PERTMIX: LEARNABLE DISTRIBUTION MIXING FOR UNSEEN PERTURBATION PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Single-cell perturbation experiments profile how genetic and chemical interventions reshape cellular states at transcriptome scale, and predicting the responses to unseen perturbations is a central problem in building virtual cells. Existing methods learn a condition-to-state map from the responses of observed perturbations, which leaves the paths between observed conditions without training signal; the shape of the map in these regions is therefore unconstrained, which limits generalization to unseen perturbations. We study the local geometry of this map and characterize the difficulty as directional-derivative mismatch and curvature anomaly, and we show that both grow as observation coverage decreases. Building on this view, we introduce mixup to perturbation response prediction and use intermediate conditions together with distribution-level supervision to constrain the map between observed points. Because the supervision target is a cell population distribution rather than a label, the condition and the response cannot share a single interpolation coefficient. We therefore propose PertMix, which adjusts the location and the shape of the synthesized condition with a learnable mixing rule under an endpoint-consistency constraint, and learns this rule from the feedback of the predictor on observed perturbations. The method provides two optimization variants: bilevel gradients, which differentiate through a one-step virtual update, and policy gradients, which avoid second-order computation and adapt to predictors of different complexity. On two unseen-perturbation tasks, gene and chemical perturbation prediction, we compare a common set of baselines under a unified fixed split and six metrics. The results indicate that distribution-space mixing supervision offers a viable route to perturbation generalization.

1 INTRODUCTION

Combining single-cell sequencing with pooled perturbation screens makes it possible to measure transcriptional states under genetic or chemical interventions at single-cell resolution. Perturb-seq links guide RNAs to transcriptome readouts and therefore connects gene function directly to expression phenotypes (Dixit et al., 2016), and subsequent work extends the strategy to genome scale, genetic interaction screens, and multimodal readouts (Adamson et al., 2016; Norman et al., 2019; Frangieh et al., 2021; Replogle et al., 2022). Chemical perturbations are represented by platforms such as Sci-Plex and L1000, which profile compounds and their doses across cell lines (Srivatsan et al., 2020; Subramanian et al., 2017), and the recently released Tahoe-100M atlas extends compound treatment to a billion-level cell count, fifty cell lines, and hundreds of compounds at multiple doses (Zhang et al., 2026). These resources, together with efforts that harmonize single-cell perturbation data across studies (Peidli et al., 2024), motivate a problem setting that has attracted growing attention: given a cell background, can a model predict the transcriptional change caused by a perturbation that never appears during training?

The problem is usually posed as distribution prediction: a model receives the control population of a cell line together with a perturbation condition representation and outputs the single-cell expression distribution under that condition. Recent systematic evaluations and community challenges show that deep models have limited advantage in this setting, since simple baselines match them on several metrics and the gains concentrate in the mean expression shift rather than in response structure (Wei et al., 2025; Li et al., 2026; Roohani et al., 2025). We look for the cause in the local properties of the condition-to-state map. Under discrete perturbation supervision there is no training signal between

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.

observed conditions, so the shape of the map along these paths is determined entirely by the inductive bias of the model. One extreme is to approximate the path by a rigid straight line, which over-linearizes the responses of intermediate conditions; the other leaves local variation unconstrained, so that the response curvature oscillates in a way that disagrees with the true geometry. Both are directly measurable on held-out perturbations, and we call them directional-derivative mismatch and curvature anomaly.

Prior work on perturbation response prediction follows three main directions. Latent-variable and additive-decomposition methods split a cell state into a background component and a perturbation component and drive prediction with a condition-specific latent shift, including scVI, scGen, CPA, SAMS-VAE, BioLORD, and scLAMBDA (Lopez et al., 2018; Lotfollahi et al., 2019; 2023; Bereket & Karaletsos, 2023; Piran et al., 2024; Qi et al., 2024). Condition-representation and structural-prior methods inject external knowledge into the perturbation encoding, such as the gene relation graph of GEARS (Roohani et al., 2023), the pretrained gene semantic representations of GenePT and GenePert (Chen & Zou, 2023), and the multi-target compound conditions of PRNet. Distribution-transport and generative methods learn a transformation from the control distribution to the perturbed distribution (Bunne et al., 2023; Driessen et al., 2026), including conditional flow matching in CellFlow (Klein et al., 2025), diffusion in PerturbDiff (Yuan et al., 2026), and the set-level state transition of STATE (Adduri et al., 2026). These methods differ in how they model responses, but they share the same supervision target, namely the responses of observed perturbations, and none of them explicitly constrains the shape of the map between observed conditions.

Based on these observations, we introduce mixup to construct intermediate conditions in the condition space, so that the paths between observed points receive supervision. Applying the idea directly is not straightforward. Standard mixup interpolates conditions and responses with the same coefficient, which assumes that responses vary linearly with the condition, whereas perturbation responses are cell population distributions and a cell-wise linear mixture of two populations does not correspond to the true distribution at the intermediate condition. The supervision target is a distribution rather than a label, so the quality of a synthetic target cannot be read off directly. PertMix therefore treats the mixing rule itself as the learning object, and drives it with the prediction feedback of the predictor on observed perturbations, as detailed in Section 4.

These observations lead to three contributions.

- **A geometric account of unseen-perturbation generalization.** We characterize the difficulty through the local geometry of the condition-to-state map, introduce two measurable diagnostics, directional-derivative mismatch and curvature anomaly, and examine their trend on STATE at three levels of observation coverage (25%, 50%, and 100%).
- **A learnable distribution mixer that generalizes mixup from label space to distribution space.** Under an endpoint-consistency constraint, the mixer adjusts the center and the shape of the synthesized target distribution, so that it departs from linear interpolation between observed conditions while reducing exactly to the original supervision at the endpoints; it is updated by the prediction feedback of the predictor through bilevel-gradient and policy-gradient variants.
- **A unified evaluation for two unseen-perturbation tasks.** A fixed split and six metrics for gene and chemical perturbation prediction, with a comparison of representative baselines on both tasks and an examination of the effect of distribution-space mixing supervision.

2 PRELIMINARY

2.1 RNA-SEQ DATA

Single-cell RNA sequencing data record the expression counts of cells across genes. Consider an experiment with n cells and G genes, and denote the observation matrix by $X \in \mathbb{R}^{n \times G}$, where each row is the expression vector of one cell. A cell line c describes the background state of the cell, and a perturbation p describes the intervention applied; in gene perturbation p is a target gene, and in chemical perturbation p is a compound identity. Cells that receive no specific intervention form the control population, which is used to estimate the baseline expression distribution of the cell line under no perturbation. The absolute scale of single-cell counts depends on sequencing depth, so comparing expression directly across cells or experiments is unreliable; before modeling,

we therefore normalize each cell by its total count to 10^4 counts per cell (CP10K) and apply a $\log(1+\cdot)$ transform to compress the dynamic range of highly expressed genes. Model outputs and all evaluation quantities are defined in this normalized log space, and the expression space is further restricted to genes with high variance to reduce dimensionality and suppress noise from low-information genes.

2.2 Unseen Perturbation Prediction Task

Given the control distribution Y_c^{ctrl} of cell line c and a perturbation condition representation e_p, the task is to predict the post-perturbation distribution under that condition. Let f_θ denote the prediction model; then

$$\hat{Y}_{c,p} = f_\theta\big(Y_c^{\mathrm{ctrl}},\ e_p,\ c\big). \tag{1}$$

Condition representations come from a fixed encoding of the perturbation identity; both gene and compound embeddings remain frozen during training. Generalization happens entirely at the perturbation level: every cell of an observed perturbation is visible during training and none of an unseen perturbation is, so response structure transfers to a new condition only through the condition representation. The objective is therefore to reduce the distribution-level discrepancy on unseen perturbations,

$$\min_\theta\ \mathbb{E}_{p\sim\mathcal{P}_{\mathrm{unseen}}}\ \mathcal{D}\big(\hat{Y}_{c,p},\ Y_{c,p}\big), \tag{2}$$

where $\mathcal{D}$ denotes a distribution distance and the expectation is taken over the set of unseen perturbations. The metrics cover the expression magnitude, the effect correlation, and the differential expression response, and their definitions and aggregation settings are given in Appendix D.

2.3 The Basic Form of Mixup

mixup constructs virtual training samples by linearly interpolating pairs of examples (Zhang et al., 2018):

$$\tilde{x} = \lambda x_a + (1-\lambda)x_b, \qquad \tilde{y} = \lambda y_a + (1-\lambda)y_b, \qquad \lambda \sim \mathrm{Beta}(\alpha,\alpha). \tag{3}$$

The shared coefficient λ implicitly assumes that the supervision target varies linearly along the segment between the two examples. Manifold Mixup moves the interpolation to an intermediate representation and shows that the operation acts as smoothing at the representation level (Verma et al., 2019); in terms of directional derivatives, it applies a first-order smoothness constraint to the loss along the interpolation direction (Zou et al., 2023). In perturbation response prediction, the supervision target is a distribution rather than a label, and the condition and the response live in different spaces. Interpolating two perturbation populations cell by cell produces a convex combination of endpoint samples whose statistical structure depends on how cells are paired, and this combination differs from the true response distribution at the intermediate condition. Applying mixup here therefore requires redesigning the construction of the interpolation.

3 Motivation

We view unseen perturbation generalization as a question about the response geometry between conditions. A model has to fit the finite set of observed perturbations, and it also has to form, between these discrete conditions, a response map that generalizes to unseen perturbations. Since the training supervision lies mainly on observed conditions, the shape of this map is determined to a large extent by the inductive bias of the model itself.

