Because Z is unobserved, learning based on the observed data requires the marginal likelihood $p(X|\theta) = \int p(X, Z|\theta)\mathrm{d}Z$, where $\int \mathrm{d}Z$ should be interpreted as integration for continuous latent variables and summation for discrete latent variables. However, directly maximizing the marginal likelihood $p(X|\theta)$ is often intractable due to the integral over the latent variables Z, especially when Z is high-dimensional or $p(X|Z, \theta)$ has a complex form (Neal, 1993).

The Expectation-Maximization (EM) algorithm provides an efficient iterative framework to address this issue (Dempster et al., 1977). Instead of directly maximizing the intractable marginal likelihood $\log p(X|\theta)$, the key idea of the EM algorithm is to iteratively optimize a surrogate objective based on the complete-data log-likelihood $\log p(X, Z|\theta)$. At each iteration, the EM algorithm constructs the function

$$Q(\theta|\theta^{(t)}) = \mathbb{E}_{p(Z|X,\theta^{(t)})}[\log p(X, Z|\theta)].$$

The E-step involves the posterior distribution of the latent variables under the current parameter $\theta^{(t)}$, and takes the expectation of the complete-data log-likelihood with respect to this posterior. The M-step then treats this expected complete-data log-likelihood as a function of θ and updates the parameters by maximizing:

$$\theta^{(t+1)} = \arg\max_{\theta} Q(\theta|\theta^{(t)}).$$

This procedure is effective only when both the posterior $p(Z|X, \theta^{(t)})$ and the maximization of $Q(\theta|\theta^{(t)})$ are tractable. This typically holds for models with conjugate exponential-family structure (McLachlan and Krishnan, 2008), such as Gaussian Mixture models (GMMs) (Bishop, 2006), hidden Markov models (HMMs) (Rabiner, 1989), and probabilistic PCA (PPCA) (Tipping and Bishop, 1999), where both the E-step and M-step admit closed-form updates.

In many modern models, this conjugacy is broken, making the posterior intractable, which in turn makes classical EM infeasible. This limitation motivates the use of variational inference.

2.2 Variational Inference

Variational Inference (VI) is a general framework for approximate posterior inference in latent variable models, particularly when the posterior distribution $p(Z|X,\theta)$ is intractable (Jordan et al., 1999; Wainwright and Jordan, 2008; Blei et al., 2017). Instead of relying on the exact posterior computation, VI introduces a tractable family of distributions $q(Z)$ to approximate $p(Z|X,\theta)$.

This approximation is formulated as an optimization problem, where the goal is to find a distribution $q(Z)$ within the chosen family that is closest to the true posterior by minimizing the Kullback–Leibler (KL) divergence: $\mathrm{KL}(q(Z)||p(Z|X,\theta))$(Bishop, 2006). This objective depends on the true posterior $p(Z|X,\theta)$, which involves the marginal likelihood $p(X|\theta)$ and is therefore difficult to evaluate directly. To obtain a computable objective, variational inference instead relies on a decomposition of the log marginal likelihood (Jordan et al., 1999). For any distribution $q(Z)$, we can write

$$\log p(X|\theta) = \mathcal{L}(q,\theta) + \mathrm{KL}\big(q(Z) \,\|\, p(Z|X,\theta)\big),$$

where

$$\mathcal{L}(q,\theta) = \mathbb{E}_{q(Z)}[\log p(X,Z|\theta)] - \mathbb{E}_{q(Z)}[\log q(Z)]$$

is the evidence lower bound (ELBO); see the derivation in Appendix A.1. Since the KL divergence is non-negative, $\mathcal{L}(q,\theta)$ provides a lower bound on $\log p(X|\theta)$. In addition, because $\log p(X|\theta)$ does not depend on $q(Z)$, minimizing $\mathrm{KL}(q(Z) \,\|\, p(Z|X,\theta))$ is equivalent to maximizing ELBO with respect to $q(Z)$ (Blei et al., 2017). This decomposition also clarifies the relationship between VI and the EM algorithm: EM can be viewed as exact coordinate ascent on the same ELBO, with the E-step setting $q(Z) = p(Z|X,\theta)$, which drives the KL term to zero and makes the bound tight, and the M-step maximizing $\mathcal{L}(q,\theta)$ with respect to θ. Variational inference addresses the setting in which this exact E-step is unavailable: when $p(Z|X,\theta)$ has no tractable form, $q(Z)$ is restricted to a tractable family $\mathcal{Q}$, and the E-step is replaced by maximizing $\mathcal{L}(q,\theta)$ over $q \in \mathcal{Q}$, so that posterior approximation reduces to maximizing the ELBO. In practice, this optimization requires specifying a tractable family for $q(Z)$.

To this end, a common choice is mean-field variational inference (MFVI) (Bishop, 2006; Blei et al., 2017), which assumes a factorized form

$$q(Z) = \prod_i q_i(Z_i).$$

This leads to coordinate ascent variational inference (CAVI) (Bishop, 2006; Blei et al., 2017), where each variational factor is updated by maximizing the ELBO while holding the others fixed:

$$\log q_i^*(Z_i) = \mathbb{E}_{q_{-i}(Z_{-i})}\big[\log p(X,Z|\theta)\big] + \text{const},$$

which is derived in Appendix A.1. Compared with the exact E-step in EM, restricting $q(Z)$ to a factorized family is what makes the resulting updates tractable, but it typically leads to underestimating the posterior variance; see Appendix A.2 for the illustration.

More generally, we refer to the approach above as vanilla variational inference, that is, the closed-form coordinate updates are available in conditionally conjugate models after deriving the

required model-specific expectations (Blei et al., 2017). However, in models with global variables and many observation-specific local variables, a global update typically requires revisiting all observations and refreshing the corresponding local factors. In non-conjugate models, the relevant expectations and coordinate updates are usually unavailable in closed form, so ELBO maximization is often reformulated as stochastic gradient optimization over a parameterized variational distribution $q_\lambda(Z)$:

$$\nabla_\lambda \mathcal{L}(\lambda) = \nabla_\lambda \mathbb{E}_{q_\lambda(Z)}\big[\log p(X, Z|\theta) - \log q_\lambda(Z)\big],$$

where differentiating the ELBO requires accounting for changes in both the integrand and the sampling distribution. These issues create three computational bottlenecks: full-data updates, non-conjugate or model-specific derivations, and per-observation local variational optimization. These challenges motivate the development of modern scalable variational inference methods with AI tools that aim to reduce the computational burden of inference.

3 Scalable Variational Inference with Modern Machine Learning Tools

In the previous section, we introduced the ELBO as the objective function of variational inference and discussed settings in which it can be optimized directly by coordinate ascent. This is often feasible when the model is conditionally conjugate, so that the relevant expectations are available in closed form. Even in such conjugate settings, however, vanilla variational inference typically requires model-specific derivations of the variational updates, and full-data updates can be computationally expensive in large-scale applications. In more general non-conjugate settings, such updates are often unavailable in closed form. This motivates us to reformulate ELBO maximization as a gradient-based optimization problem. Recent advances in machine learning have addressed these challenges by combining variational inference with stochastic optimization, automatic differentiation, and neural-network-based approximation. This enables scalable inference in complex probabilistic models and forms the foundation of many recent methods used in machine learning for single-cell genomics.

3.1 Stochastic Variational Inference

Let us motivate the problem with global and local latent variables simultaneously, which happens quite often, including but not limited to single-cell genomics. Recall that $X = \{x_1, \cdots, x_N\}$ is the observed data (e.g., for every single cell $i = 1, \cdots, N$). We assume $Z = \{z_1, \cdots, z_N\}$ denotes the collection of latent representations for each cell i. Denote β as the global latent variable shared across observations (e.g., gene-level decoder parameters, dispersion parameters, or batch-specific effects). As mentioned before, the mean-field approximation takes the form

$$q(\beta, Z) = q(\beta|\lambda) \prod_{i=1}^{N} q(z_i|\phi_i),$$

where λ are global variational parameters and ϕ_i are local variational parameters. This leads to the ELBO given by

$$\mathcal{L}(\lambda, \phi_{1:N}) = \mathbb{E}_q\Big\{ \log p(\beta) - \log q(\beta|\lambda)\Big\} + \sum_{i=1}^{N} \mathbb{E}_q\Big\{ \log p(x_i, z_i|\beta) - \log q(z_i|\phi_i)\Big\}.$$

For a conditionally conjugate exponential-family model, the complete conditional distribution of the global variable β can be written as

$$p(\beta|X, Z) \propto h(\beta) \exp\left\{\Big(\alpha + \sum_{i=1}^{N} s(x_i, z_i)\Big)^\top t(\beta)\right\},$$

where α is the prior natural parameter, $t(\beta)$ is the sufficient statistic of β, and $s(x_i, z_i)$ denotes the contribution of observation i to the natural parameter. This leads to the global update

$$\lambda^{(t+1)} = \alpha + \sum_{i=1}^{N} \mathbb{E}_{q_{\phi_i^*(\lambda^{(t)})}}\{s(x_i, z_i)\}$$

and the local update

$$\phi_i^*(\lambda^{(t)}) = \arg\max_{\phi_i} \mathbb{E}_{q_{\lambda^{(t)}}(\beta) q_{\phi_i}(z_i)}\{ \log p(x_i, z_i|\beta) - \log q_{\phi_i}(z_i)\}$$

for every local variational parameter ϕ_i given the current global variational parameter $\lambda^{(t)}$. This can also be exactly derived by optimizing the profiled ELBO by natural gradient ascent (Amari, 1998); see Appendix A.3 for the details. Note that this is exactly the computational bottleneck of CAVI, as updating the global variational parameter λ once requires refreshing the local variational parameters for all N observations. When N is very large, most of the computation is spent updating local quantities before the global parameter can move even once.

As an efficient alternative, stochastic variational inference (SVI) replaces this full-data natural-gradient step with a mini-batch stochastic natural-gradient update (Hoffman et al., 2013):

$$\widehat{\lambda}_t = \alpha + \frac{N}{|\mathcal{B}_t|} \sum_{i \in \mathcal{B}_t} \mathbb{E}_{q_{\phi_i^*(\lambda^{(t)})}}\{s(x_i, z_i)\},$$

where $\mathcal{B}_t$ is the subsampled mini-batch dataset. Then $\widehat{g}_t = \widehat{\lambda}_t - \lambda^{(t)}$ is an unbiased estimate of the natural gradient, and the global variational parameter is updated by $\lambda^{(t+1)} = \lambda^{(t)} + \rho_t \widehat{g}_t = (1 - \rho_t)\lambda^{(t)} + \rho_t \widehat{\lambda}_t$, where $\rho_t \in (0, 1]$ is the learning rate. Therefore, unlike CAVI, SVI no longer waits for a full sweep through all N local variational parameters before updating the global variational parameter. Instead, it updates the local variational parameters only for the current mini-batch, uses them to construct a noisy but unbiased estimate of the natural gradient, and immediately updates the global variational parameter. This makes variational inference scalable to massive datasets and is particularly attractive in single-cell genomics, where the number of cells can be very large.

