scROBM: Role-Guided Bottleneck Model for Single-Cell Latent Representation Learning

Anonymous authors
Paper under double-blind review

Abstract

Single-cell expressions mix cell identity, condition response, inter-individual background, and technical variation, and reconstruction alone gives no account of which latent coordinates carry which role. We present scROBM , a role-objective-guided bottleneck model for expression-only single-cell representation learning. scROBM assigns three cell-level latent variables to cell identity, perturbation semantics, and donor context, together with a shared donor-level center, and organizes training around two coupled bottlenecks: a label-free count decoder that forces every label-dependent reconstruction signal through the assigned latents, and a conditional-invariance (CI) penalty that suppresses condition information in the joint background means within supported donor–cell-type strata while leaving donor and cell-type differences unconstrained. We evaluate scROBM on four single cell data sets. Donor-associated structure shows the clearest separation, whereas cell-type and condition remain in a high, dataset-dependent range; because every latent is supervised for a single role, the three exports stay interpretable, and the donor-context export resolves donors while exposing a cell-type-dependent condition structure concentrated in the treatment's target compartment.

1 Introduction

Single-cell RNA sequencing records cell identity, condition-dependent response, inter-individual background, and technical variation in the same expression vector. These factors are statistically coupled. Donors can differ in cell-type composition, perturbation effects can be cell-type specific, and protocol can track donor or cohort. A shared embedding may preserve information about all of these factors while giving no account of which latent coordinates carry which role.

This makes latent allocation an algorithmic problem. A reconstruction-only variational autoencoder can place donor, condition, and cell-type information wherever it reduces count loss. A label-conditioned inference network introduces a second concern: metadata used during inference can become circular evidence when the same label is evaluated from the exported representation. Donor context also requires an open-set protocol because independent test donors have no training-time donor classes.

scROBM addresses this problem with three cell-level latent variables and one donor-level latent variable. The basal variable $\mathbf{z}^{(b)}$ is assigned major cell identity, the stimulus variable $\mathbf{z}^{(s)}$ is assigned perturbation semantics, and the donor-context variable $\mathbf{z}^{(c)}$ is assigned donor-associated structure. Each donor has a shared latent center $\boldsymbol{\nu}_d$. A chained hierarchical prior lets the basal state condition the donor-context distribution, while the basal state and donor context jointly condition the stimulus distribution.

In addition, scROBM adopts two coupled bottlenecks to accompany the role guidance. The first is the *architectural reconstruction bottleneck*: the decoder receives the latent variables and a fixed exposure but no cell-type, condition, or donor label, so any label-dependent reconstruction signal must be transmitted through the latent representation, beyond what the fixed exposure already reveals. The second is the *conditional semantic bottleneck*: a conditional-invariance penalty on the joint background means applies training pressure that removes condition information from the background channel, while the stimulus role loss rewards keeping that information readable. The penalty leaves donor and cell-type differences between strata unconstrained.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2 Related work

Single-cell generative representation learning. Variational autoencoders provide the probabilistic backbone for many single-cell representation models Kingma & Welling (2014). scVI models count data, library size, and latent variation with a probabilistic decoder Lopez et al. (2018); Gayoso et al. (2022). scANVI adds cell-type labels for semi-supervised annotation and integration Xu et al. (2021). These methods provide complete-representation baselines for our evaluation Luecken et al. (2022). CellPLM uses pretraining to model cell-cell context beyond an isolated cell Wen et al. (2024). Large pretrained cell models such as scGPT and scFoundation extend this line by learning transferable representations across tens of millions of cells Cui et al. (2024); Hao et al. (2024), while a recent biology-driven benchmark reports that no single model dominates and that simpler baselines remain competitive under resource constraints Wu et al. (2025).

Perturbation and condition-aware modeling. scGen represents perturbation as latent shifts in a variational autoencoder Lotfollahi et al. (2019). scDFM formulates single-cell perturbation prediction as conditional flow matching and aligns generated and observed perturbed-cell distributions in unpaired data Yu et al. (2026). scDisInFact learns shared and condition-specific factors for multi-batch, multi-condition data, while contrastiveVI isolates treatment-enriched variation from shared variation Zhang et al. (2024); Weinberger et al. (2023). CFGen uses cell type, tissue, donor, and batch attributes to guide single-cell generation Palma et al. (2025). TarDis reserves latent spaces for multiple covariates Inecik et al. (2026). GEARS predicts multigene perturbation outcomes through gene-graph priors Roohani et al. (2024), Scouter replaces these priors with large language model embeddings Zhu & Li (2026), and State scales perturbation modeling to population level Adduri et al. (2025); a recent benchmark, however, finds that such deep models do not yet consistently outperform additive or mean baselines on perturbation prediction Ahlmann-Eltze et al. (2025).

Latent disentanglement in single-cell data. For single-cell data, assigning separate meanings to cell identity, perturbation semantics, and donor context requires observed metadata or explicit architectural constraints. scDisInFact separates shared and condition-specific factors, contrastiveVI isolates treatment-enriched variation, biolord separates known attributes from residual variation, and TarDis allocates spaces to multiple covariates Zhang et al. (2024); Weinberger et al. (2023); Piran et al. (2024); Inecik et al. (2026). Celcomen studies causal disentanglement for spatial single-cell and tissue perturbation modeling Megas et al. (2025). CausCell and scCausalVI instead build causal or structural graphs to separate cellular state from treatment effect Gao et al. (2025); An et al. (2025), and spVIPES decomposes unpaired datasets into shared and private factors Novella-Rausell et al. (2026); batch signal nonetheless persists in several pretrained embeddings Wang et al. (2025).

Information bottleneck and selective invariance. The information bottleneck trades representation complexity against label sufficiency Tishby et al. (2000), whereas invariant-representation objectives remove a designated nuisance factor Moyer et al. (2018). scROBM follows the latter direction: its CI term suppresses condition information in the background channel within donor–cell-type strata while retaining donor and cell identity. It instead optimizes a distributional discrepancy on the exported means. A recent survey consolidates this line of work across theory, optimization, and applications Hu et al. (2024).