3.1 Three Shapes of the Response Geometry

Figure 1 shows this structure. In the condition space the model observes only a few discrete perturbations, and it has to generate the corresponding cell population response for unobserved conditions. Along a path between conditions, three typical response geometries can be distinguished. In the underconstrained case, the model fits the observed conditions but produces local deviations between them that are unrelated to a reasonable response trend. In the overly linearized case, the model imposes a strong local linear relation between conditions, so that an intermediate response is determined mainly by the linear relation of the two endpoints, and the magnitude change and

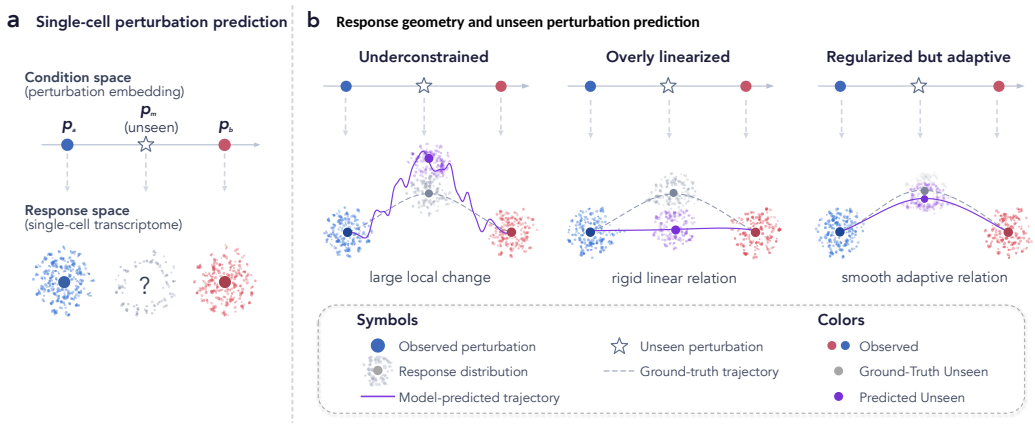


Figure 1: The unseen perturbation prediction task and the response geometry along condition paths. The left column shows the observed and unseen perturbations and the distribution to be predicted, in the condition space (perturbation representation) and the response space (single-cell transcriptome). The right column contrasts three shapes: underconstrained, overly linearized, and regularized but adaptive. Dashed curves denote the true response path and solid curves the model prediction.

the population heterogeneity of the true response are difficult to retain. A more reasonable response map keeps a regular and smooth variation between observed conditions while retaining the nonlinear response structure that the data require. Figure 1(b) contrasts the three cases. They point to the same core question: the prediction of unseen perturbations depends not only on how the model fits the observed conditions, but also on how it varies between them, and discrete perturbation supervision does not constrain that region directly.

3.2 Observation Coverage and Response Geometry

To relate unseen perturbation prediction to the local response geometry, we vary the perturbation coverage of the training data and examine prediction performance together with two response-level geometric diagnostics: the first-order mismatch between the local directional derivative of the model and the true endpoint response change, and the finite-step directional curvature of the response map along the local condition direction. Their definitions and computation are given in Appendix C.

We use three nested observation coverage levels, 25%, 50%, and 100%, on STATE, and compare the original model with a geometric-regularization variant that constrains the directional derivative and the curvature. The two branches use the same data, split, random seed, and training budget and differ only in whether the geometric regularizer is applied. The experiment asks whether a change in training perturbation coverage is accompanied by systematic changes in unseen perturbation prediction and in the local response geometry, and whether explicitly improving these geometric properties further improves unseen perturbation prediction.

For the model without additional geometric constraints, prediction on unseen perturbations degrades as the perturbation coverage decreases, and both geometric diagnostics grow systematically. The normalized curvature anomaly rises from 8.9 at 100% coverage to 13.8 at 50% and 29.1 at 25%, and the first-order mismatch rises from 1.14 and 1.25 to 1.58. Adding the geometric regularizer improves these diagnostics substantially: the normalized curvature anomaly at the three coverage levels falls to 2.0, 1.5, and 4.2, and the first-order diagnostic also decreases clearly. This geometric improvement does not lead to a corresponding prediction gain. Among the six prediction metrics, only expression MSE at full coverage improves, by 12.5%, while every other metric decreases at all three coverage levels; the Jaccard overlap of the significant gene sets drops by 17.9%, 22.3%, and 14.6% at 25%, 50%, and 100% coverage.

These results indicate that a lower perturbation coverage is accompanied both by worse prediction and by a less regular inter-condition response geometry, which suggests that when discrete perturbation supervision is insufficient, the behavior of the model between observed conditions is more easily left unconstrained. Further geometric regularization improves the diagnostics clearly but does

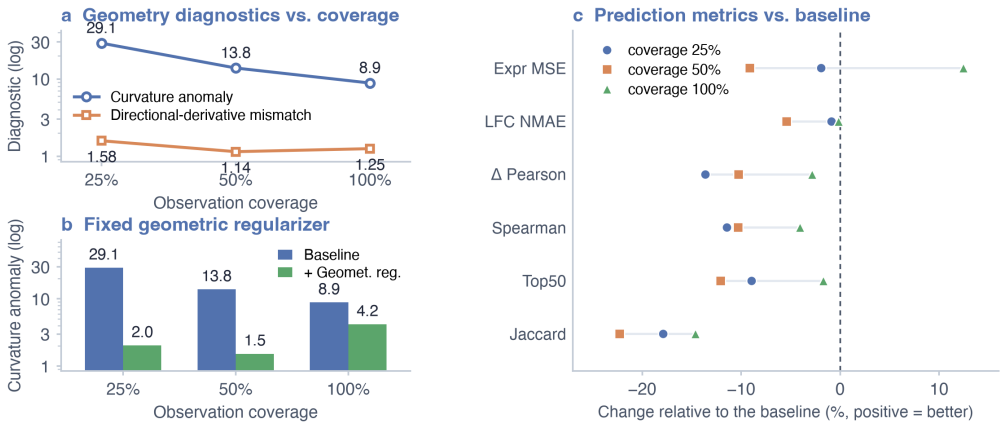


Figure 2: Response geometry and prediction performance under observation coverage. (a) Normalized geometric diagnostics of the baseline branch, on a logarithmic vertical axis. (b) Macro average of the normalized curvature anomaly over three cell lines before and after the fixed geometric regularizer. (c) Relative change of the regularized branch over the baseline on the six prediction metrics; positive is better.

not bring better unseen perturbation generalization. Effective inter-condition supervision should therefore not simply pursue a lower curvature or a stronger local linearity, but should regularize the behavior between conditions while retaining the data-dependent nonlinear response. Based on this observation, we propose PertMix, which constrains the model between observed perturbations through a Mixup-based interpolation supervision and, with a constrained and learnable distribution correction, allows the intermediate supervision to adapt according to prediction feedback.

4 Method

4.1 Overview

Sections 3.1 and 3.2 show that the difficulty lies in the supervision between observed conditions rather than in the fit at those conditions, and that flattening the response path does not remove it. PertMix therefore makes the construction of that supervision learnable, which raises two challenges. The response at an intermediate condition is in general not a linear combination of the endpoint distributions, so the mixing rule must be expressive enough to carry condition-specific magnitude and heterogeneity while still reducing to the original supervision at the endpoints. Intermediate conditions are unobserved, so a mixed target has no direct label and can only be judged through the predictor.

PertMix addresses the two challenges with two components, described in Sections 4.2 and 4.3. A constrained learnable mixer corrects the location and the shape of a linearly interpolated distribution, and the prediction feedback on observed perturbations provides the supervision that updates it. Figure 3 summarizes the pipeline.

4.2 Constrained Learnable Distribution Mixing

Let the conditions of a pair of observed perturbations be e_a and e_b, the corresponding cell populations be Y_a and Y_b, and the mixing coefficient be λ.

Condition interpolation. Condition interpolation is performed in the perturbation representation space:

$$\tilde{e}_\lambda = \lambda\, e_a + (1-\lambda)\, e_b. \tag{4}$$

Because the form depends only on the condition representation, it applies to gene and compound conditions alike. The corresponding linear population interpolation is

$$Y_\lambda^{\mathrm{lin}} = \lambda\, Y_a + (1-\lambda)\, Y_b. \tag{5}$$

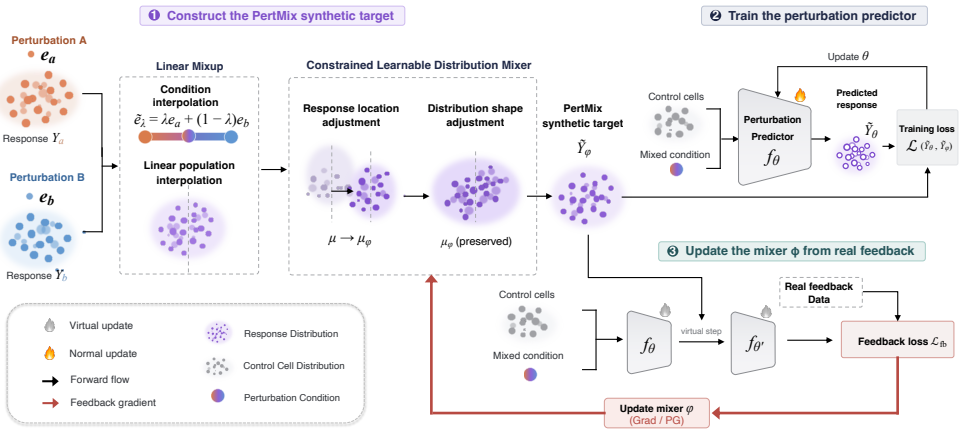


Figure 3: Overall pipeline of PertMix. (1) For a pair of observed perturbations, condition and linear population interpolation give the intermediate condition and a linear reference distribution; the constrained learnable mixer then adjusts the response location and shape to form the synthetic target, while endpoint supervision stays unchanged. (2) The synthetic target replaces the original response supervision for training the predictor. (3) The predictor is virtually updated for one step and evaluated on real feedback conditions, and the feedback loss updates the mixer by the bilevel-gradient or the policy-gradient estimator.

It gives a convex combination of the two endpoint distributions; since a response distribution is determined by population statistics and cell-by-cell pairing carries no biological meaning, we use Y_λ^{lin} as a reference rather than as a supervision target.

Learnable mixer. Linear population interpolation depends only on the endpoints and cannot express the response magnitude and heterogeneity of the intermediate condition itself, so we introduce the mixer on top of it. The mixer ϕ maps the interpolation coefficient together with the endpoint condition statistics to a 10-dimensional mixing action. Its last layer is initialized to zero, so that at the beginning of training the output equals linear interpolation and the method learns a deviation from standard mixup instead of replacing that supervision; Appendix B.6 gives the network and the quantities recorded during training.