3.2 Black-Box Variational Inference

For non-conjugate models, we should remark that the conditional distribution is typically not tractable in closed form anymore. Recall that our target in variational inference is to maximize the ELBO. Therefore, we can avoid model-specific derivations by parameterizing the variational distribution. Instead, we then maximize the ELBO by gradient-based optimization. More specifically, we have the ELBO

$$\mathcal{L}(\phi) = \mathbb{E}_{q_\phi(Z)}\Big\{ \log p(X, Z) - \log q_\phi(Z) \Big\},$$

where $q_\phi(Z)$ is the variational distribution. Here ϕ is the variational parameter, for example the mean and variance of the variational distribution. Note that we can compute the gradient with $\nabla_\phi \mathcal{L}(\phi) = \mathbb{E}_{q_\phi}\big[\nabla_\phi \log q_\phi(Z)\{\log p(X, Z) - \log q_\phi(Z)\}\big]$; see Appendix A.4 for the derivation details. This enables us to estimate the gradients using Monte Carlo methods as

$$\widehat{\nabla_\phi \mathcal{L}} = B^{-1} \sum_{b=1}^{B} \Big[\nabla_\phi \log q_\phi(Z_b) \Big\{ \log p(X, Z_b) - \log q_\phi(Z_b) \Big\} \Big],$$

as long as we can (1) sample Z_b from the variational distribution $q_\phi(Z)$ for $b = 1, \cdots, B$, (2) evaluate the score function $\nabla_\phi \log q_\phi(Z)$, and (3) calculate the joint density $\log p(X, Z)$. Then, stochastic gradient methods can be applied to optimize the ELBO (Ranganath et al., 2014). Additionally, although the estimated gradient $\widehat{\nabla_\phi \mathcal{L}}$ is an easily obtained unbiased estimator for the true gradient, Ranganath et al. (2014) noted that its variance can be a serious problem. To address this issue, methods such as control variates (Nelson, 1990), Rao-Blackwellization (Ranganath et al., 2014), and the reparameterization trick for continuous latent variables (Kingma and Welling, 2014; Rezende et al., 2014) are developed and applied.

3.3 Amortized Variational Inference

In models with many local latent variables, even when stochastic gradients are available, classical MFVI still requires optimizing a separate local variational parameter ϕ_i for the latent variable z_i associated with each observation x_i. This can be computationally expensive. Amortized variational inference replaces these free local parameters with a shared function

$$\phi_i = f_\psi(x_i),$$

where f_ψ is typically a neural network parameterized by ψ. The variational family then becomes $q_\psi(z_i|x_i)$, so that the cost of inference is shifted from per-observation optimization to learning the shared map f_ψ. Then the ELBO becomes

$$\begin{aligned}\mathcal{L}(\lambda, \psi, \omega) = {} & \mathbb{E}_q\big\{ \log p_\omega(\beta) - \log q_\lambda(\beta) \big\} \\ & + \sum_{i=1}^{N} \mathbb{E}_q\Big\{ \log p_\omega(z_i|\beta) + \log p_\omega(x_i|z_i, \beta) - \log q_\psi(z_i|x_i) \Big\},\end{aligned}$$

where $q = q_{\lambda,\psi}(\beta, Z|X)$. Thus, instead of optimizing N separate local variational parameters $\phi_1, \ldots, \phi_N$, we optimize a single shared map f_ψ. After training, the posterior approximation for a new cell requires only a forward pass through the inference network. By sharing information across observations through the neural network, amortized inference makes posterior approximation much more scalable for large datasets.

Example: Variational Autoencoders. The canonical example of amortized VI is the variational autoencoder (VAE), whose encoder-decoder structure has become one of the most common modeling frameworks in single-cell genomics (Lopez et al., 2018; Grønbech et al., 2020; Xu et al., 2021). For each cell i, let $x_i = (x_{i1}, \ldots, x_{iG})$ denote its observed gene-count vector and let $z_i \in \mathbb{R}^K$ denote its latent representation. Consider

$$z_i \sim \mathcal{N}(0, I_K),$$
$$x_{ig}|z_i \sim \text{Poisson}(\rho_{\omega g}(z_i)), \qquad g = 1, \ldots, G,$$

where I_K refers to the $K \times K$ identity matrix, and $\rho_\omega(z_i) = (\rho_{\omega 1}(z_i), \ldots, \rho_{\omega G}(z_i))$ is the decoder network output parameterized by ω, with positive entries. Thus, the latent-variable generative model is $p_\omega(x_i, z_i) = p(z_i)p_\omega(x_i|z_i)$, where $p(z_i)$ is the prior and $p_\omega(x_i|z_i)$ is the Poisson decoder. Because the posterior distribution $p_\omega(z_i|x_i)$ is generally intractable, we introduce the variational approximation $q_\psi(z_i|x_i)$, where ψ parameterizes an encoder network that maps the observed count vector x_i to the parameters of the approximate posterior. A common choice is a Gaussian encoder, that is,

$$(\mu_\psi(x_i), \sigma^2_\psi(x_i)) = g_\psi(x_i),$$
$$q_\psi(z_i|x_i) = \mathcal{N}\Big(z_i; \mu_\psi(x_i), \text{diag}\{\sigma^2_\psi(x_i)\}\Big),$$

where g_ψ is the encoder network.

Then, the corresponding ELBO is

$$\mathcal{L}(x_i; \omega, \psi) = \mathbb{E}_{q_\psi(z_i|x_i)}\Big\{\log p_\omega(x_i|z_i)\Big\} - \text{KL}\Big(q_\psi(z_i|x_i)\Big\|p(z_i)\Big).$$

This objective has a clear interpretation. The first term is a reconstruction term, which encourages the decoder to explain the observed count vector x_i well from the latent variable z_i. Under the Poisson decoder, this term is the conditional Poisson log-likelihood of the observed gene counts. The second term regularizes the encoder by keeping the approximate posterior close to the prior, thereby stabilizing the latent space and enabling generation by sampling from $p(z_i)$. To optimize the ELBO efficiently, the VAE uses the reparameterization trick:

$$z_i = \mu_\psi(x_i) + \sigma_\psi(x_i) \odot \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, I),$$

where $\odot$ denotes elementwise multiplication and ε is an auxiliary noise variable independent of x_i. This expresses sampling from $q_\psi(z_i|x_i)$ as a differentiable transformation of parameter-free noise, allowing low-variance stochastic gradients of the ELBO with respect to both ω and ψ. Then, for a dataset $X = \{x_1, \ldots, x_N\}$, we train this VAE by maximizing $\sum_{i=1}^N \mathcal{L}(x_i; \omega, \psi)$, usually with

mini-batches and automatic differentiation. In this way, the VAE learns the decoder parameters ω and the inference network parameters ψ simultaneously, so that amortized inference becomes a procedure for "learning to infer." Thus, the VAE provides both a generative model for high-dimensional single-cell count data and a scalable posterior approximation for cell-specific latent states.

3.4 Probabilistic Programming with Pyro

Modern probabilistic programming provides a practical interface for applying the gradient-based variational inference methods introduced above to general probabilistic models. Pyro is a probabilistic programming language built on PyTorch (Bingham et al., 2019). Its goal is to separate model specification from posterior inference: the user specifies a generative model and a variational approximation, while Pyro traces the corresponding stochastic computation. This is particularly useful for non-conjugate likelihoods, constrained latent variables, hierarchical random effects, and amortized guides involving neural networks.

In Pyro, the generative model $p_\theta(X, Z)$ is written as a Python function, denoted by `model`, and the variational approximation $q_\phi(Z|X)$ is written as another Python function, denoted by `guide`. A random variable (e.g., the latent variable) is introduced by `pyro.sample`, a trainable parameter is registered by `pyro.param` for the point estimates of unknown parameters, and repeated conditionally independent observations are represented by `pyro.plate`. Once the model and guide are specified, variational inference is carried out by

```
optimizer = pyro.optim.Adam({"lr": 1e-2})
svi = pyro.infer.SVI(model, guide, optimizer, loss=pyro.infer.Trace_ELBO())
for step in range(num_steps):
    loss = svi.step(data)
```

Here, we should remark that the term `SVI` in Pyro is different from the conjugacy-based SVI discussed in Section 3.1. Instead, `SVI` here is a more general interface for stochastic optimization of an ELBO estimator in the black-box variational inference framework using automatic differentiation.

4 Simulation Studies

The simulation studies are designed to validate and illustrate the features of variational inference. Study 1 uses a conjugate Gaussian model to show that mean-field CAVI can estimate posterior means accurately but may underestimate posterior uncertainty. Study 2 uses a topic model with many cell-specific local variables to show why mini-batch SVI improves scalability. Study 3 uses a non-conjugate generalized linear mixed model to show how black-box gradient-based VI and probabilistic programming make inference possible when closed-form updates are unavailable.