3 Problem setup

Denote the condition label space by $\mathcal{Y} = \{0, 1\}$. Let $\mathbf{X} \in \mathbb{N}_0^{N \times G}$ be the *training* raw-count matrix, with cell index set $\mathcal{I} = \{1, \ldots, N\}$. Every training cell has a donor identifier d_n, a cell-type label t_n, and a condition label y_n. For donor d, cell type t, and condition y, define $\mathcal{I}_d = \{n \in \mathcal{I} : d_n = d\}, \mathcal{I}_{d,t,y} = \{n \in \mathcal{I} : d_n = d, t_n = t, y_n = y\}$, with $n_d = |\mathcal{I}_d|$ and $n_{d,t,y} = |\mathcal{I}_{d,t,y}|$.

In scROBM, we assign three role-guided latent variables for cells. The basal variable $\mathbf{z}^{(b)}$ represents major cell identity, the stimulus variable $\mathbf{z}^{(s)}$ represents perturbation semantics, and the donor-context variable $\mathbf{z}^{(c)}$ represents donor-associated structure. The conditional-invariance term acts on the deterministic basal and donor-context means exported by the mean cascade. In addition, we design a shared donor center $\boldsymbol{\nu}_d \in \mathbb{R}^{d_c}$ for every training donor. Therefore, scROBM considers three core issues. **RQ1:** Does $\mathbf{z}^{(b)}$ preserve cell identity? **RQ2:** Does $\mathbf{z}^{(s)}$ preserve condition semantics? **RQ3:** Does $\mathbf{z}^{(c)}$ preserve donor structure across conditions?

4 METHOD

4.1 FRAMEWORK OVERVIEW

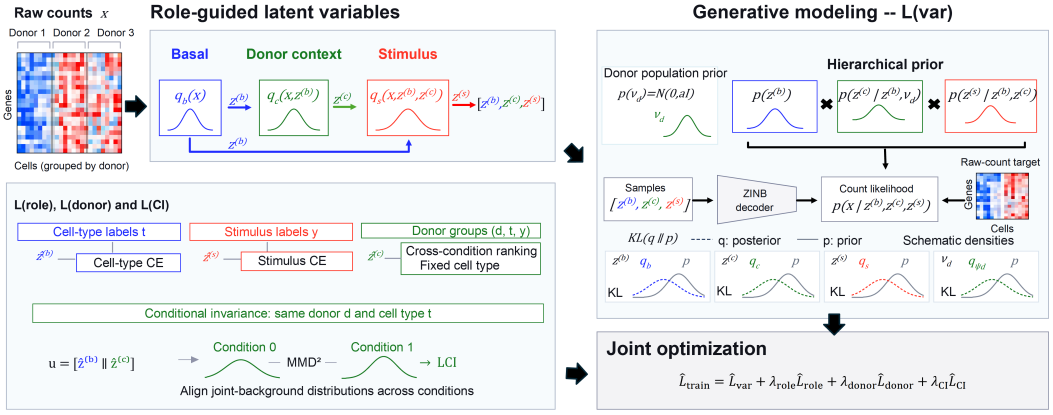


Figure 1: **Overview of scROBM.**

Figure 1 summarizes scROBM. A serial posterior encodes raw counts into the basal, donor-context, and stimulus variables (top left). Role supervision, cross-condition donor ranking, and the conditional-invariance (CI) penalty act on these variables (bottom left). A hierarchical prior with a donor-level center and a label-free ZINB count decoder define the variational objective (right). The four terms are combined in the joint objective $\widehat{\mathcal{L}}_{\text{train}}$ (bottom right).

scROBM maps expression $\mathbf{x}_n$ to three representations through a serial posterior. The inference module receives $T(\mathbf{x}_n)$ and produces Gaussian parameters. Posterior samples enter a label-free count decoder, scored against raw $\mathbf{x}_n$; a deterministic mean cascade supplies the role objectives. The hierarchical prior regularizes this inference path through the ELBO. Training metadata (t_n, y_n, d_n) provide supervision, comparison groups, and donor-center indexing. The label-free decoder is the architectural reconstruction bottleneck of this design, and the CI branch supplies the conditional semantic one.

We use *bottleneck* as a path restriction : it limits which inputs a prescribed part of the model may see, or which distributional information a prescribed channel may carry. scROBM contains two such bottlenecks with distinct sources. The *architectural reconstruction bottleneck* is imposed by the label-free count decoder: expression reconstruction is mediated entirely by the latent variables, so the latent allocation determines what the model can rebuild and where each factor of variation can reside. The *conditional semantic bottleneck* is imposed by the conditional-invariance penalty on the background channel: within a supported donor–cell-type stratum, condition information is treated as nuisance and suppressed in the joint basal and donor-context means. The two bottlenecks are coupled: the architectural bottleneck forces label-dependent signal into the shared latent variables, and only then does the semantic bottleneck have a channel to act on. The architectural bottleneck decides through which variables information can flow; the semantic bottleneck decides what the background channel should not carry.

The frozen basal output answers RQ1 through cell-type prediction, and the stimulus output answers RQ2 through perturbation-label prediction. RQ3 uses donor-conditioned means of the donor-context output for matching and clustering. All three representations share the reconstruction objective and are exported from expression alone.

4.2 GENERATIVE MODULES: HIERARCHICAL PRIOR AND COUNT DECODER

The basal state conditions donor context, and both condition the stimulus variable; a shared center $\boldsymbol{\nu}_d$ links cells from donor d and supports the donor role in RQ3. Let $\boldsymbol{\nu} = \{\boldsymbol{\nu}_d\}_{d \in \mathcal{D}_{\text{train}}}$ and $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^{N}$, we set up the full hierarchical prior as $p(\boldsymbol{\nu}_d) = \mathcal{N}(\mathbf{0}, a\mathbf{I}), p(\mathbf{z}_n^{(b)}) = \mathcal{N}(\mathbf{0}, \mathbf{I}), p(\mathbf{z}_n^{(c)} \mid$