The mixing action is divided into three parts. The first component adjusts the mixing coefficient of the synthesized condition:

$$\beta = \lambda + \tfrac{1}{2}\,\lambda(1-\lambda)\tanh(a_0), \qquad z = \beta\, a + (1-\beta)\, b. \tag{6}$$

The factor $\lambda(1-\lambda)$ concentrates the adjustment between the two observed conditions and produces no shift at the endpoints. The second group of components acts on the distribution center. With $B \in \mathbb{R}^{G\times 8}$ denoting the response bases along the gene dimension and with the weight $w = 4\lambda(1-\lambda)$, which is zero at the endpoints and maximal in the middle, the adjustment of the center is

$$\Delta = \tanh\left(a_{1:9} B^\top\right), \qquad \mu^\star = \overline{z} \odot \exp\left(0.2\, w\, \Delta\right), \tag{7}$$

where $\overline{z}$ is the population mean. This form modulates the center shift on an exponential scale, so that the synthetic target has a condition-specific mean at intermediate conditions. The third component adjusts the response magnitude. Let

$$p = \exp\left(0.3\, w\, \tanh(a_9)\right), \qquad v = \exp\left(p\, \ln(1+z)\right) - 1. \tag{8}$$

The synthetic target is then

$$\hat{Y}_\phi = \mu^\star \odot \frac{v}{\overline{v}}. \tag{9}$$

The power transform acts on the nonnegative expression space and changes the magnitude distribution while preserving the relative structure between genes, and the normalization term returns the transformed population mean to $\mu^\star$. The synthetic target therefore has both a condition-specific center and a magnitude and dispersion that vary with the condition.

Endpoint consistency. The constraint is implemented by direct substitution: $\hat{Y}_\phi = Y_a$ when $\lambda = 1$, and $\hat{Y}_\phi = Y_b$ when $\lambda = 0$, so the mixer does not change the supervision at observed conditions.

4.3 Supervision Signal and Optimization of the Mixer

Virtual update. Intermediate conditions have no observations, so the mixing result cannot be evaluated directly and is scored through the predictor. The synthetic target acts on the predictor through a single virtual update:

$$\theta' = \theta - \eta \nabla_\theta \mathcal{L}\big(f_\theta,\ \hat{Y}_\phi\big). \tag{10}$$

The single step makes the outer signal reflect the immediate usability of the synthetic target for the predictor. The updated predictor is then evaluated on observed conditions, which gives the optimization objective of the mixer:

$$\max_{\phi}\ s(\phi) - 0.01\,\|a\|^2, \tag{11}$$

where $s(\phi)$ is a response-level prediction score computed on validation conditions, and the quadratic penalty suppresses the magnitude of the action and prevents the mixer from deviating excessively. All feedback signals come from observed perturbations, and test conditions do not participate.

Two optimization variants. The differentiability of the virtual update determines how the mixer gradient is computed. The bilevel-gradient variant follows the meta-learning formulation in which an inner update is differentiated through with implicit gradients (Finn et al., 2017; Rajeswaran et al., 2019) and reparameterizes and samples K actions for each mixed condition:

$$a_k = \mu_\phi + \sigma\,\epsilon_k, \qquad \epsilon_k \sim \mathcal{N}(0, I), \qquad \sigma = 0.2. \tag{12}$$

Each sample corresponds to one differentiable virtual update. The objective averages the scores and the penalty over the samples, and the gradient is back-propagated to ϕ through the virtual update, which requires retaining the second-order computation graph. The policy-gradient variant treats the action as a sample from a Gaussian policy with standard continuous reparameterization (Williams, 1992; Schulman et al., 2015; Kingma & Welling, 2014; Jang et al., 2017; Maddison et al., 2017) and estimates the reward with a leave-one-out baseline (Schulman et al., 2016; 2017):

$$r_k = -\big(s_k + 0.01\|a_k\|^2\big), \qquad b_k = \frac{\sum_j r_j - r_k}{K-1}, \tag{13}$$

$$\mathcal{L}_{\mathrm{PG}} = -\frac{1}{K}\sum_k (r_k - b_k)\log \pi\big(a_k \mid \mu_\phi\big). \tag{14}$$

Here the virtual update does not need to be differentiable, and the gradient depends only on the forward score. The two variants share the same mixer structure and differ only in how the gradient is obtained, so the choice can be made according to the size of the predictor and the memory budget. The action space also has a 1-dimensional form that learns only the mixing coefficient; we use the 10-dimensional form by default and reserve the 1-dimensional form for examining the contribution of the center adjustment alone.

4.4 Training Procedure

Training has a warm-up stage and a joint stage. The warm-up stage optimizes only the predictor, so that the subsequent prediction feedback has reference value; afterwards the predictor continues to train while the mixer is updated at a fixed interval according to the response score of the predictor on observed perturbations, which also controls the outer-loop cost. The main hyperparameters are given in Appendix B.5.

5 Experiments

5.1 Datasets

We use two single-cell perturbation datasets covering genetic and chemical interventions. The gene data come from the Replogle–Nadig study (643,413 source cells, four cell lines, and 6,642 measured genes); after filtering perturbation populations by target-gene knockdown efficiency and ranking cell lines and perturbations by control-centered effect, we retain three cell lines, HepG2, Jurkat, and RPE1, with 150 perturbation conditions each. The chemical data come from the Tahoe-100M compound atlas (about 100.6 million source cells, fifty cell lines, 380 compounds, and 1,138 compound–dose conditions); for each cell line and compound we take the dose with the largest mean absolute

Table 1: Gene perturbation task: each baseline with its PertMix variant (macro average ± standard deviation over three cell lines; best among the variants listed here in bold). Arrows indicate the optimization direction. Table 15 reports both PertMix variants of every method.

Model	Expr MSE ↓	LFC NMAE ↓	Δ Pearson ↑	Spearman ↑	Top50 ↑	Jaccard ↑
BioLORD	$0.0102_{\pm 0.0019}$	$0.761_{\pm 0.082}$	$0.778_{\pm 0.084}$	$0.655_{\pm 0.098}$	$0.580_{\pm 0.110}$	$0.266_{\pm 0.062}$
+ PertMix (PG)	$0.0081_{\pm 0.0015}$	$0.700_{\pm 0.110}$	$\mathbf{0.872}_{\pm 0.086}$	$\mathbf{0.740}_{\pm 0.100}$	$\mathbf{0.660}_{\pm 0.110}$	$0.298_{\pm 0.063}$
scLAMBDA	$0.159_{\pm 0.019}$	$0.850_{\pm 0.100}$	$0.535_{\pm 0.092}$	$0.450_{\pm 0.100}$	$0.145_{\pm 0.063}$	$0.266_{\pm 0.062}$
+ PertMix (Grad)	$0.0075_{\pm 0.0019}$	$0.777_{\pm 0.081}$	$0.811_{\pm 0.075}$	$0.674_{\pm 0.095}$	$0.631_{\pm 0.095}$	$0.301_{\pm 0.062}$
CellFlow	$0.0093_{\pm 0.0019}$	$0.865_{\pm 0.099}$	$0.473_{\pm 0.052}$	$0.394_{\pm 0.060}$	$0.362_{\pm 0.066}$	$0.197_{\pm 0.048}$
+ PertMix (Grad)	$0.0083_{\pm 0.0018}$	$0.778_{\pm 0.094}$	$0.605_{\pm 0.065}$	$0.505_{\pm 0.074}$	$0.464_{\pm 0.085}$	$0.223_{\pm 0.043}$
STATE	$0.0148_{\pm 0.0040}$	$0.719_{\pm 0.086}$	$0.662_{\pm 0.095}$	$0.519_{\pm 0.097}$	$0.510_{\pm 0.100}$	$0.331_{\pm 0.087}$
+ PertMix (Grad)	$\mathbf{0.0046}_{\pm 0.0016}$	$\mathbf{0.647}_{\pm 0.087}$	$0.780_{\pm 0.093}$	$0.630_{\pm 0.120}$	$0.610_{\pm 0.110}$	$0.364_{\pm 0.075}$
PRNet	$0.041_{\pm 0.016}$	$0.930_{\pm 0.110}$	$0.570_{\pm 0.470}$	$0.570_{\pm 0.160}$	$0.265_{\pm 0.095}$	$0.350_{\pm 0.380}$
+ PertMix (Grad)	$0.0168_{\pm 0.0031}$	$0.839_{\pm 0.098}$	$0.670_{\pm 0.520}$	$0.625_{\pm 0.048}$	$0.320_{\pm 0.110}$	$\mathbf{0.380}_{\pm 0.330}$

shift relative to the matched control as the representative treatment condition, and we then retain the cell lines CVCL_1531, CVCL_1571, and CVCL_1577. All cells are CP10K-normalized and $\log(1 + \cdot)$-transformed, and the 2,000 highly variable genes are selected from training expression only. The gene task contains 55,962 cells and the chemical task 18,875, each with 450 cell line–perturbation instances; after de-duplication this gives 315 perturbation identities for the gene task and 255 compound identities for the chemical task. Every cell line is split into 80 training, 20 validation, and 50 test conditions with a fixed seed of 20250920. Data sources, the filtering pipeline, and all statistics are given in Appendix A.

5.2 Baselines

We compare representative models from the main families of perturbation response prediction. BioLORD and scLAMBDA are latent-variable and additive-decomposition methods; CellFlow and STATE are distribution-transport and generative methods; PRNet is a structural-prior method that models multi-target compound conditions through the compound identity. All models share the same fixed split, the same 2,000-gene expression space, and the same frozen perturbation representations—3,072-dimensional gene embeddings, 768-dimensional compound representations, and the control population as the cell background anchor—so the comparison isolates how responses are modeled rather than differences in representation or data. The biological assumptions, condition representations, and decoding schemes of the individual models are given in Appendix B.

5.3 Training Details

All models are trained and evaluated on the same split and in the same expression space. Control cells take part in fitting as the cell background reference, validation perturbations are used for model selection and feedback, and test perturbations are used only for the final report. The six metrics are computed by a single evaluation framework with settings that are identical across models and runs, including the differential expression test and the multiple-testing correction; the exact settings are given in Appendix D.

Tables 1 and 2 compare each baseline with its PertMix variant on the two tasks, and Appendix E reports the complete results for both variants of every method. Each value is the macro average and standard deviation over three cell lines: a metric is first computed on each cell line–perturbation pair, then averaged over perturbations within a cell line with equal weights, and finally macro-averaged over cell lines, so that each cell line contributes equally to the overall result. Within each table the best value of a column is boldfaced according to the optimization direction of the metric, and PertMix rows are shaded in gray.

5.4 Gene Perturbation Task

Table 1 reports the gene task. Among the baselines, BioLORD gives the lowest expression MSE together with the highest Δ Pearson, Spearman, and Top50, while STATE gives the lowest LFC NMAE and the highest Jaccard. PertMix improves all six metrics of each baseline it is applied to,

Table 2: Chemical perturbation task: each baseline with its PertMix variant (macro average ± standard deviation over three cell lines; best among the variants listed here in bold). Arrows indicate the optimization direction. Columns follow Table 1, and Table 16 reports both PertMix variants of every method.