Study 1: Linear Mixed Model. The objective of this study is to illustrate the accurate estimation of posterior means using CAVI in a conjugate Gaussian model, but leads to un-

derestimate the posterior variance and hence a lower coverage rate. To this end, we consider the linear mixed model as an example to compare VI and the EM algorithm (Dempster et al., 1977) and its variants (parameter-expanded EM (PX-EM) algorithm (Liu et al., 1998) and minorization-maximization (MM) algorithm (Zhou et al., 2019)). Specifically, assume

$$y = W\alpha + X\beta + \varepsilon, \qquad \beta \sim \mathcal{N}(0, \sigma_b^2 I_p), \qquad \varepsilon \sim \mathcal{N}(0, \sigma_e^2 I_n),$$

where $W \in \mathbb{R}^{n\times q}$ and $X \in \mathbb{R}^{n\times p}$ are fixed and random effect covariates, respectively. We use the mean-field variational family $q(\beta) = \prod_{j=1}^p \mathcal{N}(\beta_j; m_j, s_j^2)$ to approximate the posterior of β. Then, we have the CAVI updates

$$\widehat{s}_j^2 = \left(\frac{x_j^\top x_j}{\widehat{\sigma}_e^2} + \frac{1}{\widehat{\sigma}_b^2}\right)^{-1}, \qquad \widehat{m}_j = \frac{\widehat{s}_j^2}{\widehat{\sigma}_e^2} x_j^\top \left(y - W\widehat{\alpha} - \sum_{\ell\neq j} x_\ell \widehat{m}_\ell\right),$$

where x_j is the jth column of X for $j = 1, \cdots, p$. Given the variational parameters, the model parameter updates are given by $\widehat{\alpha} = (W^\top W)^{-1} W^\top (y - X\widehat{m})$, $\widehat{\sigma}_b^2 = p^{-1}\sum_{j=1}^p (\widehat{m}_j^2 + \widehat{s}_j^2)$, and $\widehat{\sigma}_e^2 = n^{-1}\{\|y - W\widehat{\alpha} - X\widehat{m}\|^2 + \sum_{j=1}^p \widehat{s}_j^2 x_j^\top x_j\}$. The algorithm alternates between these variational updates and parameter updates until convergence with detailed derivation in Appendix A.5.1. In the simulation, this mean-field CAVI estimator is compared with EM, PX-EM, and MM algorithms under the same data-generating mechanism with fixed effect dimension $q = 10$, random effect dimension $p = 10$, and an AR(1) correlated design matrix. The numerical simulation is then replicated $R = 100$ times, and the results are summarized in Figure 1A. Specifically, for a parameter $\theta \in \mathbb{R}^d$ (e.g., fixed parameter α or random variable realization $\beta^{(r)}$ for $r = 1, \cdots, R$), we calculate $\mathrm{MSE}(\widehat{\theta}) = d^{-1}\|\widehat{\theta} - \theta\|_2^2$ and report the log-transformed values. The nominal 95% posterior credible intervals for various methods are also shown in Figure 1A. We find that CAVI achieves similar estimation accuracy compared with other algorithms. We also note that, the coverage rate of 95% credible interval of EM-type algorithms is well calibrated. The coverage of CAVI is lower, since MFVI typically underestimates the posterior variance.

Study 2: Single-cell Latent Dirichlet Allocation. The objective of this study is to illustrate the usefulness of SVI when a probabilistic model contains many local latent variables. Here, we consider a single-cell latent Dirichlet allocation (LDA) model (Pritchard et al., 2000; Blei et al., 2003). Specifically, let $X = (x_{ig}) \in \mathbb{N}^{N\times G}$ be a single-cell count matrix, where x_{ig} is the number of transcripts of gene g in cell i, and $m_i = \sum_{g=1}^G x_{ig}$ is the library size. Assume there exist a total of K gene programs. For each gene program k, we have the gene composition of program k as $\xi_k \sim \mathrm{Dirichlet}(\omega)$. For each cell i, $\eta_i \sim \mathrm{Dirichlet}(a)$ is the program mixture. For every transcript j in cell i, we first select the gene program by $z_{ij}|\eta_i \sim \mathrm{Categorical}(\eta_i)$, following by the observed gene identity $w_{ij}|z_{ij} = k, \xi \sim \mathrm{Categorical}(\xi_k)$. The observed count data is then generated by $x_{ig} = \sum_{j=1}^{m_i} I(w_{ij} = g)$.

Note that the exact posterior for this model requires summing over K^{m_i} possible topic assignments for cell i, which is infeasible. For the variational inference, we use the mean-field

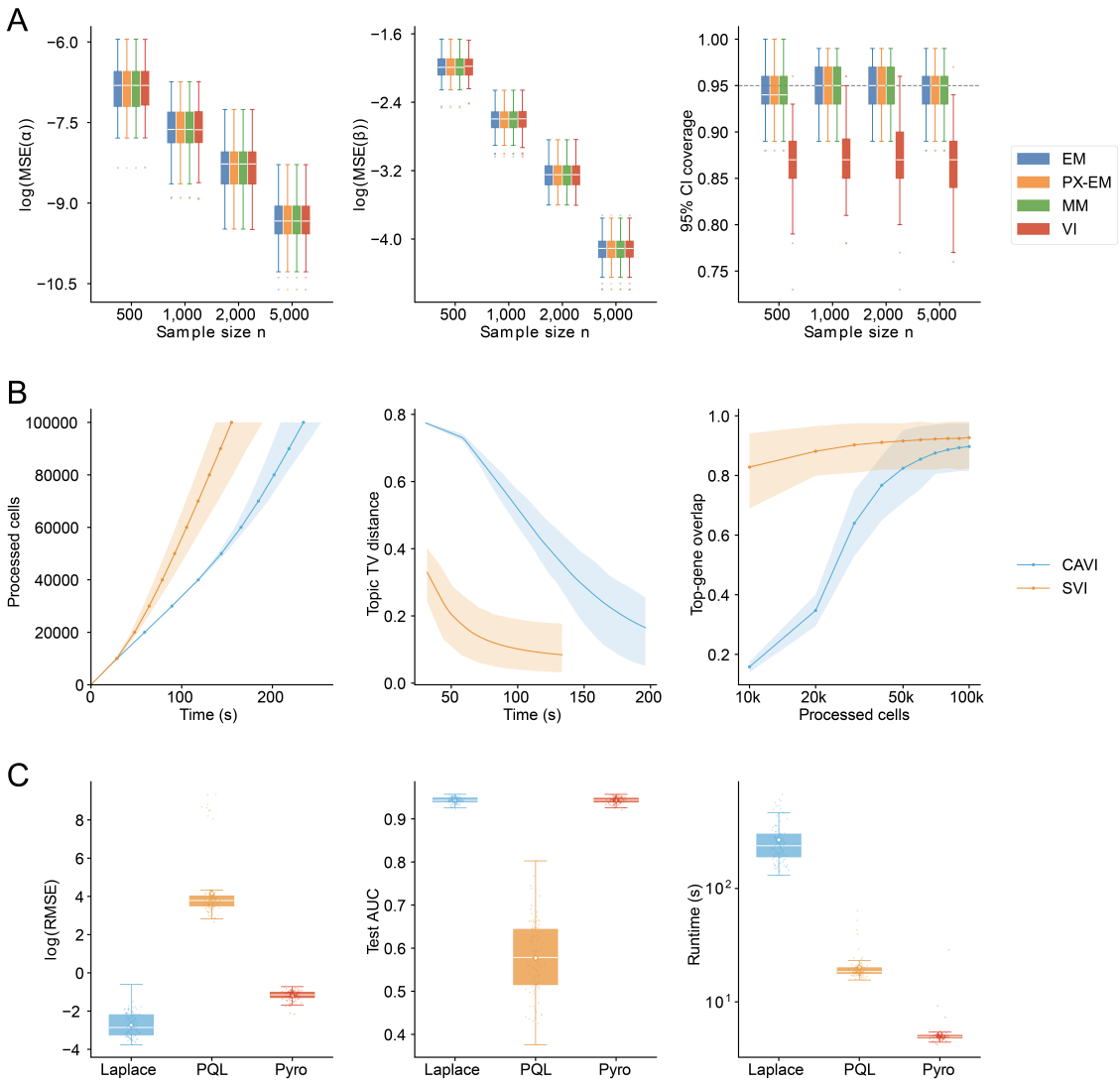

Figure 1: Simulation studies comparing variational inference and alternative inference methods in various models. (A) Comparison of CAVI, EM, PX-EM, and MM for the linear mixed model simulation. Left: mean squared errors of fixed-effect estimates. Middle: mean squared errors of random-effect estimates. Right: coverage of 95% credible interval for the random effect. (B) Comparison of CAVI and SVI in the single-cell LDA simulation. Left: processed cells versus running time. Middle: topic total-variation distance versus running time. Right: top gene recovery versus processed cells. Shaded regions denote the 10th–90th percentile intervals across simulation replicates. (C) Comparison of Laplace approximation, PQL, and Pyro in the GLMM. Left: estimation error of the random effect. Middle: AUC evaluated on the test dataset. Right: running time of different methods.

family

$$q(\xi, \eta, z) = \prod_{k=1}^{K} \text{Dirichlet}(\xi_k; \lambda_k) \prod_{i=1}^{N} \left\{ \text{Dirichlet}(\eta_i; \gamma_i) \prod_{j=1}^{m_i} \text{Categorical}(z_{ij}; \varphi_{ij}) \right\}$$

to approximate the posterior. Given the current global parameter λ, let $r_{igk} = \text{softmax}_k(\mathbb{E} \log \eta_{ik} + \mathbb{E} \log \xi_{kg})$ denote the variational probability that a transcript of gene g in cell i is assigned to topic k. The corresponding full-data CAVI update for the global topics is $\lambda_{kg}^{\text{CAVI}} = \omega_g + \sum_{i=1}^{N} x_{ig} r_{igk}$.

This update is statistically natural but computationally expensive, since CAVI must refresh the local variational parameters for all N cells before one global update of λ. As an alternative, SVI replaces the full expected topic-gene counts with a rescaled mini-batch estimate. For a mini-batch $\mathcal{B}_t$ of size b_t, we have

$$\widehat{\lambda}_{kg}^{(t)} = \omega_g + \frac{N}{b_t} \sum_{i \in \mathcal{B}_t} x_{ig} r_{igk}^{(t)}, \qquad \lambda_{kg}^{(t+1)} = (1-\rho_t)\lambda_{kg}^{(t)} + \rho_t \widehat{\lambda}_{kg}^{(t)},$$

which can update the global variational parameter λ faster.

In the simulation, we generate count matrices from the well-specified LDA model with $N =$ 10,000 cells, $G =$ 5,000 genes, and $K = 10$ topics. The true topic-gene distributions are generated from Dirichlet($0.05\mathbf{1}_G$), the cell-specific topic mixtures from Dirichlet($0.2\mathbf{1}_K$), and the library sizes from a Poisson distribution with mean 500. We compare CAVI and SVI under the same processed-cell budget. Specifically, CAVI is run for 10 full passes over all cells, whereas SVI is run for 100 mini-batch updates with batch size 1,000. We report top-gene overlap and topic total-variation distance $\mathrm{TV}(\widehat{\xi}_k, \xi_k^0) = 2^{-1} \sum_{g=1}^{G} |\widehat{\xi}_{kg} - \xi_{kg}^0|$ in Figure 1B, showing that SVI updates the global topics more efficiently than full-data CAVI. This computational gain also leads to faster topic recovery, as SVI reduces the topic total-variation distance more quickly in wall-clock time and achieves high top-gene overlap after fewer processed cells (Figure 1B). This shows that SVI reduces much computational efforts by updating the global parameters more quickly.