$\mathbf{z}_n^{(b)}, \boldsymbol{\nu}_{d_n}) = \mathcal{N}\Big(h_c(\mathbf{z}_n^{(b)}) + \boldsymbol{\nu}_{d_n}, \sigma_c^2\mathbf{I}\Big)$, and $p(\mathbf{z}_n^{(s)} \mid \mathbf{z}_n^{(b)}, \mathbf{z}_n^{(c)}) = \mathcal{N}\Big(h_s(\mathbf{z}_n^{(b)}, \mathbf{z}_n^{(c)}), \sigma_s^2\mathbf{I}\Big)$. (Detailed in Appendix A). Conditional on the observed donor indices, the joint distribution is

$$p(\mathbf{X}, \mathbf{Z}, \boldsymbol{\nu}) = \prod_{d \in \mathcal{D}_{\text{train}}} p(\boldsymbol{\nu}_d) \prod_{n=1}^{N} p(\mathbf{z}_n^{(b)}) p(\mathbf{z}_n^{(c)} \mid \mathbf{z}_n^{(b)}, \boldsymbol{\nu}_{d_n}) p(\mathbf{z}_n^{(s)} \mid \mathbf{z}_n^{(b)}, \mathbf{z}_n^{(c)}) p_\theta(\mathbf{x}_n \mid \mathbf{z}_n). \tag{1}$$

The decoder connects all three latent variables to the observed expression profile through a label-free zero-inflated negative-binomial likelihood, conditioned on a fixed observed exposure ℓ_n (the raw-count library $\sum_g x_{ng}$ over the model's selected genes). Because the decoder receives the latent variables and the fixed exposure but no cell-type, condition, or donor label, any label-dependent reconstruction signal must be transmitted through the latent representation itself, beyond what the fixed exposure already reveals. The exposure is an unlabelled count summary that may co-vary with donor and condition; it is not a label input, so the statement above concerns label inputs rather than statistical independence from the metadata. This is the *architectural reconstruction bottleneck*.

4.3 Inference module: serial posterior and mean cascade

Inference supplies cell representations and estimates the donor centers used in the prior. The joint approximation (Appendix A) combines a diagonal Gaussian factor per training donor with a serial cell-level posterior

$$q_\phi(\mathbf{z}_n \mid \mathbf{x}_n) = q_{\phi_b}(\mathbf{z}_n^{(b)} \mid \mathbf{x}_n) q_{\phi_c}(\mathbf{z}_n^{(c)} \mid \mathbf{x}_n, \mathbf{z}_n^{(b)}) \times q_{\phi_s}(\mathbf{z}_n^{(s)} \mid \mathbf{x}_n, \mathbf{z}_n^{(b)}, \mathbf{z}_n^{(c)}). \tag{2}$$

Each factor is diagonal Gaussian, receives expression and its indicated upstream variables, and passes samples downstream, so sequential sampling carries upstream uncertainty through the posterior. The separate donor-center factor acts through the conditional-prior KL term. Training and unseen donors therefore share the same expression-only encoding path.

In algorithm instantiation, we use the following deterministic exports $\widehat{\mathbf{z}}_n^{(\cdot)}$:

$$\widehat{\mathbf{z}}_n^{(b)} = \boldsymbol{\mu}^{(b)}(\mathbf{x}_n), \widehat{\mathbf{z}}_n^{(c)} = \boldsymbol{\mu}^{(c)}(\mathbf{x}_n, \widehat{\mathbf{z}}_n^{(b)}), \widehat{\mathbf{z}}_n^{(s)} = \boldsymbol{\mu}^{(s)}(\mathbf{x}_n, \widehat{\mathbf{z}}_n^{(b)}, \widehat{\mathbf{z}}_n^{(c)}). \tag{3}$$

Here, all $\boldsymbol{\mu}^{(\cdot)}$ are variational parameters defined for all $\mathbf{z}^{(\cdot)}$s' posterior distributions, i.e., $q_{\phi_b}(\mathbf{z}_n^{(b)} \mid \mathbf{x}_n) = \mathcal{N}(\boldsymbol{\mu}_n^{(b)}, \text{diag}(\mathbf{v}_n^{(b)})), q_{\phi_c}(\mathbf{z}_n^{(c)} \mid \mathbf{x}_n, \mathbf{z}_n^{(b)}) = \mathcal{N}(\boldsymbol{\mu}_n^{(c)}, \text{diag}(\mathbf{v}_n^{(c)})), q_{\phi_s}(\mathbf{z}_n^{(s)} \mid \mathbf{x}_n, \mathbf{z}_n^{(b)}, \mathbf{z}_n^{(c)}) = \mathcal{N}(\boldsymbol{\mu}_n^{(s)}, \text{diag}(\mathbf{v}_n^{(s)}))$, as detailed in eq. S5. These plug-in conditional means generally differ from marginal posterior expectations for nonlinear encoders, and make the implementation efficient.

4.4 Learning objectives and their RQ roles

Shared variational objective. The generative modules (eq. 1) and the inference module give the normalized negative ELBO

$$\mathcal{L}_{\text{var}} = -\frac{1}{N}\sum_{n=1}^{N} \mathcal{E}_n + \frac{1}{N}\mathcal{K}_\nu, \qquad \mathcal{K}_\nu = \sum_{d \in \mathcal{D}_{\text{train}}} \text{KL}(q_{\psi_d} \| p(\boldsymbol{\nu}_d)). \tag{4}$$

The local term $\mathcal{E}_n$ combines count reconstruction with the three cell-level KL penalties (As detailed in eq. S9, $\mathcal{E}_n = \mathbb{E}_{q_b q_c q_s} \log p_\theta(\mathbf{x}_n \mid \mathbf{z}_n) - \mathcal{K}_n^{(b)} - \mathbb{E}_{q_b}\overline{\mathcal{K}}_n^{(c)}(\mathbf{z}_n^{(b)}) - \mathbb{E}_{q_b q_c}\mathcal{K}_n^{(s)}(\mathbf{z}_n^{(b)}, \mathbf{z}_n^{(c)})$). And, $\mathcal{K}_\nu$ counts each donor center once. Appendix A.3 gives the analytic center integration and serial estimator $\widehat{\mathcal{E}}_n$.

Conditional semantic bottleneck. The CI term is what implements the conditional semantic bottleneck: it is the only objective term that acts on the distribution of the joint background means rather than on their labels, and it suppresses condition information within each supported donor–cell-type stratum.