Model	Expr MSE ↓	LFC NMAE ↓	Δ Pearson ↑	Spearman ↑	Top50 ↑	Jaccard ↑
BioLORD	$0.0039_{\pm 0.0014}$	$0.750_{\pm 0.160}$	$0.320_{\pm 0.220}$	$0.288_{\pm 0.023}$	$0.320_{\pm 0.130}$	$0.0138_{\pm 0.0092}$
+ PertMix (Grad)	$\mathbf{0.0035}_{\pm 0.0016}$	$\mathbf{0.650}_{\pm 0.150}$	$0.340_{\pm 0.220}$	$0.311_{\pm 0.033}$	$\mathbf{0.380}_{\pm 0.130}$	$0.0169_{\pm 0.0094}$
scLAMBDA	$0.035_{\pm 0.019}$	$0.888_{\pm 0.083}$	$0.170_{\pm 0.130}$	$0.158_{\pm 0.017}$	$0.205_{\pm 0.035}$	$0.015_{\pm 0.010}$
+ PertMix (Grad)	$0.0043_{\pm 0.0017}$	$0.782_{\pm 0.091}$	$0.250_{\pm 0.160}$	$0.198_{\pm 0.023}$	$0.250_{\pm 0.140}$	$0.019_{\pm 0.010}$
CellFlow	$0.0195_{\pm 0.0091}$	$0.967_{\pm 0.088}$	$0.160_{\pm 0.120}$	$0.175_{\pm 0.018}$	$0.211_{\pm 0.067}$	$0.0141_{\pm 0.0099}$
+ PertMix (PG)	$0.0143_{\pm 0.0092}$	$0.851_{\pm 0.077}$	$0.240_{\pm 0.150}$	$0.211_{\pm 0.022}$	$0.243_{\pm 0.060}$	$0.017_{\pm 0.010}$
STATE	$0.030_{\pm 0.015}$	$2.880_{\pm 0.690}$	$0.210_{\pm 0.180}$	$0.260_{\pm 0.024}$	$0.253_{\pm 0.095}$	$0.018_{\pm 0.012}$
+ PertMix (PG)	$0.00430_{\pm 0.00090}$	$1.800_{\pm 0.170}$	$0.310_{\pm 0.180}$	$0.280_{\pm 0.041}$	$0.308_{\pm 0.095}$	$0.022_{\pm 0.010}$
PRNet	$0.0159_{\pm 0.0061}$	$1.020_{\pm 0.100}$	$0.240_{\pm 0.200}$	$0.294_{\pm 0.083}$	$0.290_{\pm 0.110}$	$0.020_{\pm 0.021}$
+ PertMix (PG)	$0.0068_{\pm 0.0010}$	$0.943_{\pm 0.100}$	$\mathbf{0.360}_{\pm 0.210}$	$\mathbf{0.330}_{\pm 0.025}$	$0.320_{\pm 0.130}$	$\mathbf{0.025}_{\pm 0.019}$

Table 3: Comparison of Native, Grad, and the simulated Ceof method on BioLORD.

Method	Expr MSE ↓	Top50 ↑	Jaccard ↑
Native	0.0102 ± 0.0019	0.580 ± 0.110	0.266 ± 0.062
Grad	$\mathbf{0.0080 \pm 0.0014}$	$\mathbf{0.630 \pm 0.110}$	$\mathbf{0.302 \pm 0.063}$
Ceof	0.0091 ± 0.0017	0.605 ± 0.110	0.284 ± 0.063

and the largest absolute changes appear on the baselines with the weakest native predictions: for PRNet the expression MSE drops from 0.041 to 0.0168 and the LFC NMAE from 0.930 to 0.839, and for STATE the expression MSE drops from 0.0148 to 0.0046. The best value of every column is obtained by a PertMix variant. The policy-gradient and bilevel-gradient variants differ by less than one standard deviation on all six metrics, and Appendix E reports both.

5.5 Chemical Perturbation Task

Table 2 reports the chemical task. The native baselines are less consistent here than on the gene task: BioLORD gives the lowest LFC NMAE (0.750) and the highest Top50 (0.320), PRNet the highest Spearman (0.294) and Jaccard (0.020), and STATE the largest LFC NMAE (2.880). PertMix improves all six metrics of every baseline. For STATE the expression MSE drops from 0.030 to 0.0043 and the LFC NMAE from 2.880 to 1.800, and for scLAMBDA the expression MSE drops from 0.035 to 0.0043. As on the gene task, the best value of every column is obtained by a PertMix variant.

5.6 Ablation Study

We choose BioLORD as a representative perturbation prediction model and compare a coefficient-only mixer with the full PertMix under the same backbone, training budget, and gradient-based optimization. The coefficient-only model only adapts the effective interpolation position, whereas PertMix additionally allows constrained distribution-level correction.

As shown in Table 3, PertMix consistently improves the three representative metrics over coefficient-only mixing, indicating that distribution-level correction provides additional benefit beyond adapting the interpolation position alone.

6 Conclusion

We studied the prediction of single-cell transcriptional responses to unseen perturbations. We characterized the generalization difficulty through the local geometry of the condition-to-state map as directional-derivative mismatch and curvature anomaly, and found that both grow as observation coverage decreases. This view motivates PertMix, whose mixing rule is learnable: under an endpoint-consistency constraint the mixer adjusts the center and the shape of the synthesized target distribution, and the prediction feedback of the predictor on observed perturbations updates it. Bilevel-gradient and policy-gradient variants cover differentiable and second-order-free predictors. Under a fixed split and six metrics, we compare representative baselines on both tasks, and the results indicate that consistent distribution-level supervision along condition paths is a viable route to

unseen-perturbation generalization. The method applies to paired endpoint perturbations; extending the mixing supervision to combined perturbations and larger atlases remains open.

AI USE STATEMENT

Generative AI tools were used in the preparation of this paper for two purposes only: (i) assistance with writing and language polishing, and (ii) literature retrieval and discovery of related work. Every AI-assisted output was checked item by item by the authors: any reference surfaced by AI was verified against the original publication for existence and relevance, and the polishing did not alter any technical claim or conclusion. The authors take full responsibility for all content of this work.

ETHICS STATEMENT

All experimental data used in this work come from publicly available datasets, whose provenance and processing pipeline are described in the appendix. The authors are not aware of any potential risks in this work concerning bias and fairness, privacy and security, legal compliance, conflicts of interest, or research integrity.

REPRODUCIBILITY STATEMENT

All experimental details are fully specified in the main text and the appendix, covering dataset construction and preprocessing, model architecture and key implementation aspects, training hyperparameters, and the evaluation protocol.

REFERENCES

Britt Adamson, Thomas M. Norman, Marco Jost, et al. A multiplexed single-cell CRISPR screening platform enables systematic dissection of the unfolded protein response. *Cell*, 2016. doi: 10.1016/j.cell.2016.11.048.

Abhinav Adduri, Pavan Gautam, Beatrice Bevilacqua, Mohammadreza Naghipourfar, et al. Predicting cellular responses to perturbation across diverse contexts with State. *Cell*, 2026. doi: 10.1016/j.cell.2026.07.052.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B*, 1995. doi: 10.1111/j.2517-6161.1995.tb02031.x.

Michael Bereket and Theofanis Karaletsos. Modelling cellular perturbations with the sparse additive mechanism shift variational autoencoder. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Charlotte Bunne, Stefan G. Stark, Gabriele Gut, et al. Learning single-cell perturbation responses using neural optimal transport. *Nature Methods*, 2023. doi: 10.1038/s41592-023-01969-x.

Yiqun Chen and James Zou. GenePT: A simple but effective foundation model for genes and cells built from ChatGPT. *bioRxiv*, 2023. doi: 10.1101/2023.10.16.562533.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2013.

Atray Dixit, Oren Parnas, Bi Li, et al. Perturb-Seq: Dissecting molecular circuits with scalable single-cell RNA profiling of pooled genetic screens. *Cell*, 2016. doi: 10.1016/j.cell.2016.11.038.

Driessen et al. Conditional Monge gap enables generalizable single-cell perturbation modelling. *Nature Machine Intelligence*, 2026. doi: 10.1038/s42256-026-01242-8.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, 2017.

Chris J. Frangieh, Johannes C. Melms, Pratiksha I. Thakore, et al. Multimodal pooled Perturb-CITE-seq screens in patient models define mechanisms of cancer immune evasion. *Nature Genetics*, 2021. doi: 10.1038/s41588-021-00779-1.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations (ICLR)*, 2017.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

Dominik Klein, Jonas Simon Fleck, Daniel Bobrovskiy, et al. CellFlow enables generative single-cell phenotype modeling with flow matching. *bioRxiv*, 2025. doi: 10.1101/2025.04.11.648220.

Li et al. A systematic comparison of single-cell perturbation response prediction models. *Science Advances*, 2026. doi: 10.1126/sciadv.aed3414.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

Romain Lopez, Jeffrey Regier, Michael B. Cole, Michael I. Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 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*, 2019. doi: 10.1038/s41592-019-0494-8.

Mohammad Lotfollahi, Anna Klimovskaia Susmelj, Carlo De Donno, et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular Systems Biology*, 2023. doi: 10.15252/msb.202211517.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *International Conference on Learning Representations (ICLR)*, 2017.

Ajay Nadig, Joseph M. Replogle, Angela N. Pogson, et al. Transcriptome-wide analysis of differential expression in perturbation atlases. *Nature Genetics*, 2025. doi: 10.1038/s41588-025-02169-3.

Thomas M. Norman, Max A. Horlbeck, Joseph M. Replogle, et al. Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. *Science*, 2019. doi: 10.1126/science.aax4438.

Stefan Peidli, Tessa D. Green, Chi Shen, et al. scPerturb: Harmonized single-cell perturbation data. *Nature Methods*, 2024. doi: 10.1038/s41592-023-02144-y.

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

X. Qi, L. Zhao, C. Tian, et al. Predicting transcriptional responses to novel chemical perturbations using deep generative model for drug discovery. *Nature Communications*, 2024. doi: 10.1038/s41467-024-53457-1.

