DISCOLORA: VARIATIONAL DISENTANGLEMENT OF PRETRAINED REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Pretrained models learn representations that can be reused across different tasks, but these representations often mix distinct sources of variation. Given attribute labels, we seek to separate variation associated with those attributes from structure shared across groups. This separation can help us understand how attributes are encoded and vary them during generation while preserving shared structure. We introduce DISCOLORA, a generative framework that learns complementary shared and attribute-associated representations through low-rank updates to a frozen pretrained encoder–decoder. Across synthetic, image, and gene-expression data, DISCOLORA separates these sources of variation using pretrained backbones. We identify interpretable directions in the attribute-associated representation and trace their decoded effects to observed features. These results demonstrate how pretrained representations can be adapted to separate, interpret, and manipulate attribute-associated variation across modalities.

1 INTRODUCTION

In the Monty Python sketch *The Ministry of Silly Walks*, John Cleese demonstrates an improbable collection of ways to walk: long steps, sideways steps, kicks, pauses, and contortions (Gaesser et al., 2022). To an alien that has already learned to represent and reproduce human movement, these walks pose an inference problem: they share the same body and joints, yet each has distinctive features. Given labels for the walking styles, can the alien separate the shared structure from the features of each style without learning human movement all over again?

Pretrained encoder–decoder models present a similar challenge. An encoder maps observations to latent representations, and a decoder reconstructs the observations from them (Kingma & Welling, 2022). Expressive models learn rich structure in complex data (Rombach et al., 2021; Peebles & Xie, 2023; Hoffmann et al., 2022; Brown et al., 2020), but developing them requires substantial data and computation. When labels for a new attribute become available, we would like to reuse the pretrained model to understand which variation is associated with that attribute and which structure is shared across groups.

Parameter-efficient adaptation methods can specialize pretrained models while updating relatively few parameters (Houlsby et al., 2019; Hu et al., 2021). Representation learning methods can separate shared and attribute-associated information through reconstruction and regularization (Slavutsky et al., 2025). Combining these solutions requires learning a decomposition within the representations and decoding capabilities of a pretrained model. Predicting an attribute does not establish how it appears in the observations, while suppressing attribute information can discard useful shared structure. We therefore seek complementary representations that separate these sources of variation while retaining the information needed for reconstruction.

We introduce DISCOLORA, a parameter-efficient generative framework that separates shared and attribute-associated variation in pretrained encoder–decoder models. The method learns two paths through a frozen backbone using separate LoRA updates (Hu et al., 2021). One path infers a shared representation z from x and reconstructs the observation from z alone. The other infers an attribute-associated representation w from (x, y) and reconstructs the observation using both z and w. The two reconstruction objectives encourage z to retain shared structure and w to supply attribute-associated information. Conditional priors and adversarial regularization guide this separation (Slavutsky et al., 2025), with classifiers discouraging attribute information in both z and its reconstruction.

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.

This decomposition lets us examine attribute-associated variation in the original data space. We can hold z fixed and vary w to study decoded changes, or compare marginal and joint reconstructions of the same observation. Using integrated gradients (Sundararajan et al., 2017), we attribute decoded changes to latent coordinates and changes in attribute predictions to observed features. Our contributions are threefold: *1) Parameter-efficient decomposition.* We adapt frozen pretrained encoders and decoders with separate low-rank updates to learn shared and attribute-associated representations. This builds on pretrained features and decoding, with far fewer trainable parameters than disentanglement models trained from scratch and competitive performance; *2) Interpretation in observation space.* We combine latent traversals and reconstruction contrasts with feature attribution to show how the learned variation appears in the data, from attribute-associated regions in images to genes associated with a biological condition; *3) Disentangling foundation model architectures.* We show that a single-cell foundation model (scGPT) can be decomposed post hoc into cell identity and a stimulation response without retraining. The stimulation representation recovers canonical IFN-β genes (ISG15, IFIT1, CXCL10) and enrichment for chemokine and antiviral programs, and the shared representation improves cell-type structure over the unadapted backbone.

2 Related Work

Our work connects learning disentangled representations, adapting pretrained models, and interpreting learned variation. These areas address complementary questions: how to separate sources of variation, how to reuse an existing model, and how to relate its representations to observed features. We bring these questions together in a generative decomposition of a pretrained encoder–decoder.

Variational disentanglement. Supervised variational models use labels to specify which sources of variation should be separated. CSVAE assigns label-associated variation to a dedicated subspace and adversarially suppresses label information in the remaining representation (Klys et al., 2018). HCSVAE (*Patches*) models shared and condition-specific transcriptional programs hierarchically (Beker et al., 2026). DisCoVR combines marginal and joint reconstruction objectives with conditional priors and adversarial regularization (Slavutsky et al., 2025). The two reconstruction objectives give the latent components complementary roles in explaining the observations.