For every cell define the deterministic background representation $\mathbf{u}_n = [\widehat{\mathbf{z}}_n^{(b)} \| \widehat{\mathbf{z}}_n^{(c)}] \in \mathbb{R}^{d_b + d_c}$. Let $P_{d,t,y}^{bc}$ be its population distribution conditional on donor d, cell type t, and condition y. For supported binary condition pairs with both conditions observed, the invariance that RQ3's reading presupposes is $P_{d,t,0}^{bc} = P_{d,t,1}^{bc}$, which is equivalent to

$$\mathbf{U} \perp Y \mid (\mathsf{D}, \mathsf{T}). \tag{5}$$

Here D, T are random donor and cell-type labels. We then express the *conditional semantic bottleneck* through the loss $\mathcal{L}_{\mathrm{CI}}$, built on the maximum mean discrepancy (MMD) $\mathcal{L}_{\mathrm{CI}} = \frac{1}{M}\sum_{(d,t)\in\mathcal{S}_{\mathrm{CI}}} \mathrm{MMD}^2_k(P^{bc}_{d,t,0}, P^{bc}_{d,t,1})$ (Detailed in eq. S13). The kernel k is defined in eq. S12 and has three modes, which define the three CI implementations of scROBM: a *Gaussian* mode, a matched-scale product of the two per-block Gaussian kernels; a *linear* mode, a bandwidth-scaled inner product of the two blocks; and a *marginal* mode, the average of the two per-block Gaussian kernels. The three modes share the same objective and differ only in k. Support, kernel, and estimator details are in Appendix B, and the bandwidth pilot in Appendix C.4.

Cell identity and perturbation supervision (RQ1 and RQ2). Affine heads predict cell type from $\widehat{\mathbf{z}}^{(b)}_n$ and condition from the detached stimulus embedding $\breve{\mathbf{z}}^{(s)}_n$. The cell-level role loss $\mathcal{L}_{\mathrm{role},n}$ sums the two cross-entropies; its first term updates only the basal encoder and its head, and the second only the stimulus encoder and its head under the prescribed detached paths (definitions in Appendix A.3). RQ1 and RQ2 use separate downstream probes on the frozen hatted representations. Detached embeddings coincide numerically with the hatted exports when dropout is disabled; only their gradient routes differ.

Cross-condition donor ranking (RQ3). Alongside the shared donor prior, ranking trains same-donor cross-condition matching within each cell type; its comparisons correspond to the pairwise evaluation in RQ3. Batch centroids of the detached donor-context embedding $\breve{\mathbf{z}}^{(c)}_n$ are scored by cosine similarity against an epoch-refreshed, stop-gradient global candidate bank over all training donor–cell-type–condition groups, and an InfoNCE-style loss $\widehat{\mathcal{L}}_{\mathrm{donor}}$ (Appendix A.3) increases the softmax share of same-donor cross-condition positives against other-donor negatives. Gradients reach only the current query's ϕ_c path; the candidate bank and upstream basal input are detached. RQ3 evaluates this matching relation on unseen donor means and separately clusters them to assess donor identity across groups; the latter is an evaluation, with no additional clustering loss. The hierarchical KL complements ranking by pulling each shared donor offset toward the donor-level prior.

Complete objective. Let $\pi_n = \Pr(n \in \mathcal{B}) > 0$. Both additive cell-level losses use Horvitz–Thompson (HT) weights: the variational term is $\widehat{\mathcal{L}}_{\mathrm{var}} = -\frac{1}{N}\sum_{n\in\mathcal{B}} \widehat{\mathcal{E}}_n/\pi_n + \frac{1}{N}\mathcal{K}_\nu$, which also carries the all-donor KL, and the role term is $\widehat{\mathcal{L}}_{\mathrm{role}} = \frac{1}{N}\sum_{n\in\mathcal{B}} \mathcal{L}_{\mathrm{role},n}/\pi_n$. The complete objective is

$$\widehat{\mathcal{L}}_{\mathrm{train}} = \underbrace{\widehat{\mathcal{L}}_{\mathrm{var}}}_{\text{(architectural)}} + \lambda_{\mathrm{role}}\widehat{\mathcal{L}}_{\mathrm{role}} + \lambda_{\mathrm{donor}}\widehat{\mathcal{L}}_{\mathrm{donor}} + \lambda_{\mathrm{CI}} \underbrace{\widehat{\mathcal{L}}_{\mathrm{CI}}}_{\text{(semantic)}} . \tag{6}$$

The two underbraces mark where the bottlenecks of Section 4.1 enter the objective: the architectural bottleneck through the label-free reconstruction term inside $\widehat{\mathcal{L}}_{\mathrm{var}}$, and the conditional semantic bottleneck through $\widehat{\mathcal{L}}_{\mathrm{CI}}$. The KL terms retain unit weight, including the donor-center term ($\lambda_\nu = 1$), at the end of KL warm-up. Each step evaluates $\mathcal{K}_\nu$ over all training donors. The HT normalization targets the full training-cell mean; the donor ranking loss targets the sampled valid-anchor average. In practice these four terms share computation, and three lemmas make the resulting estimators tractable. First, the donor center $\boldsymbol{\nu}_d$ is the only latent variable outside the cell plate: Lemma 1 shows that, when the structured posterior makes q_c independent of $\boldsymbol{\nu}_{d_n}$ at a fixed basal sample, the expectation of the donor-context KL over $q(\boldsymbol{\nu}_d)$ has a closed form in which center uncertainty enters as the additive term $\sum_j s^2_{dj}/(2\sigma^2_c)$ and the conditional prior variance is unchanged, so this ELBO term is evaluated analytically without extra sampling of $\boldsymbol{\nu}_d$. Second, each step processes one stratified minibatch: Lemma 2 shows that, with $\pi_n > 0$ and within-batch conditional unbiasedness, the two additive terms weighted by $1/\pi_n$ (variational and role) equal the full training-set means in expectation, so the minibatch implementation does not change the declared objective; this is why $\widehat{\mathcal{L}}_{\mathrm{var}}$ and $\widehat{\mathcal{L}}_{\mathrm{role}}$ use Horvitz–Thompson normalization. Third, CI has a different structure: it is a within-stratum, donor–cell-type paired quantity rather than an additive term; Lemma 3, the unbiased finite-set estimator, shows that, with the within-group V-statistic carrying the finite-population factor $(n_{h,y}-1)/n_{h,y}$ and with normalization by the fixed M rather than the batch's eligible-stratum count, this estimator and its gradient are unbiased for the finite-training-set objective $\mathcal{L}_{\mathrm{CI}}$ (eq. S16), and it also explains why negative minibatch values must not be truncated.

