VIRTUAL CELL MODELS MAKE SYSTEMATIC AND TRANSFERABLE PREDICTION ERRORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Single-cell perturbation models predict transcriptional responses to unseen genetic or chemical interventions, but their prediction errors are commonly treated as residual noise. We present the first systematic study of whether these errors recur across perturbations and transfer to unseen interventions. Across diverse predictors and genetic, combinatorial, and drug perturbation datasets, error fingerprints estimated from training perturbations recur on held-out perturbations and anticipate which genes and biological pathways will be misestimated. The fingerprints are stable across training seeds, and much of their model-specific structure reflects persistent gene-wise over- and under-estimation. We also introduce Gap Transferability Correction (GTC), a lightweight post-hoc framework that reuses historical errors to improve unseen-perturbation predictions without retraining the base model. GTC improves differentially expressed gene recovery in all three perturbation settings. Experiments across multiple cell lines show that fingerprints transfer more reliably within a cell line than across cell lines, identifying cellular context as a boundary condition for error transfer. These results establish prediction error as a structured, transferable, and reusable property of virtual-cell models.

1 INTRODUCTION

Single-cell perturbation screens measure how genetic and chemical interventions reshape cellular states (Bunne et al., 2024). Because experimental screens can cover only a small fraction of the possible interventions, computational methods seek to predict transcriptional responses to perturbations that have not been measured (Roohani et al., 2024; Lotfollahi et al., 2023; Bunne et al., 2023; Cui et al., 2024; Piran et al., 2024; Yu et al., 2025; He et al., 2025). Generalization nevertheless remains difficult. Recent evaluations report substantial performance variation and, in some genetic-perturbation settings, no consistent advantage over simple linear baselines (Ahlmann-Eltze et al., 2025; Wei et al., 2026; Li et al., 2026). Even models with strong aggregate performance can fail on individual perturbations, motivating methods that quantify predictive uncertainty or model perturbation effects probabilistically (Cheng et al., 2025; Xing & Yau, 2025). Most existing work has therefore focused on improving predictions or identifying when they are unreliable.

We ask a different question: are prediction errors themselves reproducible across perturbations? If a predictor repeatedly overestimates or underestimates the same genes, its past errors may reveal how it will fail on unseen interventions. We refer to recurring gene-wise error patterns that persist from seen to unseen perturbations as transferable error fingerprints (Figure 1). Analogous model-specific fingerprints have been studied in computer vision (Marra et al., 2019; Yu et al., 2019; 2021), but whether virtual-cell predictors leave transferable fingerprints in their errors has not been examined.

In this paper,we show that virtual-cell perturbation predictors indeed leave transferable error fingerprints. Across diverse models and genetic, combinatorial, and drug perturbation datasets, error patterns observed on training perturbations recur on held-out interventions and anticipate which genes will be misestimated (Figure 1a). Technical factors account for part of this signal. Much of the model-specific structure, however, reflects persistent gene-wise tendencies to overestimate or underestimate particular genes. These patterns remain reproducible across training runs.

To test whether these historical errors can be reused, we introduce **Gap Transferability Correction (GTC)**, a lightweight post-hoc framework that transfers training-set errors to unseen perturbations

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.

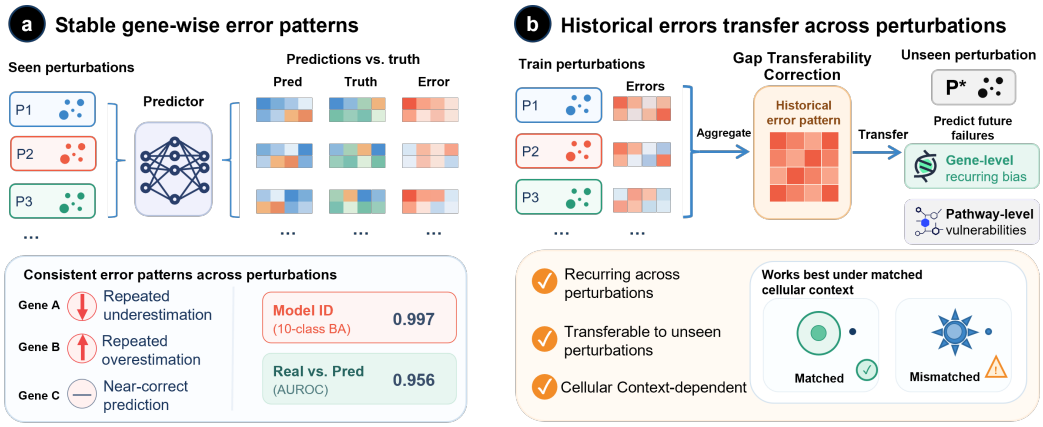

Figure 1: **Transferable error fingerprints in virtual-cell predictors.** (a) Across perturbations, the same predictor repeatedly overestimates or underestimates particular genes, producing stable gene-wise error patterns. (b) Historical errors aggregate into an error fingerprint that transfers to unseen perturbations, where it anticipates gene- and pathway-level failures and can be reused for post-hoc correction through GTC. Transfer is stronger within the same cell line than across cell lines.

without retraining the underlying predictor. GTC improves perturbation-response recovery in genetic, combinatorial, and drug perturbation settings. Experiments across multiple cell lines further show that fingerprints transfer more reliably within the same cell line than across cell lines, revealing an important limitation on transfer between cellular contexts (Figure 1b).

We make three contributions. First, to our knowledge, this is the first systematic study of whether prediction errors in virtual-cell models are reproducible and transferable across perturbations. Second, we characterize the structure of these fingerprints, showing that much of their model-specific signal reflects persistent gene-wise over- and under-estimation and remains stable across training runs. Third, we demonstrate that historical errors can improve unseen-perturbation predictions through simple post-hoc transfer without modifying the base model. Our cross-cellline experiments further establish that error transfer is stronger within the same cell line than across cell lines.

2 Related Work

2.1 Single-Cell Perturbation Prediction and Reliability

Many methods predict unseen perturbations. A broad class of methods predicts responses to unmeasured perturbations using latent-variable, graph-based, transport-based, or pretrained representations (Lotfollahi et al., 2023; Roohani et al., 2024; Bunne et al., 2023; Cui et al., 2024). Recent work further studies generalization across perturbations, experimental conditions, and modalities (Yu et al., 2025; He et al., 2025), as well as variation across drugs, doses, and combinations (Driessen et al., 2026). Complementary approaches model population-level responses or explicitly incorporate cellular context and biological priors (Yuan et al., 2026; Su et al., 2026). Whereas these studies develop response predictors, we analyze the systematic errors that remain after prediction.

A complementary literature evaluates reliability. Uncertainty-aware methods quantify confidence in perturbation responses (Cheng et al., 2025; Xing & Yau, 2025), while benchmarking studies assess generalization across datasets, contexts, metrics, and baseline classes (Ahlmann-Eltze et al., 2025; Wenteler et al., 2025; Wei et al., 2026). Recent analyses identify inflation in conventional metrics due to shared variation (Viñas Torné et al., 2026) and decompose responses into global, cell-line, perturbation, and interaction components (Molina & Zhang, 2026). Our study addresses a distinct question: whether recurring errors anticipate future failures and support post-hoc correction.

2.2 Model Fingerprints and Systematic Prediction Errors

Fingerprints capture recurring model signatures. Model fingerprinting uses persistent output or input–output signatures for source attribution, verification, and generated-content detection (Marra et al., 2019; Yu et al., 2019; Lukas et al., 2021; Wang et al., 2020; Pasquini et al., 2025; Bai et al., 2025; Nasery et al., 2025; Tong et al., 2026). In contrast, our fingerprints are estimated directly from observed prediction errors and evaluated for transfer across perturbations, anticipation of gene- and pathway-level failures, and post-hoc correction.

3 Transferable Prediction Errors in Virtual-Cell Models

We define gene-level error fingerprints and evaluate their reproducibility, transfer, and composition.

3.1 Preliminaries

Let $\Delta^{\text{real}}_{p,c} \in \mathbb{R}^G$ denote the observed transcriptional effect of perturbation p under cellular context c, and let $\Delta^{\text{pred}}_{m,p,c}$ be the corresponding prediction from model m. We define the prediction gap as

$$g_{m,p,c} = \Delta^{\text{real}}_{p,c} - \Delta^{\text{pred}}_{m,p,c}. \tag{1}$$

A positive gap denotes underestimation and a negative gap overestimation (Figure 1a).

We define the **Transferable Error Fingerprint (TEF)** of model m under cellular context c as its mean prediction gap across the training perturbations:

$$F_{m,c} = \frac{1}{|\mathcal{P}_{\text{train}}|} \sum_{p \in \mathcal{P}_{\text{train}}} g_{m,p,c}. \tag{2}$$

Thus, $F_{m,c}$ summarizes recurring gene-wise biases in context c. Estimated from training perturbations only, it is evaluated on unseen perturbations (Figure 1b) and used for correction in Section 4.

3.2 Experimental Settings

Datasets. We consider three complementary perturbation settings: **Norman2019** (Norman et al., 2019) for combinatorial genetic perturbations in K562 cells, **Replogle** (Replogle et al., 2022) for unseen single-gene perturbations in K562 cells, and **SciPlex3** (Srivatsan et al., 2020) for drug perturbations across K562, A549, and MCF7 cells. Norman2019 provides both a component-visible held-out-cell diagnostic and a strict complete-label-unseen combination split (Appendix C.1). SciPlex3 provides shared and matched cell-line settings for studying context dependence (Appendix C.2).

Baselines. Following VCBench (Mao et al., 2026), we evaluate ten diverse perturbation predictors: BiolordStar (Piran et al., 2024), CPA (Lotfollahi et al., 2023), DecoderOnly (Wu et al., 2024), GEARS (Roohani et al., 2024), GenePert (Chen & Zou, 2024), LatentAdditive (Wu et al., 2024), LinearAdditive (Wu et al., 2024), SparseAdditiveVAE (Bereket & Karaletsos, 2023), PRNet (Qi et al., 2024), and scLAMBDA (Wang et al., 2024). Model configurations, optimization settings, and training procedures are provided in Appendix C.4.

Evaluation. Fingerprint analyses use AUROC for real–prediction separability, balanced accuracy for model attribution, and cosine similarity, top-gene recovery, and pathway recovery for transfer. The primary correction endpoint, **DEG@50**, is the overlap between predicted and measured top-50 genes ranked by absolute effect, divided by 50. We also report Spearman correlation, L1 perturbation discrimination score (PDS), energy distance, and RBF maximum mean discrepancy (MMD); definitions are in Appendix H.5. Metrics are aggregated within model–context units and compared using model-level replication; hierarchical bootstrap intervals resample contexts and models. Additional implementation, null-model, and statistical details are in Appendices D–F.

Training details. We average cells into perturbation-level profiles in the common 2,000-gene VCBench space and subtract matched controls (Mao et al., 2026) (Appendix C.3). Fingerprints use training perturbations only; GTC uses internal training splits distinct from base-model validation, and test outcomes are used only for evaluation (Appendices C.4 and D).

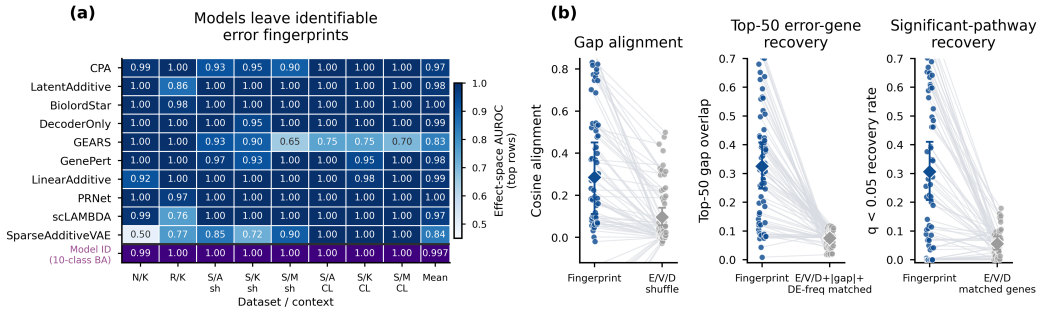

Figure 2: **Virtual-cell predictors exhibit distinctive outputs and transferable error structure.** (a) Across eight dataset–context settings, predicted responses are highly model-specific (purple; 10-class model-identity balanced accuracy, chance 0.10 in every setting) and remain systematically distinguishable from real responses on held-out perturbations (AUROC). (b) Train-derived error fingerprints predict held-out error direction, high-error genes, and significantly affected pathways. E/V/D denote expression mean, variance, and detection rate; matched controls preserve these properties while shuffling gene identity, and the gene-recovery control additionally matches training error magnitude and DEG frequency. All controls use training data only.

3.3 Do Virtual-Cell Errors Recur on Unseen Perturbations?

Held-out predictions reveal their source models and remain distinct from real responses. On held-out perturbations, a 10-class classifier identifies the predictor with mean balanced accuracy 0.997 across eight dataset-context settings (chance 0.10 throughout). These comprise five shared settings and three per-cell-line SciPlex3 settings, each with all ten models. A separate real-versus-predicted classifier achieves mean AUROC 0.956 (Figure 2a). Thus, predictions retain model-specific structure and differ systematically from measured responses.

Training errors predict held-out failures from genes to pathways. We compare each training-derived mean error fingerprint with held-out errors (Figure 2b). The true fingerprint outperforms its matched control for error-direction alignment (0.285 [95% CI 0.112–0.450] vs. 0.096 [0.057–0.141]), top-error-gene recovery (0.325 [0.247–0.408] vs. 0.075 [0.065–0.085]), and pathway recovery (0.306 [0.208–0.410] vs. 0.056 [0.035–0.081]). The corresponding absolute lifts are 0.189, 0.250, and 0.250, respectively. Historical errors therefore reveal where the predictor is likely to fail on unseen perturbations, from genes to biological pathways.

Matched gene properties cannot account for error transfer. Covariate-matched null fingerprints control gene-selection biases (Young et al., 2010; Zhang et al., 2022). For direction alignment, values are shuffled among genes matched on expression mean, variance, and detection rate. Top-error-gene recovery additionally matches typical training error magnitude and top-50 DEG frequency. For pathway recovery, the fingerprint top-100 genes are replaced by expression-, variance-, and detection-matched genes before repeating the same pathway analysis (Appendix E). A stricter five-covariate null preserves the ordering for alignment (0.285 vs. 0.109) and pathway recovery (0.306 vs. 0.103). These controls disrupt gene identity while preserving the matched properties.

3.4 What Constitutes an Error Fingerprint?

Transferability alone does not reveal what an error fingerprint contains. We therefore dissect its underlying structure using the constructions detailed in Appendix G.

Coarse response statistics explain most prediction–data separability. Using six response-level summary statistics, real and predicted effects are separated with mean AUROC 0.99, comparable to 0.96 for the full effect vector (Figure 3a). Fold-safe residualization lowers AUROC by 0.41 to 0.58, only 0.08 above chance. Thus, raw differences largely reflect coarse distributional properties rather than fine-grained gene identity across the evaluated model–context settings.

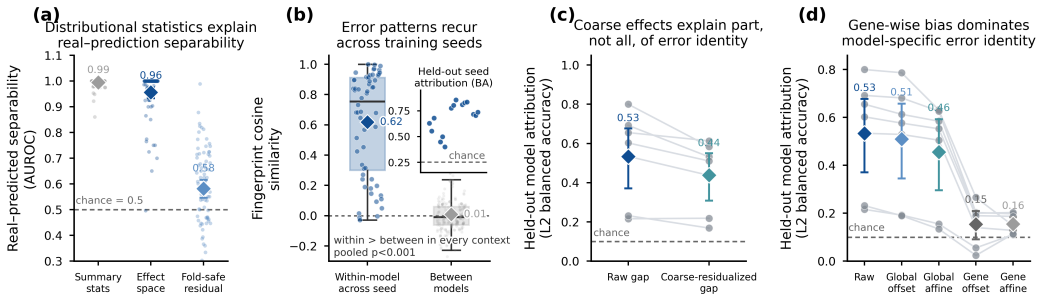

Figure 3: **Dissecting the structure of error fingerprints.** (a) Simple response-level statistics—including expression level, sparsity, variability, and effect magnitude—capture much of the separability between real and predicted responses; fold-safe residualization substantially reduces this signal. (b) Error fingerprints are more similar across independent seeds of the same model than across different models, and retain model identity on held-out seeds (four model families per context, chance 0.25). (c) Removing error components associated with the same response-level statistics reduces, but does not eliminate, model attribution on held-out perturbations. (d) Global calibration has a limited effect on attribution, whereas removing training-derived gene-wise biases reduces held-out model identity close to the 0.10 chance level.

Error fingerprints reproduce across independent training runs. Fingerprints from the same model trained with different seeds are substantially more similar than fingerprints from different models (mean cosine similarity 0.62 vs. 0.01; Figure 3b). Held-out-seed attribution ranges from 0.448 to 0.825 across contexts, consistently above four-class chance (0.25), confirming reproducible model-level structure rather than seed-specific variation across model families and contexts.

Coarse statistics cannot explain model-specific error structure. Under the train-to-held-out attribution protocol on the seven complete-label-disjoint contexts (ten models each; chance 0.10 in every context), raw error gaps identify the underlying model with mean balanced accuracy 0.53. Removing the component of each gene's error associated with these whole-response statistics reduces model attribution to 0.44, but leaves it 0.34 above chance (Figure 3c). Thus, model-specific error structure cannot be explained by coarse response statistics alone.