Aravind Rajeswaran, Chelsea Finn, Sham Kakade, and Sergey Levine. Meta-learning with implicit gradients. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Joseph M. Replogle, Reuben A. Saunders, Angela N. Pogson, et al. Mapping information-rich genotype-phenotype landscapes with genome-scale Perturb-seq. *Cell*, 2022. doi: 10.1016/j.cell.2022.05.013.

Yusuf Roohani, Kexin Huang, and Jure Leskovec. Predicting transcriptional outcomes of novel multigene perturbations with GEARS. *Nature Biotechnology*, 2023. doi: 10.1038/s41587-023-01905-6.

Yusuf Roohani, Tony Hua, Yuhui Tung, et al. Virtual cell challenge: Toward a Turing test for the virtual cell. *Cell*, 2025. doi: 10.1016/j.cell.2025.06.008.

John Schulman, Nicolas Heess, Theophane Weber, and Pieter Abbeel. Gradient estimation using stochastic computation graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

John Schulman, Philipp Moritz, Sergey Levine, et al. High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations (ICLR)*, 2016.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Sanjay R. Srivatsan, José L. McFaline-Figueroa, Vinay Ramani, et al. Massively multiplex chemical transcriptomics at single-cell resolution. *Science*, 2020. doi: 10.1126/science.aax6234.

Aravind Subramanian, Rajiv Narayan, Steven M. Corsello, et al. A next generation connectivity map: L1000 platform and the first 1,000,000 profiles. *Cell*, 2017. doi: 10.1016/j.cell.2017.10.049.

Vikas Verma, Alex Lamb, Christopher Beckham, et al. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning (ICML)*, 2019.

Wei et al. Benchmarking algorithms for generalizable single-cell perturbation response prediction. *Nature Methods*, 2025. doi: 10.1038/s41592-025-02980-0.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 1992. doi: 10.1007/BF00992696.

X. Yuan, X. Liu, Y. S. Zhang, Z. Zhang, H. Guo, and J. Tang. PerturbDiff: Functional diffusion for single-cell perturbation modeling. *arXiv preprint arXiv:2602.19685*, 2026.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2018.

X. Zhang, A. A. Ubas, V. Svensson, et al. Tahoe-100m: Mapping drug-induced molecular phenotypes at single-cell resolution. *Cell*, 2026. doi: 10.1016/j.cell.2026.08.035.

Yifei Zou, Vikas Verma, Sarthak Mittal, et al. MixupE: Understanding and improving mixup from directional derivative perspective. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2023.

CONTENTS OF THE APPENDIX

A Datasets **14**

A.1 Data Sources . . . 14

A.1.1 Gene Perturbation Data . . . 14

A.1.2 Chemical Perturbation Data . . . 14

A.1.3 Compound-Level Perturbation Representation . . . 15

A.1.4 Unified Gene and Chemical Perturbation Task . . . 15

A.2 Data Processing . . . 15

A.2.1 Raw Count Preprocessing . . . 15

A.2.2 Highly Variable Gene Selection . . . 16

A.2.3 Task Splits . . . 16

A.3 Data Statistics . . . 16

B Model details **17**

B.1 Comparison Models . . . 17

B.1.1 STATE . . . 17

B.1.2 CellFlow . . . 17

B.1.3 PRNet . . . 17

B.1.4 scLAMBDA . . . 17

B.1.5 BioLORD . . . 18

B.2 Unified Experimental Formulation . . . 18

B.3 Construction of Perturbation Condition Representations . . . 19

B.4 Training Hyperparameters of the Comparison Models . . . 19

B.5 PertMix Training Hyperparameters . . . 19

B.6 The Mixer Network . . . 20

B.7 Observation-Coverage Experiment Configuration . . . 20

C Response Geometry Diagnostics **21**

C.1 Model Analysis . . . 21

C.1.1 Response Geometry . . . 21

C.1.2 Learned Distributional Correction . . . 21

D Metric **22**

D.1 Expression Magnitude and Correlation . . . 22

D.1.1 Expression MSE (Expr MSE) . . . 22

D.1.2 Δ Pearson Correlation . . . 22

D.1.3 Spearman Rank Correlation . . . 22

D.2 Differential Gene Expression (DGE) Metrics . . . 23

D.2.1 DGE Fold-Change Accuracy (LFC NMAE) . . . 23

D.2.2 Top-50 DE Recall (Top50) 23
D.2.3 DGE Significant-Set Agreement (Jaccard) 23
D.3 Metric Aggregation 23
D.4 Statistical Testing and Evaluation Settings 23

E Complete Results **24**

A DATASETS

This appendix describes the dataset definitions, the preprocessing and splitting procedure, the core principles of the comparison models, the training configurations, and the six evaluation metrics used for unseen-perturbation single-cell transcriptional response prediction. Gene and chemical perturbations differ in the semantics of the condition, but both are formulated as a mapping from a cell background and a perturbation condition to the post-perturbation cell state. All statistics come from the current experimental data and from a unified evaluation setting.

The appendix compares all methods in the same 2,000-gene expression space and with the same perturbation-level split. The training set estimates model parameters, and the highly variable genes are also determined from training expression only; the validation set is used for model selection; and the test set reports the responses of perturbations that did not participate in fitting.

A.1 DATA SOURCES

A.1.1 GENE PERTURBATION DATA

The gene perturbation experiments come from the Replogle–Nadig single-cell gene perturbation study. The study knocks down the expression of target genes with pooled CRISPR interventions and measures the resulting transcriptional states at single-cell resolution. A perturbation condition is defined by the target gene identity, a cell line represents the background state at the time of intervention, and non-targeting constructs form the control population used to estimate the baseline expression without a specific gene intervention (Replogle et al., 2022; Nadig et al., 2025). The source data contain 643,413 cells, four cell lines, and 6,642 measured genes.

Target-gene suppression efficiency is used to filter intervention responses. We assess on-target knockdown efficiency from the residual expression of the target gene in perturbed cells and retain perturbation populations with stable cell support. This step improves the correspondence between the perturbation label and the observed response and reduces the influence of weak interventions and small populations on training. We then rank cell lines and perturbations by control-centered effect, and finally include the three cell lines HepG2, Jurkat, and RPE1 with 150 perturbation conditions each.

A.1.2 CHEMICAL PERTURBATION DATA

The chemical perturbation experiments come from the Tahoe ARC CTC data. The study applies compound treatments in multiple cell lines and records the post-treatment cell states with single-cell transcriptomics. The cell line describes the background of the drug action, the compound defines the intervention identity, the dose describes the exposure strength, and DMSO or an equivalent vehicle condition forms the matched control (Zhang et al., 2026). The source data cover about 100.6 million cells, fifty cell lines, 380 compounds, and 1,138 compound–dose conditions.

The multi-dose design gives different response strengths for the same compound. For each cell line and compound we compute the mean absolute transcriptional shift of each dose relative to the matched control and select the dose with the largest shift as the representative treatment condition. The three doses 0.05, 0.5, and 5.0 micromolar are finally selected in 142, 157, and 151 cell line–compound pairs, which shows that the representative response is not fixed at the highest dose but depends on the compound and the cell background together.

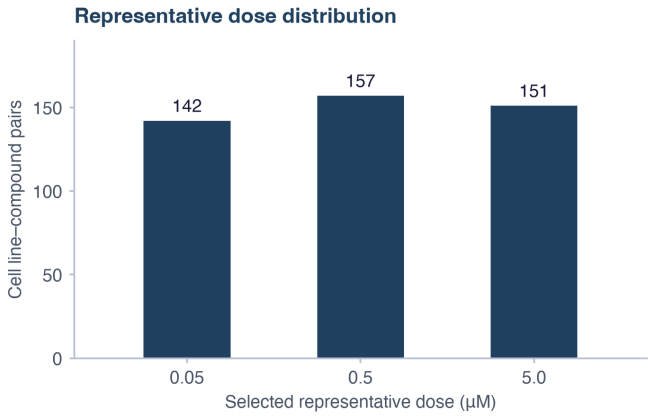


Figure 4: Distribution of the representative dose over 450 cell line–compound pairs. The doses 0.05, 0.5, and 5.0 micromolar are selected by 142, 157, and 151 pairs respectively; the three shares are close, which shows that the representative response depends on the compound and the cell background together rather than being fixed at the highest dose.

Aspect	Gene perturbation	Chemical perturbation
Condition definition	Target gene or gene perturbation identity	Compound identity
Biological mechanism	Intervention at the target gene	Multi-target and pathway-level response
Dose	Not applicable	Representative response dose selected
Unit of evaluation	Cell line–gene perturbation pair	Cell line–compound pair
Prediction target	Post-perturbation single-cell expression distribution	Post-perturbation single-cell expression distribution

Table 4: Unified task definition for gene and chemical perturbations.

A.1.3 Compound-Level Perturbation Representation

A compound usually affects several direct targets, downstream signaling pathways, and feedback loops at the same time. The observed transcriptional response is further influenced by cell-line-specific receptor expression, metabolic state, and pathway availability. Chemical perturbations therefore use the compound identity as the condition and are not reduced to a single target gene, whereas gene perturbations use the intervened target gene as the condition. The two kinds of conditions carry different biological semantics but correspond to the same comparable task: predicting the distribution of the post-perturbation cell state for a given cell background.

A.1.4 Unified Gene and Chemical Perturbation Task

The input of the unified task consists of a cell background and a perturbation condition, and the output is the post-perturbation single-cell expression distribution. Every prediction uses the control population of the same cell line as a reference and is evaluated in the fixed 2,000-gene expression space. The model must generalize to perturbation conditions that do not appear during training while preserving the background structure of the corresponding cell line.

A.2 Data Processing

A.2.1 Raw Count Preprocessing

For cell line c, perturbation p, and gene g, the perturbation effect is defined as the difference between the mean expression of the perturbed population and the mean expression of the control population of the same cell line. All cells are first CP10K-normalized by total count, so that the total expression of each cell lies on a common scale, and then transformed by $\log(1 + \cdot)$ to compress the dynamic

Statistic	Gene perturbation	Chemical perturbation
Source cells	643,413	100,648,790
Source cell lines	4	50
Final cells	55,962	18,875
Final control cells	28,474	4,935
Final cell lines	3	3
Cell line–perturbation instances	450	450
De-duplicated perturbation identities	315	255
Expression dimension	2,000 HVGs	2,000 HVGs
Mean HVG effect strength	0.1505	0.0671

Table 5: Final dataset scale.

range of highly expressed genes. Model training and all evaluation metrics are computed in this normalized log expression space.