The direct loss-gradient destinations are summarized in Table S1 (Appendix A.3): each auxiliary loss updates only its prescribed path, and CI differentiates through the basal and donor-context means, including the basal-to-context mean path. Optimizer weight decay is separate from these loss-gradient statements. CI must be able to reshape the background, whereas condition supervision must not; this asymmetry is why CI keeps the attached mean path while the stimulus and ranking branches detach it.

5 Experimental setup

5.1 Datasets and donor-aware protocol

The evaluation covers four donor-structured single-cell datasets. Each dataset provides a declared two-condition contrast, donor identifiers, cell-type labels, and paired donor–condition support for the RQ3 centroid analysis. The condition names are taken from the dataset manifest; the binary coding does not imply that $y = 1$ always denotes a stimulation or a treatment. The four datasets are scBloodNL-MTB (3hMTB versus 24hMTB; 11 donors, 88 eligible centroids, and 88 cross-condition queries), GSE297329 (manifest-defined contrast; 4 donors, 16 eligible centroids, and 16 queries), Zhang2021 blood (Pre versus Post; 3 donors, 24 eligible centroids, and 24 queries), and GSE236581 (colorectal-cancer anti-PD-1 neoadjuvant time series; baseline (Pre) versus on-treatment (Post); 22 eligible baseline donors and 24 queries, with a separate four-donor scROBM preview). The detailed data preparation and support rules appear in Appendix C.

5.2 Baselines and evaluation tasks

PCA, scVI, scANVI, and scDisInFact are fresh baselines evaluated through a generic frozen embedding. For scROBM, RQ1 probes $\widehat{\mathbf{z}}^{(b)}$ for cell type, RQ2 probes $\widehat{\mathbf{z}}^{(s)}$ for condition, and RQ3 evaluates donor matching, retrieval, and clustering from $\widehat{\mathbf{z}}^{(c)}$. All probes use the same scoring rules, and all donor comparisons keep cell type fixed.

5.3 Training and frozen export

Training samples stratified donor–group minibatches, applies KL warm-up followed by CI ramp-in under a scheduled objective, and selects checkpoints by development RQ3 matching AUROC. The selected checkpoint exports three expression-only representations through the mean cascade (eq. 3): the decoder output supports count modeling, while the frozen representations feed the RQ1 and RQ2 probes and the RQ3 centroid evaluations scored in Experiments. The network architectures, stratified optimization, and training state machine are specified in Appendix C (Algorithm 1); the frozen evaluation protocol, including centroid construction and scoring, is given in Appendix D, and Appendix F analyzes the computational cost of training, frozen export, and downstream evaluation.

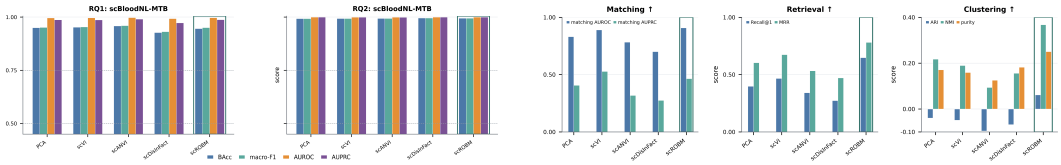
6 Results

We first report the scBloodNL-MTB benchmark, then the ablation study, the three additional datasets, and the donor-context application. Complete results are reported in Appendix E.

6.1 Main benchmark on scBloodNL-MTB

Figure 2(a) reports the RQ1 cell-type and RQ2 condition exports, and Figure 2(b) reports the RQ3 donor-associated metrics, for scROBM and the four generic baselines. The scBloodNL-MTB RQ3 values are retained in Table 1; complete tables for all four datasets are given in Appendix E. Bars report balanced accuracy, macro-F1, AUROC, and AUPRC for the four generic baselines and scROBM; the y-axes are zoomed to reveal the small residual differences. In (b), Panels report donor matching, retrieval, and clustering, with the outlined bar group marking scROBM. Exact values for this dataset and the three additional datasets are reported in Tables S3- S6.

Cell-type and condition tasks in RQ 1 and 2. On scBloodNL-MTB, RQ1 and RQ2 are close to saturation. scANVI has the strongest cell-type scores, but scROBM remains in the same range;



(a) RQ1/RQ2 exports **(b)** RQ3 donor structure

Figure 2: Main benchmark on scBloodNL-MTB. **(a)** Cell-type (RQ1) and condition (RQ2) prediction is close to saturation. **(b)** Donor-associated structure under RQ3.

Table 1: RQ3 donor-associated performance comparison on scBloodNL-MTB.

Method	matching AUROC ↑	matching AUPRC ↑	Recall@1 ↑	MRR ↑	single mAP ↑	clustering ARI ↑	clustering NMI ↑	clustering purity ↑
PCA	0.8308	0.4059	0.3977	0.6034	0.6034	-0.0399	0.2171	0.1705
scVI	0.8901	**0.5268**	0.4659	0.6735	0.6735	-0.0489	0.1897	0.1591
scANVI	0.7829	0.3183	0.3409	0.5326	0.5326	-0.0960	0.0936	0.1250
scDisInFact generic	0.7000	0.2747	0.2727	0.4707	0.4707	-0.0682	0.1556	0.1818
scROBM (Gaussian)	0.9074	0.4641	0.6477	0.7811	0.7811	0.0613	0.3673	0.2500
scROBM (linear)	**0.9085**	0.4674	0.6477	**0.7915**	**0.7915**	**0.1220**	**0.4549**	**0.3068**
scROBM (marginal)	0.9039	0.4809	**0.6591**	0.7896	0.7896	0.0997	0.4109	0.2955

condition scores exceed 0.99 for all methods, with the generic scDisInFact probe marginally ahead. These tasks provide a consistency check rather than a strong method separator.