Persistent gene-wise biases carry most model identity. Raw gaps yield mean balanced accuracy 0.53. Global offset and affine calibration reduce it to 0.51 and 0.46 (drops of 0.02–0.07), whereas both gene-wise variants reduce it to 0.15 (a 0.38 drop; Figure 3d). Thus, persistent gene-specific over- and under-estimation carries most identifiable model-specific error structure.

Coarse statistics distinguish predictions from data; gene biases distinguish models. Overall, coarse distributional differences mainly distinguish predictions from real responses, whereas stable gene-wise biases distinguish the errors of different predictors. These are distinct signatures: prediction–data mismatch is largely global, while model identity is predominantly gene-specific.

4 Gap Transferability Correction

Historical errors contain stable information about future failures. We now ask whether transferring this information improves predictions on unseen perturbations.

4.1 How Can Historical Errors Be Reused?

Estimate the missing error, then add it back. GTC estimates an unseen perturbation's error from historical gaps and applies the post-hoc correction

$$\Delta^{\mathrm{corr}}_{m,p,c} = \Delta^{\mathrm{pred}}_{m,p,c} + \rho\, \widehat{g}_{m,p,c}, \tag{3}$$

where $\widehat{g}_{m,p,c}$ is estimated from training information and ρ controls correction strength. Sensitivity to ρ is evaluated in Appendix A.3 and Table S10.

Reuse the average error. Global GTC reuses persistent error shared across perturbations:

$$\widehat{g}^{\text{global}}_{m,p,c} = F_{m,c}. \tag{4}$$

This directly corrects the recurring gene-wise bias captured by the training fingerprint.

With $\rho = 1$, Global GTC is per-gene intercept calibration. GTC also supports local and compositional error transfer; Appendix H.1 compares it with affine, Ridge, and Lasso calibration.

Reuse errors from similar perturbations. Errors may depend on perturbation structure. For an unseen perturbation p, we transfer gaps from similar training perturbations $\mathcal{N}_K(p)$:

$$\widehat{g}^{\text{local}}_{m,p,c} = \sum_{i \in \mathcal{N}_K(p)} w_{pi} g_{m,i,c}, \qquad \sum_i w_{pi} = 1. \tag{5}$$

Neighborhoods use either perturbation representations or the model's predicted response Δ^{pred}, exploiting external perturbation similarity or model-specific response geometry (Appendix A.1).

Combine average and local errors. We further decompose each training gap into the global fingerprint and a perturbation-specific residual, $r_{m,i,c} = g_{m,i,c} - F_{m,c}$, and estimate

$$\widehat{g}^{\text{G+L}}_{m,p,c} = F_{m,c} + \eta \sum_{i \in \mathcal{N}_K(p)} w_{pi} r_{m,i,c}, \tag{6}$$

where η controls the contribution of local residual structure; sensitivity is evaluated in Appendix A.3. This formulation separates persistent model bias from perturbation-specific deviations.

Add errors from individual components. For combination perturbations whose components are observed during training, we additionally test whether prediction errors are compositional. Given a combination $A + B$, we estimate

$$\widehat{g}^{\text{add}}_{m,A+B,c} = g_{m,A,c} + g_{m,B,c}. \tag{7}$$

Correction follows Eq. 3 and requires both components in training (baselines in Appendix A.2).

Choose settings without test data. Global transfer uses prespecified unit strength $\rho = 1$ in the primary benchmark. Neighborhood and compositional hyperparameters are selected on training-only validation splits; a separate comparator selects global ρ likewise (Appendix A.3).

Theoretical grounding. Let $\mathcal{I}_p$ denote information available for an unseen perturbation before observing its true response. For any additive correction $h(\mathcal{I}_p)$, define $R(h) = \mathbb{E}\|g_p - h(\mathcal{I}_p)\|_2^2$.

Proposition 4.1 (Optimal error transfer). *If* $\mathbb{E}\|g_p\|_2^2 < \infty$, *then*

$$h^\star(\mathcal{I}_p) = \mathbb{E}[g_p \mid \mathcal{I}_p] = \arg\min_h R(h), \tag{8}$$

and

$$R(h) - R(h^\star) = \mathbb{E}\big\|h(\mathcal{I}_p) - \mathbb{E}[g_p \mid \mathcal{I}_p]\big\|_2^2 \geq 0.$$

Thus, GTC estimates the predictable component of future error from historical gaps: Global GTC approximates $\mathbb{E}[g]$, while local variants condition this estimate on perturbation-specific information. The proof and finite-sample analysis are given in Appendix B.

4.2 Can Historical Errors Improve Unseen Perturbation Prediction?

We next ask whether historical errors improve predictions on unseen perturbations.

Historical errors help on gene combinations. On Norman2019, GTC helps in component-visible and label-unseen settings. On `combo_unseen0`, Global improves mean DEG@50 by 1.4 percentage points (pp) across 9/10 predictors. Perturbation-embedding retrieval yields a 7.6 pp gain (10/10), and compositional transfer yields 6.3 pp. In contrast, Simple Additive reduces mean DEG@50 by 5.7 pp (Table 1). On `combo_unseen1`, all 43 combination labels are unseen but one component is training-visible; Global and perturbation-embedding retrieval improve sign-aware DEG@50 by 1.4 pp (9/10) and 3.9 pp (10/10), respectively (Table S3). Thus, GTC corrects unseen combinations.

Table 1: **DEG@50 on held-out cells for Norman2019 combinations.** Combination labels and components are training-visible; parentheses show absolute percentage-point changes from Base.

Model	Base	Baseline	GTC					
		Simple Add.	Global	Local–Pert.	Local–Δ_{pred}	G+L–Pert.	G+L–Δ_{pred}	Compositional
BiolordStar	0.716	0.629 (-8.7)	0.736 (+2.0)	0.754 (+3.8)	0.742 (+2.6)	0.753 (+3.7)	0.742 (+2.6)	0.735 (+1.9)
CPA	0.493	0.489 (-0.4)	0.493 (+0.0)	0.706 (+21.3)	0.641 (+14.8)	0.706 (+21.3)	0.639 (+14.6)	0.684 (+19.1)
DecoderOnly	0.725	0.635 (-9.0)	0.747 (+2.2)	0.757 (+3.2)	0.759 (+3.4)	0.757 (+3.2)	0.758 (+3.3)	0.754 (+2.9)
GEARS	0.526	0.473 (-5.2)	0.552 (+2.6)	0.670 (+14.4)	0.676 (+15.0)	0.666 (+14.1)	0.676 (+15.0)	0.635 (+10.9)
GenePert	0.726	0.656 (-7.0)	0.745 (+1.9)	0.756 (+3.0)	0.754 (+2.8)	0.755 (+2.8)	0.753 (+2.7)	0.752 (+2.6)
LatentAdditive	0.745	0.657 (-8.8)	0.758 (+1.3)	0.763 (+1.8)	0.762 (+1.7)	0.763 (+1.9)	0.762 (+1.7)	0.761 (+1.6)
LinearAdditive	0.634	0.562 (-7.2)	0.665 (+3.0)	0.698 (+6.3)	0.715 (+8.1)	0.698 (+6.3)	0.717 (+8.3)	0.696 (+6.1)
SparseAdditiveVAE	0.745	0.702 (-4.3)	0.745 (-0.1)	0.750 (+0.5)	0.747 (+0.1)	0.748 (+0.3)	0.747 (+0.1)	0.740 (-0.6)
PRNet	0.704	0.648 (-5.6)	0.709 (+0.5)	0.739 (+3.5)	0.715 (+1.2)	0.731 (+2.8)	0.726 (+2.2)	0.733 (+2.9)
scLAMBDA	0.523	0.511 (-1.2)	0.527 (+0.4)	0.703 (+17.9)	0.531 (+0.8)	0.703 (+17.9)	0.528 (+0.5)	0.678 (+15.5)
Mean	0.654	0.596 (-5.7)	0.668 (+1.4)	**0.729 (+7.6)**	0.704 (+5.0)	0.728 (+7.4)	0.705 (+5.1)	0.717 (+6.3)
Wins	–	0/10	9/10	10/10	10/10	10/10	10/10	9/10

Table 2: **DEG@50 on unseen single-gene perturbations in Replogle.** Parentheses show absolute percentage-point changes from Base.

Model	Base	GTC				
		Global	Local–Pert.	Local–Δ_{pred}	G+L–Pert.	G+L–Δ_{pred}
BiolordStar	0.262	0.278 (+1.5)	0.274 (+1.2)	0.275 (+1.3)	0.277 (+1.5)	0.272 (+1.0)
CPA	0.220	0.251 (+3.2)	0.259 (+3.9)	0.245 (+2.6)	0.265 (+4.5)	0.245 (+2.6)
DecoderOnly	0.262	0.277 (+1.5)	0.277 (+1.5)	0.281 (+1.9)	0.279 (+1.7)	0.281 (+1.9)
GEARS	0.230	0.233 (+0.3)	0.242 (+1.2)	0.262 (+3.2)	0.241 (+1.1)	0.261 (+3.1)
GenePert	0.132	0.274 (+14.3)	0.265 (+13.3)	0.276 (+14.4)	0.273 (+14.1)	0.275 (+14.3)
LatentAdditive	0.276	0.277 (+0.1)	0.288 (+1.2)	0.280 (+0.4)	0.286 (+1.0)	0.280 (+0.4)
LinearAdditive	0.213	0.245 (+3.2)	0.245 (+3.2)	0.266 (+5.3)	0.248 (+3.5)	0.266 (+5.3)
SparseAdditiveVAE	0.243	0.245 (+0.2)	0.260 (+1.7)	0.257 (+1.4)	0.256 (+1.3)	0.257 (+1.4)
PRNet	0.275	0.276 (+0.0)	0.277 (+0.2)	0.274 (-0.1)	0.275 (+0.0)	0.276 (+0.0)
scLAMBDA	0.252	0.254 (+0.2)	0.265 (+1.3)	0.265 (+1.3)	0.265 (+1.4)	0.265 (+1.3)
Mean	0.236	0.261 (+2.5)	0.265 (+2.9)	**0.268 (+3.2)**	0.267 (+3.0)	0.268 (+3.1)
Wins	–	10/10	10/10	9/10	9/10	10/10

Historical errors help on unseen genes. The same principle extends to Replogle, where test genes are not observed during training. All GTC variants improve the mean DEG@50 by 2.5–3.2 pp, with at least 9/10 predictors benefiting from each strategy (Table 2). Positive median gains of 1.7–2.3 pp further show that the improvement is not driven solely by a small number of large gains.

Historical errors help on unseen drugs. On SciPlex3 K562, GTC also improves unseen-compound predictions (Table 3). Global transfer raises the mean by 3.5 pp, local transfer reaches 7.0 pp with perturbation fingerprints and 6.1 pp with response-informed neighbors, and Global+Local reaches 6.8 pp. The response-informed variant benefits all 10 predictors (+6.4 pp). Median gains remain positive but vary substantially across base models.

More specific errors can improve the correction. Global fingerprints provide useful corrections across these settings, with additional gains from local or compositional transfer. We next test whether cellular-context mismatch weakens fingerprint attribution and correction gains.

GTC improves some metrics, but not all. Across 80 model–context units, GTC reduced energy distance and MMD in 74 units but did not improve Spearman correlation or PDS on average. Calibration, data-size, seed, local-neighbor, and pathway checks are reported in Appendix H.

4.3 When Does Error Transfer Fail?

The gains above are not uniform across settings, suggesting that error fingerprints may depend on cellular context. We therefore ask when historical errors cease to transfer.

Table 3: **DEG@50 on unseen compounds in SciPlex3.** Results use the shared K562 setting; parentheses show absolute percentage-point changes from Base.

Model	Base	GTC				
		Global	Local–Pert.	Local–Δ_{pred}	G+L–Pert.	G+L–Δ_{pred}
BiolordStar	0.403	0.409 (+0.6)	0.427 (+2.4)	0.434 (+3.1)	0.433 (+2.9)	0.423 (+1.9)
CPA	0.421	0.424 (+0.3)	0.429 (+0.8)	0.427 (+0.7)	0.423 (+0.3)	0.434 (+1.3)
DecoderOnly	0.300	0.320 (+2.0)	0.371 (+7.1)	0.372 (+7.2)	0.370 (+7.0)	0.372 (+7.2)
GEARS	0.084	0.345 (+26.1)	0.385 (+30.2)	0.324 (+24.1)	0.372 (+28.8)	0.345 (+26.1)
GenePert	0.333	0.272 (-6.1)	0.354 (+2.1)	0.368 (+3.5)	0.354 (+2.1)	0.368 (+3.5)
LatentAdditive	0.427	0.415 (-1.2)	0.417 (-0.9)	0.427 (+0.0)	0.420 (-0.7)	0.431 (+0.4)
LinearAdditive	0.361	0.354 (-0.7)	0.409 (+4.8)	0.383 (+2.2)	0.398 (+3.7)	0.383 (+2.2)
SparseAdditiveVAE	0.305	0.329 (+2.4)	0.391 (+8.6)	0.356 (+5.1)	0.385 (+8.0)	0.356 (+5.1)
PRNet	0.262	0.360 (+9.8)	0.379 (+11.7)	0.401 (+13.9)	0.379 (+11.7)	0.401 (+13.9)
scLAMBDA	0.392	0.410 (+1.8)	0.421 (+2.9)	0.406 (+1.4)	0.434 (+4.2)	0.412 (+1.9)
Mean	0.329	0.364 (+3.5)	**0.398 (+7.0)**	0.390 (+6.1)	0.397 (+6.8)	0.392 (+6.4)
Wins	–	7/10	9/10	9/10	9/10	10/10

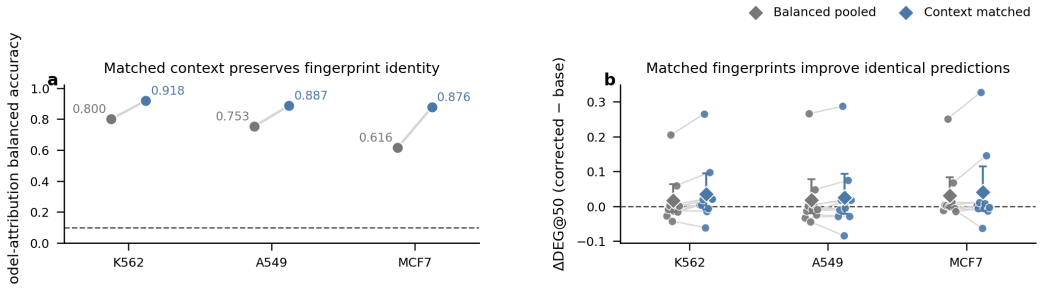

Figure 4: **Error transfer weakens when source and target cellular contexts differ.** (a) Model-attribution balanced accuracy for context-matched and balanced-pooled fingerprints in SciPlex3; the dashed line is 10-class chance (0.10). (b) DEG@50 changes when the same held-out predictions are corrected with the two training-derived fingerprints. Lines pair the same shared checkpoint, diamonds are means, and error bars are bootstrap 95% CIs over the ten checkpoints.

Compare contexts without changing the model. To separate fingerprint estimation from per-cell-line retraining, we held each of ten shared SciPlex3 model checkpoints fixed and changed only whether its training-error fingerprint was estimated from the target cell line or from a balanced pool of all three cell lines. The model parameters and held-out predictions were therefore identical between the two conditions, isolating the effect of fingerprint context.

Matched cell contexts work better. Context-matched attribution accuracy is 0.918, 0.887, and 0.876 in K562, A549, and MCF7, respectively, compared with 0.800, 0.753, and 0.616 for balanced pooling (Figure 4a). Mean DEG@50 gains are also larger (3.40 versus 2.21 pp; paired difference 1.19 pp, 95% CI 0.16–2.22 pp; Figure 4b). Pooling remains useful but weakens both attribution and correction, establishing context mismatch as a graded limitation on error transfer. This behavior is consistent with the population analysis in Appendix B, which shows that transferring a mismatched mean error introduces an explicit context-mismatch penalty. Implementation details and per-cell-line values are in Appendix H.2.

4.4 What Biological Failures Can Error Transfer Repair?

To make these gains biologically concrete, we trace a representative SciPlex3 K562 correction (DecoderOnly, SRT2104) from genes to pathways; its gain is near the median among eligible cases.

Past errors reveal which genes will be missed. The training-derived fingerprint closely aligns with the held-out gene-wise error pattern (cosine similarity 0.972; Figure 5a). Among ground-truth

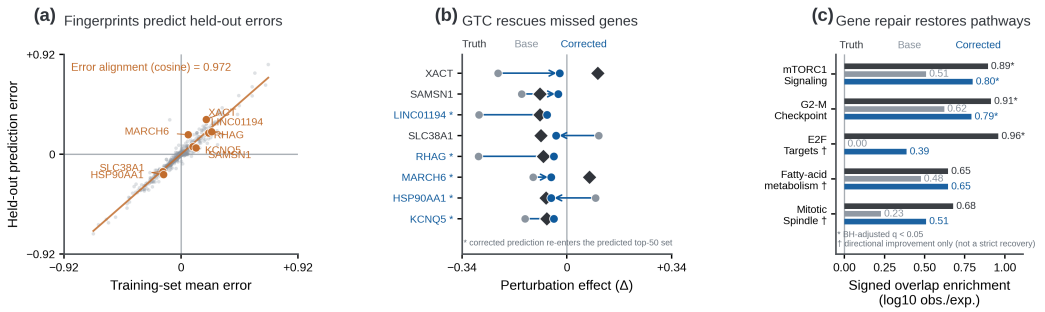

Figure 5: **Error fingerprints predict and repair recurring biological failures.** SciPlex3 K562 example (DecoderOnly, SRT2104; global GTC). (a) Training and held-out gene-wise errors align strongly (cosine $= 0.972$). (b) GTC moves all eight selected true top-50 genes missed by Base toward Truth; asterisks mark the five restored to the predicted top 50. (c) GTC significantly recovers *mTORC1 Signaling* and *G2-M Checkpoint* and shifts three other pathways toward Truth. Asterisks denote BH-adjusted $q < 0.05$; daggers denote directional improvement without strict recovery. Scores are signed log observed-to-expected overlap, not GSEA NES (Appendix F).

top-response genes missed by the base predictor, global GTC moves all eight displayed genes toward the held-out truth, with five re-entering the predicted top-50 set (Figure 5b).