Adapting pretrained representations. Parameter-efficient adaptation reuses pretrained models by restricting which parameters or representations are updated. LoRA learns additive low-rank updates to frozen weight matrices (Hu et al., 2021). LoReFT learns interventions within low-dimensional subspaces of hidden representations (Wu et al., 2024). Closely related to our goal, MultiLoReFT uses representation interventions to separate shared and modality-specific information from frozen unimodal encoders (Tonekaboni et al., 2026). Unlike previous work, our disentangled decompositions are guided by observed attributes. Our work uses a new form of adapting both encoder and decoder maps under a generative objective. This lets us examine the learned components through their reconstructions in observation space.

Generative controls. Generative models provide a way to study learned variation by changing a representation and examining the resulting observations. Concept Sliders learns attribute-specific LoRA controls with a preservation objective to reduce interference with other attributes (Gandikota et al., 2025a). SliderSpace discovers multiple controls by aligning adapter-induced changes with principal directions in semantic feature space (Gandikota et al., 2025b). These methods learn editing directions in pretrained generators. DisCoLoRA instead infers shared and attribute-associated latent variables for each observation. Holding the shared representation fixed provides a reference for interpreting changes in the attribute-associated component through the fitted decoder.

Interpreting learned representations. Concept-based methods connect hidden representations to interpretable attributes. TCAV measures classifier sensitivity along directions obtained from concept examples (Kim et al., 2017). Concept bottleneck models instead predict annotated concepts before the target, enabling interventions on these intermediate predictions (Koh et al., 2020). Integrated gradients (IG) attributes an output difference to input coordinates by integrating derivatives along a path from a reference (Sundararajan et al., 2017). In DisCoLoRA, attributes supervise a generative decomposition. We use IG to interpret attribute-associated variation through the joint decoder and through classifier-based contrasts between the marginal and full reconstructions.

3 DISCOLORA: DECOMPOSING PRETRAINED REPRESENTATIONS

Our goal is to adapt a pretrained encoder–decoder to separate shared structure from attribute-associated variation. We observe labeled data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$, where $x_i \in \mathbb{R}^d$ and $y_i \in \{1, \dots, K\}$, and are given a pretrained encoder-decoder pair (f_{ϕ_0}, h_{ψ_0}). DISCOLORA learns a shared representation z and an attribute-associated representation w through separate low-rank updates to the encoder and decoder, keeping the pretrained weights fixed.

3.1 REPRESENTATION DECOMPOSITION

We model each observation through shared and attribute-associated latent variables $z, w \in \mathbb{R}^m$. In the generative model, z is independent of (w, y), and x is conditionally independent of y given (z, w). The joint distribution therefore factorizes as (Slavutsky et al., 2025)

$$p(x, y, z, w) = p(y)\, p(w \mid y)\, p(z)\, p(x \mid z, w). \tag{1}$$

We approximate the posterior with $q_{z,w|x,y} = q_{z|x}\, q_{w|x,y}$. The shared encoder infers z from x and the attribute-associated encoder infers w from (x, y). Although the generative model assumes independence between z and y, the inferred representations need not satisfy it. The adversarial objective in Section 3.3 encourages this independence across observations.

Representations and reconstructions. The shared encoder $f_\phi^z(x)$ and attribute-associated encoder $f_\rho^w(x, y)$ output the parameters $\theta_z = (\mu_z, \sigma_z^2)$ and $\theta_w = (\mu_w, \sigma_w^2)$ of diagonal Gaussian distributions. We draw $z \sim q_{\theta_z}$ and $w \sim q_{\theta_w}$ by reparameterization and form two reconstructions:

$$\hat{x} = h_\psi(z), \qquad \tilde{x} = h_\eta^*(z, w). \tag{2}$$

Reconstructing x from z alone encourages the shared representation to retain useful information, while the joint reconstruction uses w to supply additional, attribute-associated information.

3.2 ADAPTATION OF PRETRAINED ENCODER-DECODER ARCHITECTURES

The key innovation of DISCOLORA is to separate shared and attribute-associated variation while reusing a pretrained encoder–decoder. We do this with two paths that share a frozen backbone (Figure 1A). The shared path reconstructs x from z alone. The attribute-associated path combines z and w to produce $\tilde{x}$. We learn separate low-rank updates for the two paths and keep the pretrained weights fixed (Figure 1B). For a frozen base matrix $W_0 \in \mathbb{R}^{a \times b}$, LoRA (Hu et al., 2021) defines an adapted matrix as

$$W = W_0 + \Delta W, \qquad \Delta W = s_{\text{LoRA}} AB, \qquad A \in \mathbb{R}^{a \times r}, \quad B \in \mathbb{R}^{r \times b}, \tag{3}$$

The rank r controls the update capacity. We set the scale to $s_{\text{LoRA}} = \alpha_{\text{LoRA}}/r$ and initialize $A = 0$ and B randomly. Updates start at zero and preserve the pretrained mapping at initialization.