Donor-associated structure under RQ3. RQ3 provides the clearest separation (Figure 2(b)). scROBM has the strongest matching AUROC, Recall@1, MRR, and all three clustering measures. scVI has the highest matching AUPRC. Because the scROBM export is a health-validation artifact while the fresh baselines use their declared final split, these comparisons are descriptive; the ablation tests the contribution of the design.

6.2 Ablation Study

Table 2 evaluates the contribution of the major components of scROBM on the unseen-donor matching task. The most pronounced effect comes from the donor-ranking objective. Removing donor ranking leads to a substantial degradation across all matching and retrieval metrics, with AUROC/AUPRC decreasing from $0.9074/0.4641$ to $0.8822/0.4153$, Recall@1 from 0.6477 to 0.4318, and MRR from 0.7811 to 0.6523. The degradation is also reflected in donor-level clustering, whose ARI, NMI, and purity decrease to 0.0556, 0.3596, and 0.2500, respectively. This result indicates that explicitly organizing $z^{(c)}$ around within-donor cross-condition correspondence is important for preserving donor-specific geometry beyond the generative objective alone.

Removing the CI loss produces a more moderate decrease in matching and retrieval performance, reducing AUROC from 0.9074 to 0.9029, AUPRC from 0.4641 to 0.4586, Recall@1 from 0.6477 to 0.6023, and MRR from 0.7811 to 0.7631. After removing CI, the clustering metrics change non-monotonically, so its contribution concentrates in cross-condition matching and retrieval. Similarly, removing the role loss preserves comparable matching AUROC and even yields a higher AUPRC (0.5032), but reduces Recall@1 and MRR; its clustering ARI and NMI are instead higher, so role supervision sharpens ranking-based retrieval rather than clustering. This pattern suggests that role supervision primarily improves the quality and organization of the retrieved donor structure in the latent space.

Finally, the Gaussian, linear, and marginal configurations achieve broadly comparable matching performance, with AUROC ranging from 0.9039 to 0.9085 and Recall@1 from 0.6477 to 0.6591. The linear configuration gives the strongest overall retrieval and clustering results, whereas the marginal configuration achieves the highest AUPRC and Recall@1. Thus, the observed RQ3 behavior is robust across CI parameterizations, although the choice shifts the balance between ranking-based retrieval and clustering quality.

Table 2: Ablation on scBloodNL–MTB.

Configuration	matching AUROC ↑	matching AUPRC ↑	Recall@1 ↑	MRR ↑	single mAP ↑	clustering ARI ↑	clustering NMI ↑	clustering purity ↑
scROBM (Gaussian)	0.9074	0.4641	0.6477	0.7811	0.7811	0.0613	0.3673	0.2500
scROBM (linear)	**0.9085**	0.4674	0.6477	**0.7915**	**0.7915**	**0.1220**	**0.4549**	**0.3068**
scROBM (marginal)	0.9039	0.4809	**0.6591**	0.7896	0.7896	0.0997	0.4109	0.2955
Remove CI loss	0.9029	0.4586	0.6023	0.7631	0.7631	0.0837	0.3728	0.2614
Remove role loss	0.9079	**0.5032**	0.6250	0.7744	0.7744	0.0955	0.4064	0.2841
Remove donorloss	0.8822	0.4153	0.4318	0.6523	0.6523	0.0556	0.3596	0.2500

6.3 Cross-dataset generalization

We applied scROBM to more diseases data and condition semantics, GSE297329, Zhang2021 blood, and GSE236581. Complete values are given in Tables S4–S6 in Appendix E.

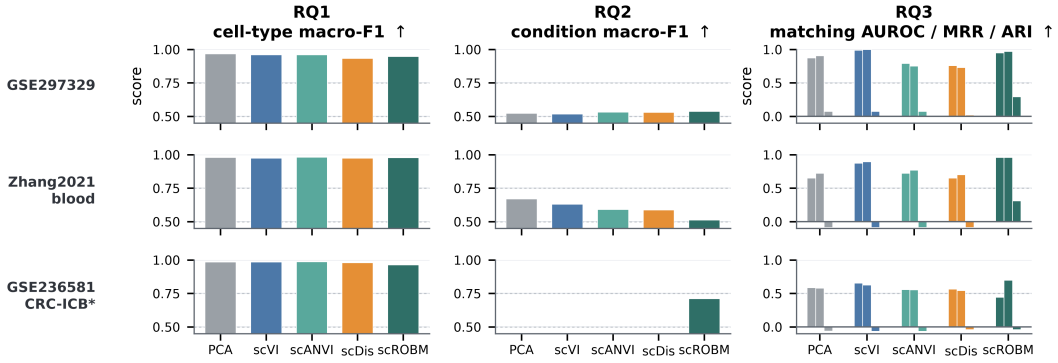


Figure 3: Cross-dataset summary beyond scBloodNL-MTB.

Across these datasets, RQ1 remains in a narrow, high range, whereas RQ2 tracks the difficulty of the condition contrast. scROBM remains usable for both tasks, without being uniformly best. Its clearest advantage is RQ3: the compact view in Figure 3 shows the strong donor-structure pattern in matching, retrieval, and clustering, while the full metric set remains in the appendix. On GSE297329 and Zhang2021 blood, scROBM has the strongest clustering ARI; it also leads the representative matching and retrieval measures on Zhang2021 blood.

6.4 Donor-context observation

Figure 4 collects the latent-space observations for the three exports of scROBM on GSE236581. In the basal representation (a), cells colored by cell type form six distinct clusters, whereas the same cells colored by condition show no separable structure; cell identity is therefore organized in $\widehat{\mathbf{z}}^{(b)}$ and condition information is not. In the stimulus representation (b), the six annotated cell types (B, Epi, ILC, Mye, Stromal, T) remain separable and, within each cell-type group, the two conditions split into distinguishable sub-clusters (Pre/Post, $y = 0/1$); condition semantics are therefore organized in $\widehat{\mathbf{z}}^{(s)}$.