Study 3: Generalized Linear Mixed Model. The objective of this study is to illustrate why black-box variational inference is useful. To this end, we consider a non-conjugate generalized linear mixed model, by extending the previous LMM to a binary response case with logit link, where the random effects enter through a grouped random-intercept design. Specifically, let $y_i \in \{0, 1\}$ denote the binary response, $w_i \in \mathbb{R}^q$ denote the fixed-effect covariates, and $g_i \in \{1, \ldots, p\}$ denote the group membership of observation i. Equivalently, $x_i \in \mathbb{R}^p$ is a one-hot vector with $x_{i,g_i} = 1$. We then consider

$$y_i | \alpha, \beta \sim \text{Bernoulli}\left\{\text{logit}^{-1}(w_i^\top \alpha + x_i^\top \beta)\right\} = \text{Bernoulli}\left\{\text{logit}^{-1}(w_i^\top \alpha + \beta_{g_i})\right\},$$

with independent Gaussian random effects $\beta_j \sim \mathcal{N}(0, \sigma_b^2)$ for $j = 1, \ldots, p$. Note that the Bernoulli likelihood is not conjugate to the Gaussian prior on β, so the closed-form EM and CAVI updates no longer apply. Instead, we should use the BBVI to optimize the ELBO for solving this model.

Regarding implementations, we use Pyro to estimate the GLMM to illustrate the practical usefulness of Pyro, which mirrors the probabilistic specification directly as follows.

```
def model(W, X, y=None):
    alpha = pyro.param("alpha", alpha0)
    sigma_b = pyro.param("sigma_b",sigma0,constraint=constraints.positive)
    beta = pyro.sample("beta",dist.Normal(zeros_p, sigma_b).to_event(1))
    fixed_eta = W.matmul(alpha)
    random_eta = torch.einsum("np,...p->...n", X, beta)
    eta = fixed_eta + random_eta
```

```
    with pyro.plate("obs", X.shape[0], dim=-1):
        pyro.sample("y", dist.Bernoulli(logits=eta), obs=y)
```

Here, `obs=y` refers to the observed binary responses, `logits=eta` means that Pyro models the Bernoulli probability as $\mathbb{P}(y_i = 1) = \text{logit}^{-1}(\eta_i)$, and the command `to_event(1)` tells Pyro to treat β as one multivariate latent variable rather than scalars. The guide is the mean-field Gaussian family as $q_\phi(\beta) = \mathcal{N}\{m, \text{diag}(s^2)\}$:

```
def guide(W, X, y=None):
    beta_loc = pyro.param("beta_loc", zeros_p)
    beta_scale = pyro.param("beta_scale",scale0,constraint=constraints.positive)
    pyro.sample("beta", dist.Normal(beta_loc, beta_scale).to_event(1))
```

Thus, Pyro handles the nonconjugate Bernoulli likelihood by combining automatic differentiation, Monte Carlo ELBO estimation, and the reparameterized Gaussian guide. The simulation study with sample size $N = 30,000$ and random effect $p = 200$ is replicated for 100 times, compared with the method of Laplace approximation that approximates the integral over random effects by second-order expansion, and the penalized quasi-likelihood (PQL) method that uses a working linear mixed model approximation (Breslow and Clayton, 1993).

We set the intercept in the simulation so that the proportion of positive responses in the dataset is about 9%. By Figure 1C, we find that the PQL method breaks down in this imbalanced setting. The Laplace approximation remains accurate but is much slower because it repeatedly solves for the conditional modes. Pyro achieves predictive accuracy comparable to Laplace while running much faster.

5 Real Data Applications on Single-cell Genomics

5.1 scPI for Dimension Reduction

In this section, we use the scPI framework (Ming et al., 2023) as a practical example to illustrate how different inference strategies affect probabilistic dimension reduction for scRNA-seq datasets. Specifically, we consider two representative latent variable models, Factor Analysis (FA) and Zero-Inflated Factor Analysis (ZIFA) (Pierson and Yau, 2015), and study how their performance changes under three inference paradigms: the classical EM algorithm, non-amortized variational inference, and amortized variational inference.

Suppose we have a gene expression matrix $X \in \mathbb{N}^{N\times D}$ for D genes measured across N cells. We transform the raw counts x_{ij} using $y_{ij} = \log(x_{ij} + 1)$, and perform dimension reduction on the resulting log-expression matrix $Y \in \mathbb{R}^{N\times D}$.

Factor Analysis. FA assumes that each observation $y_i \in \mathbb{R}^D$ is generated from a low-dimensional latent representation $z_i \in \mathbb{R}^K$, where $K \ll D$:

$$z_i \sim \mathcal{N}(0, I), \qquad y_i|z_i \sim \mathcal{N}(Wz_i + \mu, \Psi),$$

where $\mu \in \mathbb{R}^D$ is the mean vector, $W \in \mathbb{R}^{D\times K}$ is the loading matrix and $\Psi \in \mathbb{R}^{D\times D}$ is a diagonal noise covariance matrix. Equivalently, we have $y_i = Wz_i + \mu + \varepsilon_i$ with $\varepsilon_i \sim \mathcal{N}(0, \Psi)$. FA provides

a simple probabilistic formulation of dimension reduction, but does not explicitly account for the excessive zeros commonly observed in scRNA-seq data.

Zero-Inflated Factor Analysis. ZIFA extends FA by introducing a dropout mechanism. Let u_i denote the latent transformed expression before dropout and h_{ij} is a Bernoulli dropout indicator:

$$u_i|z_i \sim \mathcal{N}(Wz_i + \mu, \Psi), \qquad h_{ij}|u_{ij} \sim \text{Bernoulli}(p_{ij}), \qquad p_{ij} = \exp(-\lambda u_{ij}^2), \qquad \lambda > 0,$$

where p_{ij} denotes the dropout probability. The observed expression is then given by

$$y_{ij} = \begin{cases} 0, & h_{ij} = 1, \\ u_{ij}, & h_{ij} = 0. \end{cases}$$

This formulation allows observed zeros to arise through an explicit dropout process.

To compare the computational efficiency and scalability of different inference strategies, we conduct experiments on the 1.3 million mouse brain cell dataset (10x Genomics, 2017). We randomly subsample datasets of varying sizes and retain either the top 1,000 or 2,000 genes ranked by variance for analysis. For each probabilistic model, we compare EM, non-amortized variational inference, and amortized variational inference under the same latent dimensionality. For the variational approaches, we implement mean-field Gaussian approximations in Pyro; the amortized variant follows the encoder-based design of scPI. Since the original ZIFA implementation becomes computationally prohibitive at this scale, we instead use block_ZIFA, as recommended by Pierson and Yau (2015), as the EM-based baseline.

As shown in Figure 2, the computational cost increases substantially when moving from FA to ZIFA. While FA can be fitted efficiently using EM, the additional dropout mechanism in ZIFA leads to a much more expensive optimization procedure, resulting in substantially longer runtimes for block_ZIFA. In contrast, both variational approaches provide a unified inference framework that can be applied to different probabilistic models without deriving model-specific update equations. Among the variational methods, the amortized approach consistently achieves lower runtime than the non-amortized approach, reflecting the computational benefits of sharing an inference network across cells. These results suggest that amortization can substantially improve the practicality of probabilistic dimension reduction for large-scale single-cell datasets.

We further evaluate model performance on the mouse cortex dataset (Zeisel et al., 2015). From the dataset, we retained 558 genes with the highest variance and 2,253 cells. We assess the models on two downstream tasks: imputation and clustering. For imputation, we report the median L_1 error on the raw count scale for the 10% randomly masked nonzero entries. For clustering, we apply k-means to the inferred latent representations, with the number of clusters set to the number of provided cell-type annotations, and evaluate the resulting clusters using average silhouette width (ASW) (Rousseeuw, 1987), adjusted Rand index (ARI) (Hubert and Arabie, 1985), and normalized mutual information (NMI) (Strehl and Ghosh, 2002). Each experiment is repeated ten times. Figure 3 summarizes the downstream performance across imputation and clustering tasks. For imputation, ZIFA generally achieves lower reconstruction error than FA, reflecting the benefit of explicitly modeling dropout events, while different inference procedures

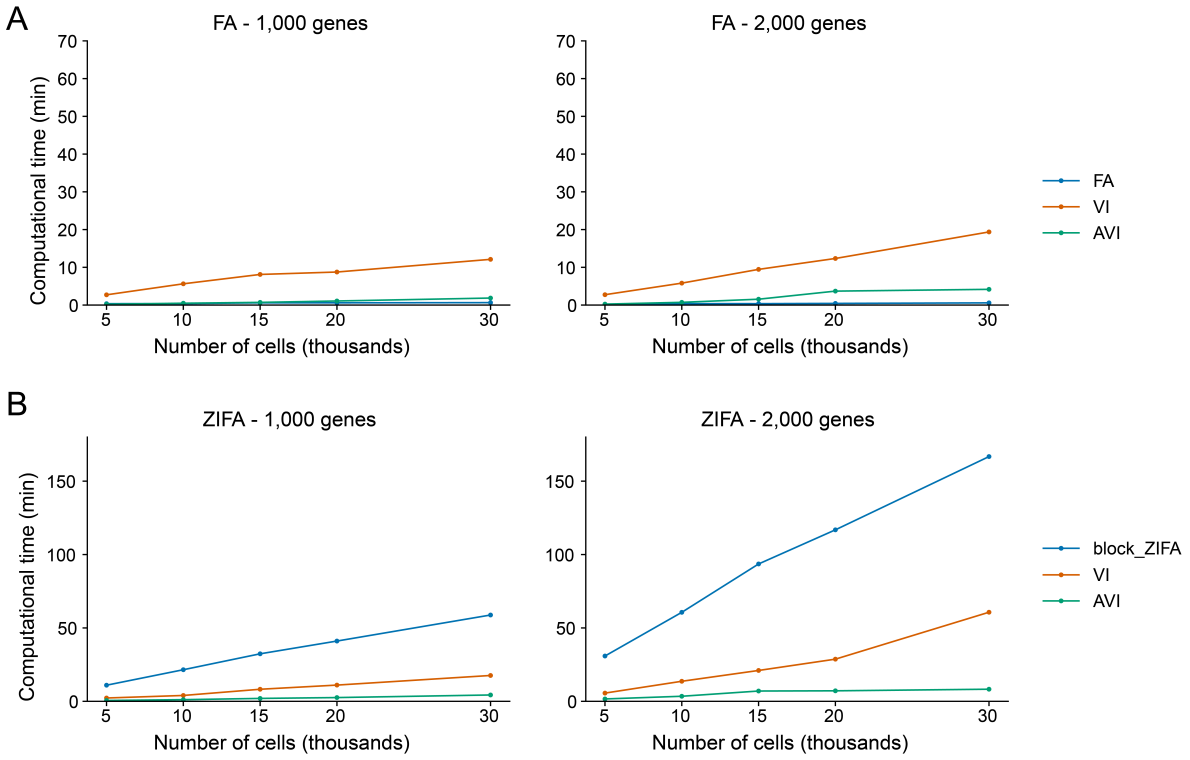

Figure 2: Runtime comparison of EM, non-amortized variational inference (VI), and amortized variational inference (AVI) under varying dataset sizes. (A) FA with 1,000 genes and 2,000 genes (B) ZIFA with 1,000 genes and 2,000 genes. The timings of FA and ZIFA were evaluated on the AMD EPYC 7282 16-Core Processor (CPU). The runtime of scPI was evaluated on an NVIDIA GeForce RTX 4090 GPU.