Correction can restore missed pathways. The gene-level corrections also restore higher-level biological signals (Figure 5c). In this case, *mTORC1 Signaling* and *G2-M Checkpoint* are significantly recovered after correction, while *E2F Targets*, *Fatty Acid Metabolism*, and *Mitotic Spindle* move toward their ground-truth pathway scores.

Pathway improvements extend beyond the case study. Across all held-out cases, GTC reduced pathway-score error in 59.8%, indicating that this example reflects a broader, though non-universal, improvement across models and contexts rather than an isolated success (Appendix H.7).

5 Conclusion and Limitations

Prediction errors recur and can be reused. Across genetic, combinatorial, and chemical perturbations, virtual-cell predictors repeat many of the same gene- and pathway-level errors on unseen interventions. These fingerprints largely arise from stable gene-wise over- and under-estimation. GTC reuses that error history to adjust new predictions without retraining the underlying model.

Cellular context constrains error transfer. Error transfer depends on cellular context: fingerprints estimated in the target cell line are more reliable than balanced fingerprints pooled across cell lines. They characterize predictive bias rather than causal mechanisms and require evaluated perturbations from a compatible context. GTC is an exploratory feasibility test of whether historical errors can improve unseen-perturbation predictions, rather than an attempt to achieve state-of-the-art performance through task-specific optimization. Its gains vary by model and metric, so correction should be evaluated against the intended biological endpoint.

AI Use Statement

Generative AI tools were used for research ideation, literature discovery, manuscript drafting, and language polishing. AI-assisted suggestions were reviewed against the experimental results and relevant literature. All experiments, numerical results, and statistical analyses were conducted and verified by the authors. The authors take full responsibility for the final manuscript.

Ethics Statement

This study uses previously released single-cell perturbation datasets and does not collect new human-subject data. The analyses are intended for computational genomics research and not for clinical diagnosis or treatment.

Reproducibility Statement

Datasets, preprocessing, model training, data splits, fingerprint construction, correction procedures, evaluation metrics, and statistical analyses are described in the main text and appendix. Fingerprints and correction parameters use training data only, while held-out perturbations are reserved for evaluation. Results are evaluated across multiple models, perturbation settings, and training seeds where applicable. Source code and reproduction scripts will be released upon publication.

References

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: 1341–1344, 2025. doi: 10.1038/s41592-025-02772-6.

Xiaofan Bai, Pingyi Hu, Xiaojing Ma, Linchen Yu, Dongmei Zhang, Qi Zhang, and Bin Benjamin Zhu. ESF: Efficient sensitive fingerprinting for black-box tamper detection of large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 10477–10494. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.546.

Michael Bereket and Theofanis Karaletsos. Modelling cellular perturbations with the sparse additive mechanism shift variational autoencoder. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Charlotte Bunne, Stefan G. Stark, Gabriele Gut, Jacobo Sarabia Del Castillo, Mitch Levesque, Kjong-Van Lehmann, Lucas Pelkmans, Andreas Krause, and Gunnar Rätsch. Learning single-cell perturbation responses using neural optimal transport. *Nature Methods*, 20(11):1759–1768, 2023. doi: 10.1038/s41592-023-01969-x.

Charlotte Bunne, Yusuf Roohani, Yanay Rosen, Ankit Gupta, Xikun Zhang, Marcel Roed, Theo Alexandrov, Mohammed AlQuraishi, Patricia Brennan, Daniel B Burkhardt, et al. How to build the virtual cell with artificial intelligence: Priorities and opportunities. *Cell*, 187(25):7045–7063, 2024.

Yiqun T. Chen and James Zou. GenePert: Leveraging GenePT embeddings for gene perturbation prediction. *bioRxiv*, 2024. doi: 10.1101/2024.10.27.620513.

Jiabei Cheng, Changxi Chi, Jingbo Zhou, Hongyi Xin, and Jun Xia. PRESCRIBE: Predicting single-cell responses with bayesian estimation. In *Advances in Neural Information Processing Systems*, volume 38, pp. 102937–102968, 2025. doi: 10.52202/085713-4868.

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(8):1470–1480, 2024. doi: 10.1038/s41592-024-02201-0.

Alice Driessen, Dhruva Abhijit Rajwade, Benedek Harsanyi, Marianna Rapsomaniki, and Jannis Born. Conditional Monge gap enables generalizable single-cell perturbation modelling. *Nature Machine Intelligence*, 8:984–996, 2026. doi: 10.1038/s42256-026-01242-8.

Yijia He, Jiaqi Zhang, Munther Dahleh, and Caroline Uhler. MORPH predicts the single-cell outcome of genetic perturbations across conditions and data modalities. *bioRxiv*, 2025. doi: 10.1101/2025.06.10.659017. Preprint.

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

Mohammad Lotfollahi, Anna Klimovskaia Susmelj, Carlo De Donno, et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular Systems Biology*, 19 (6):e11517, 2023. doi: 10.15252/msb.202211517.

Nils Lukas, Yuxuan Zhang, and Florian Kerschbaum. Deep neural network fingerprinting by conferrable adversarial examples. In *International Conference on Learning Representations*, 2021.

Xinjie Mao, Songming Zhang, Qianhong Wen, Xiangyu Wen, Kedu Jin, Hao Wu, Shuizhou Chen, Yuqiang Li, Lei Bai, Qi Liu, Ning Ding, Siqi Sun, and Zhangyang Gao. Benchmarking virtual cell models for in-the-wild perturbation response. *arXiv preprint arXiv:2604.27646*, 2026.

Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. Do GANs leave artificial fingerprints? In *IEEE Conference on Multimedia Information Processing and Retrieval*, pp. 506–511, 2019. doi: 10.1109/MIPR.2019.00103.

Alexis Molina and Xinyi Zhang. Perturbation response decomposition enables biologically aligned generalization to unseen perturbations and cellular contexts. *bioRxiv*, 2026. doi: 10.64898/2026.07.24.740459. Preprint.

Anshul Nasery, Jonathan Hayase, Creston Brooks, Peiyao Sheng, Himanshu Tyagi, Pramod Viswanath, and Sewoong Oh. Scalable fingerprinting of large language models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-4170.

Thomas M. Norman, Max A. Horlbeck, Joseph M. Replogle, Alex Y. Ge, Albert Xu, Marco Jost, Luke A. Gilbert, and Jonathan S. Weissman. Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. *Science*, 365(6455):786–793, 2019. doi: 10.1126/science.aax4438.

Dario Pasquini, Evgenios M. Kornaropoulos, and Giuseppe Ateniese. LLMmap: Fingerprinting for large language models. In *34th USENIX Security Symposium (USENIX Security 25)*, pp. 299–318. USENIX Association, 2025. URL https://www.usenix.org/conference/usenixsecurity25/presentation/pasquini.

Zoe Piran, Niv Cohen, Yedid Hoshen, and Mor Nitzan. Disentanglement of single-cell data with BioLORD. *Nature Biotechnology*, 42(11):1678–1683, 2024. doi: 10.1038/s41587-023-02079-x.

Xiaoning Qi, Lianhe Zhao, Chenyu Tian, Yueyue Li, Zhen-Lin Chen, Peipei Huo, Runsheng Chen, Xiaodong Liu, Baoping Wan, Shengyong Yang, and Yi Zhao. PRnet: Predicting transcriptional responses to novel chemical perturbations using deep generative model for drug discovery. *Nature Communications*, 15:9256, 2024. doi: 10.1038/s41467-024-53457-1.

Joseph M. Replogle, Reuben A. Saunders, Angela N. Pogson, Jeffrey A. Hussmann, Alexander Lenail, Alina Guna, Lauren Mascibroda, Eric J. Wagner, Karen Adelman, Gila Lithwick-Yanai, et al. Mapping information-rich genotype-phenotype landscapes with genome-scale Perturb-seq. *Cell*, 185(14):2559–2575.e28, 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*, 42(6):927–935, 2024. doi: 10.1038/s41587-023-01905-6.

Sanjay R. Srivatsan, José L. McFaline-Figueroa, Vijay Ramani, Lauren Saunders, Junyue Cao, Jonathan Packer, Hannah A. Pliner, Dana L. Jackson, Riza M. Daza, Lena Christiansen, et al. Massively multiplex chemical transcriptomics at single-cell resolution. *Science*, 367(6473):45–51, 2020. doi: 10.1126/science.aax6234.

Yangqi Su, Dingyu Liu, Vipin Menon, Bicna Song, Samuel Boccara, Nan Zhang, Huan Zhao, Jiahui Hazel Zhao, Lei Wang, Nan Hu, Mpathi Nzima, Alon Katz, Bharath Kumar Swargam, Seth A. Ament, Yarui Diao, Hanrui Zhang, Lumen Chao, Gary Hon, Danwei Huangfu, and Wei Li. pertTF: context-aware AI modeling for genome-scale and cross-system perturbation prediction. *bioRxiv*, 2026. doi: 10.64898/2026.03.12.711379. Preprint.

Yao Tong, Haonan Wang, Siquan Li, Kenji Kawaguchi, and Tianyang Hu. Seed-Prints: Fingerprints can even tell which seed your large language model was trained from. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/49e48a1a40628b160101c1375b8eb151-Abstract-Conference.html.

Ramon Viñas Torné, Maciej Wiatrak, Zoe Piran, Shuyang Fan, Liangze Jiang, Sarah A. Teichmann, Mor Nitzan, and Maria Brbic. Systema: a framework for evaluating genetic perturbation response prediction beyond systematic variation. *Nature Biotechnology*, 44:1050–1059, 2026. doi: 10.1038/s41587-025-02777-8.

Gefei Wang, Tianyu Liu, Jia Zhao, Youshu Cheng, and Hongyu Zhao. Modeling and predicting single-cell multi-gene perturbation responses with scLAMBDA. *bioRxiv*, 2024. doi: 10.1101/2024.12.04.626878.

Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. CNN-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8695–8704, 2020.

Zhiting Wei, Yiheng Wang, Yicheng Gao, Shuguang Wang, Ping Li, Duanmiao Si, Yuli Gao, Siqi Wu, Danlu Li, Kejing Dong, Xingbo Yang, Chen Tang, Shaliu Fu, Xiaohan Chen, Wannian Li, Yuzhou You, Chen Zhang, Aibin Liang, Guohui Chuai, and Qi Liu. Benchmarking algorithms for generalizable single-cell perturbation response prediction. *Nature Methods*, 23:451–464, 2026. doi: 10.1038/s41592-025-02980-0.

Aaron Wenteler, Martina Occhetta, Nikhil Branson, Victor Curean, Magdalena Huebner, William Dee, William Connell, Siu Pui Chung, Alex Hawkins-Hooker, Yasha Ektefaie, César Miguel Valdez Córdova, and Amaya Gallagher-Syed. PertEval-scFM: Benchmarking single-cell foundation models for perturbation effect prediction. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 66633–66677. PMLR, 2025. URL https://proceedings.mlr.press/v267/wenteler25a.html.

Yan Wu, Esther Wershof, Sebastian M. Schmon, Marcel Nassar, Błażej Osiński, Ridvan Eksi, Zichao Yan, Rory Stark, Kun Zhang, and Thore Graepel. PerturBench: Benchmarking machine learning models for cellular perturbation analysis. *arXiv preprint arXiv:2408.10609*, 2024.

Hanwen Xing and Christopher Yau. GPerturb: Gaussian process modelling of single-cell perturbation data. *Nature Communications*, 16:5423, 2025. doi: 10.1038/s41467-025-61165-7.

Matthew D. Young, Matthew J. Wakefield, Gordon K. Smyth, and Alicia Oshlack. Gene ontology analysis for RNA-seq: Accounting for selection bias. *Genome Biology*, 11:R14, 2010. doi: 10.1186/gb-2010-11-2-r14.

Hengshi Yu, Weizhou Qian, Yuxuan Song, and Joshua D. Welch. PerturbNet predicts single-cell responses to unseen chemical and genetic perturbations. *Molecular Systems Biology*, 21:960–982, 2025. doi: 10.1038/s44320-025-00131-3.

Ning Yu, Larry S. Davis, and Mario Fritz. Attributing fake images to GANs: Learning and analyzing GAN fingerprints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7556–7566, 2019.

Ning Yu, Vladislav Skripniuk, Sahar Abdelnabi, and Mario Fritz. Artificial fingerprinting for generative models: Rooting deepfake attribution in training data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Xinyu Yuan, Xixian Liu, Ya Shi Zhang, Zuobai Zhang, Hongyu Guo, and Jian Tang. PerturbDiff: Functional diffusion for single-cell perturbation modeling. *arXiv preprint arXiv:2602.19685*, 2026. doi: 10.48550/arXiv.2602.19685.

Martin Jinye Zhang, Kangcheng Hou, Kushal K. Dey, Saori Sakaue, Karthik A. Jagadeesh, Kathryn Weinand, et al. Polygenic enrichment distinguishes disease associations of individual cells in single-cell RNA-seq data. *Nature Genetics*, 54:1572–1580, 2022. doi: 10.1038/s41588-022-01167-z.

A GTC METHOD DETAILS

A.1 LOCAL TRANSFER: NEIGHBORHOOD CONSTRUCTION

Neighbor representations. Three representations are used to retrieve training perturbations similar to an unseen query perturbation p: (i) *Perturbation-KNN* uses VCBench-provided perturbation embeddings: precomputed ESM2 protein-sequence embeddings (5,120-dimensional) for genetic perturbations (Norman2019, Replogle) and 2,048-dimensional binary molecular fingerprints derived from SMILES (ECFP-style) for compounds (SciPlex3); (ii) Δ*pred-KNN* uses the model's own predicted effect vector $\Delta^{\mathrm{pred}}_{m,p,c}$, L_2-normalized; and (iii) *Hybrid* (appendix ablation only) concatenates the L_2-normalized perturbation embedding and the L_2-normalized Δ^{pred} block and re-normalizes; perturbations without an embedding contribute a zero block.

Similarity, neighbors, and weights. The reference set is the training perturbations with an available representation; validation and test perturbations are never part of it, so no self-exclusion rule is needed. Similarity is cosine similarity. The neighbor set $\mathcal{N}_K(p)$ takes the top-K reference perturbations, with K truncated at the reference-set size. Weights are a fixed-temperature softmax within the neighbor set,

$$w_{pi} = \frac{\exp\big(\mathrm{sim}(p,i)/T\big)}{\sum_{j\in\mathcal{N}_K(p)} \exp\big(\mathrm{sim}(p,j)/T\big)}, \qquad T = 0.5, \tag{9}$$

so $\sum_i w_{pi} = 1$; this is the only normalization applied to $\widehat{g}$, which is neither clipped nor rescaled. Local transfer uses raw neighbor gaps, $\widehat{g}^{\mathrm{local}} = \sum_i w_{pi} g_{m,i,c}$, and G+L uses residuals, $\widehat{g}^{\mathrm{G+L}} = F_{m,c} + \eta \sum_i w_{pi} r_{m,i,c}$ with $r_{m,i,c} = g_{m,i,c} - F_{m,c}$.

Representation coverage. Representation keys are matched case-insensitively, and a combination perturbation $A{+}B$ is represented by the sum of its component embeddings (under cosine similarity this is equivalent to their mean). With these rules, coverage is complete in all reported settings: 69/69 Norman2019 test combinations (all 43 component genes have embeddings), 100/100 Replogle test genes, and 38/38 SciPlex3 test compounds.

A.2 CORRECTION RULES

All corrections follow Eq. 3 of the main text, $\Delta^{\mathrm{corr}} = \Delta^{\mathrm{pred}} + \rho\,\widehat{g}$. Global transfer uses $\widehat{g} = F_{m,c}$ with $\rho = 1.0$ fixed (no search). Additive transfer (Norman2019 only) uses $\widehat{g}_{A+B} = g_{m,A,c} + g_{m,B,c}$ and requires both components among the training perturbations, otherwise the combination is skipped; its strength ρ_{add} is validation-selected per model from $\{0.25, 0.5, 0.75, 1.0\}$ (selected values ranged from 0.25 to 0.55 across models). The non-GTC *Simple Additive* baseline predicts the combination as $\Delta^{\mathrm{pred}}_{m,A,c} + \Delta^{\mathrm{pred}}_{m,B,c}$ with no gap and no ρ.

A.3 HYPERPARAMETER SELECTION

Hyperparameters are selected on validation perturbations only; held-out test ground truth is used exactly once with the selected values. Validation uses five random 80:20 train/validation perturbation splits (seeds 0–4) with mean validation DEG@50 as the selection metric, and selection is performed independently for each model within each dataset/context.