A.2.2 Highly Variable Gene Selection

Single-cell expression matrices are high-dimensional and sparse. We select the 2,000 genes with the largest variance from training expression in order to retain the dimensions that are most informative for the perturbation response and the cell-line background differences; model outputs and all evaluations are built on this gene set.

Feature selection is performed on training expression only, and validation and test expression do not take part in determining the gene panel. This reduces the computational cost, decreases the noise contributed by low-information genes, and makes different models comparable in the same output space.

A.2.3 Task Splits

The two datasets are ranked by cell line separately. We compute the mean effect strength of each cell line over candidate perturbations, select the three cell lines with the largest mean response, and keep the 150 perturbations with the largest effect strength in each cell line. The perturbation is then used as the unit of splitting: 80 training perturbations, 20 validation perturbations, and 50 test perturbations are randomly assigned within each cell line with a fixed random seed of 20250920. The control population takes part in model fitting as the cell background reference.

This design measures unseen-perturbation generalization within the same cell line. The training set provides the response mapping and the background distribution, the validation set is used to select hyperparameters and the stopping point, and the test set is used only for the final evaluation. Because all cells of a perturbation belong to the same split, the model cannot obtain the test response through other cells of the same perturbation.

A.3 Data Statistics

After preprocessing, the gene perturbation task contains 55,962 cells and the chemical perturbation task contains 18,875 cells. Each dataset contains three cell lines and 450 cell line–perturbation instances. Perturbation identities overlap across cell lines, so after de-duplicating by label the gene task contains 315 perturbation identities and the chemical task contains 255 compound identities.

The mean effect strength of gene perturbations is 0.1505, higher than the value of 0.0671 for chemical perturbations. This difference reflects the experimental design, the cell backgrounds, and the response sparsity of the two datasets, and it should not be read directly as evidence that one class of intervention has a stronger general biological effect. The third quartile of the chemical task is 0.0834, but the maximum reaches 0.1939, which indicates that a few compounds produce a clearly stronger transcriptional shift in specific cell lines.

Dataset	Cell line	Cells	Perturbation instances
Gene	HepG2	13,809	150
Gene	Jurkat	22,245	150
Gene	RPE1	19,908	150
Chemical	CVCL_1531	2,781	150
Chemical	CVCL_1571	1,900	150
Chemical	CVCL_1577	14,194	150

Table 6: Distribution of cell lines and perturbation instances.

Dataset	Train cells	Val. cells	Test cells	Perturbations per cell line (train/val/test)
Gene	42,623	3,849	9,490	80 / 20 / 50
Chemical	12,337	1,851	4,687	80 / 20 / 50

Table 7: Task split sizes.

B Model details

B.1 Comparison Models

All comparison models receive the cell line control distribution, the perturbation representation, and the cell line background, and output the post-perturbation expression distribution under that condition. Below we describe, for each model and in the same order, the biological assumption, the condition representation, the state representation, the decoding scheme, the training objective, the prediction target, and the applicable perturbation type.

B.1.1 STATE

STATE treats a cell population as a set of cell states and learns the population state transition under a perturbation condition (Adduri et al., 2026). A set-level encoder summarizes the structure of the control population, the perturbation representation is injected into the state transition module, and a cell set matching the target population is generated. The training objective compares the predicted population with the true perturbed population at the distribution level, with an entropy-regularized optimal-transport formulation (Cuturi, 2013), and the model handles gene and chemical conditions uniformly.

B.1.2 CellFlow

CellFlow directly learns a conditional flow from the control distribution to the perturbation distribution (Klein et al., 2025). The model takes control cells as the source distribution and true perturbed cells as the target distribution and trains a conditional velocity field that describes the continuous transport path between them (Lipman et al., 2023). The perturbation representation and the cell line background jointly modulate the velocity field, so the model can generate new cell populations for unseen conditions.

B.1.3 PRNet

PRNet is designed for compound response modeling. The model encodes the chemical condition into a latent perturbation representation, combines it with the control cell state, and generates the expression after compound treatment through a conditional decoder. It carries multi-target information in the compound identity and predicts single-cell responses under new compound conditions, and it is evaluated on both tasks.

B.1.4 scLAMBDA

scLAMBDA decomposes single-cell states with a basal branch and a salient perturbation branch (Qi et al., 2024). The basal branch captures the cell line and shared states, and the salient branch

Dataset	Minimum	First quartile	Median	Third quartile	Maximum
Gene	0.0851	0.1267	0.1517	0.1725	0.2413
Chemical	0.0215	0.0385	0.0717	0.0834	0.1939

Table 8: Quantiles of the perturbation effect strength in the HVG2000 space.

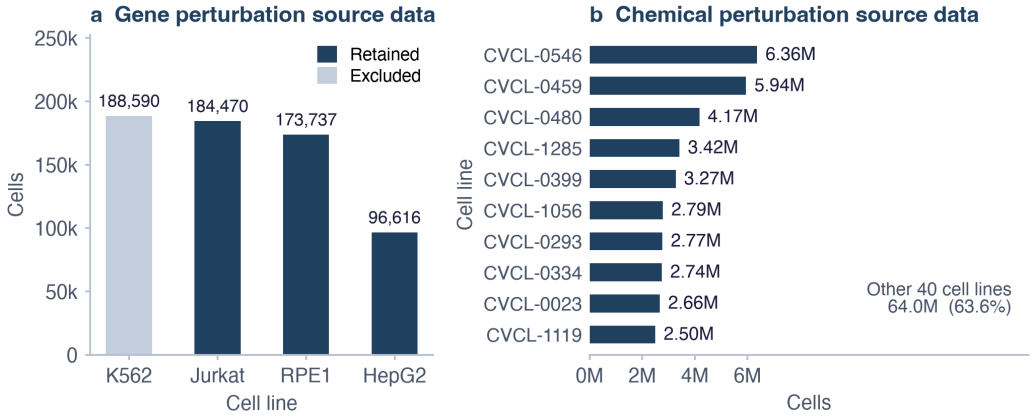


Figure 5: Cell-line composition of the source data. (a) Cell lines in the gene perturbation source data; the darker bars are retained in the task and the lighter bar is excluded. (b) Ten most frequent cell lines in the chemical perturbation source data. The remaining 40 cell lines hold 64.0M cells, that is, 63.6% of the source data, and are not shown individually.

represents the condition-specific response. The decoder combines the two latent components and generates counterfactual cells, so that background differences and perturbation effects are relatively disentangled in the latent space. The structure applies to gene and chemical conditions.

B.1.5 BioLORD

BioLORD decomposes a cell state into latent variables that can be explained by observed attributes and an unexplained basal component (Piran et al., 2024). The perturbation representation enters the model as an observed attribute, and training reduces confounding between the attribute and the basal state through reconstruction and latent-space constraints. At prediction time the perturbation attribute is replaced while the basal latent variables are kept, which produces counterfactual expression under the target condition. The method adapts to gene and compound conditions.

B.2 Unified Experimental Formulation

Here Y_c^{ctrl} denotes the control distribution of cell line c, e_p the condition representation of perturbation p, and c the cell line background; the prediction output $\hat{Y}_{c,p}$ is the post-perturbation single-cell expression distribution. Gene perturbation conditions use a 3,072-dimensional gene representation and chemical perturbation conditions a 768-dimensional compound representation; control cells provide the background anchor of the same cell line. Different models can use these inputs in mean shifts, latent variables, graph structures, flows, or diffusion processes, but all are trained and evaluated in the same output space and on the same data split.

All experiments use the 2,000 training-set highly variable genes and the random seed 20250920. The 80 training perturbations of each cell line fit the model, the 20 validation perturbations are used for hyperparameter selection, and the 50 test perturbations are used for the final report. Each method keeps its original objective function and generative mechanism. Test evaluation uses the cell line–perturbation pair as the basic unit, and gene and chemical data are aggregated separately.

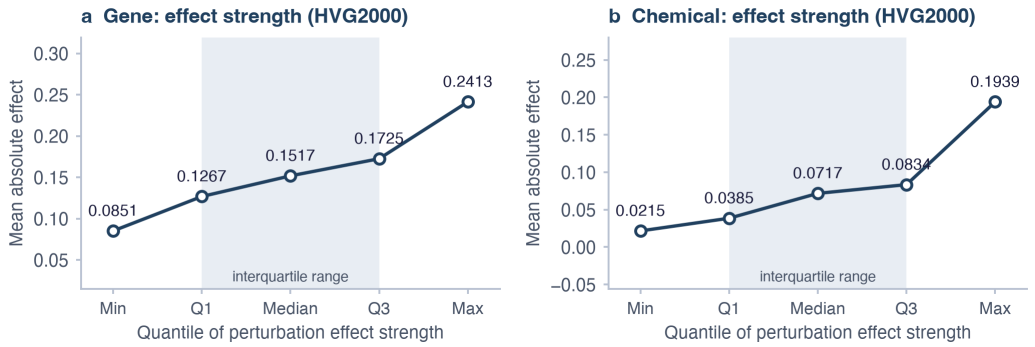


Figure 6: Quantiles of the perturbation effect strength in the HVG2000 space, with the shaded band marking the interquartile range. (a) Gene perturbations. (b) Chemical perturbations.

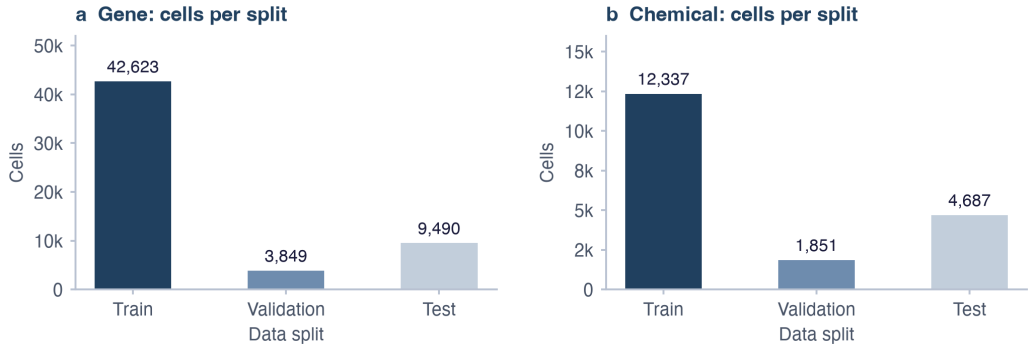


Figure 7: Number of cells in the training, validation, and test splits. (a) Gene perturbation data. (b) Chemical perturbation data.