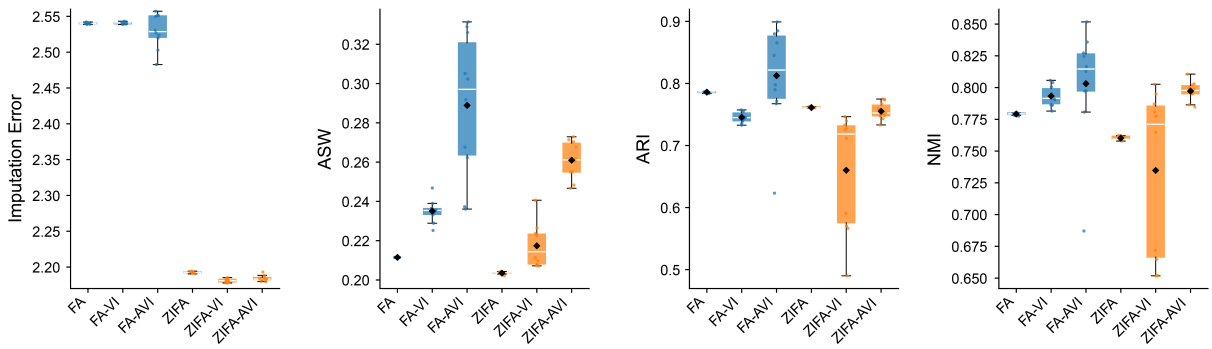

Figure 3: Downstream performance on the mouse cortex dataset. Evaluation metrics include imputation error (left panel), average silhouette width (ASW, middle left panel), adjusted Rand index (ARI, middle right panel), and normalized mutual information (NMI, right panel).

achieve broadly comparable imputation performance. In contrast, for clustering, FA-based latent representations tend to better preserve cell-type structure, suggesting that the Gaussian assumption can provide a more robust embedding for capturing cluster structure in this dataset. The differing performance of FA and ZIFA emphasizes that model selection should depend on the specific downstream objective.

Furthermore, the results demonstrate that the inference strategy itself can influence downstream analyses. Even under the same generative model specification, different inference pro-

cedures can lead to noticeably different latent representations and clustering outcomes, highlighting that the choice of inference procedure affects not only computational efficiency but also the quality and stability of learned embeddings. Greater variability is observed across repeated runs for variational approaches compared with EM-based estimation, reflecting the additional stochasticity introduced by gradient-based optimization and neural-network-based inference.

5.2 Temporal Trajectory Inference with Gaussian Process

Aging is a complex temporal process in living organisms (López-Otín et al., 2013). At the molecular level, gene expression changes across the life span can be cell-type-specific, nonlinear, and noisy. As a real-data example for temporal trajectory inference, we use the mouse subventricular zone (SVZ) aging single-cell RNA-seq data (Buckley et al., 2023) and study how the variational distribution and prior structure affect the inference of gene-level temporal trajectories within a selected cell type.

Given a cell-type-specific gene expression matrix $Y = (Y_{cg}) \in \mathbb{N}^{C\times G}$, where cell c has library size N_c and is measured at an age or time point $t_c \in \mathcal{T} = \{t_1, \ldots, t_T\}$. We model the single-cell read count using a Poisson distribution with a cell-and-gene specific normalized expression rate η_{cg}:

$$Y_{cg}|\eta_{cg} \sim \text{Poisson}(N_c\eta_{cg}), \quad \log(\eta_{cg}) = \log(\eta_{0,g}) + b_g(t_c) + \epsilon_{cg},$$

where $\log(\eta_{0,g})$ is a gene-specific baseline expression level, $b_g(t_c)$ represents the shared temporal effect of gene g at time point t_c, and ϵ_{cg} captures cell-level residual variation that is not explained by the temporal trajectory. Collecting the temporal effects over the observed time grid as $b_g = \{b_g(t_1), \ldots, b_g(t_T)\}^\top$, we place a Gaussian process prior on b_g to encode smoothness across ages and an independent Gaussian prior on the residual:

$$b_g \sim \mathcal{N}(0, \tau_g^2 K), \quad \epsilon_{cg} \sim \mathcal{N}(0, \tau_{\epsilon,g}^2),$$

where $K = (K_{tt'}) \in \mathbb{R}^{T\times T}$ is a temporal kernel matrix defining the covariance between time points with $K_{tt'} = \exp\{-(t-t')^2/\gamma\}$, τ_g^2 controls the overall scale of the temporal variation for gene g, and $\tau_{\epsilon,g}^2$ represents the variance of the non-temporal component. Under this model, cells measured at the same time point share the same temporal effect $b_g(t_c)$, while cells at nearby time points are correlated through the GP prior, allowing the model to borrow information across cells with identical or nearby temporal measurements. Since the posterior distribution of this model is analytically intractable, we use stochastic variational inference in Pyro to approximate it. Implementing the model requires specifying both the generative model and the variational guide—the likelihood, latent structure, prior, and variational family. These are modeling decisions made by the researcher, as they determine the validity and interpretation of the results. Once specified, translating them into executable Pyro code is a largely mechanical task well suited to a coding agent. Figure 4 illustrates this agent-assisted programming workflow. The researcher specifies the generative model first, including the underlying statistical assumptions, the choice of prior distributions over the relevant parameters, and the variational family used for inference, along with the required output; see the prompts in Appendix A.6. The agent generates the corresponding Pyro code and returns the fitted results. These outputs are then

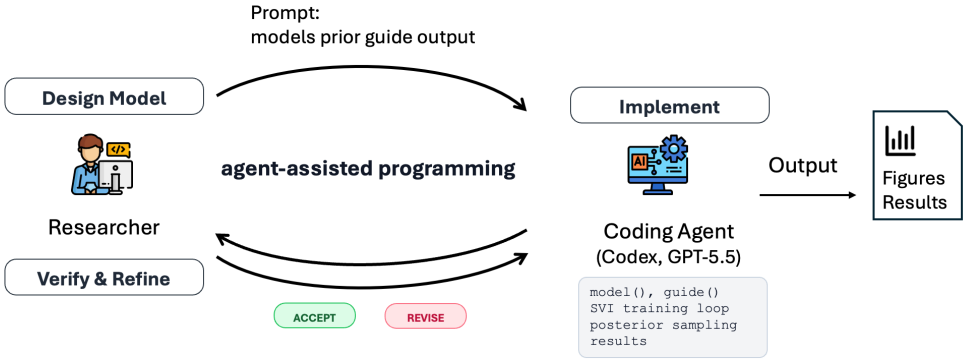

Figure 4: Agent-assisted programming workflow. The researcher designs the statistical model, specifying the priors, guide, and required outputs; the coding agent implements the corresponding Pyro code and returns the fitted results; the researcher then verifies these results against the target model, either accepting them or otherwise prompting the agent to revise. Once accepted, the agent outputs the final results.

inspected by the researcher, who examines the estimated means, variance, and other parameters to determine whether they are statistically plausible and align with the specified model and assumptions. Based on this assessment, the implementation is either accepted or returned to the agent with revision suggestions. The agent thus accelerates implementation, but the statistical judgments that determine the correctness of the inference remain with the researcher. Building on this framework, we then vary the prior and guide to examine how the prior structure and the variational family affect inference.

Specifically, we compare three settings: (i) a GP prior with a mean-field Gaussian guide (GP+MF); (ii) a GP prior with a full-rank Gaussian guide (GP+Full-rank), which captures posterior correlations across ages within each gene; and (iii) an independent Gaussian prior with a mean-field guide (Indep+MF), which removes the temporal coupling. We first use microglia as a representative cell type to examine how different inference settings affect computational runtime, posterior uncertainty, and the inferred temporal trajectories. To assess scalability, we subsample varying numbers of genes from the original gene set and measure the runtime under each setting. Runtime grows with the number of genes in all settings, with the full-rank guide being the most expensive, since it introduces $O(T^2)$ additional variational covariance parameters per gene to model posterior correlations across ages (Figure 5A). Fixing the gene set at 5,000 genes, we then quantify the width of the 95% credible bands across genes (Figure 5B). GP+MF yields the narrowest 95% credible bands—consistent with the known property of mean-field variational inference to underestimate posterior variance due to its factorized assumption that ignores posterior correlations (Blei et al., 2017), while GP+Full-rank produces wider, better-calibrated bands; Indep+MF is similarly diffuse but for a different reason, namely the absence of temporal smoothing across nearby ages (Figure 5B). This is reflected at the trajectory level, as illustrated for gene *Bc1* in Figure 5C: both GP-prior models recover smooth posterior curves for the representative gene *Bc1*, whereas Indep+MF exhibits irregular local variations. Thus, the GP prior is what enables temporal smoothing, while the guide determines how faithfully the posterior is approximated.