Two search grids are used. The Perturbation-KNN columns search $\rho \in \{0.25, 0.5, 0.75, 1.0\}$, $\eta \in \{0, 0.5, 1.0\}$, $K \in \{3, 5, 10\}$, select the best triple per split, and aggregate across the five splits (mean ρ, mean η, median K). The Δpred-KNN and Hybrid columns search a single denser grid of 336 combinations, $\rho \in \{0, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 2.0\}$, $\eta \in \{0, 0.25, 0.5, 0.75, 1.0, 1.5, 2.0\}$, $K \in \{1, 3, 5, 10, 20, 50\}$, and keep the single best combination by five-split mean validation DEG@50; the pure Local variant fixes $\eta \equiv 1$ (no global term). Temperature $T = 0.5$ and the softmax weighting are fixed in both grids.

Sensitivity. A separate study (Table S1) compares fixed recipes against validation selection and varies the weighting scheme (softmax, uniform $1/K$, inverse distance $1/(1-\mathrm{sim})$) and temperature $T \in \{0.1, 0.3, 0.5, 1.0, 2.0\}$ around the selected point. Validation selection improves over the fixed

Table S1: Fixed versus validation-selected KNN representations. "Fixed" uses $\rho = 1, \eta = 0.5, K = 50$; "Validation" selects the best combination of the 336-point grid by mean validation DEG@50 over five perturbation-level splits (seeds 0–4). Gains are DEG@50 changes from each model's own Base, averaged over the ten predictors. Seed SD is the mean within-model SD across five single-seed validation selections; Model SD is the between-model SD. The three Norman2019 combination splits are listed separately.

Dataset	Representation	n	Base	DEG@50 gain		Spread (pp)	
				Fixed	Validation	Seed SD	Model SD
Norman2019, combo_unseen0 (diagnostic)	Pert.-KNN	10	0.654	2.15	**7.59**	1.01	7.46
	Δpred-KNN	10	0.654	2.10	**5.14**	1.19	5.65
	Hybrid	10	0.654	2.19	**8.80**	0.92	9.55
Norman2019 (combo_unseen1)	Pert.-KNN	10	0.575	1.80	**3.81**	0.41	3.31
	Δpred-KNN	10	0.575	1.94	**2.41**	0.45	2.71
	Hybrid	10	0.575	1.86	**4.04**	0.43	3.65
Norman2019, combo_unseen2	Pert.-KNN	10	0.532	0.65	**-0.02**	0.64	1.47
	Δpred-KNN	10	0.532	0.80	**0.28**	0.67	2.80
	Hybrid	10	0.532	0.77	**0.00**	0.68	2.95
Replogle K562	Pert.-KNN	10	0.236	2.62	**3.08**	0.28	4.20
	Δpred-KNN	10	0.236	2.83	**3.13**	0.36	4.23
	Hybrid	10	0.236	2.91	**3.15**	0.43	4.22
SciPlex3 K562 (shared)	Pert.-KNN	10	0.329	4.38	**6.23**	0.94	8.48
	Δpred-KNN	10	0.329	3.68	**6.36**	0.53	7.99
	Hybrid	10	0.329	4.18	**7.35**	0.85	9.66

recipe (overall gain $+0.030 \rightarrow +0.058$ for Perturbation-KNN, $+0.029 \rightarrow +0.049$ for Δpred-KNN), and the importance ranking is $\rho \gg \eta \approx K \gg$ weighting scheme $>$ temperature; $\eta > 1$ is consistently harmful and $\eta = 0$ (pure global) is nearly optimal, consistent with the bounded local gains reported in the main text. Table S1 lists the three Norman2019 combination splits separately and adds the spread of every endpoint across validation seeds: the seed SD stays at or below 1.2 pp whereas the between-predictor SD spans 1.5–9.7 pp, so the reported gains are governed by which predictor is corrected rather than by the validation seed. Validation selection is most useful when the tested combination, or at least one component, is represented in training; on `combo_unseen2`, the fixed recipe matches or exceeds it.

B Theoretical Foundations of Gap Transferability Correction

We provide a statistical interpretation of Gap Transferability Correction (GTC) and characterize when transferring historical errors can improve predictions on unseen perturbations. The analysis concerns squared error in perturbation-effect space; our discrete DEG-based endpoints are evaluated empirically.

B.1 Setup

Fix a predictor and cellular context and suppress these indices. For perturbation p, let

$$Y_p = \Delta_p^{\text{real}}, \qquad \widehat{Y}_p = \Delta_p^{\text{pred}}, \qquad G_p = Y_p - \widehat{Y}_p.$$

GTC applies

$$\widehat{Y}_p^{\text{corr}} = \widehat{Y}_p + \rho \widehat{G}_p,$$

where $\widehat{G}_p$ is estimated without access to the held-out response and $\rho \geq 0$ controls correction strength.

B.2 When Does Error Transfer Help?

Proposition B.1 (Exact improvement condition)**.** *For any perturbation,*

$$\|Y - \widehat{Y}^{\text{corr}}\|_2^2 = \|G\|_2^2 - 2\rho\langle G, \widehat{G}\rangle + \rho^2 \|\widehat{G}\|_2^2.$$

Thus, for $\rho > 0$, GTC improves squared prediction error if and only if

$$\boxed{\langle G, \widehat{G} \rangle > \frac{\rho}{2} \|\widehat{G}\|_2^2.}$$

Equivalently, when both vectors are nonzero,

$$\cos(G, \widehat{G}) > \frac{\rho}{2} \frac{\|\widehat{G}\|_2}{\|G\|_2}.$$

Proof. Since $Y - \widehat{Y}^{\mathrm{corr}} = G - \rho\widehat{G}$, the result follows by expanding $\|G - \rho\widehat{G}\|_2^2$. □

This result connects the fingerprint–gap alignment measured in our experiments to correction performance. A transferred gap should point in the direction of the unseen error, while ρ prevents excessive correction.

B.3 GTC AS CONDITIONAL ERROR ESTIMATION

Let $\mathcal{I}$ denote information available before observing the held-out response, such as training errors, a perturbation embedding, or the base model's predicted response.

Theorem B.2 (Optimal additive correction)**.** *Among all corrections $h(\mathcal{I})$, the one minimizing*

$$R(h) = \mathbb{E}\|G - h(\mathcal{I})\|_2^2$$

is

$$\boxed{h^\star(\mathcal{I}) = \mathbb{E}[G \mid \mathcal{I}].}$$

Moreover,

$$R(h) = R(h^\star) + \mathbb{E}\left\|h(\mathcal{I}) - \mathbb{E}[G \mid \mathcal{I}]\right\|_2^2.$$

Proof. Let $\mu(\mathcal{I}) = \mathbb{E}[G \mid \mathcal{I}]$ and write

$$G - h(\mathcal{I}) = (G - \mu(\mathcal{I})) + (\mu(\mathcal{I}) - h(\mathcal{I})).$$

After expanding the squared norm, the cross term vanishes because $\mathbb{E}[G - \mu(\mathcal{I}) \mid \mathcal{I}] = 0$. □

The theorem gives a unified interpretation of the GTC variants.

Global transfer. Without perturbation-specific information, the optimal constant correction is

$$\mathbb{E}[G].$$

The training fingerprint

$$F = \frac{1}{|\mathcal{P}_{\mathrm{train}}|} \sum_{p \in \mathcal{P}_{\mathrm{train}}} G_p$$

is its empirical estimator.

Local transfer. For a perturbation representation Z, the optimal representation-dependent correction is

$$\mathbb{E}[G \mid Z].$$

Local GTC approximates this quantity from neighboring training perturbations. At the population level,

$$\boxed{R(\mathbb{E}[G]) - R(\mathbb{E}[G \mid Z]) = \mathbb{E}\left\|\mathbb{E}[G \mid Z] - \mathbb{E}[G]\right\|_2^2 \geq 0.}$$

Thus, local information is useful precisely when mean prediction errors vary with Z.

Global+Local transfer. Writing $\mu = \mathbb{E}[G]$ and $R = G - \mu$ gives

$$\mathbb{E}[G \mid Z] = \mu + \mathbb{E}[R \mid Z].$$

Global+Local GTC directly mirrors this decomposition, while η shrinks the estimated local residual toward the global component.

Compositional transfer. If a combination perturbation satisfies

$$G_{A+B} = G_A + G_B + \Xi_{A,B}, \qquad \mathbb{E}[\Xi_{A,B} \mid G_A, G_B] = 0,$$

then

$$\mathbb{E}[G_{A+B} \mid G_A, G_B] = G_A + G_B.$$

Hence additive gap transfer is the optimal squared-error correction based only on the two component gaps under this assumption.

B.4 Finite-Sample Behavior of Global GTC

To characterize the empirical fingerprint, assume

$$G_i = b + \varepsilon_i,$$

where $b \in \mathbb{R}^G$ is a persistent gene-wise bias, $\mathbb{E}[\varepsilon_i] = 0$, $\mathrm{Cov}(\varepsilon_i) = \Sigma$, and residuals are independent across the n training perturbations and an unseen test perturbation. No independence between genes is required.

Let

$$F_n = \frac{1}{n}\sum_{i=1}^{n} G_i.$$

Theorem B.3 (Expected risk of Global GTC)**.** *For an independent unseen gap $G_\star$,*

$$\mathbb{E}\|G_\star - \rho F_n\|_2^2 = (1-\rho)^2\|b\|_2^2 + \mathrm{tr}(\Sigma) + \frac{\rho^2}{n}\mathrm{tr}(\Sigma).$$

Therefore, the expected reduction relative to the uncorrected predictor is

$$\boxed{(2\rho - \rho^2)\|b\|_2^2 - \frac{\rho^2}{n}\mathrm{tr}(\Sigma).}$$

Proof. Since

$$F_n = b + \bar{\varepsilon},$$

the corrected test gap is

$$G_\star - \rho F_n = (1-\rho)b + \varepsilon_\star - \rho\bar{\varepsilon}.$$

The residual terms are zero mean and independent, with

$$\mathbb{E}\|\varepsilon_\star\|_2^2 = \mathrm{tr}(\Sigma), \qquad \mathbb{E}\|\bar{\varepsilon}\|_2^2 = \frac{\mathrm{tr}(\Sigma)}{n}.$$

Expanding the squared norm gives the result. □

For the primary choice $\rho = 1$,

$$\boxed{\text{Global GTC improves expected squared error} \iff \|b\|_2^2 > \frac{\mathrm{tr}(\Sigma)}{n}.}$$

Thus, the persistent fingerprint signal must exceed the sampling noise in its finite-sample estimate.

The risk-minimizing shrinkage strength is

$$\boxed{\rho^\star = \frac{\|b\|_2^2}{\|b\|_2^2 + \mathrm{tr}(\Sigma)/n}.}$$

This approaches 1 as the fingerprint becomes more precisely estimated and approaches 0 when persistent bias is weak.

B.5 Why Transfer Can Weaken Across Cell Lines

Let C denote cell line and define

$$\mu_c = \mathbb{E}[G \mid C = c], \qquad \Sigma_c = \mathrm{Cov}(G \mid C = c).$$

Proposition B.4 (Penalty from a mismatched fingerprint)**.** *For any constant correction a applied in cell line c,*

$$\boxed{\mathbb{E}[\|G - a\|_2^2 \mid C = c] = \mathrm{tr}(\Sigma_c) + \|\mu_c - a\|_2^2.}$$

Hence the population-optimal constant correction is μ_c. If μ_{pool} is a fingerprint pooled across cell lines, then

$$\boxed{R_c(\mu_{\mathrm{pool}}) - R_c(\mu_c) = \|\mu_{\mathrm{pool}} - \mu_c\|_2^2.}$$

Proof. Write

$$G - a = (G - \mu_c) + (\mu_c - a).$$

The conditional cross term is zero because $\mathbb{E}[G - \mu_c \mid C = c] = 0$, yielding the stated decomposition. □

Thus, differences in mean error structure across cell lines create an explicit population penalty for transferring a mismatched fingerprint. Pooling may nevertheless remain useful if the pooled fingerprint is closer to the target mean error than zero:

$$\|\mu_c - \mu_{\mathrm{pool}}\|_2^2 < \|\mu_c\|_2^2.$$

In finite samples, pooling may additionally reduce estimator variance; the result above isolates the bias caused specifically by cell-line mismatch.

B.6 Scope of the Guarantees

These guarantees concern continuous squared error in effect space. They do not imply that every GTC correction must improve a discrete ranking metric such as DEG@50. Improvements in DEG recovery therefore remain an empirical result of our experiments.

Overall, GTC can be viewed as estimating

$$\boxed{\mathbb{E}[\text{prediction gap} \mid \text{information available before test truth}].}$$

Global GTC estimates the mean gap, Local GTC estimates a representation-conditional mean gap, Global+Local combines the two, and compositional transfer exploits component errors when their combination error is additive in conditional expectation.

C Datasets, splits, and base-model training

C.1 Datasets and split accounting

All perturbation counts below are non-control perturbations in the version-pinned VCBench data. Validation and test assignments are fixed by the VCBench splits; no perturbation was re-assigned for this study.

Norman2019 (K562; combinatorial genetic). The component-visible diagnostic uses `combo_unseen0`: 137 training perturbations (68 single genes + 69 combinations), and validation and test sets containing the same 69 combination labels with disjoint cells. Both components of every test combination are visible during training. Strict complete-label-unseen evidence uses `combo_unseen1`: 43 test combinations (13 validation) whose complete labels never appear in training and for which exactly one component is training-visible.

Replogle K562 (single-gene). 399 training, 100 validation, and 100 test perturbations. All 100 test genes are fully unseen (0/100 in training); validation and test contain the same 100 genes with disjoint cells.

SciPlex3 (K562/A549/MCF7; chemical). Per cell line: 149 training, 38 validation, and 38 test compounds. All 38 test compounds are fully unseen (0/38 in training); validation and test contain the same 38 compounds with disjoint cells. The three cell lines share the same perturbation partition.

C.2 SciPlex3 shared versus matched per-cell-line models

Shared models are trained once on the pooled three-cell-line data with cell-line identity as a covariate, so training perturbations (and control cells) are pooled across cell lines. *Matched per-cell-line* models are trained independently on per-cell-line subsets of the same data while keeping the original split assignments, so no training data cross cell lines (30 independently trained models: 10 architectures $\times$ 3 cell lines). The perturbation sets and split partitions are identical between the two settings; only the training data and models differ.

C.3 Expression space, preprocessing, and control policy

All analyses use the VCBench 2,000-gene space: the top 2,000 highly variable genes per dataset (Seurat v3, Scanpy), library-size normalized to 10,000 counts per cell and $\log(1 + x)$ transformed. No further gene filtering or transformation is applied. The control profile x_{ctrl} is the mean of control cells from the training split of the same cellular context (a fixed context-specific policy); for SciPlex3 per-cell-line analyses the control mean is recomputed within the cell line.

C.4 Base-model training

All ten base predictors are trained with the default VCBench training configuration of each dataset: at most 50 epochs, batch size 64, one-cycle learning-rate scheduling, and early stopping (patience 50, monitored on validation loss) at seed 42. No model-specific hyperparameter changes are permitted beyond the data path, logging, and device. We audited every training run against these defaults and retrained the small number of non-conforming runs; all reported results use audited runs.

Separation of roles. Base-model training and early stopping use the dataset validation split above. GTC hyperparameters (Appendix A.3) are selected on 80/20 perturbation-level splits of the *training* perturbations only (five seeds), so dataset validation/test perturbations never enter GTC fitting, and test ground truth is used exactly once for the reported metric.

D Evaluation protocol and metrics

Pseudo-bulk effects. All cells sharing a perturbation are aggregated by the arithmetic mean (at least one cell required), giving one pseudo-bulk profile per perturbation in the 2,000-gene VCBench space. Effects are differences of means,

$$\Delta_p^{\mathrm{real}} = \bar{x}_p^{\mathrm{real}} - x_{\mathrm{ctrl}}, \qquad \Delta_{m,p}^{\mathrm{pred}} = \bar{x}_{m,p}^{\mathrm{pred}} - x_{\mathrm{ctrl}}, \tag{10}$$

computed directly on the normalized expression values; no additional log-fold-change transform is applied. The control profile x_{ctrl} follows Appendix C.3. The prediction gap is $g_{m,p} = \Delta_p^{\mathrm{real}} - \Delta_{m,p}^{\mathrm{pred}}$.

DEG@50. For a held-out perturbation, let $T_{50}(\cdot)$ denote the 50 genes with the largest absolute effect. The primary endpoint is the top-50 recall

$$\mathrm{DEG@50} = \frac{\left| T_{50}(\Delta_p^{\mathrm{real}}) \cap T_{50}(\widehat{\Delta}_{m,p}) \right|}{50}, \tag{11}$$

where $\widehat{\Delta}$ is the base or corrected prediction. The sign-aware variant counts only genes whose effect sign also agrees, $\sum_{g \in \mathcal{I}} \mathbb{1}[\operatorname{sign} \Delta_{p,g}^{\mathrm{real}} = \operatorname{sign} \widehat{\Delta}_{m,p,g}]/50$ with $\mathcal{I}$ the intersection above. Effect-space cosine ($\cos(\Delta_p^{\mathrm{real}}, \widehat{\Delta}_{m,p})$) is reported as a secondary metric. Table entries report the metric with the change from Base in percentage points (pp), i.e. $(\mathrm{corrected} - \mathrm{base}) \times 100$; a model is counted as a win when the corrected value strictly exceeds its own base value.