B.3 Construction of Perturbation Condition Representations

All condition models share the same set of frozen perturbation condition representations, which are not updated during training. Gene perturbation conditions use the 3,072-dimensional gene embeddings released with GenePT v2, and chemical perturbation conditions use the 768-dimensional compound representations of ChemBERTa (Chen & Zou, 2023).

The control condition has no released representation vector, so following the convention of the reference implementation we take the mean of all non-control perturbation representations as the representation of the control condition. Every non-control perturbation in the task has a corresponding condition representation.

B.4 Training Hyperparameters of the Comparison Models

Table 9 summarizes the key training hyperparameters of the comparison models. Parameters that are not listed explicitly follow the defaults of the official implementations of each method.

B.5 PertMix Training Hyperparameters

The predictor and the endpoint supervision of PertMix follow the training configuration of the comparison models. The mixer is updated at a fixed interval after the warm-up stage ends; its network size, action dimension, and penalty coefficient are given in Table 10.

Model	Key hyperparameters	Training budget	Batch size
STATE	State-transition set Transformer; cell set length 128; energy-distance and Sinkhorn objective	20,000 steps	4
CellFlow	Hidden dimension 256; condition dimension 64; optimal-transport flow-matching objective	1,000 iterations	1024
PRNet	Compound condition encoding and response decoding; ChemBERTa condition representation; Adam	Official default	Official default
scLAMBDA	Basal and salient dual branch; latent and hidden dimensions follow the official default	10 epochs	Official default
BioLORD	Latent 32; hidden 512; 3 layers; dropout 0.1; penalty coefficient 1.0; learning rate 6.72×10^{-5}; weight decay 7.90×10^{-9}	50 epochs	300

Table 9: Key training hyperparameters of the comparison models.

Hyperparameter	Value	Role
Warm-up fraction	0.05	Only the predictor is trained in the first 5% of steps, so that the prediction feedback has reference value
Mixer update interval	40	The mixer is updated every 40 steps, which controls the outer-loop cost
Action standard deviation	0.2	Standard deviation of the reparameterized sampling and the policy distribution
Cell set size	128	Cell set size used for each virtual update
Response-base dimension	8	Dimension of the response bases along the gene axis
Action dimension	10	One mixing-coefficient component, eight response-base components, and one power-exponent component
Action penalty coefficient	0.01	Quadratic penalty that suppresses the magnitude of the mixing action
Random seed	20250920	Same as all comparison models

Table 10: Main training hyperparameters of PertMix.

B.6 The Mixer Network

The mixer is a small feed-forward network. Its input is a 65-dimensional feature vector built from the interpolation coefficient λ and the summary statistics of the two endpoint conditions, and its output is the 10-dimensional action a used in Equations equation 6 to equation 8. The network has two 64-dimensional hidden layers with SiLU activations, and its last layer is initialized to zero, so that at the beginning of training the synthetic target equals the linear interpolation and the learned update is a deviation from it. The response bases B in Equation equation 7 are learned jointly with the network weights.

To monitor the mixer we record the effective coefficient β, its offset from λ, the power exponent p, the root-mean-square center displacement, the root-mean-square center error, and the trace of the variation of the synthetic target. These quantities indicate whether the mixer departs from linear interpolation and how the correction is distributed. Table 13 reports the corresponding statistics.

B.7 Observation-Coverage Experiment Configuration

The experiment compares a baseline without geometric constraints and a branch that penalizes the directional derivative and the curvature on STATE, and the two branches differ only in the regularization term. The three coverage levels use nested splits, and the low-coverage settings reuse perturbations that already appear in the high-coverage settings, so that the trend comparison is not affected by split randomness. The configuration is given in Table 11.

Configuration	Value
Dataset	Replogle, gene perturbation task
Observation coverage	25% / 50% / 100% (nested coverage)
Random seed	20250920
Training steps	20,000
Batch size	4
Cell set length	128
Control branch	Baseline without geometric constraints
Motivation branch	Penalty on the directional derivative and the curvature (5 neighbors, gradient weight 0.5, curvature weight 0.5, difference step 0.25)
Output per branch	Predictions, held-out perturbation response geometry, strict prediction contract validation, six-metric evaluation

Table 11: Observation-coverage experiment configuration.

Table 12: Prediction quality and response geometry under different training strategies.

Method	PDS ↑	G_{resp} ↓	C_{resp} ↓
BioLORD	0.7510	0.02507	0.08967
+ Geom. Reg.	0.5227	0.02697	**0.00810**
+ PertMix-Grad	**0.7770**	**0.02191**	0.04707

C Response Geometry Diagnostics

C.1 Model Analysis

We further use BioLORD as a representative model to analyze how PertMix affects the learned response geometry and how its learned mixer departs from direct linear interpolation.

C.1.1 Response Geometry

We compare BioLORD, Direct Geometry Regularization, and PertMix using PDS together with the first-order inconsistency G_{resp} and directional curvature C_{resp}.

As summarized in Table 12 and visualized in Fig. 8(a–b), direct geometry regularization strongly reduces curvature but degrades prediction quality, whereas PertMix improves prediction while simultaneously reducing both geometric diagnostics. This shows that simply enforcing a more linear response map is insufficient; PertMix instead induces a better-behaved geometry while preserving predictive flexibility.

C.1.2 Learned Distributional Correction

We compare PertMix-generated populations with populations obtained by directly interpolating endpoint cells. The learned corrections remain small overall, while larger actions produce larger distributional deviations.

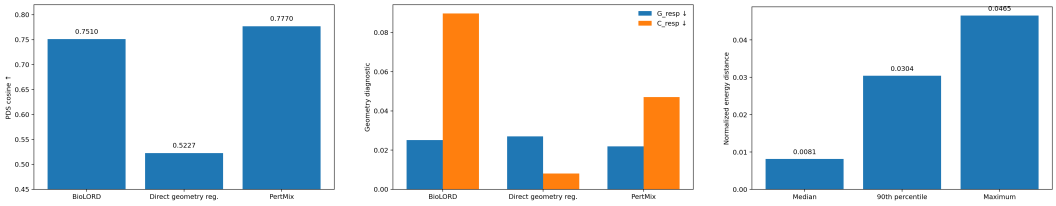
As shown in Table 13 and Fig. 8(c), PertMix learns controlled, action-dependent deviations from direct linear interpolation, with the correction mainly expressed through gene-dependent redistribution of population variation.

To describe the shapes of the response map quantitatively, we probe the map locally in the condition representation space. For each observed condition we take its k nearest neighbors as a partner set and use a pair of neighbors to span a local path and its direction d. At the center e of the path we estimate the directional derivative and the curvature of the map along that direction by central differences:

$$\mathcal{D}_d f(e) \approx \frac{f(e+hd) - f(e-hd)}{2h}, \qquad \mathcal{C}_d f(e) \approx \frac{f(e+hd) - 2f(e) + f(e-hd)}{h^2}, \tag{15}$$

Table 13: Mechanistic statistics of the learned mixer.

Quantity	Value	Pattern
Median energy distance	0.0081	Small deviation
$\mathrm{corr}(\lvert\Delta\beta\rvert, \mathrm{energy})$	0.267	Grows with action
Variance ratio / mean $\lvert$ log-var shift$\rvert$	1.0000 / 0.0177	Gene-dependent variation



(a) Prediction quality. (b) Response geometry. (c) Learned deviation.

Figure 8: Model analysis of PertMix with BioLORD. (a) Prediction quality under different training strategies. (b) First-order response inconsistency and directional curvature. (c) Distributional deviation of PertMix from direct linear interpolation.

where h is the difference step. Comparing the estimates with the response geometry measured on held-out perturbations yields two scalars, the directional-derivative mismatch and the curvature anomaly. Both can be used directly as regularization terms during training.

D METRIC

Six metrics measure the expression magnitude, the effect correlation, and the differential expression response. Every metric is computed in the same 2,000-gene space. Expr MSE and LFC NMAE are error terms, for which a lower value means a better agreement with the true response; the other four are better when higher.

D.1 EXPRESSION MAGNITUDE AND CORRELATION

D.1.1 EXPRESSION MSE (EXPR MSE)

Expr MSE measures the squared error between the predicted and the true perturbation expression profiles, normalized by the magnitude of the true deviation from control. The metric corrects for the mean estimation noise caused by finite cell sampling and limits the amount of noise subtraction available to the predicted distribution. A lower value means a more accurate expression magnitude, and 0 corresponds to an almost error-free population expression.

D.1.2 Δ PEARSON CORRELATION

Δ Pearson computes the Pearson correlation between the predicted and the true mean expression effects over the 2,000 genes. Both sides subtract the true control mean of the same cell line, so only the shape of the expression change caused by the perturbation is evaluated. A value closer to 1 means a stronger linear agreement of the up- and down-regulation patterns.

D.1.3 SPEARMAN RANK CORRELATION

Spearman first computes the log2 fold changes of the predicted and the true responses relative to control and then compares the Spearman correlation of the two gene orderings. A fixed pseudocount of 0.1 stabilizes the ratios of lowly expressed genes. The metric emphasizes the relative order of gene responses and is less sensitive to an overall scale change than the Pearson correlation.

Metric	Direction	Main object of evaluation
Expr MSE	Low	Normalized squared error of the expression profile
LFC NMAE	Low	Normalized absolute error of the log fold changes on significant genes
Δ Pearson	High	Linear agreement of the control-centered expression effect
Spearman	High	Agreement of the gene fold-change ordering
Top50	High	Recovery rate of the strongly responding genes
Jaccard	High	Overlap of the significant differential expression sets

Table 14: The six evaluation metrics and their optimization directions.

D.2 Differential Gene Expression (DGE) Metrics

D.2.1 DGE Fold-Change Accuracy (LFC NMAE)

LFC NMAE computes the normalized mean absolute error between the predicted and the true log2 fold changes on the genes that are identified as significantly differentially expressed in the true data by a Wilcoxon test. The denominator is the sum of the absolute true effects of these genes, which makes the metric comparable across perturbations of different response strength. A lower value means a more accurate differential expression magnitude, and a value of 1 corresponds to predicting no fold change.

D.2.2 Top-50 DE Recall (Top50)

Top50 selects the 50 genes with the strongest mean absolute expression effect in the predicted and the true response respectively and computes the fraction of the true top 50 that is recovered by the predicted set. It does not evaluate direction again; it asks whether the model found the most important response genes. The value ranges from 0 to 1, and higher is better.