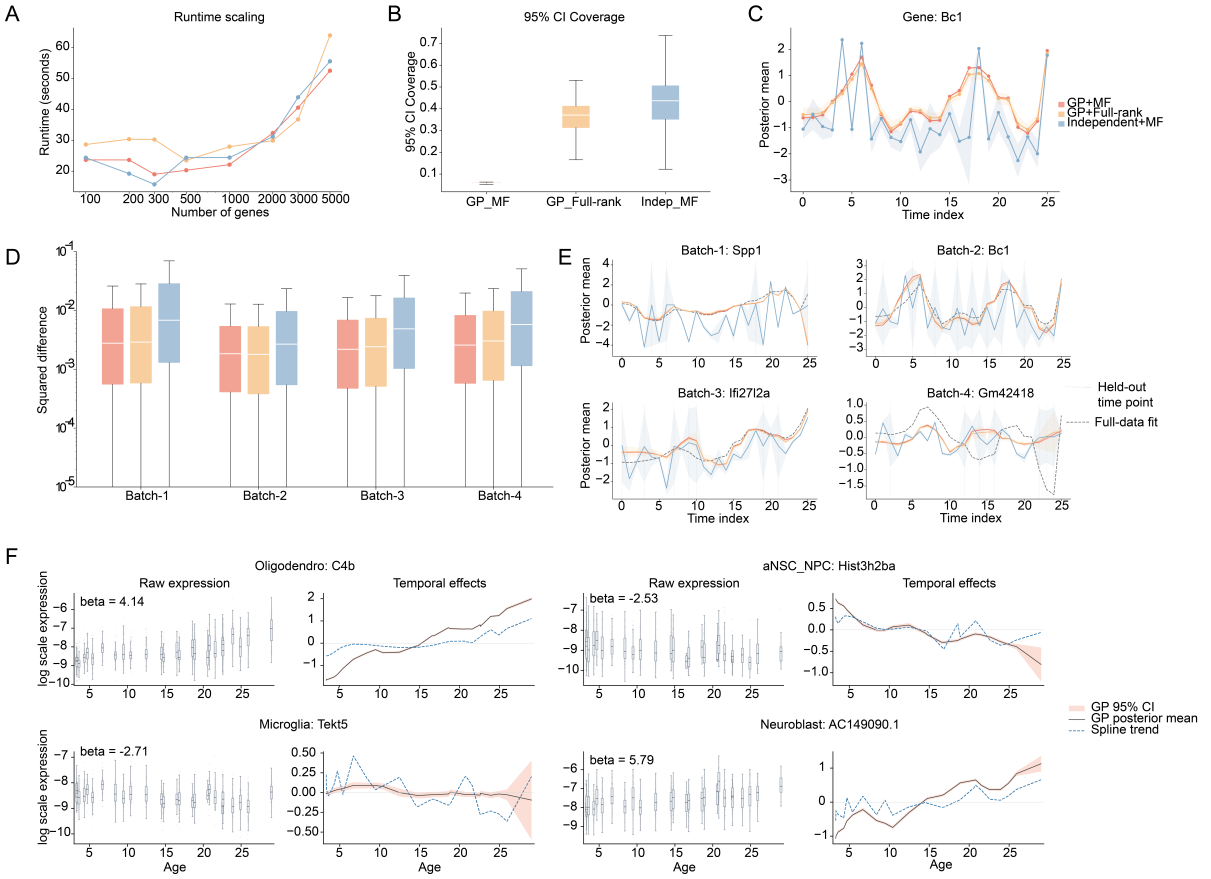

Figure 5: Temporal trajectory inference in the SVZ aging data. (A) Runtime scaling with the number of genes. (B) Distribution of mean 95% credible intervals across 5,000 genes. (C) Posterior trajectory for *Bc1* under three inference settings. (D) Leave-one-cohort-out interpolation error, measured by squared deviation from the full-data fit at held-out time points. (E) Representative held-out-cohort trajectories; dashed curves denote full-data fits and shaded regions mark held-out windows. (F) Comparison with external aging-clock genes and spline trends for representative cell-type-gene pairs.

To further examine whether the GP prior allows the model to borrow information across nearby ages, we adopt a leave-one-cohort-out design: in each run one of the four SVZ cohorts (Buckley et al., 2023) is held out while the model is still fitted over the full time axis, so held-out ages must be recovered by interpolation. Treating the full-data fit as a reference, we compute the squared deviation of each gene at each held-out time point. The distribution of these deviations is summarized for each held-out cohort in Figure 5D. Across cohorts, the two GP-prior models tend to produce smaller deviations from the full-data baseline than Indep+MF, indicating more stable interpolation at held-out ages. Representative genes from each held-out cohort (Figure 5E) show the same pattern: GP-prior trajectories closely track the full-data baseline within held-out windows by leveraging information from neighboring ages, whereas Indep+MF fits each age more independently and can deviate noticeably at held-out time points.

Finally, under the GP+MF setting, we compare the inferred trajectories with external aging-clock coefficients reported in the original study (Buckley et al., 2023) and with spline trends fitted to per-age mean expression. Figure 5F shows four representative cell-type–gene pairs spanning different reported aging directions: *C4b* in Oligodendrocytes ($\beta = 4.14$), *Hist3h2ba* in aNSC/NPCs ($\beta = -2.53$), *Tekt5* in Microglia ($\beta = -2.71$), and *AC149090.1* in Neuroblasts ($\beta =$

5.79). For *C4b* and *AC149090.1*, the GP posterior means show increasing temporal patterns that are consistent with their positive aging-clock coefficients. In contrast, the negative-clock genes show more heterogeneous behavior. *Hist3h2ba* exhibits a broadly decreasing trajectory, whereas *Tekt5* does not show a clear monotone decline in the inferred GP trajectory and has substantially wider posterior uncertainty at later ages. Thus, although the reported clock coefficient suggests a negative aging association for *Tekt5*, the posterior trajectory indicates that its temporal change is less certain and should not be interpreted simply as a steadily decreasing gene. Compared with the spline fit, the GP model provides a smoother trajectory together with uncertainty estimates, which helps distinguish robust temporal patterns from age-specific noise or weakly supported trends.

6 Concluding Remarks

We provide a unified and practical introduction to VI for single-cell genomics. Many widely used single-cell models can be formulated as hierarchical latent variable models, but their posterior distributions are often intractable because of non-Gaussian likelihoods, hierarchical dependencies, and large-scale datasets. Starting from the EM algorithm and classical mean-field VI, we developed VI as an optimization-based alternative to exact posterior inference, then extended it to modern settings through stochastic VI for scalability, black-box VI for non-conjugate models, and amortized VI for efficient per-observation inference. We also showed how probabilistic programming, using Pyro as a representative framework, connects these ideas in practice by separating model specification from inference. In the simulations, we benchmarked CAVI, SVI, BBVI, and amortized VI against EM-based likelihood baselines on hierarchical latent variable models. The real-data analyses illustrated the practical use of VI in dimension reduction using scPI and the temporal trajectory inference with a Gaussian process latent variable model. We hope that this provides readers with both the conceptual understanding and the practical orientation needed to apply variational inference thoughtfully to their own single-cell analyses.

A Technical Details

In this appendix, we provide the technical details for the derivations in the main text.

A.1 Details on ELBO and CAVI Updates

Note that the KL divergence from distribution q to p is defined as $\mathrm{KL}\{q(Z) \,\|\, p(Z)\} = \int q(Z) \log \{q(Z)/p(Z)\} \mathrm{d}Z$. We then have

$$\begin{aligned}
\log p(X|\theta) &= \int q(Z) \log p(X|\theta) \mathrm{d}Z \\
&= \int q(Z) \log \frac{p(X, Z|\theta)}{p(Z|X, \theta)} \mathrm{d}Z \\
&= \int q(Z) \log \frac{p(X, Z|\theta)}{q(Z)} \mathrm{d}Z + \int q(Z) \log \frac{q(Z)}{p(Z|X, \theta)} \mathrm{d}Z \\
&= \mathcal{L}(q, \theta) + \mathrm{KL}\{q(Z) \,\|\, p(Z|X, \theta)\}
\end{aligned}$$

in Section 2.2.

Next, for mean-field variational inference, suppose $q(Z) = \prod_{i=1}^{I} q_i(Z_i)$. Holding $q_{-i}(Z_{-i}) = \prod_{\ell \neq i} q_\ell(Z_\ell)$ fixed, the part of the ELBO depending on q_i is

$$\mathcal{L}_i(q_i) = \int q_i(z_i)\mathbb{E}_{q_{-i}}\{\log p(X, Z|\theta)\}\mathrm{d}z_i - \int q_i(z_i)\log q_i(z_i)\mathrm{d}z_i + \text{const}.$$

Imposing the normalization constraint $\int q_i(z_i)\mathrm{d}z_i = 1$ and using a Lagrange multiplier gives

$$\mathbb{E}_{q_{-i}}\{\log p(X, Z|\theta)\} - \log q_i(z_i) - 1 + \kappa = 0.$$

Therefore,

$$\log q_i^*(Z_i) = \mathbb{E}_{q_{-i}}\{\log p(X, Z|\theta)\} + \text{const},$$

which is the coordinate-ascent variational inference update used in Section 2.2.

A.2 Posterior Variance Underestimation of Mean-Field Variational Inference

We illustrate the posterior variance underestimation of mean-field variational inference by the following example. Suppose the true posterior is a correlated bivariate Gaussian

$$p(Z_1, Z_2) \;=\; \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right), \qquad |\rho| < 1,$$

whose marginals satisfy $\mathrm{Var}(Z_1) = \mathrm{Var}(Z_2) = 1$. We approximate it by a factorized Gaussian $q(Z_1, Z_2) = \mathcal{N}(Z_1; m_1, s_1^2)\mathcal{N}(Z_2; m_2, s_2^2)$ and apply the CAVI update. Using the standard conditional $p(Z_1|Z_2) = \mathcal{N}(\rho Z_2,\, 1-\rho^2)$ and averaging its log density over $q(Z_2)$ gives

$$\log q^*(Z_1) \;=\; \mathbb{E}_{q(Z_2)}[\log p(Z_1|Z_2)] + \text{const} \;=\; -\frac{(Z_1 - \rho m_2)^2}{2(1-\rho^2)} + \text{const},$$

so $q^*(Z_1) = \mathcal{N}(\rho m_2,\, 1-\rho^2)$, and by symmetry $q^*(Z_2) = \mathcal{N}(\rho m_1,\, 1-\rho^2)$. At the fixed point $m_1 = m_2 = 0$, the variational marginal variances become

$$s_1^2 \;=\; s_2^2 \;=\; 1-\rho^2 \;<\; 1 \;=\; \mathrm{Var}(Z_i), \qquad \rho \neq 0.$$

The factorized q therefore reports a marginal variance smaller than the truth by exactly the factor $1-\rho^2$, and this gap increases as the posterior dependence $|\rho|$ grows. Intuitively, the factorization forces q to describe each coordinate without correlation, so the uncertainty that is shared between Z_1 and Z_2 in the true posterior cannot be captured.