The donor-context export $\widehat{\mathbf{z}}^{(c)}$ completes the picture (Figure 4(c)). Colored by donor across all donors, it forms compact clusters that largely separate the individual donors, so donor-level grouping is well induced in $\widehat{\mathbf{z}}^{(c)}$. Fixing a single donor and coloring by cell type, the major cell types remain separable within that donor, so the donor view retains within-donor cell identity. Zooming into one cell type at a time and coloring by condition then shows that the separation between the two conditions is cell-type dependent: for T cells the baseline and on-treatment cells fall into two distinct clusters, whereas for Epi cells they remain entangled. This cell-type dependence is consistent with the biology of the treatment. GSE236581 is a colorectal-cancer anti-PD-1 neoadjuvant time series in which the baseline (Pre) and on-treatment (Post) samples of each donor are profiled together Chen et al. (2024), and T cells are the direct target of the blockade: anti-PD-1 reinvigorates exhausted CD8 T cells and expands tumor-reactive-like CD8 T cells Sade-Feldman et al. (2018), so their programs

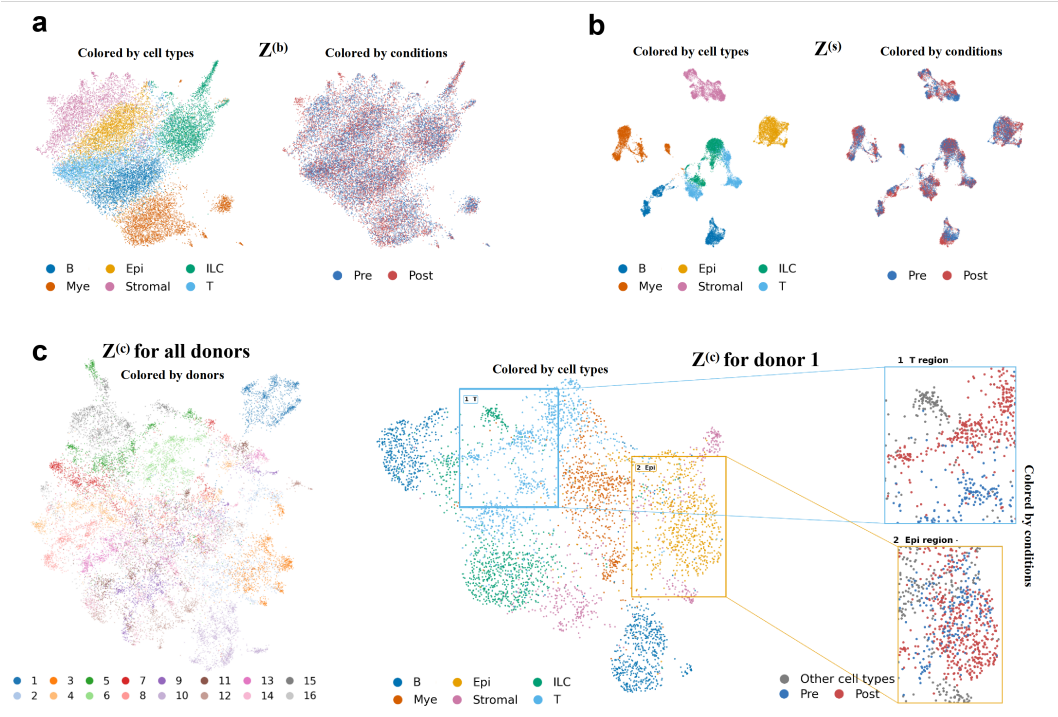


Figure 4: **Latent-space observations on GSE236581.**

shift across the two time points and the conditions separate in the latent space as expected, whereas the epithelial compartment is not the direct target of the blockade and changes less coherently.

The results suggest that role-guided design enables each cell-level latent variable to be supervised for a distinct role, the basal block (a) answers RQ1, the stimulus block (b) answers RQ2, and the donor-context block (c) answers RQ3.

7 Discussion and Conclusion

scROBM pairs a label-free count decoder with a conditional-invariance penalty. The decoder creates the architectural reconstruction bottleneck, forcing every label-dependent reconstruction signal through the latent variables; CI then constrains the background channel, suppressing condition information in the joint basal and donor-context means within supported donor–cell-type strata. The two are complementary: CI aligns conditions within a donor and cell type, whereas donor ranking separates donors, and a constant background can satisfy CI while losing donor or cell identity, so both objectives are needed.

Computationally, the all-donor KL costs $O(Dd_c)$ at every step and the CI pair terms scale with the number of eligible strata, whereas frozen export needs no prior, decoder, donor table, or kernel (Appendix F); training cost thus grows mainly with training-donor and stratum counts rather than with downstream inference.

Limitations. CI requires the same donor and cell type to appear in both conditions; missing pairing, or confounding between condition and donor, therefore bounds how far the invariance can be identified. Although we prove that the minibatch CI estimator is unbiased for the finite training-set objective, it is not for a population criterion, so residual condition readability remains possible even when the training MMD is small (Appendix B.5). Scaling to cohorts with many donors remains an open question.

AI USE STATEMENT

Large language model (LLM) and AI tools were used during this work, and the authors disclose their use here. Their role was confined to four purposes: (i) assisting and polishing the writing; (ii) retrieval and discovery, for example locating related work; (iii) research ideation and execution, for example evaluating the contribution of the ideas and reviewing related literatures; and (iv) drafting portions of the text, which the authors subsequently edited and verified. The authors take full responsibility for the content of this paper.

REFERENCES

Abhinav K. Adduri et al. Predicting cellular responses to perturbation across diverse contexts with STATE. bioRxiv, 2025.

Constantin Ahlmann-Eltze, Wolfgang Huber, and Simon Anders. Deep-learning-based gene perturbation effect prediction does not yet outperform simple linear baselines. *Nature Methods*, 22: 1657–1661, 2025. doi: 10.1038/s41592-025-02772-6.

Shaokun An, Jae-Won Cho, Kai Cao, Jiankang Xiong, Martin Hemberg, and Lin Wan. scCausalVI disentangles single-cell perturbation responses with causality-aware generative model. *Cell Systems*, 16(11):101443, 2025. doi: 10.1016/j.cels.2025.101443.

Y. Chen, D. Wang, Y. Li, L. Qi, et al. Spatiotemporal single-cell analysis decodes cellular dynamics underlying different responses to immunotherapy in colorectal cancer. *Cancer Cell*, 42(7):1268–1285, 2024.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scGPT: toward building a foundation model for single-cell multi-omics using generative ai. *Nature Methods*, 21:1470–1480, 2024. doi: 10.1038/s41592-024-02201-0.