D.2.3 DGE Significant-Set Agreement (Jaccard)

Jaccard compares the predicted and the true sets of significantly differentially expressed genes. The intersection is identified by both sides and the differences correspond to misses or extra predictions, so the Jaccard coefficient is symmetric with respect to the two kinds of set error. A value of 1 means that the two sets are identical and a value of 0 means that they do not overlap.

D.3 Metric Aggregation

The six metrics are first computed on each cell line–perturbation pair. The results are then averaged over all test perturbations within a cell line with equal weights, and finally macro-averaged over the three cell lines, so that each cell line contributes equally to the overall result. Gene and chemical perturbations are reported separately, and the chemical results are interpreted as the consistency of compound-level responses.

Expr MSE and LFC NMAE are better when lower, and Δ Pearson, Spearman, Top50, and Jaccard are better when higher. Reporting several metrics jointly prevents a model from optimizing only the population mean or a small number of significant genes, and the evaluation covers the expression magnitude, the effect correlation, and the differential expression response at the same time.

D.4 Statistical Testing and Evaluation Settings

The six metrics are computed by a single evaluation framework with settings that are identical across models and runs, in the same 2,000-gene space and against the same true control reference.

The differential expression test is a two-sided Wilcoxon rank-sum test (equivalent to the Mann–Whitney U test with tie correction). Multiple-testing correction uses the Benjamini–Hochberg procedure to control the false discovery rate within each perturbation independently, with a significance threshold of corrected $p < 0.05$; no additional lower bound is imposed on the effect size, that is,

Table 15: Gene perturbation task: complete results, with the native baseline and both PertMix variants of every method (best in bold). Arrows indicate the optimization direction.

Model	Variant	Expr MSE ↓	LFC NMAE ↓	Δ Pearson ↑	Spearman ↑	Top50 ↑	Jaccard ↑
BioLORD	Native	$0.0102_{\pm 0.0019}$	$0.761_{\pm 0.082}$	$0.778_{\pm 0.084}$	$0.655_{\pm 0.098}$	$0.580_{\pm 0.110}$	$0.266_{\pm 0.062}$
	PertMix (PG)	$0.0081_{\pm 0.0015}$	$0.700_{\pm 0.110}$	**0.872**$_{\pm 0.086}$	**0.740**$_{\pm 0.100}$	**0.660**$_{\pm 0.110}$	$0.298_{\pm 0.063}$
	PertMix (Grad)	$0.0080_{\pm 0.0014}$	$0.670_{\pm 0.100}$	$0.841_{\pm 0.087}$	$0.710_{\pm 0.100}$	$0.630_{\pm 0.110}$	$0.302_{\pm 0.063}$
scLAMBDA	Native	$0.159_{\pm 0.019}$	$0.850_{\pm 0.100}$	$0.535_{\pm 0.092}$	$0.450_{\pm 0.100}$	$0.145_{\pm 0.063}$	$0.266_{\pm 0.062}$
scLAMBDA	PertMix (PG)	$0.0086_{\pm 0.0016}$	$0.742_{\pm 0.079}$	$0.797_{\pm 0.079}$	$0.660_{\pm 0.099}$	$0.600_{\pm 0.110}$	$0.296_{\pm 0.061}$
scLAMBDA	PertMix (Grad)	$0.0075_{\pm 0.0019}$	$0.777_{\pm 0.081}$	$0.811_{\pm 0.075}$	$0.674_{\pm 0.095}$	$0.631_{\pm 0.095}$	$0.301_{\pm 0.062}$
CellFlow	Native	$0.0093_{\pm 0.0019}$	$0.865_{\pm 0.099}$	$0.473_{\pm 0.052}$	$0.394_{\pm 0.060}$	$0.362_{\pm 0.066}$	$0.197_{\pm 0.048}$
	PertMix (PG)	$0.0082_{\pm 0.0017}$	$0.779_{\pm 0.095}$	$0.606_{\pm 0.064}$	$0.504_{\pm 0.075}$	$0.435_{\pm 0.079}$	$0.224_{\pm 0.042}$
	PertMix (Grad)	$0.0083_{\pm 0.0018}$	$0.778_{\pm 0.094}$	$0.605_{\pm 0.065}$	$0.505_{\pm 0.074}$	$0.464_{\pm 0.085}$	$0.223_{\pm 0.043}$
STATE	Native	$0.0148_{\pm 0.0040}$	$0.719_{\pm 0.086}$	$0.662_{\pm 0.095}$	$0.519_{\pm 0.097}$	$0.510_{\pm 0.100}$	$0.331_{\pm 0.087}$
	PertMix (PG)	$0.0047_{\pm 0.0017}$	**0.646**$_{\pm 0.086}$	$0.779_{\pm 0.094}$	$0.640_{\pm 0.110}$	$0.600_{\pm 0.120}$	$0.365_{\pm 0.074}$
	PertMix (Grad)	**0.0046**$_{\pm 0.0016}$	$0.647_{\pm 0.087}$	$0.780_{\pm 0.093}$	$0.630_{\pm 0.120}$	$0.610_{\pm 0.110}$	$0.364_{\pm 0.075}$
PRNet	Native	$0.041_{\pm 0.016}$	$0.930_{\pm 0.110}$	$0.570_{\pm 0.470}$	$0.570_{\pm 0.160}$	$0.265_{\pm 0.095}$	$0.350_{\pm 0.380}$
	PertMix (PG)	$0.0174_{\pm 0.0026}$	$0.840_{\pm 0.099}$	$0.680_{\pm 0.510}$	$0.624_{\pm 0.049}$	$0.290_{\pm 0.120}$	**0.390**$_{\pm 0.320}$
	PertMix (Grad)	$0.0168_{\pm 0.0031}$	$0.839_{\pm 0.098}$	$0.670_{\pm 0.520}$	$0.625_{\pm 0.048}$	$0.320_{\pm 0.110}$	$0.380_{\pm 0.330}$

Table 16: Chemical perturbation task: complete results, with the native baseline and both PertMix variants of every method (best in bold). Arrows indicate the optimization direction.

Model	Variant	Expr MSE ↓	LFC NMAE ↓	Δ Pearson ↑	Spearman ↑	Top50 ↑	Jaccard ↑
BioLORD	Native	$0.0039_{\pm 0.0014}$	$0.750_{\pm 0.160}$	$0.320_{\pm 0.220}$	$0.288_{\pm 0.023}$	$0.320_{\pm 0.130}$	$0.0138_{\pm 0.0092}$
	PertMix (PG)	$0.0036_{\pm 0.0017}$	$0.680_{\pm 0.150}$	**0.370**$_{\pm 0.220}$	$0.323_{\pm 0.031}$	$0.350_{\pm 0.130}$	$0.0170_{\pm 0.0093}$
	PertMix (Grad)	**0.0035**$_{\pm 0.0016}$	**0.650**$_{\pm 0.150}$	$0.340_{\pm 0.220}$	$0.311_{\pm 0.033}$	**0.380**$_{\pm 0.130}$	$0.0169_{\pm 0.0094}$
scLAMBDA	Native	$0.035_{\pm 0.019}$	$0.888_{\pm 0.083}$	$0.170_{\pm 0.130}$	$0.158_{\pm 0.017}$	$0.205_{\pm 0.035}$	$0.015_{\pm 0.010}$
	PertMix (PG)	$0.0040_{\pm 0.0016}$	$0.817_{\pm 0.090}$	$0.220_{\pm 0.150}$	$0.214_{\pm 0.021}$	$0.280_{\pm 0.140}$	$0.016_{\pm 0.010}$
	PertMix (Grad)	$0.0043_{\pm 0.0017}$	$0.782_{\pm 0.091}$	$0.250_{\pm 0.160}$	$0.198_{\pm 0.023}$	$0.250_{\pm 0.140}$	$0.019_{\pm 0.010}$
CellFlow	Native	$0.0195_{\pm 0.0091}$	$0.967_{\pm 0.088}$	$0.160_{\pm 0.120}$	$0.175_{\pm 0.018}$	$0.211_{\pm 0.067}$	$0.0141_{\pm 0.0099}$
	PertMix (PG)	$0.0143_{\pm 0.0092}$	$0.851_{\pm 0.077}$	$0.240_{\pm 0.150}$	$0.211_{\pm 0.022}$	$0.243_{\pm 0.060}$	$0.017_{\pm 0.010}$
	PertMix (Grad)	$0.0142_{\pm 0.0091}$	$0.890_{\pm 0.079}$	$0.210_{\pm 0.160}$	$0.225_{\pm 0.024}$	$0.247_{\pm 0.060}$	$0.016_{\pm 0.011}$
STATE	Native	$0.030_{\pm 0.015}$	$2.880_{\pm 0.690}$	$0.210_{\pm 0.180}$	$0.260_{\pm 0.024}$	$0.253_{\pm 0.095}$	$0.018_{\pm 0.012}$
	PertMix (PG)	$0.00430_{\pm 0.00090}$	$1.800_{\pm 0.170}$	$0.310_{\pm 0.180}$	$0.280_{\pm 0.041}$	$0.308_{\pm 0.095}$	$0.022_{\pm 0.010}$
	PertMix (Grad)	$0.00420_{\pm 0.00060}$	$1.860_{\pm 0.160}$	$0.280_{\pm 0.170}$	$0.291_{\pm 0.040}$	$0.301_{\pm 0.097}$	$0.0188_{\pm 0.0098}$
PRNet	Native	$0.0159_{\pm 0.0061}$	$1.020_{\pm 0.100}$	$0.240_{\pm 0.200}$	$0.294_{\pm 0.083}$	$0.290_{\pm 0.110}$	$0.020_{\pm 0.021}$
	PertMix (PG)	$0.0068_{\pm 0.0010}$	$0.943_{\pm 0.100}$	$0.360_{\pm 0.210}$	**0.330**$_{\pm 0.025}$	$0.320_{\pm 0.130}$	**0.025**$_{\pm 0.019}$
	PertMix (Grad)	$0.0065_{\pm 0.0012}$	$0.902_{\pm 0.098}$	$0.330_{\pm 0.210}$	$0.327_{\pm 0.025}$	$0.350_{\pm 0.130}$	$0.022_{\pm 0.019}$

genes are not filtered by $|\log_2$ fold change$|$ (Benjamini & Hochberg, 1995). Non-finite predicted fold changes are treated as zero, which is equivalent to predicting no change for that gene.

E Complete Results

The two tables below report the native baseline and both PertMix variants for every method and task.