A.3 Natural Gradient Perspective of Variational Inference

For the conditionally conjugate exponential family model described in Section 3.1, assume that the global variational factor is chosen in the same exponential family as the complete

conditional of β:

$$q(\beta|\lambda) = h(\beta)\exp\left\{\lambda^\top t(\beta) - A_\beta(\lambda)\right\},$$

where $A_\beta(\lambda)$ is the log-partition function. The ELBO is then given by taking the expectation as follows:

$$\mathcal{L}(\lambda,\phi) = \nabla A_\beta(\lambda)^\top\Big\{\alpha + \sum_{i=1}^N \mathbb{E}_{\phi_i} s(x_i,z_i) - \lambda\Big\} + A_\beta(\lambda),$$

up to some constants. Differentiating this w.r.t. λ, we have

$$\nabla_\lambda \mathcal{L}(\lambda,\phi) = \nabla^2 A_\beta(\lambda)\Big\{\alpha + \sum_{i=1}^N \mathbb{E}_{\phi_i} s(x_i,z_i) - \lambda\Big\}.$$

The derivatives w.r.t. ϕ_is can be derived similarly. Therefore, the natural gradient of $\mathcal{L}(\lambda,\phi)$ has a much simpler form as:

$$\widetilde{\nabla}_\lambda \mathcal{L}(\lambda,\phi) = \alpha + \sum_{i=1}^N \mathbb{E}_{\phi_i} s(x_i,z_i) - \lambda.$$

When the local variational parameters are set to their optima $\phi_i^*(\lambda)$, the envelope theorem allows us to ignore the derivative of $\phi_i^*(\lambda)$ when differentiating the profiled ELBO. Thus the same natural-gradient expression holds with $\mathbb{E}_{\phi_i} s(x_i,z_i)$ replaced by $\mathbb{E}_{\phi_i^*(\lambda)} s(x_i,z_i)$.

A.4 Black-Box Variational Inference Gradients

Consider the ELBO as follows

$$\mathcal{L}(\phi) = \mathbb{E}_{q_\phi(Z)}\left[\log p(X,Z) - \log q_\phi(Z)\right],$$

where $q_\phi(Z)$ is a variational distribution parameterized by ϕ. To optimize the ELBO $\mathcal{L}(\phi)$, we compute its gradient as follows:

$$\begin{aligned}
\nabla_\phi \mathcal{L}(\phi) &= \nabla_\phi \mathbb{E}_{q_\phi}\Big\{\log p(X,Z) - \log q_\phi(Z)\Big\} \\
&= \int \Big[\nabla_\phi q_\phi(Z)\Big\{\log p(X,Z) - \log q_\phi(Z)\Big\} - q_\phi(Z)\nabla_\phi \log q_\phi(Z)\Big]\mathrm{d}Z \qquad (1)\\
&= \int \Big[q_\phi(Z)\nabla_\phi \log q_\phi(Z)\Big\{\log p(X,Z) - \log q_\phi(Z)\Big\} - q_\phi(Z)\nabla_\phi \log q_\phi(Z)\Big]\mathrm{d}Z \qquad (2)\\
&= \mathbb{E}_{q_\phi}\Big[\nabla_\phi \log q_\phi(Z)\Big\{\log p(X,Z) - \log q_\phi(Z) - 1\Big\}\Big] \\
&= \mathbb{E}_{q_\phi}\Big[\nabla_\phi \log q_\phi(Z)\Big\{\log p(X,Z) - \log q_\phi(Z)\Big\}\Big], \qquad (3)
\end{aligned}$$

where (1) follows by interchanging differentiation and integration, (2) is due to the identity $\nabla_\phi \log q_\phi(Z) = \{q_\phi(Z)\}^{-1}\nabla_\phi q_\phi(Z)$, and (3) holds because $q_\phi(Z)$ is a probability distribution with $\mathbb{E}_{q_\phi}\{\nabla_\phi \log q_\phi(Z)\} = 0$.

A.5 Derivations in Simulation Studies

A.5.1 Derivations for the Linear Mixed Model

The linear mixed model in simulation Study 1 is

$$y = W\alpha + X\beta + \varepsilon, \qquad \beta \sim \mathcal{N}(0, \sigma_b^2 I_p), \qquad \varepsilon \sim \mathcal{N}(0, \sigma_e^2 I_n).$$

Let x_j denote the jth column of X, and define $r = y - W\alpha$. Then, the complete-data log density is given by

$$\log p(y, \beta|\alpha, \sigma_b^2, \sigma_e^2) = -\frac{n}{2}\log\sigma_e^2 - \frac{1}{2\sigma_e^2}\|r - X\beta\|^2 - \frac{p}{2}\log\sigma_b^2 - \frac{1}{2\sigma_b^2}\beta^\top\beta$$

up to constants independent of the unknown parameters.

Using the mean-field family $q(\beta) = \prod_{j=1}^p \mathcal{N}(\beta_j; m_j, s_j^2)$ for variational approximation, the coordinate update for $q_j(\beta_j)$ follows from

$$\log q_j^*(\beta_j) = \mathbb{E}_{q_{-j}}\{\log p(y, \beta|\alpha, \sigma_b^2, \sigma_e^2)\} + \text{const}.$$

Keeping only terms depending on β_j,

$$\log q_j^*(\beta_j) = -\frac{1}{2\sigma_e^2}\left[\beta_j^2 x_j^\top x_j - 2\beta_j x_j^\top\Big\{y - W\alpha - \sum_{\ell\neq j} x_\ell m_\ell\Big\}\right] - \frac{1}{2\sigma_b^2}\beta_j^2 + \text{const}.$$

Therefore we have $q_j^*(\beta_j) = \mathcal{N}(m_j, s_j^2)$, with

$$s_j^2 = \left(\frac{x_j^\top x_j}{\sigma_e^2} + \frac{1}{\sigma_b^2}\right)^{-1}, \quad m_j = \frac{s_j^2}{\sigma_e^2}x_j^\top\left(y - W\alpha - \sum_{\ell\neq j} x_\ell m_\ell\right),$$

which corresponds to the CAVI updates in Section 4.

Given $q(\beta)$, the model-parameter updates are obtained by maximizing the expected complete-data log likelihood, with $E_q(\beta) = m$ and $E_q(\beta_j^2) = m_j^2 + s_j^2$. We then obtain

$$\widehat{\alpha} = (W^\top W)^{-1}W^\top(y - Xm),$$
$$\widehat{\sigma}_b^2 = \frac{1}{p}\sum_{j=1}^p (m_j^2 + s_j^2), \quad \widehat{\sigma}_e^2 = \frac{1}{n}\Big(\|y - W\widehat{\alpha} - Xm\|^2 + \sum_{j=1}^p s_j^2 x_j^\top x_j\Big).$$

We should also note that, for fixed $(\alpha, \sigma_b^2, \sigma_e^2)$, the exact posterior by the EM algorithm is multivariate Gaussian:

$$\beta|y \sim \mathcal{N}(\mu_\beta, V_\beta), \qquad V_\beta = \left(\frac{X^\top X}{\sigma_e^2} + \frac{I_p}{\sigma_b^2}\right)^{-1}, \qquad \mu_\beta = V_\beta\frac{X^\top(y - W\alpha)}{\sigma_e^2}.$$

At convergence of the mean-field coordinate equations, the variational mean m solves the same

linear system as μ_β, but the marginal variational variance is

$$s_j^2 = \left\{ \frac{x_j^\top x_j}{\sigma_e^2} + \frac{1}{\sigma_b^2} \right\}^{-1},$$

which is the reciprocal of the jth diagonal entry of the exact posterior precision. Note that $s_j^2 \leq (V_\beta)_{jj}$. This explains why the CAVI posterior means can be accurate while the nominal credible intervals have a shorter coverage when the random effects are posterior-correlated.

A.5.2 Derivations for Single-Cell LDA

Recall the single-cell LDA model in Study 2 is

$$\xi_k \sim \text{Dirichlet}(\omega), \qquad \eta_i \sim \text{Dirichlet}(a),$$

and, for transcript j in cell i, we have

$$z_{ij}|\eta_i \sim \text{Categorical}(\eta_i), \qquad w_{ij}|z_{ij} = k, \xi \sim \text{Categorical}(\xi_k).$$

The mean-field variational family is

$$q(\xi, \eta, z) = \prod_{k=1}^{K} \text{Dirichlet}(\xi_k; \lambda_k) \prod_{i=1}^{N} \left\{ \text{Dirichlet}(\eta_i; \gamma_i) \prod_{j=1}^{m_i} \text{Categorical}(z_{ij}; \varphi_{ij}) \right\}.$$

We then have the ELBO as follows:

$$\begin{aligned} \mathcal{L} = &\sum_{k=1}^{K} \mathbb{E}_q\{\log p(\xi_k|\omega) - \log q(\xi_k|\lambda_k)\} + \sum_{i=1}^{N} \mathbb{E}_q\{\log p(\eta_i|a) - \log q(\eta_i|\gamma_i)\} \\ &+ \sum_{i=1}^{N} \sum_{j=1}^{m_i} \sum_{k=1}^{K} \varphi_{ijk} \left\{ \mathbb{E}_q(\log \eta_{ik}) + \mathbb{E}_q(\log \xi_{k,w_{ij}}) - \log \varphi_{ijk} \right\}. \end{aligned}$$

Let $\psi(\cdot)$ denote the digamma function (Abramowitz et al., 1966). Then, by the properties of the Dirichlet distribution, we have

$$\mathbb{E}_q\{\log \eta_{ik}\} = \psi(\gamma_{ik}) - \psi\left(\sum_{\ell=1}^{K} \gamma_{i\ell}\right), \quad \mathbb{E}_q\{\log \xi_{kg}\} = \psi(\lambda_{kg}) - \psi\left(\sum_{h=1}^{G} \lambda_{kh}\right).$$

The part of the ELBO involving φ_{ij} is

$$\sum_{k=1}^{K} \varphi_{ijk} \left[\mathbb{E}_q\{\log \eta_{ik}\} + \mathbb{E}_q\{\log \xi_{k,w_{ij}}\} - \log \varphi_{ijk} \right],$$

with the constraint $\sum_k \varphi_{ijk} = 1$. Therefore

$$\varphi_{ijk} \propto \exp\left[\mathbb{E}_q\{\log \eta_{ik}\} + \mathbb{E}_q\{\log \xi_{k,w_{ij}}\} \right].$$

Note that all transcripts in cell i with the same observed gene identity g share the same

variational topic probability

$$r_{igk} = \frac{\exp\left[\psi(\gamma_{ik}) - \psi(\sum_{\ell=1}^{K}\gamma_{i\ell}) + \psi(\lambda_{kg}) - \psi(\sum_{h=1}^{G}\lambda_{kh})\right]}{\sum_{\ell=1}^{K}\exp\left[\psi(\gamma_{i\ell}) - \psi(\sum_{r=1}^{K}\gamma_{ir}) + \psi(\lambda_{\ell g}) - \psi(\sum_{h=1}^{G}\lambda_{\ell h})\right]}.$$

Then the local update becomes $\gamma_{ik} = a_k + \sum_{g=1}^{G} x_{ig} r_{igk}$, and the full-data global update becomes $\lambda_{kg}^{\mathrm{CAVI}} = \omega_g + \sum_{i=1}^{N} x_{ig} r_{igk}$.