Fingerprint transfer metrics. For a model–context unit with training fingerprint F (Eq. 2 of the main text) and held-out gap g_p: (i) *gap alignment* is $\cos(F, g_p)$ per held-out perturbation, averaged within the unit (a fingerprint with norm below 10^{-7} scores 0); (ii) *missing-gene recovery* is $|T_{50}(F) \cap T_{50}(g_p)|/50$ (by $|F|$ and $|g_p|$), with a sign-aware variant that additionally requires matching signs; (iii) *significant-pathway recovery* is defined in Appendix F; and (iv) *model attribution* is the nearest-centroid balanced accuracy defined in Appendix G.

Aggregation and statistics. Metrics are computed per held-out perturbation, averaged within each model×context unit, and reported as macro-averages over the ten models; cross-context aggregation and all uncertainty statements use the model×context unit as the replicate (hierarchical bootstrap as in Appendix E). Per-context method-versus-base comparisons use a perturbation-level paired bootstrap (2,000 replicates, seed 42) with a Wilcoxon signed-rank test as a secondary descriptor.

D.1 Supplementary performance results

Tables S2, S4, and S5 retain the secondary endpoints for their frozen settings; primary DEG@50 blocks are omitted where duplicated. Parentheses give changes from Base (absolute cosine differences or DEG@50 percentage points). Perturbation-KNN uses ESM2 or molecular fingerprints, Δpred-KNN uses normalized predicted responses, and Hybrid combines both representations. Table S3 applies the identical protocol to complete-label-unseen combinations (`combo_unseen1`, 43 test combinations absent from training with exactly one component singly perturbed there). Compositional (Additive) columns are not defined on this split. Tables S6 and S7 then report the full SciPlex3 context analysis separately for matched and pooled training.

Table S2: Secondary metrics for the Norman2019 component-visible combination diagnostic (combo_unseen0). Parentheses show changes from Base; DEG-like metrics use percentage points. Hybrid is an appendix-only ablation.

Model	Base	Non-GTC baseline	Global transfer	Local transfer		Global + Local		Compositional transfer	Auxiliary ablation
		Simple Additive	Global	Local (Pert.-KNN)	Local (Δpred-KNN)	G+L (Pert.-KNN)	G+L (Δpred-KNN)	Additive	G+L (Hybrid)
logFC cosine									
BiolordStar	0.893	0.828 (-0.065)	**0.901 (+0.007)**	**0.915 (+0.022)**	**0.903 (+0.010)**	**0.912 (+0.019)**	**0.901 (+0.008)**	**0.908 (+0.015)**	**0.904 (+0.011)**
CPA	0.647	0.634 (-0.013)	0.647 (-0.000)	**0.876 (+0.229)**	**0.807 (+0.160)**	**0.876 (+0.229)**	**0.807 (+0.160)**	**0.864 (+0.217)**	**0.917 (+0.270)**
DecoderOnly	0.892	0.823 (-0.068)	**0.911 (+0.019)**	**0.918 (+0.026)**	**0.916 (+0.024)**	**0.917 (+0.026)**	**0.918 (+0.026)**	**0.914 (+0.023)**	**0.918 (+0.026)**
GEARS	0.609	0.552 (-0.057)	**0.636 (+0.027)**	**0.780 (+0.171)**	**0.809 (+0.200)**	**0.773 (+0.163)**	**0.809 (+0.200)**	**0.733 (+0.124)**	**0.807 (+0.197)**
GenePert	0.858	0.769 (-0.090)	**0.896 (+0.038)**	**0.910 (+0.052)**	**0.909 (+0.051)**	**0.908 (+0.050)**	**0.907 (+0.049)**	**0.912 (+0.054)**	**0.909 (+0.050)**
LatentAdditive	0.910	0.850 (-0.060)	**0.918 (+0.008)**	**0.922 (+0.012)**	**0.921 (+0.011)**	**0.922 (+0.012)**	**0.921 (+0.011)**	**0.919 (+0.009)**	**0.921 (+0.011)**
LinearAdditive	0.796	0.731 (-0.065)	**0.825 (+0.030)**	**0.860 (+0.064)**	**0.877 (+0.081)**	**0.858 (+0.062)**	**0.879 (+0.083)**	**0.843 (+0.048)**	**0.878 (+0.082)**
SparseAdditiveVAE	0.907	0.870 (-0.037)	0.906 (-0.000)	**0.911 (+0.005)**	**0.908 (+0.001)**	**0.910 (+0.004)**	**0.908 (+0.001)**	**0.908 (+0.001)**	**0.908 (+0.002)**
PRNet	0.859	0.809 (-0.051)	**0.872 (+0.012)**	**0.899 (+0.040)**	**0.881 (+0.022)**	**0.895 (+0.035)**	**0.892 (+0.033)**	**0.891 (+0.031)**	**0.900 (+0.041)**
scLAMBDA	0.660	0.657 (-0.002)	**0.668 (+0.008)**	**0.878 (+0.218)**	**0.678 (+0.018)**	**0.878 (+0.218)**	**0.670 (+0.010)**	**0.858 (+0.198)**	**0.903 (+0.243)**
Mean	0.803	0.752 (-0.051)	**0.818 (+0.015)**	**0.887 (+0.084)**	**0.861 (+0.058)**	**0.885 (+0.082)**	**0.861 (+0.058)**	**0.875 (+0.072)**	**0.896 (+0.093)**
Median	0.859	0.789 (-0.070)	**0.884 (+0.025)**	**0.905 (+0.046)**	**0.892 (+0.033)**	**0.902 (+0.043)**	**0.897 (+0.038)**	**0.899 (+0.040)**	**0.906 (+0.047)**
Win rate	–	0/10	8/10	10/10	10/10	10/10	10/10	10/10	10/10
sign-aware DEG@50									
BiolordStar	0.714	0.627 (-8.7)	**0.733 (+1.9)**	**0.752 (+3.8)**	**0.740 (+2.6)**	**0.751 (+3.7)**	**0.739 (+2.6)**	**0.733 (+1.9)**	**0.741 (+2.7)**
CPA	0.470	0.465 (-0.4)	**0.470 (+0.1)**	**0.705 (+23.5)**	**0.633 (+16.3)**	**0.705 (+23.5)**	**0.632 (+16.2)**	**0.683 (+21.4)**	**0.766 (+29.6)**
DecoderOnly	0.721	0.630 (-9.0)	**0.745 (+2.4)**	**0.755 (+3.4)**	**0.757 (+3.6)**	**0.755 (+3.4)**	**0.756 (+3.6)**	**0.752 (+3.1)**	**0.758 (+3.7)**
GEARS	0.523	0.473 (-5.0)	**0.550 (+2.7)**	**0.668 (+14.5)**	**0.673 (+15.0)**	**0.664 (+14.1)**	**0.673 (+15.0)**	**0.632 (+10.9)**	**0.675 (+15.2)**
GenePert	0.724	0.651 (-7.4)	**0.744 (+2.0)**	**0.755 (+3.0)**	**0.753 (+2.8)**	**0.754 (+2.9)**	**0.752 (+2.8)**	**0.751 (+2.7)**	**0.752 (+2.8)**
LatentAdditive	0.743	0.654 (-8.9)	**0.757 (+1.4)**	**0.761 (+1.8)**	**0.761 (+1.8)**	**0.762 (+1.9)**	**0.761 (+1.8)**	**0.760 (+1.7)**	**0.761 (+1.9)**
LinearAdditive	0.630	0.554 (-7.6)	**0.662 (+3.2)**	**0.694 (+6.3)**	**0.712 (+8.1)**	**0.694 (+6.4)**	**0.716 (+8.5)**	**0.694 (+6.4)**	**0.721 (+9.0)**
SparseAdditiveVAE	0.743	0.698 (-4.5)	0.742 (-0.1)	**0.748 (+0.5)**	**0.744 (+0.1)**	**0.746 (+0.3)**	**0.744 (+0.1)**	0.738 (-0.5)	**0.744 (+0.1)**
PRNet	0.702	0.646 (-5.6)	**0.706 (+0.4)**	**0.737 (+3.6)**	**0.713 (+1.2)**	**0.730 (+2.8)**	**0.724 (+2.2)**	**0.732 (+3.0)**	**0.738 (+3.6)**
scLAMBDA	0.496	0.490 (-0.6)	**0.502 (+0.6)**	**0.701 (+20.4)**	**0.510 (+1.3)**	**0.701 (+20.4)**	**0.503 (+0.6)**	**0.677 (+18.1)**	**0.741 (+24.5)**
Mean	0.647	0.589 (-5.8)	**0.661 (+1.5)**	**0.728 (+8.1)**	**0.699 (+5.3)**	**0.726 (+8.0)**	**0.700 (+5.3)**	**0.715 (+6.9)**	**0.740 (+9.3)**
Median	0.708	0.629 (-7.9)	**0.720 (+1.2)**	**0.743 (+3.5)**	**0.727 (+1.9)**	**0.738 (+3.0)**	**0.732 (+2.4)**	**0.733 (+2.5)**	**0.743 (+3.5)**
Win rate	–	0/10	9/10	10/10	10/10	10/10	10/10	9/10	10/10

Table S3: Results on complete-label-unseen Norman2019 combinations (combo_unseen1; 43 test pairs with one training-visible component). Parentheses show changes from Base; DEG-like metrics use percentage points. Secondary metrics continue on the next page.

Model	**Base**	**Global transfer**	**Local transfer**		**Global + Local**		**Auxiliary ablation**
		Global	Local (Pert.-KNN)	Local (Δpred-KNN)	G+L (Pert.-KNN)	G+L (Δpred-KNN)	G+L (Hybrid)
DEG@50 (primary)							
BiolordStar	0.560	**0.576** (+1.6)	**0.574** (+1.4)	**0.576** (+1.6)	**0.581** (+2.1)	**0.578** (+1.8)	**0.580** (+2.0)
CPA	0.529	**0.530** (+0.1)	**0.640** (+11.1)	**0.557** (+2.7)	**0.640** (+11.1)	**0.556** (+2.7)	**0.636** (+10.7)
DecoderOnly	0.604	**0.634** (+3.0)	**0.637** (+3.3)	**0.637** (+3.3)	**0.634** (+3.0)	**0.634** (+3.0)	**0.636** (+3.2)
GEARS	0.471	**0.516** (+4.6)	**0.544** (+7.3)	**0.558** (+8.7)	**0.554** (+8.4)	**0.558** (+8.7)	**0.566** (+9.5)
GenePert	0.623	**0.634** (+1.1)	**0.636** (+1.3)	**0.635** (+1.2)	**0.636** (+1.3)	**0.635** (+1.2)	**0.636** (+1.3)
LatentAdditive	0.610	**0.613** (+0.3)	**0.618** (+0.7)	**0.618** (+0.7)	**0.618** (+0.8)	**0.618** (+0.7)	**0.618** (+0.7)
LinearAdditive	0.554	**0.587** (+3.3)	**0.604** (+5.0)	**0.603** (+4.9)	**0.603** (+4.9)	**0.606** (+5.2)	**0.610** (+5.6)
SparseAdditiveVAE	0.627	**0.628** (+0.1)	**0.638** (+1.1)	**0.631** (+0.4)	**0.638** (+1.1)	**0.631** (+0.4)	**0.633** (+0.6)
PRNet	0.612	0.608 (-0.4)	**0.631** (+1.9)	0.609 (-0.3)	**0.626** (+1.3)	**0.619** (+0.7)	**0.626** (+1.4)
scLAMBDA	0.557	**0.560** (+0.2)	**0.626** (+6.9)	0.555 (-0.2)	**0.626** (+6.9)	0.555 (-0.2)	**0.610** (+5.3)
Mean	0.575	**0.589** (+1.4)	**0.615** (+4.0)	**0.598** (+2.3)	**0.616** (+4.1)	**0.599** (+2.4)	**0.615** (+4.0)
Median	0.582	**0.598** (+1.6)	**0.628** (+4.6)	**0.606** (+2.4)	**0.626** (+4.4)	**0.612** (+3.0)	**0.622** (+4.0)
Win rate	–	9/10	10/10	8/10	10/10	9/10	10/10

Table S3 (continued).

Model	Base	Global transfer	Local transfer		Global + Local		Auxiliary ablation
		Global	Local (Pert.-KNN)	Local (Δpred-KNN)	G+L (Pert.-KNN)	G+L (Δpred-KNN)	G+L (Hybrid)
Effect-space cosine							
BiolordStar	0.655	**0.662** (+0.007)	**0.673** (+0.018)	**0.661** (+0.006)	**0.671** (+0.016)	**0.661** (+0.006)	**0.663** (+0.008)
CPA	0.656	**0.656** (+0.000)	**0.749** (+0.093)	0.655 (-0.001)	**0.749** (+0.093)	0.654 (-0.002)	**0.727** (+0.071)
DecoderOnly	0.730	**0.741** (+0.011)	**0.745** (+0.015)	**0.744** (+0.014)	**0.744** (+0.014)	**0.744** (+0.014)	**0.747** (+0.017)
GEARS	0.509	**0.542** (+0.034)	**0.589** (+0.081)	**0.614** (+0.105)	**0.594** (+0.085)	**0.614** (+0.105)	**0.629** (+0.120)
GenePert	0.709	**0.738** (+0.030)	**0.743** (+0.034)	**0.741** (+0.033)	**0.743** (+0.034)	**0.741** (+0.033)	**0.742** (+0.034)
LatentAdditive	0.717	**0.723** (+0.007)	**0.725** (+0.008)	**0.724** (+0.008)	**0.726** (+0.009)	**0.724** (+0.008)	**0.724** (+0.008)
LinearAdditive	0.672	**0.696** (+0.024)	**0.721** (+0.049)	**0.721** (+0.049)	**0.721** (+0.049)	**0.723** (+0.051)	**0.728** (+0.056)
SparseAdditiveVAE	0.734	**0.735** (+0.001)	**0.739** (+0.005)	**0.736** (+0.002)	**0.738** (+0.004)	**0.736** (+0.002)	**0.736** (+0.002)
PRNet	0.698	**0.711** (+0.013)	**0.736** (+0.038)	**0.719** (+0.021)	**0.732** (+0.034)	**0.724** (+0.026)	**0.732** (+0.034)
scLAMBDA	0.671	**0.672** (+0.001)	**0.737** (+0.066)	**0.674** (+0.003)	**0.737** (+0.066)	**0.673** (+0.002)	**0.713** (+0.041)
Mean	0.675	**0.688** (+0.013)	**0.716** (+0.041)	**0.699** (+0.024)	**0.715** (+0.040)	**0.699** (+0.024)	**0.714** (+0.039)
Median	0.685	**0.704** (+0.019)	**0.737** (+0.052)	**0.720** (+0.035)	**0.735** (+0.050)	**0.724** (+0.039)	**0.727** (+0.042)
Win rate	–	10/10	10/10	9/10	10/10	9/10	10/10
sign-aware DEG@50							
BiolordStar	0.534	**0.549** (+1.5)	**0.550** (+1.6)	**0.550** (+1.5)	**0.556** (+2.1)	**0.551** (+1.7)	**0.553** (+1.9)
CPA	0.510	**0.511** (+0.1)	**0.620** (+11.1)	**0.527** (+1.8)	**0.620** (+11.1)	**0.527** (+1.8)	**0.617** (+10.7)
DecoderOnly	0.587	**0.615** (+2.8)	**0.619** (+3.2)	**0.618** (+3.1)	**0.616** (+2.9)	**0.617** (+3.0)	**0.618** (+3.1)
GEARS	0.454	**0.495** (+4.1)	**0.521** (+6.7)	**0.534** (+8.0)	**0.531** (+7.7)	**0.534** (+8.0)	**0.542** (+8.8)
GenePert	0.607	**0.620** (+1.3)	**0.621** (+1.4)	**0.621** (+1.4)	**0.622** (+1.5)	**0.620** (+1.3)	**0.621** (+1.5)
LatentAdditive	0.587	**0.591** (+0.5)	**0.594** (+0.8)	**0.596** (+0.9)	**0.597** (+1.1)	**0.596** (+0.9)	**0.596** (+1.0)
LinearAdditive	0.537	**0.570** (+3.3)	**0.585** (+4.8)	**0.587** (+5.0)	**0.585** (+4.8)	**0.589** (+5.2)	**0.593** (+5.6)
SparseAdditiveVAE	0.609	**0.610** (+0.1)	**0.617** (+0.8)	**0.612** (+0.3)	**0.617** (+0.8)	**0.612** (+0.3)	**0.613** (+0.5)
PRNet	0.591	0.590 (-0.1)	**0.612** (+2.1)	0.591 (+0.0)	**0.607** (+1.5)	**0.600** (+0.9)	**0.608** (+1.7)
scLAMBDA	0.537	**0.540** (+0.3)	**0.606** (+6.9)	0.535 (-0.2)	**0.606** (+6.9)	0.535 (-0.1)	**0.587** (+5.0)
Mean	0.555	**0.569** (+1.4)	**0.595** (+3.9)	**0.577** (+2.2)	**0.596** (+4.0)	**0.578** (+2.3)	**0.595** (+4.0)
Median	0.562	**0.580** (+1.8)	**0.609** (+4.7)	**0.589** (+2.7)	**0.606** (+4.4)	**0.592** (+3.0)	**0.602** (+4.0)
Win rate	–	9/10	10/10	8/10	10/10	9/10	10/10

Table S4: Secondary metrics for unseen single-gene perturbations in Replogle K562. Parentheses show changes from Base; DEG-like metrics use percentage points. Hybrid is an appendix-only ablation.