Yicheng Gao, Kejing Dong, Caihua Shan, Dongsheng Li, and Qi Liu. Causal disentanglement for single-cell representations and controllable counterfactual generation. *Nature Communications*, 16:6775, 2025. doi: 10.1038/s41467-025-62008-1.

Adam Gayoso, Romain Lopez, Galen Xing, Pierre Boyeau, Valeh Valiollah Pour Amiri, Justin Hong, Katherine Wu, Michael Jayasuriya, Edouard Mehlman, Maxime Langevin, et al. A python library for probabilistic analysis of single-cell omics data. *Nature Biotechnology*, 40(2):163–166, 2022.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012.

Minsheng Hao, Jing Gong, Xin Zeng, Chiming Liu, Yucheng Guo, Xingyi Cheng, Taifeng Wang, Jianzhu Ma, Xuegong Zhang, and Le Song. Large-scale foundation model on single-cell transcriptomics. *Nature Methods*, 21:1481–1491, 2024. doi: 10.1038/s41592-024-02305-7.

Shizhe Hu, Zhengzheng Lou, Xiaoqiang Yan, and Yangdong Ye. A survey on information bottleneck. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8), 2024. doi: 10.1109/TPAMI.2024.3366344.

Kemal Inecik, Aleyna Kara, Antony Rose, Muzlifah Haniffa, and Fabian J. Theis. TarDis: Achieving robust and structured disentanglement of multiple covariates. *Cell Systems*, 17(5):101573, 2026. doi: 10.1016/j.cels.2026.101573.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *Proceedings of the 2nd International Conference on Learning Representations*, 2014. URL https://arxiv.org/abs/1312.6114.

Romain Lopez, Jeffrey Regier, Michael B. Cole, Michael I. Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15:1053–1058, 2018. doi: 10.1038/s41592-018-0229-2.

Mohammad Lotfollahi, F. Alexander Wolf, and Fabian J. Theis. scgen predicts single-cell perturbation responses. *Nature Methods*, 16(8):715–721, 2019.

Malte D. Luecken, Maren Büttner, Kridsadakorn Chaichoompu, Anna Danese, Marta Interlandi, et al. Benchmarking atlas-level data integration in single-cell genomics. *Nature Methods*, 19(1): 41–50, 2022. doi: 10.1038/s41592-021-01336-8.

Stathis Megas, Daniel G. Chen, Krzysztof Polanski, Moshe Eliasof, Carola-Bibiane Schoenlieb, and Sarah A. Teichmann. Estimation of single-cell and tissue perturbation effect in spatial transcrip-

tomics via spatial causal disentanglement. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Tqdsruwyac.

Daniel Moyer, Shuyang Gao, Rob Brekelmans, Aram Galstyan, and Greg Ver Steeg. Invariant representations without adversarial training. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Claudio Novella-Rausell, David J. Peters, and Ahmed Mahfouz. Integrative learning of disentangled representations from single-cell RNA-sequencing datasets. *Computational and Structural Biotechnology Journal*, 35(1):0015, 2026. doi: 10.34133/csbj.0015.

Alessandro Palma, Till Richter, Hanyi Zhang, Manuel Lubetzki, Alexander Tong, Andrea Dittadi, and Fabian Theis. Multi-modal and multi-attribute generation of single cells with CFGen. In *Proceedings of the 13th International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/13079a5dd5c994a7e5f0cc691e0338e8-Abstract-Conference.html.

Zoe Piran, Niv Cohen, Yedid Hoshen, Mor Nitzan, et al. Disentanglement of single-cell data with biolord. *Nature Biotechnology*, 42:1678–1683, 2024. doi: 10.1038/s41587-023-02079-x.

Yusuf Roohani, Kexin Huang, and Jure Leskovec. Predicting transcriptional outcomes of novel multigene perturbations with GEARS. *Nature Biotechnology*, 42(6):927–935, 2024. doi: 10.1038/s41587-023-01905-6.

Moshe Sade-Feldman, Keren Yizhak, Shannon L. Bjorgaard, John P. Ray, Christina G. de Boer, Russell W. Jenkins, David J. Lieb, Jonathan H. Chen, Dennie T. Frederick, Michal Barzily-Rokni, et al. Defining t cell states associated with response to checkpoint immunotherapy in melanoma. *Cell*, 175(4):998–1013, 2018.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Linting Wang, Chihao Zhang, and Shi-Hua Zhang. Batch effects remain a fundamental barrier to universal embeddings in single-cell foundation models. bioRxiv, 2025.

Ethan Weinberger, Chris Lin, and Su-In Lee. Isolating salient variations of interest in single-cell data with contrastiveVI. *Nature Methods*, 20(9):1336–1345, 2023. doi: 10.1038/s41592-023-01955-3.

Hongzhi Wen, Wenzhuo Tang, Xinnan Dai, Jiayuan Ding, Wei Jin, Yuying Xie, and Jiliang Tang. CellPLM: Pre-training of cell language model beyond single cells. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=BKXvPDekud.

Jialu Wu, Qing Ye, Yilin Wang, Renling Hu, Yiheng Zhu, Mingze Yin, Tianyue Wang, Jike Wang, Chang-Yu Hsieh, and Tingjun Hou. Biology-driven insights into the power of single-cell foundation models. *Genome Biology*, 26(1):334, 2025. doi: 10.1186/s13059-025-03781-6.

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):e9620, 2021. doi: 10.15252/msb.20209620.

Chenglei Yu, Chuanrui Wang, Bangyan Liao, and Tailin Wu. scDFM: Distributional flow matching for robust single-cell perturbation prediction. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=QSGanMEcUV.

Ziqi Zhang, Xinye Zhao, Mehak Bindra, Peng Qiu, and Xiuwei Zhang. scdisinfact: disentangled learning for integration and prediction of multi-batch multi-condition single-cell rna-sequencing data. *Nature Communications*, 15(1):912, 2024.

Ouyang Zhu and Jun Li. Scouter predicts transcriptional responses to genetic perturbations with large language model embeddings. *Nature Computational Science*, 6:21–28, 2026. doi: 10.1038/s43588-025-00912-8.