For SVI, let $\mathcal{B}_t$ be a mini-batch of b_t cells. The unbiased estimate of the full expected topic-gene count is

$$\frac{N}{b_t}\sum_{i\in\mathcal{B}_t} x_{ig} r_{igk}^{(t)}.$$

Thus

$$\widehat{\lambda}_{kg}^{(t)} = \omega_g + \frac{N}{b_t}\sum_{i\in\mathcal{B}_t} x_{ig} r_{igk}^{(t)},$$

and the stochastic natural-gradient update is

$$\lambda_{kg}^{(t+1)} = (1-\rho_t)\lambda_{kg}^{(t)} + \rho_t \widehat{\lambda}_{kg}^{(t)}.$$

A.6 Details for Probabilistic Programming in Temporal Trajectory Inference

Box 1 shows a prompt for implementing the temporal GP model using Pyro with a mean-field Gaussian variational guide. The prompt specifies the prior, variational distribution, trainable parameters, and diagnostic outputs, which the researcher can check before using the fitted trajectories in the analysis. In our implementation, we used Codex as the coding agent, with GPT-5.5 as the underlying large language model.

Box 1. Example prompt for agent-assisted Pyro implementation

```
Write Pyro code for the GP+MF temporal trajectory model.

Inputs are the count matrix Y, library sizes N, age indices age_idx, and a
    precomputed temporal kernel matrix K.

Use the model

Y[c,g] ~ Poisson(N[c] * exp(alpha[g] + b[g, age_idx[c]] + eps[c,g])),
where b[g,:] ~ MVN(0, tau[g]^2 K + jitter I) and eps[c,g] ~ Normal(0, sigma[g]^2).

Treat alpha[g], tau[g], and sigma[g] as trainable parameters, with positive
    constraints on tau[g] and sigma[g].

Use a mean-field Gaussian guide for b and eps.
```

```
Return the Pyro model, guide, SVI training loop, ELBO trace, runtime, posterior
    mean trajectory, and pointwise 95\% credible bands for b.

Include checks for tensor shapes, positive rates and scales, finite ELBO values,
    and posterior summary shapes.
```

Code Availability

The code for reproducing the simulation and real data analysis results is available at: https://anonymous.4open.science/r/Variational_Inference_Tutorial-F5AC.

References

10x Genomics. 1.3 million brain cells from e18 mice, 2017.

Milton Abramowitz, Irene A Stegun, and Jacques E Romain. Handbook of mathematical functions, with formulas, graphs, and mathematical tables, 1966.

Divyansh Agarwal, Jingshu Wang, and Nancy R Zhang. Data denoising and post-denoising corrections in single cell rna sequencing. *Statistical Science*, 35(1):112–128, 2020.

Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2): 251–276, 1998.

David J Bartholomew, Martin Knott, and Irini Moustaki. *Latent Variable Models and Factor Analysis: A Unified Approach*. John Wiley & Sons, 2011.

Eli Bingham, Jonathan P Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D Goodman. Pyro: Deep universal probabilistic programming. *Journal of Machine Learning Research*, 20(28):1–6, 2019.

Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

David M Blei. Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application*, 1(1):203–232, 2014.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

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.

Pierre Boyeau, Justin Hong, Adam Gayoso, Martin Kim, José L McFaline-Figueroa, Michael I Jordan, Elham Azizi, Can Ergen, and Nir Yosef. Deep generative modeling of sample-level heterogeneity in single-cell genomics. *Nature Methods*, 22(11):2264–2274, 2025.

Norman E Breslow and David G Clayton. Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association*, 88(421):9–25, 1993.

Stephen Brooks. Markov chain monte carlo method and its application. *Journal of the Royal Statistical Society: Series D (the Statistician)*, 47(1):69–100, 1998.

Matthew T. Buckley, Eric D. Sun, Benson M. George, Ling Liu, Nicholas Schaum, Lucy Xu, Jaime M. Reyes, Margaret A. Goodell, Irving L. Weissman, Tony Wyss-Coray, Thomas A. Rando, and Anne Brunet. Cell-type-specific aging clocks to quantify aging and rejuvenation in neurogenic regions of the brain. *Nature Aging*, 3(1):121–137, 2023.

Smriti Chawla, Anja Rockstroh, Melanie Lehman, Ellca Ratther, Atishay Jain, Anuneet Anand, Apoorva Gupta, Namrata Bhattacharya, Sarita Poonia, Priyadarshini Rai, et al. Gene expression based inference of cancer drug sensitivity. *Nature Communications*, 13(1):5680, 2022.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

Bradley Efron. *Large-Scale Inference: Empirical Bayes Methods for Estimation, Testing, and Prediction*, volume 1. Cambridge University Press, 2012.

Charles Gawad, Winston Koh, and Stephen R Quake. Single-cell genome sequencing: current state of the science. *Nature Reviews Genetics*, 17(3):175–188, 2016.

Adam Gayoso, Zoë Steier, Romain Lopez, Jeffrey Regier, Kristopher L Nazor, Aaron Streets, and Nir Yosef. Joint probabilistic modeling of single-cell multi-omic data with totalvi. *Nature Methods*, 18(3):272–282, 2021.

Christopher Heje Grønbech, Maximillian Fornitz Vording, Pascal N Timshel, Casper Kaae Sønderby, Tune H Pers, and Ole Winther. scvae: variational auto-encoders for single-cell gene expression data. *Bioinformatics*, 36(16):4415–4422, 2020.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 2013.

Philippe Huber, Elvezio Ronchetti, and Maria-Pia Victoria-Feser. Estimation of generalized linear latent variable models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 66(4):893–908, 2004.

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2(1): 193–218, 1985.

Karthik A Jagadeesh, Kushal K Dey, Daniel T Montoro, Rahul Mohan, Steven Gazal, Jesse M Engreitz, Ramnik J Xavier, Alkes L Price, and Aviv Regev. Identifying disease-critical cell types and cellular processes by integrating single-cell rna-sequencing and human genetics. *Nature Genetics*, 54(10):1479–1492, 2022.

Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada*, 2014.

Ilya Korsunsky, Nghia Millard, Jean Fan, Kamil Slowikowski, Fan Zhang, Kevin Wei, Yuriy Baglaenko, Michael Brenner, Po-ru Loh, and Soumya Raychaudhuri. Fast, sensitive and accurate integration of single-cell data with harmony. *Nature Methods*, 16(12):1289–1296, 2019.

Yingxin Lin, Shila Ghazanfar, Kevin YX Wang, Johann A Gagnon-Bartsch, Kitty K Lo, Xianbin Su, Ze-Guang Han, John T Ormerod, Terence P Speed, Pengyi Yang, et al. scmerge leverages factor analysis, stable expression, and pseudoreplication to merge multiple single-cell rna-seq datasets. *Proceedings of the National Academy of Sciences*, 116(20):9775–9784, 2019.

Chuanhai Liu, Donald B Rubin, and Ying Nian Wu. Parameter expansion to accelerate em: the px-em algorithm. *Biometrika*, pages 755–770, 1998.

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, 2018.

Carlos López-Otín, Maria A Blasco, Linda Partridge, Manuel Serrano, and Guido Kroemer. The hallmarks of aging. *Cell*, 153(6):1194–1217, 2013.

Mohammad Lotfollahi, Mohsen Naghipourfar, Malte D Luecken, Matin Khajavi, Maren Büttner, Marco Wagenstetter, Žiga Avsec, Adam Gayoso, Nir Yosef, Marta Interlandi, et al. Mapping single-cell data to reference atlases by transfer learning. *Nature Biotechnology*, 40(1):121–130, 2022.

Lars Maaløe, Marco Fraccaro, Valentin Liévin, and Ole Winther. Biva: A very deep hierarchy of latent variables for generative modeling. *Advances in Neural Information Processing Systems*, 32, 2019.

Geoffrey J. McLachlan and Thriyambakam Krishnan. *The EM Algorithm and Extensions*. Wiley, 2008.

Jingsi Ming, Jia Zhao, and Can Yang. scpi: A scalable framework for probabilistic inference in single-cell rna-sequencing data analysis. *Statistics in Biosciences*, 15(3):633–656, 2023.

Radford M Neal. *Probabilistic Inference Using Markov Chain Monte Carlo Methods*. Department of Computer Science, University of Toronto, Toronto, ON, Canada, 1993.

Barry L Nelson. Control variate remedies. *Operations Research*, 38(6):974–992, 1990.

Emma Pierson and Christopher Yau. Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biology*, 16(1):241, 2015.

Jonathan K Pritchard, Matthew Stephens, and Peter Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959, 2000.

Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial Intelligence and Statistics*, pages 814–822. PMLR, 2014.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, pages 1278–1286. PMLR, 2014.

Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3(Dec):583–617, 2002.

Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B*, 61(3):611–622, 1999.

Martin J. Wainwright and Michael I. Jordan. *Graphical Models, Exponential Families, and Variational Inference*. Now Publishers, 2008.

Chenling Xu, Romain Lopez, Edouard Mehlman, Jeffrey Regier, Michael I Jordan, and Nir Yosef. Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Molecular Systems Biology*, 17(1):MSB20209620, 2021.

Yunpei Xu, Shaokai Wang, Qilong Feng, Jiazhi Xia, Yaohang Li, Hong-Dong Li, and Jianxin Wang. sccad: Cluster decomposition-based anomaly detection for rare cell identification in single-cell expression data. *Nature Communications*, 15(1):7561, 2024.

Amit Zeisel, Ana B Muñoz-Manchado, Simone Codeluppi, Peter Lönnerberg, Gioele La Manno, Anna Juréus, Sueli Marques, Hermany Munguba, Liqun He, Christer Betsholtz, et al. Cell types in the mouse cortex and hippocampus revealed by single-cell rna-seq. *Science*, 347 (6226):1138–1142, 2015.

Chengwei Zhong, Kok Siong Ang, and Jinmiao Chen. Interpretable spatially aware dimension reduction of spatial transcriptomics with stamp. *Nature Methods*, 21(11):2072–2083, 2024.

Hua Zhou, Liuyi Hu, Jin Zhou, and Kenneth Lange. Mm algorithms for variance components models. *Journal of Computational and Graphical Statistics*, 28(2):350–361, 2019.