Model	Base	Global transfer	Local transfer		Global + Local		Auxiliary ablation
		Global	Local (Pert.-KNN)	Local (Δpred-KNN)	G+L (Pert.-KNN)	G+L (Δpred-KNN)	G+L (Hybrid)
logFC cosine							
BiolordStar	0.398	0.396 (-0.001)	0.381 (-0.017)	0.382 (-0.015)	0.388 (-0.010)	0.378 (-0.020)	0.385 (-0.012)
CPA	0.316	**0.397 (+0.081)**	**0.369 (+0.053)**	**0.366 (+0.050)**	**0.391 (+0.075)**	**0.366 (+0.050)**	**0.387 (+0.071)**
DecoderOnly	0.385	**0.394 (+0.009)**	**0.395 (+0.010)**	**0.393 (+0.008)**	**0.395 (+0.010)**	**0.393 (+0.008)**	**0.393 (+0.008)**
GEARS	0.336	0.335 (-0.002)	**0.342 (+0.006)**	**0.355 (+0.018)**	**0.344 (+0.007)**	**0.351 (+0.014)**	**0.349 (+0.013)**
GenePert	0.269	**0.393 (+0.124)**	**0.386 (+0.117)**	**0.392 (+0.123)**	**0.393 (+0.124)**	**0.393 (+0.124)**	**0.393 (+0.124)**
LatentAdditive	0.425	**0.431 (+0.006)**	0.415 (-0.010)	0.410 (-0.015)	**0.425 (+0.001)**	0.410 (-0.015)	0.415 (-0.009)
LinearAdditive	0.313	**0.387 (+0.074)**	**0.377 (+0.064)**	**0.385 (+0.072)**	**0.391 (+0.078)**	**0.385 (+0.072)**	**0.379 (+0.067)**
SparseAdditiveVAE	0.391	**0.395 (+0.004)**	**0.402 (+0.011)**	**0.395 (+0.004)**	**0.402 (+0.011)**	**0.395 (+0.004)**	0.374 (-0.017)
PRNet	0.412	**0.416 (+0.004)**	0.412 (-0.000)	0.405 (-0.006)	**0.416 (+0.004)**	0.410 (-0.002)	**0.414 (+0.002)**
scLAMBDA	0.387	**0.399 (+0.012)**	0.387 (+0.000)	0.384 (-0.002)	**0.387 (+0.001)**	0.384 (-0.002)	0.329 (-0.058)
Mean	0.363	**0.394 (+0.031)**	**0.387 (+0.023)**	**0.387 (+0.024)**	**0.393 (+0.030)**	**0.386 (+0.023)**	**0.382 (+0.019)**
Median	0.386	**0.396 (+0.010)**	**0.387 (+0.001)**	**0.389 (+0.003)**	**0.392 (+0.007)**	**0.389 (+0.003)**	0.386 (+0.000)
Win rate	–	8/10	7/10	6/10	9/10	6/10	6/10
sign-aware DEG@50							
BiolordStar	0.246	**0.259 (+1.3)**	**0.254 (+0.8)**	**0.255 (+0.9)**	**0.257 (+1.1)**	**0.250 (+0.4)**	**0.252 (+0.6)**
CPA	0.205	**0.236 (+3.1)**	**0.240 (+3.6)**	**0.223 (+1.8)**	**0.249 (+4.4)**	**0.223 (+1.8)**	**0.244 (+4.0)**
DecoderOnly	0.246	**0.258 (+1.2)**	**0.258 (+1.2)**	**0.261 (+1.5)**	**0.259 (+1.3)**	**0.261 (+1.5)**	**0.260 (+1.4)**
GEARS	0.216	**0.217 (+0.2)**	**0.226 (+1.0)**	**0.244 (+2.9)**	**0.225 (+1.0)**	**0.243 (+2.7)**	**0.246 (+3.0)**
GenePert	0.120	**0.256 (+13.6)**	**0.246 (+12.6)**	**0.255 (+13.5)**	**0.254 (+13.4)**	**0.255 (+13.5)**	**0.255 (+13.5)**
LatentAdditive	0.260	**0.262 (+0.2)**	**0.269 (+0.9)**	**0.261 (+0.1)**	**0.271 (+1.0)**	**0.261 (+0.1)**	0.260 (-0.1)
LinearAdditive	0.189	**0.228 (+3.9)**	**0.224 (+3.5)**	**0.246 (+5.7)**	**0.231 (+4.2)**	**0.246 (+5.7)**	**0.235 (+4.6)**
SparseAdditiveVAE	0.220	**0.224 (+0.3)**	**0.240 (+2.0)**	**0.237 (+1.7)**	**0.234 (+1.4)**	**0.237 (+1.7)**	**0.239 (+1.9)**
PRNet	0.257	0.258 (+0.0)	**0.258 (+0.1)**	0.255 (-0.3)	0.257 (-0.0)	0.257 (-0.1)	0.256 (-0.2)
scLAMBDA	0.235	**0.238 (+0.3)**	**0.246 (+1.1)**	**0.244 (+0.8)**	**0.247 (+1.1)**	**0.244 (+0.8)**	0.226 (-1.0)
Mean	0.219	**0.243 (+2.4)**	**0.246 (+2.7)**	**0.248 (+2.9)**	**0.248 (+2.9)**	**0.248 (+2.8)**	**0.247 (+2.8)**
Median	0.228	**0.247 (+1.9)**	**0.246 (+1.8)**	**0.250 (+2.2)**	**0.252 (+2.4)**	**0.248 (+2.0)**	**0.249 (+2.1)**
Win rate	–	10/10	10/10	9/10	9/10	9/10	7/10

Table S5: Secondary metrics for unseen compounds in the shared SciPlex3 K562 setting. Parentheses show changes from Base; DEG-like metrics use percentage points. Hybrid is an appendix-only ablation.

Model	Base	Global transfer	Local transfer		Global + Local		Auxiliary ablation
		Global	Local (Pert.-KNN)	Local (Δpred-KNN)	G+L (Pert.-KNN)	G+L (Δpred-KNN)	G+L (Hybrid)
logFC cosine							
BiolordStar	0.301	**0.363** (+0.063)	**0.344** (+0.044)	**0.347** (+0.046)	**0.351** (+0.050)	**0.359** (+0.058)	**0.360** (+0.059)
CPA	0.340	**0.359** (+0.019)	**0.362** (+0.021)	**0.354** (+0.013)	**0.363** (+0.023)	**0.367** (+0.026)	**0.361** (+0.020)
DecoderOnly	0.229	**0.298** (+0.068)	**0.336** (+0.107)	**0.314** (+0.084)	**0.338** (+0.109)	**0.314** (+0.084)	**0.314** (+0.084)
GEARS	0.093	**0.169** (+0.076)	**0.278** (+0.185)	0.035 (-0.058)	**0.262** (+0.169)	**0.169** (+0.076)	**0.169** (+0.075)
GenePert	0.223	**0.227** (+0.004)	**0.296** (+0.073)	**0.237** (+0.014)	**0.296** (+0.073)	**0.237** (+0.014)	**0.294** (+0.071)
LatentAdditive	0.337	**0.376** (+0.040)	**0.356** (+0.019)	0.337 (+0.000)	**0.355** (+0.018)	**0.352** (+0.016)	**0.349** (+0.013)
LinearAdditive	0.241	**0.257** (+0.016)	**0.301** (+0.061)	**0.279** (+0.038)	**0.295** (+0.054)	**0.279** (+0.038)	**0.282** (+0.042)
SparseAdditiveVAE	0.205	**0.230** (+0.026)	**0.320** (+0.116)	**0.270** (+0.065)	**0.312** (+0.108)	**0.270** (+0.065)	**0.312** (+0.107)
PRNet	0.146	**0.207** (+0.061)	**0.251** (+0.106)	**0.276** (+0.131)	**0.251** (+0.106)	**0.276** (+0.131)	**0.323** (+0.178)
scLAMBDA	0.322	**0.371** (+0.048)	**0.356** (+0.033)	**0.345** (+0.023)	**0.372** (+0.050)	**0.367** (+0.045)	**0.377** (+0.055)
Mean	0.244	**0.286** (+0.042)	**0.320** (+0.076)	**0.279** (+0.036)	**0.320** (+0.076)	**0.299** (+0.055)	**0.314** (+0.070)
Median	0.235	**0.277** (+0.042)	**0.328** (+0.093)	**0.296** (+0.061)	**0.325** (+0.090)	**0.296** (+0.061)	**0.318** (+0.083)
Win rate	–	10/10	10/10	8/10	10/10	10/10	10/10
sign-aware DEG@50							
BiolordStar	0.316	**0.341** (+2.4)	**0.352** (+3.6)	**0.350** (+3.4)	**0.354** (+3.8)	**0.351** (+3.4)	**0.352** (+3.5)
CPA	0.348	**0.356** (+0.8)	**0.363** (+1.5)	**0.360** (+1.2)	**0.363** (+1.5)	**0.366** (+1.8)	**0.362** (+1.4)
DecoderOnly	0.241	**0.264** (+2.4)	**0.316** (+7.5)	**0.302** (+6.1)	**0.315** (+7.4)	**0.302** (+6.1)	**0.324** (+8.4)
GEARS	0.063	**0.237** (+17.4)	**0.309** (+24.6)	**0.179** (+11.6)	**0.292** (+22.9)	**0.237** (+17.4)	**0.267** (+20.4)
GenePert	0.254	0.208 (-4.6)	**0.286** (+3.2)	**0.277** (+2.3)	**0.286** (+3.2)	**0.277** (+2.3)	**0.295** (+4.1)
LatentAdditive	0.346	**0.355** (+0.9)	**0.350** (+0.4)	0.346 (+0.0)	**0.350** (+0.4)	**0.352** (+0.6)	**0.350** (+0.4)
LinearAdditive	0.283	0.279 (-0.4)	**0.333** (+5.0)	**0.303** (+2.1)	**0.323** (+4.0)	**0.303** (+2.1)	**0.305** (+2.2)
SparseAdditiveVAE	0.235	**0.255** (+2.1)	**0.322** (+8.7)	**0.282** (+4.7)	**0.316** (+8.1)	**0.282** (+4.7)	**0.328** (+9.4)
PRNet	0.179	**0.256** (+7.7)	**0.284** (+10.5)	**0.308** (+12.9)	**0.284** (+10.5)	**0.308** (+12.9)	**0.316** (+13.7)
scLAMBDA	0.321	**0.347** (+2.7)	**0.359** (+3.8)	**0.336** (+1.5)	**0.367** (+4.6)	**0.348** (+2.8)	**0.355** (+3.5)
Mean	0.258	**0.290** (+3.1)	**0.327** (+6.9)	**0.304** (+4.6)	**0.325** (+6.6)	**0.312** (+5.4)	**0.325** (+6.7)
Median	0.268	**0.272** (+0.3)	**0.327** (+5.9)	**0.306** (+3.7)	**0.319** (+5.1)	**0.306** (+3.7)	**0.326** (+5.8)
Win rate	–	8/10	10/10	9/10	10/10	10/10	10/10

Table S6: SciPlex3 results for models trained within the matched cell line. Parentheses are DEG@50 percentage-point changes from Base.

Dataset	Cell line	Setting	Model	Base	Global	Local (Δpred-KNN)	G+L (Δpred-KNN)	G+L (Hybrid)
SciPlex3	A549	Matched per-cell-line	BiolordStar	0.427	0.436 (+0.9)	0.444 (+1.7)	0.442 (+1.5)	0.445 (+1.8)
SciPlex3	A549	Matched per-cell-line	CPA	0.447	0.463 (+1.6)	0.464 (+1.7)	0.460 (+1.3)	0.486 (+3.9)
SciPlex3	A549	Matched per-cell-line	DecoderOnly	0.501	0.506 (+0.5)	0.502 (+0.1)	0.502 (+0.1)	0.501 (-0.1)
SciPlex3	A549	Matched per-cell-line	GEARS	0.094	0.379 (+28.5)	0.359 (+26.5)	0.376 (+28.2)	0.463 (+36.9)
SciPlex3	A549	Matched per-cell-line	GenePert	0.437	0.446 (+0.9)	0.494 (+5.7)	0.494 (+5.7)	0.498 (+6.1)
SciPlex3	A549	Matched per-cell-line	LatentAdditive	0.416	0.464 (+4.7)	0.462 (+4.6)	0.473 (+5.7)	0.464 (+4.8)
SciPlex3	A549	Matched per-cell-line	LinearAdditive	0.487	0.476 (-1.1)	0.474 (-1.3)	0.475 (-1.2)	0.475 (-1.2)
SciPlex3	A549	Matched per-cell-line	PRNet	0.262	0.425 (+16.3)	0.453 (+19.1)	0.453 (+19.1)	0.451 (+18.9)
SciPlex3	A549	Matched per-cell-line	SparseAdditiveVAE	0.457	0.444 (-1.3)	0.459 (+0.2)	0.464 (+0.7)	0.461 (+0.4)
SciPlex3	A549	Matched per-cell-line	scLAMBDA	0.471	0.474 (+0.3)	0.485 (+1.4)	0.485 (+1.4)	0.483 (+1.2)
SciPlex3	K562	Matched per-cell-line	BiolordStar	0.414	0.426 (+1.3)	0.432 (+1.8)	0.427 (+1.4)	0.434 (+2.0)
SciPlex3	K562	Matched per-cell-line	CPA	0.405	0.409 (+0.4)	0.414 (+0.9)	0.414 (+0.9)	0.414 (+0.8)
SciPlex3	K562	Matched per-cell-line	DecoderOnly	0.453	0.468 (+1.5)	0.464 (+1.1)	0.462 (+0.9)	0.465 (+1.2)
SciPlex3	K562	Matched per-cell-line	GEARS	0.115	0.335 (+22.0)	0.305 (+19.0)	0.335 (+22.0)	0.365 (+25.0)
SciPlex3	K562	Matched per-cell-line	GenePert	0.389	0.411 (+2.2)	0.433 (+4.4)	0.429 (+4.1)	0.432 (+4.3)
SciPlex3	K562	Matched per-cell-line	LatentAdditive	0.425	0.418 (-0.7)	0.425 (+0.0)	0.424 (-0.1)	0.422 (-0.3)
SciPlex3	K562	Matched per-cell-line	LinearAdditive	0.447	0.433 (-1.4)	0.433 (-1.4)	0.433 (-1.4)	0.435 (-1.2)
SciPlex3	K562	Matched per-cell-line	PRNet	0.247	0.414 (+16.6)	0.428 (+18.1)	0.429 (+18.2)	0.423 (+17.5)
SciPlex3	K562	Matched per-cell-line	SparseAdditiveVAE	0.405	0.403 (-0.2)	0.408 (+0.4)	0.407 (+0.2)	0.412 (+0.7)
SciPlex3	K562	Matched per-cell-line	scLAMBDA	0.393	0.423 (+3.0)	0.417 (+2.4)	0.417 (+2.4)	0.428 (+3.5)
SciPlex3	MCF7	Matched per-cell-line	BiolordStar	0.458	0.473 (+1.5)	0.457 (-0.1)	0.472 (+1.4)	0.470 (+1.2)
SciPlex3	MCF7	Matched per-cell-line	CPA	0.421	0.459 (+3.8)	0.445 (+2.4)	0.445 (+2.4)	0.448 (+2.7)
SciPlex3	MCF7	Matched per-cell-line	DecoderOnly	0.461	0.467 (+0.6)	0.467 (+0.7)	0.469 (+0.8)	0.467 (+0.6)
SciPlex3	MCF7	Matched per-cell-line	GEARS	0.061	0.371 (+31.0)	0.342 (+28.1)	0.382 (+32.2)	0.417 (+35.6)
SciPlex3	MCF7	Matched per-cell-line	GenePert	0.404	0.425 (+2.2)	0.437 (+3.3)	0.437 (+3.3)	0.437 (+3.3)
SciPlex3	MCF7	Matched per-cell-line	LatentAdditive	0.459	0.472 (+1.3)	0.467 (+0.8)	0.466 (+0.7)	0.461 (+0.2)
SciPlex3	MCF7	Matched per-cell-line	LinearAdditive	0.462	0.464 (+0.2)	0.464 (+0.2)	0.462 (-0.1)	0.462 (-0.1)
SciPlex3	MCF7	Matched per-cell-line	PRNet	0.254	0.395 (+14.1)	0.409 (+15.5)	0.423 (+16.8)	0.443 (+18.9)
SciPlex3	MCF7	Matched per-cell-line	SparseAdditiveVAE	0.436	0.442 (+0.5)	0.446 (+1.0)	0.444 (+0.7)	0.453 (+1.6)
SciPlex3	MCF7	Matched per-cell-line	scLAMBDA	0.452	0.448 (-0.4)	0.456 (+0.4)	0.452 (-0.1)	0.451 (-0.1)

Table S7: SciPlex3 results for shared models trained on pooled cell lines. Parentheses are DEG@50 percentage-point changes from Base.

Dataset	Cell line	Setting	Model	Base	Global	Local (Δpred-KNN)	G+L (Δpred-KNN)	G+L (Hybrid)
SciPlex3	A549	Shared	BiolordStar	0.466	0.385 (-8.1)	0.392 (-7.4)	0.411 (-5.5)	0.416 (-5.0)
SciPlex3	A549	Shared	CPA	0.460	0.437 (-2.3)	0.453 (-0.7)	0.444 (-1.6)	0.444 (-1.6)
SciPlex3	A549	Shared	DecoderOnly	0.393	0.379 (-1.4)	0.311 (-8.2)	0.311 (-8.2)	0.370 (-2.3)
SciPlex3	A549	Shared	GEARS	0.095	0.289 (+19.4)	0.273 (+17.7)	0.289 (+19.4)	0.104 (+0.8)
SciPlex3	A549	Shared	GenePert	0.435	0.371 (-6.5)	0.320 (-11.5)	0.320 (-11.5)	0.373 (-6.3)
SciPlex3	A549	Shared	LatentAdditive	0.480	0.467 (-1.3)	0.480 (+0.0)	0.478 (-0.2)	0.479 (-0.1)
SciPlex3	A549	Shared	LinearAdditive	0.424	0.378 (-4.6)	0.283 (-14.2)	0.283 (-14.2)	0.351 (-7.3)
SciPlex3	A549	Shared	PRNet	0.330	0.283 (-4.7)	0.269 (-6.1)	0.269 (-6.1)	0.251 (-7.9)
SciPlex3	A549	Shared	SparseAdditiveVAE	0.413	0.376 (-3.7)	0.369 (-4.4)	0.369 (-4.4)	0.386 (-2.7)
SciPlex3	A549	Shared	scLAMBDA	0.452	0.412 (-3.9)	0.351 (-10.1)	0.386 (-6.6)	0.406 (-4.6)
SciPlex3	K562	Shared	BiolordStar	0.403	0.409 (+0.6)	0.434 (+3.1)	0.423 (+1.9)	0.426 (+2.3)
SciPlex3	K562	Shared	CPA	0.421	0.424 (+0.3)	0.427 (+0.7)	0.434 (+1.3)	0.427 (+0.7)
SciPlex3	K562	Shared	DecoderOnly	0.300	0.320 (+2.0)	0.372 (+7.2)	0.372 (+7.2)	0.389 (+8.9)
SciPlex3	K562	Shared	GEARS	0.084	0.345 (+26.1)	0.324 (+24.1)	0.345 (+26.1)	0.402 (+31.8)
SciPlex3	K562	Shared	GenePert	0.333	0.272 (-6.1)	0.368 (+3.5)	0.368 (+3.5)	0.366 (+3.3)
SciPlex3	K562	Shared	LatentAdditive	0.427	0.415 (-1.2)	0.427 (+0.0)	0.431 (+0.4)	0.430 (+0.3)
SciPlex3	K562	Shared	LinearAdditive	0.361	0.354 (-0.7)	0.383 (+2.2)	0.383 (+2.2)	0.375 (+1.4)
SciPlex3	K562	Shared	PRNet	0.262	0.360 (+9.8)	0.401 (+13.9)	0.401 (+13.9)	0.394 (+13.2)
SciPlex3	K562	Shared	SparseAdditiveVAE	0.305	0.329 (+2.4)	0.356 (+5.1)	0.356 (+5.1)	0.399 (+9.4)
SciPlex3	K562	Shared	scLAMBDA	0.392	0.410 (+1.8)	0.406 (+1.4)	0.412 (+1.9)	0.415 (+2.3)
SciPlex3	MCF7	Shared	BiolordStar	0.447	0.393 (-5.4)	0.413 (-3.4)	0.414 (-3.3)	0.415 (-3.2)
SciPlex3	MCF7	Shared	CPA	0.448	0.438 (-0.9)	0.453 (+0.5)	0.446 (-0.2)	0.449 (+0.1)
SciPlex3	MCF7	Shared	DecoderOnly	0.438	0.409 (-2.9)	0.354 (-8.5)	0.354 (-8.5)	0.371 (-6.7)
SciPlex3	MCF7	Shared	GEARS	0.059	0.223 (+16.4)	0.197 (+13.8)	0.223 (+16.4)	0.074 (+1.5)
SciPlex3	MCF7	Shared	GenePert	0.438	0.404 (-3.4)	0.363 (-7.5)	0.363 (-7.5)	0.369 (-6.9)
SciPlex3	MCF7	Shared	LatentAdditive	0.452	0.442 (-1.0)	0.452 (+0.0)	0.448 (-0.3)	0.449 (-0.3)
SciPlex3	MCF7	Shared	LinearAdditive	0.432	0.404 (-2.8)	0.323 (-10.8)	0.323 (-10.8)	0.377 (-5.4)
SciPlex3	MCF7	Shared	PRNet	0.259	0.228 (-3.1)	0.210 (-4.9)	0.210 (-4.9)	0.205 (-5.4)
SciPlex3	MCF7	Shared	SparseAdditiveVAE	0.435	0.401 (-3.4)	0.399 (-3.5)	0.399 (-3.5)	0.394 (-4.1)
SciPlex3	MCF7	Shared	scLAMBDA	0.473	0.426 (-4.7)	0.393 (-8.0)	0.419 (-5.4)	0.426 (-4.7)

E MATCHED NULL CONTROLS AND STATISTICAL TESTING

Inference and uncertainty. Each model×context configuration is one independent unit. We average all per-perturbation metrics within that unit before comparison, so individual perturbations are not treated as independent replicates. The full panel contains 80 units (8 dataset×context settings × 10 models); the strict label-disjoint primary set contains 70 units and excludes the Norman2019 `combo_unseen0` diagnostic, whose training metadata contain complete combination labels. Aggregate confidence intervals use a two-stage hierarchical bootstrap that first resamples context clusters (dataset, context, and model-source strata) and then model units within selected contexts (10,000 replicates, seed 42; percentile 95% intervals). Comparisons against nulls are paired at the unit level: the per-unit lift (observed − null mean) is computed first and then bootstrapped. Leave-one-model-out (LOMO) and leave-one-context-out (LOCO) sensitivities are reported in Table S8.

Cluster-sensitivity detail. Table S8 summarizes leave-one-model-out (LOMO) and leave-one-context-out (LOCO) analyses by their mean and full range. This compact form shows whether a conclusion depends on one model or context; individual deletion rows are retained in the source below for auditability but omitted from the compiled appendix because they repeat the same pattern.

Table S8: Cluster-deletion sensitivity. Means and ranges summarize the result after omitting one model (LOMO) or one context (LOCO) at a time.

Endpoint	Omission	n	Mean	Range
Alignment lift vs expression null	LOMO	10	0.1888	[0.1594, 0.2044]
	LOCO	7	0.1888	[0.1344, 0.2392]
Alignment lift vs effect null	LOMO	10	0.1706	[0.1440, 0.1847]
	LOCO	7	0.1705	[0.1166, 0.2203]
Missing-gene lift vs DE-frequency null	LOMO	10	0.2499	[0.2188, 0.2682]
	LOCO	7	0.2499	[0.2215, 0.2709]
Global utility, shared contexts	LOMO	10	0.0090	[-0.0044, 0.0121]
	LOCO	5	0.0090	[0.0025, 0.0155]
Global utility, shared A549/MCF7	LOMO	10	-0.0142	[-0.0356, -0.0082]
	LOCO	2	-0.0142	[-0.0170, -0.0114]
Global utility, matched cell line	LOMO	10	0.0503	[0.0257, 0.0567]
	LOCO	3	0.0503	[0.0481, 0.0531]

Alignment null (Figure 2b, left). Fingerprint values are shuffled within joint strata of three training-only gene covariates: mean expression, expression variance, and detection rate (fraction of training cells with non-zero expression), each discretized into 4 equal-frequency rank bins (joint strata, at most 4^3). Twenty shuffles (seed 42) are performed per unit. The endpoint is the cosine similarity between the (shuffled) training fingerprint and each held-out true gap, averaged within the unit; the null is the mean over shuffles. On the strict 70 units, the fingerprint scores 0.285 [95% CI $0.112, 0.450$] against 0.096 [$0.057, 0.141$] for the null; mean lift 0.189 [$0.037, 0.340$], positive in 87% of units.

Missing-gene null (Figure 2b, middle). Replacement gene sets match five training-only covariates: the three above plus the mean training gap magnitude $\overline{|g|}$ and the training top-50 DEG frequency (the fraction of training perturbations for which the gene is among the top-50 by $|\Delta^{\mathrm{real}}|$). Each covariate is discretized into 3 equal-frequency rank bins; a replacement gene is sampled uniformly without replacement from the same joint stratum (excluding genes of the true top-50 set; if a stratum is exhausted, sampling falls back to any remaining gene). Twenty null sets (seed 42) are evaluated per unit. The endpoint is top-50 overlap, $|T_{50}(F) \cap T_{50}(g_p)|/50$, for held-out true gaps g_p. The fingerprint scores 0.325 [$0.247, 0.408$] against 0.075 [$0.065, 0.085$] for the null; mean lift 0.250 [$0.174, 0.332$], positive in 93% of units.

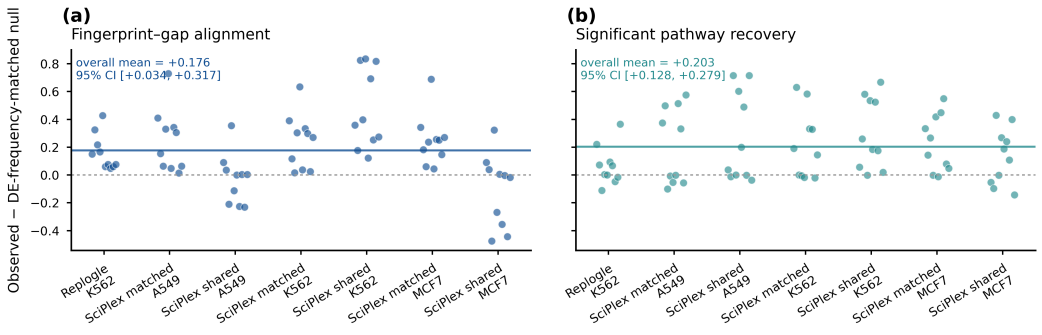

Figure S1: **DE-frequency-matched null controls.** Each point is one model×context unit from the strict complete-label-disjoint analysis. The vertical axis shows the observed endpoint minus the mean endpoint from a null that additionally matches the training-set frequency with which a gene appears among the top-50 response genes, together with expression mean, expression variance, detection rate, and mean training gap magnitude. (a) Fingerprint–held-out-gap alignment. (b) Recovery of pathways significant for the held-out true gap. Horizontal lines and intervals show the aggregate mean and hierarchical bootstrap 95% interval; the null matching uses training data only.

Pathway null (Figure 2b, right). The fingerprint top genes are replaced by expression-matched genes (the three covariates above, 4 bins), and the identical pathway analysis (Appendix F) is rerun; five null draws per unit. The endpoint is the significant-pathway recovery rate defined in Appendix F. The fingerprint scores 0.306 $[0.208, 0.410]$ against 0.056 $[0.035, 0.081]$ for the null. In the main-figure null, 50 matched replacement genes are scored against the true top-100, while the observed endpoint uses the fingerprint top-100 against the true top-100; the supplementary DE-frequency-matched variant uses a symmetric top-100 construction (five covariates, three bins) and gives the same qualitative ordering (recovery 0.306 vs. null 0.103).

DE-frequency-matched variants (Figure S1). The supplementary alignment and pathway endpoints match all five covariates (expression mean, variance, detection rate, mean training gap magnitude, and training DEG frequency; 3 bins each) with 20 draws. Strict-70 lifts: alignment 0.285 vs. 0.109; pathway recovery 0.306 vs. 0.103. All matching uses training data only; held-out truth enters only the final endpoint.

F Pathway analysis

Gene sets. We use the MSigDB Hallmark 2020 collection (50 pathways, HGNC symbols, 32–200 genes per pathway). Pathways are mapped to the 2,000-gene evaluation universe by symbol, and only pathways with at least five mapped genes are retained (44 for Norman2019, 47 for Replogle, 35 for SciPlex3).

Enrichment test. For a gene set S of size n (the top-100 genes by $|F|$ for fingerprints, or by $|g_p|$ for held-out gaps) and a pathway P of size K with overlap $k = |S \cap P|$, significance is the one-sided hypergeometric tail $p = \Pr[X \geq k]$ with $X \sim \mathrm{Hypergeom}(N{=}2000, K, n)$, and the effect size is the overlap-based score $\mathrm{NES} = \log_{10} \frac{k+0.1}{\mathbb{E}[k]+0.1}$ with $\mathbb{E}[k] = Kn/N$; this is an overlap score, not a GSEA NES. Pathways with zero overlap are not tested. p-values are Benjamini–Hochberg corrected within each enrichment run over the tested family (pathways with non-zero overlap for the population analyses; all eligible pathways with $5 \leq |P| \leq 500$ for the case study), and significance means $q < 0.05$.

Significant-pathway recovery endpoint. For each model×context unit, let $\mathcal{S}_{\mathrm{true}}$ be the pathways significant for the held-out true gap and $\mathcal{S}_{\mathrm{fp}}$ those significant for the training fingerprint. The endpoint is $|\mathcal{S}_{\mathrm{true}} \cap \mathcal{S}_{\mathrm{fp}}| \,/\, |\mathcal{S}_{\mathrm{true}}|$ (defined as 0 when $\mathcal{S}_{\mathrm{true}} = \emptyset$), i.e. the fraction of truly significant pathways that the training fingerprint already flags. The case-study criterion is stricter: a pathway is

reported as recovered when it is significant for both the truth and the corrected prediction ($q < 0.05$), not significant for the base prediction (or truth−base NES ≥ 0.10), and its corrected NES is closer to the truth than the base NES.

G FINGERPRINT STRUCTURE ANALYSES

This section gives the exact construction of Figure 3 of the main text (fingerprint structure). Panel (a) separates measured from predicted responses; panels (b)–(d) attribute an error pattern to the model that produced it.

Common representation. Cells were averaged to pseudo-bulk responses in the 2,000-gene VCBench space. For perturbation p,

$$\Delta_p^{\text{real}} = x_p^{\text{real}} - x_{\text{ctrl}}, \qquad \Delta_p^{\text{pred}} = x_p^{\text{pred}} - x_{\text{ctrl}}, \qquad g_p = \Delta_p^{\text{real}} - \Delta_p^{\text{pred}}. \tag{12}$$

A positive gap denotes under-prediction at that gene. A training fingerprint is the mean gap over training perturbations (Eq. 2 of the main text).

Response-level summary statistics. The six response-level statistics used in Figure 3a,c are, for an effect vector Δ with control mean x_{ctrl} and expression profile $x = \Delta + x_{\text{ctrl}}$: the expression sum $\sum_g x_g$ (library size), the detected-gene count $\sum_g \mathbb{1}[x_g > 0]$, the zero fraction $\frac{1}{G}\sum_g \mathbb{1}[x_g = 0]$, the expression mean, the expression standard deviation, and the effect norm $\|\Delta\|_2$. Paired differences subtract the predicted from the measured statistic of the same perturbation.

Panel (a): real versus predicted responses. Each model–context configuration contributed one measured row and one predicted row per perturbation. The two rows from a perturbation remained in the same split. Five random 80:20 perturbation splits were evaluated with a training-only StandardScaler and L_2-regularized logistic regression. The score is AUROC for distinguishing measured from predicted rows. The three inputs were the six response-level summaries, the complete effect vector, and a fold-safe residualized response (below). Mean AUROC was 0.994, 0.956, and 0.581, respectively. This is a binary real–prediction test, not a model-identity test.

Fold-safe residualization. Within each train split only, every gene of the pseudo-bulk response was regressed on the six response-level summaries (least squares with intercept); the fitted coefficients were then applied to the held-out rows of the same split. No held-out value entered any fit. The residualized response retains whatever real–predicted structure is not linearly explained by the coarse statistics.

Panel (b): reproducibility across seeds. CPA, GEARS, GenePert, and SparseAdditiveVAE were trained with seeds 42, 43, and 44 in five contexts (Norman2019, Replogle K562, and SciPlex3 K562/A549/MCF7 per-cell-line), using the configurations of Appendix C.4. For each model and seed, the fingerprint was the mean training gap. Cosine similarity was compared for same-model fingerprint pairs across seeds (n=60) and different-model pairs within a context (n=270). The mean similarities were 0.618 and 0.008 (one-sided Mann–Whitney pooled $p = 2.1 \times 10^{-27}$; each context individually $p \leq 6.4 \times 10^{-6}$). For the inset, one seed was held out; the other two seeds formed one centroid per model, and held-out gaps were assigned to the nearest centroid. The inset reports four-class balanced accuracy (chance $= 0.25$); per-context means ranged from 0.448 to 0.825.

Seed-held-out diagnostics. Table S9 provides the per-held-out-seed class-recall and confusion diagnostics under the seed robustness experiment, together with the corresponding model- and seed-level variance fractions. The numbers of evaluated rows are 172 for Norman2019, 400 for Replogle, and 152 for each SciPlex3 cell line.

Attribution protocol for panels (c) and (d). Panels (c) and (d) use row-wise L_2 normalization, model centroids computed from each model's own training rows, and nearest-centroid assignment (mean squared distance) of held-out rows. The endpoint is balanced accuracy. Panels (c) and (d) both use the seven complete-label-disjoint contexts (Replogle K562; SciPlex3 K562/A549/MCF7, each shared and per-cell-line) with ten models per context, so the chance level is 0.10 in every context.

Table S9: Seed-held-out model-attribution diagnostics for the four model families retained across training seeds (chance $= 0.25$). Error rate is one minus mean class recall; variance fractions are repeated across seeds because they are estimated once per context.

Context	Held-out seed	Mean recall	Worst recall	Error rate	Model variance	Seed variance
Norman/K562	42	0.6279	0.3256	0.3721	0.7777	0.2223
Norman/K562	43	0.5523	0.3721	0.4477	0.7777	0.2223
Norman/K562	44	0.6802	0.3256	0.3198	0.7777	0.2223
Replogle/K562	42	0.4475	0.0400	0.5525	0.4759	0.5241
Replogle/K562	43	0.4975	0.3600	0.5025	0.4759	0.5241
Replogle/K562	44	0.4000	0.0200	0.6000	0.4759	0.5241
SciPlex/K562	42	0.7763	0.5263	0.2237	0.8249	0.1751
SciPlex/K562	43	0.8026	0.6316	0.1974	0.8249	0.1751
SciPlex/K562	44	0.8553	0.7105	0.1447	0.8249	0.1751
SciPlex/A549	42	0.8092	0.6316	0.1908	0.9079	0.0921
SciPlex/A549	43	0.8289	0.5789	0.1711	0.9079	0.0921
SciPlex/A549	44	0.8355	0.5789	0.1645	0.9079	0.0921
SciPlex/MCF7	42	0.7171	0.5000	0.2829	0.9490	0.0510
SciPlex/MCF7	43	0.7039	0.3684	0.2961	0.9490	0.0510
SciPlex/MCF7	44	0.7368	0.5526	0.2632	0.9490	0.0510

The Norman2019 component-visible setting is excluded because complete combination labels occur in its training metadata; Figure 2a is the only identity panel that includes it, and it applies the same ten-model protocol in all eight of its contexts (chance 0.10).

Panel (c): residualized model attribution. The input for each model–perturbation row was the error gap g. A gene-wise linear residualizer was fitted on pooled model$\times$perturbation training rows using the six paired differences between measured and predicted summary statistics, then applied to held-out rows. Mean balanced accuracy was 0.53 for raw gaps and 0.44 after residualization (chance $= 0.10$; seven contexts, ten models each). Held-out truth is needed to construct the evaluation gap and its paired covariate, but not to fit residualizer coefficients.

Panel (d): global versus gene-wise calibration. On training perturbations only, predicted effects were calibrated against measured effects using five pre-specified rules. Writing x for predicted and y for measured training effects with n training perturbations:

$$\text{Global offset:} \quad a_g = 1,\ b_g = \overline{y - x}, \tag{13}$$

$$\text{Global affine:} \quad a_g = \tfrac{n}{n+10}\,\hat{s} + \tfrac{10}{n+10}, \quad b_g = \bar{y} - a_g\bar{x}, \tag{14}$$

$$\text{Gene-wise offset:} \quad a_g = 1,\ b_g = \overline{y_g - x_g}, \tag{15}$$

$$\text{Gene-wise affine:} \quad a_g = \tfrac{n}{n+10}\,\hat{s}_g + \tfrac{10}{n+10}, \quad b_g = \bar{y}_g - a_g\bar{x}_g, \tag{16}$$

where $\hat{s}$ (resp. $\hat{s}_g$) is the least-squares slope of y on x pooled over all genes (resp. per gene), and slopes are shrunk toward one by the fixed factor $n/(n+10)$; genes with near-zero training prediction variance fall back to slope one. Residual gaps $y-(a\odot x+b)$ were used for held-out attribution. Mean balanced accuracy was 0.53, 0.51, 0.46, 0.15, and 0.15 for Raw, Global offset, Global affine, Gene-wise offset, and Gene-wise affine (chance $= 0.10$). Gene-wise offset/affine remove the training mean gap by construction; degenerate zero centroids are reported at the fixed random baseline rather than amplified by rescaling.

Aggregation. Panel (a) aggregates model–context AUROCs. The left plot in panel (b) aggregates fingerprint pairs, its inset aggregates held-out-seed scores, and panels (c) and (d) aggregate one

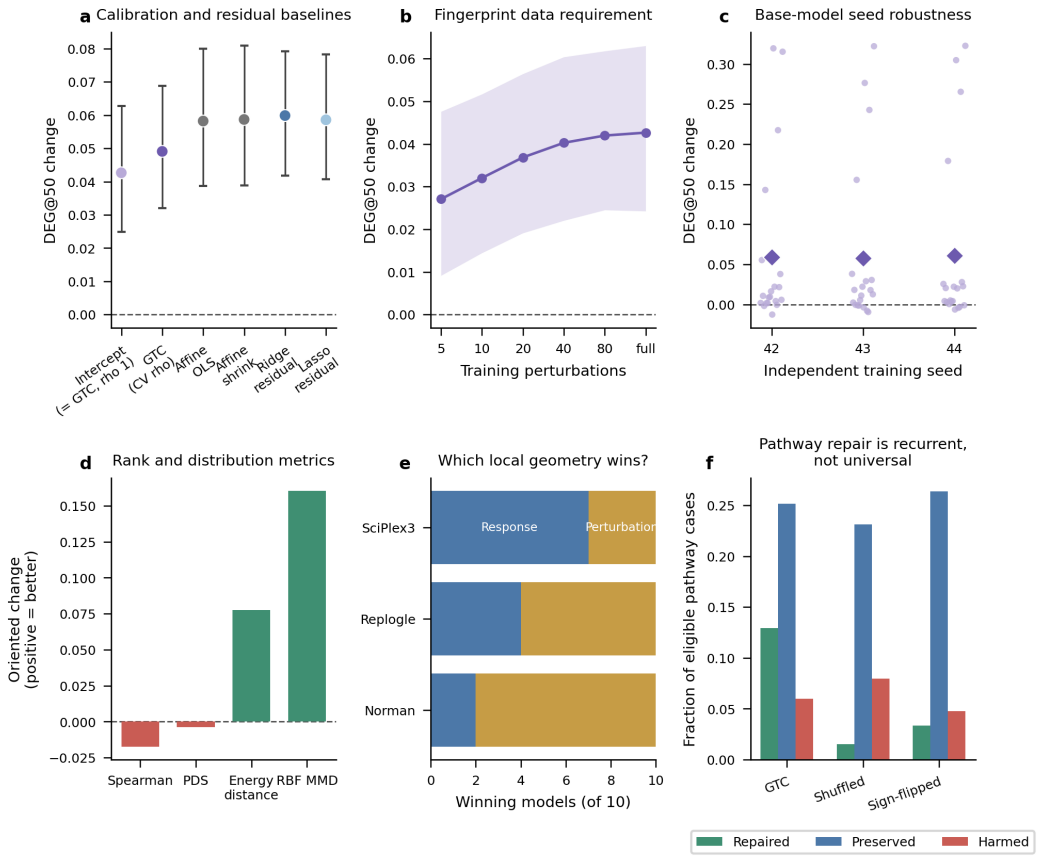

Figure S2: **Additional evaluation of GTC.** (a) Calibration and residual-learning baselines. (b) Gain versus the number of perturbations used to estimate the fingerprint. (c) Gain across independently trained model seeds. (d) Rank and distribution metrics; positive values indicate improvement. (e) Wins for the two local-neighbor representations. (f) Pathway outcomes for GTC and negative controls. Bars and shaded regions show bootstrap 95% confidence intervals.

balanced-accuracy score per context. Diamonds show arithmetic means. Error bars are percentile 95% intervals from 10,000 bootstrap resamples of the plotted units, using seed 42.

H Additional analyses

H.1 Affine calibration and residual-learning baselines

Across 80 model–context units, mean DEG@50 gains were 4.27 pp for intercept GTC, 4.91 pp for validation-selected GTC, 5.82 pp for affine calibration, 5.99 pp for Ridge, and 5.86 pp for Lasso (Figure S2a). The flexible learners added about one percentage point on average.

All additional estimators use training perturbations only and are applied unchanged to held-out perturbations. For predicted effect $x_{p,g}$ and measured effect $y_{p,g}$, the per-gene intercept baseline sets $a_g = 1$ and $b_g = \overline{y_g - x_g}$; this is exactly Global GTC at $\rho = 1$. Per-gene affine OLS estimates a_g and b_g independently by least squares. A stabilized affine variant shrinks every slope toward one by $n/(n+10)$ and recomputes $b_g = \bar{y}_g - a_g \bar{x}_g$. Genes with near-zero training prediction variance use $a_g = 1$.

Ridge fits $\Delta^{\text{pred}} \mapsto g$ and selects $\alpha \in \{10^{-4}, \ldots, 10^{4}\}$ using five repeated 80:20 splits of the training perturbations. Lasso uses a training-only PCA with at most 32 components and selects

its penalty in the same way. The global-ρ comparator selects $\rho \in \{0, 0.25, 0.5, 0.75, 1, 1.25, 1.5\}$. Selection maximizes validation DEG@50; held-out data are used only for final evaluation.

H.2 Fixed-checkpoint SciPlex3 implementation details

For the comparison in Figure 4, we generated predictions separately for K562, A549, and MCF7 from each of the ten shared checkpoints without updating the model. Checkpoint hashes were identical across the three runs. A context-matched fingerprint averages the 149 training gaps from the target cell line, whereas the balanced-pooled fingerprint gives each cell line equal weight. Each fingerprint corrects the same held-out predictions with $\rho = 1$. For model attribution, genes are standardized using training gaps only, and each held-out gap is assigned to the nearest Euclidean fingerprint centroid. Per-cell-line values are reported in Table S12.

H.3 Fingerprint data requirements

Mean DEG@50 gain rose from 2.71 pp with 5 training perturbations to 3.68 pp with 20 and 4.20 pp with 80, close to the full-data gain of 4.27 pp. Among units with a positive full-data gain, the median sample size needed to recover 90% of that gain was 40 perturbations (Figure S2b).

For each model–context unit, we drew 50 random permutations of the training perturbations (seeds 1000–1049). Within each permutation, the subsets at $n \in \{5, 10, 20, 40, 80\}$ were nested; the largest available subset used all training perturbations. We recomputed $F_{m,c}$ for every subset and applied Global GTC with the prespecified $\rho = 1$. We report held-out DEG@50 gain, cosine similarity to the full-data fingerprint, and cosine alignment with held-out gaps. Repeated subsets are averaged within a unit before uncertainty is calculated across model–context units. The n_{90} descriptor is the smallest sampled size attaining at least 90% of that unit's positive full-data gain; it is undefined when the full-data gain is non-positive.

H.4 Independently trained base-model seeds

Across 60 independently trained checkpoints, 47 improved after GTC. After averaging seeds within each architecture and context, the mean gain was 5.92 pp (95% CI 1.97–10.74 pp; Figure S2c).

We independently trained CPA, GEARS, GenePert, and SparseAdditiveVAE at seeds 42, 43, and 44 in five contexts: strict complete-label-unseen Norman2019, Replogle K562, and SciPlex3 K562, A549, and MCF7. This gives 20 architecture–context units and 60 independently trained checkpoint runs. The primary summary first averages the three seed gains within each architecture–context unit and then bootstraps these 20 units; within-unit seed standard deviations quantify checkpoint sensitivity.

H.5 Rank-based and distributional metrics

GTC reduced energy distance by 0.0777 and MMD by 0.1603, with 74/80 units improving for both metrics. Mean changes in Spearman correlation (-0.0175) and PDS (-0.0037) were slightly negative (Figure S2d).

Spearman correlation is computed between predicted and measured 2,000-gene effect vectors for each held-out perturbation. PDS is $1 - (r_p - 1)/n_{\text{test}}$, where r_p is the one-based L1-distance rank of the matched measured perturbation among all measured held-out effects queried by the predicted effect. Directly targeted genes are excluded from the query distances for genetic perturbations. Energy distance treats the held-out perturbation effects as empirical multivariate distributions and divides Euclidean distances by $\sqrt{G}$. MMD uses the biased empirical RBF MMD^2 estimator with the bandwidth set by the pooled median squared distance. Spearman and PDS are higher-is-better; energy distance and MMD are lower-is-better. Each endpoint is computed or aggregated within a model–context unit before paired analysis.

Table S10: Training-only calibration and residual-learning baselines. The reporting unit is a model–context combination. Intervals are percentile bootstrap 95% confidence intervals over reporting units. Global GTC with unit strength is algebraically identical to per-gene intercept-only calibration.

Method	n	Mean gain (pp)	95% CI (pp)	Improved
Per-gene intercept (= Global GTC, $\rho = 1$)	80	4.27	[2.49, 6.27]	60/80
Global GTC (training-CV ρ)	80	4.91	[3.20, 6.88]	67/80
Per-gene affine OLS	80	5.82	[3.87, 7.99]	61/80
Per-gene affine, slope shrinkage	80	5.87	[3.88, 8.09]	61/80
Ridge residual ($\Delta^{\mathrm{pred}} \to g$)	80	5.99	[4.19, 7.92]	65/80
Lasso residual ($\Delta^{\mathrm{pred}} \to g$)	80	5.86	[4.07, 7.83]	60/80

Table S11: Rank-based and effect-profile-distribution metrics. Improvement is raw change for Spearman and PDS and the negative raw change for the lower-is-better energy distance and MMD. Metrics are first aggregated within each model–context unit.

Correction	Metric	n	Raw change	Improvement	Improved
GTC ($\rho = 1$)	Spearman	80	-0.0288	-0.0288	36/80
GTC ($\rho = 1$)	PDS	80	-0.0057	-0.0057	32/80
GTC ($\rho = 1$)	Energy distance	80	-0.0797	0.0797	80/80
GTC ($\rho = 1$)	RBF MMD	80	-0.1832	0.1832	80/80
GTC (CV ρ)	Spearman	80	-0.0175	-0.0175	38/80
GTC (CV ρ)	PDS	80	-0.0037	-0.0037	31/80
GTC (CV ρ)	Energy distance	80	-0.0777	0.0777	74/80
GTC (CV ρ)	RBF MMD	80	-0.1603	0.1603	74/80

H.6 Response-informed versus perturbation-embedding KNN

Response-informed KNN won 13/30 comparisons: 2/10 in Norman2019, 4/10 in Replogle, and 7/10 in SciPlex3 (Figure S2e). Training-set size did not predict the winner; the most useful signal was how smoothly training errors varied within each neighbor graph.

For each of 30 model–dataset units, we compare the best validation gain from response-informed and perturbation-embedding KNN. Training-only descriptors include gap smoothness, neighbor overlap, predicted-response spread, and training-set size. A leave-one-architecture-out Ridge model tests whether these descriptors predict the better representation.

H.7 Pathway-level recurrence analysis

Across all held-out cases, GTC repaired 12.9% of missed or wrong-direction pathways, preserved 25.1%, and harmed 6.0%; pathway-score error decreased in 59.8% (Figure S2f). Repair rates were lower with shuffled (1.5%) or sign-flipped (3.4%) fingerprints.

For every held-out perturbation in every available model–context unit, we identify Hallmark pathways enriched among the measured top-100 absolute effects by a hypergeometric test with Benjamini–Hochberg control across the 50 pathways at $q < 0.05$. A pathway is detected correctly when it is significant in the prediction and the sign of its mean pathway effect agrees with the measurement. Relative to Base, a truth-significant pathway is classified as repaired, preserved, or harmed when detection changes from incorrect to correct, remains correct, or changes from correct to incorrect, respectively. We also record whether the absolute pathway-score error decreases. Gene-shuffled and sign-flipped fingerprints serve as negative controls. Rates are computed within model–context units before macro-averaging.

Table S12: SciPlex3 matched versus balanced-pooled fingerprints with the shared base checkpoint held fixed. All fingerprint estimators and attribution standardization use only training perturbations. BA denotes balanced accuracy; chance is 0.10.

Fingerprint	Cell line	n models	DEG@50 gain (pp)	Attribution BA
Context-matched	K562	10	3.56	0.918
Context-matched	A549	10	2.60	0.887
Context-matched	MCF7	10	4.05	0.876
Balanced pooled	K562	10	1.69	0.800
Balanced pooled	A549	10	1.88	0.753
Balanced pooled	MCF7	10	3.05	0.616

Table S13: Fingerprint subsampling curves. Fifty nested subsamples were drawn at each available size. Values are unit-level means with bootstrap 95% confidence intervals across model–context units.

Training perturbations	n units	DEG@50 gain (pp)	Fingerprint–full cosine
5	80	2.71 [0.91, 4.76]	0.736 [0.679, 0.791]
10	80	3.20 [1.44, 5.17]	0.808 [0.760, 0.856]
20	80	3.68 [1.91, 5.64]	0.867 [0.828, 0.903]
40	80	4.03 [2.20, 6.04]	0.918 [0.889, 0.944]
80	80	4.20 [2.45, 6.18]	0.962 [0.945, 0.977]
Full	80	4.27 [2.43, 6.30]	1.000 [1.000, 1.000]

Table S14: Global GTC across independently trained base-model seeds. The across-seed interval resamples the 20 architecture–context units after averaging their three training seeds.

Training seed	n units	Mean gain (pp)	Positive units
42	20	5.88	17/20
43	20	5.80	14/20
44	20	6.08	16/20
Across seeds	20	5.92 [1.97, 10.74]	47/60 runs

Table S15: Frequency with which validation-selected response-informed KNN outperformed perturbation-embedding KNN. Ties were absent.

Dataset	Response-KNN wins	Perturbation-KNN wins
norman2019	2/10	8/10
replogle_k562	4/10	6/10
sciplex3	7/10	3/10

Table S16: Pathway outcomes across all held-out perturbation–Hallmark cases significant in the truth. Rates are first computed within model–context units and then macro-averaged.

Correction	n units	Repaired	Preserved	Harmed
GlobalGTC	80	12.9%	25.1%	6.0%
ShuffledGap	80	1.5%	23.1%	8.0%
SignFlippedGap	80	3.4%	26.4%	4.7%