For encoders that already output Gaussian means and variances, we retain and adapt those output heads. For deterministic encoders, we add a head that maps their features to the mean and variance of a diagonal Gaussian, following work on variational information bottlenecks and probabilistic embeddings (Alemi et al., 2017; Shi et al., 2019). In both cases, we keep the base head weights fixed and learn low-rank updates through LoRA (Appendix A).

Augmented inputs. The joint decoder takes $[z, w]$ as input. We extend the encoder and decoder input matrices with zero columns for y and w, respectively. This preserves the pretrained mappings at initialization. Low-rank updates then learn how to use the additional inputs. Writing the base parameters with the necessary zero extensions as $(\bar{\phi}_0, \bar{\psi}_0)$; the two paths are parameterized as $\phi = \phi_0 + \Delta\phi$, $\psi = \psi_0 + \Delta\psi$, respectively $\rho = \bar{\phi}_0 + \Delta\rho$, and $\eta = \bar{\psi}_0 + \Delta\eta$. We apply LoRA to all linear layers in the MLP encoders and decoders, included the added posterior heads. We use the same parameterization for attention projections. The pretrained weights remain fixed; we train only the LoRA factors and the auxiliary classifiers. The classifiers have no low-rank constraint (Appendix A).

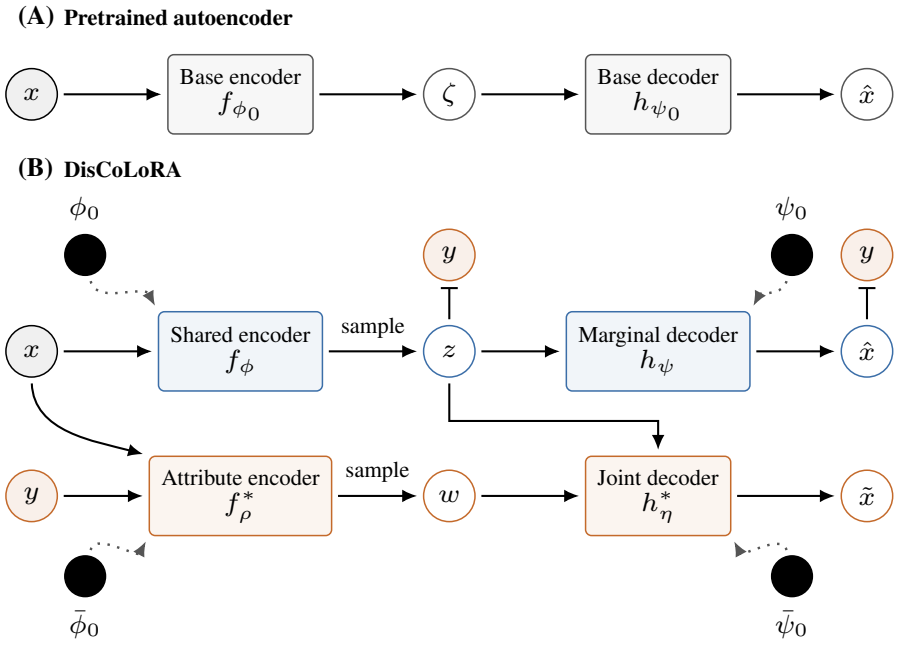


Figure 1: **A)** Pretrained encoder–decoder networks learn rich representations of high-dimensional data but can be costly to train. **B)** When attribute labels become available, DISCOLORA uses low-rank updates to disentangle shared and attribute specific variation while keeping pretrained weights fixed.

3.3 TRAINING OBJECTIVE

We train the model to reconstruct x from (z, w) while encouraging z alone to generate reconstructions $\hat{x}$ which minimize information about attributes y. As a result, a classifier trained only on $\hat{x}$ should have difficulty in distinguishing between y labels.

Latent priors. We model the prior for z and w as a standard Gaussian and a class-conditional Gaussian respectively: $p_z = \mathcal{N}(0, I), p_{w|y=k} = \mathcal{N}(\mu_k, I)$. The conditional mean is the expected shared representation within class k, $\mu_k = \mathbb{E}_{p_{x|y=k}} \mathbb{E}_{q_{z|x}}[z]$. We estimate it from the latent samples in each mini-batch $\mathcal{B}$: $\hat{\mu}_k = \frac{1}{|\mathcal{B}_k|} \sum_{i \in \mathcal{B}_k} z_i, \mathcal{B}_k = \{i \in \mathcal{B} : y_i = k\}$.

Variational objective. We use two complementary reconstruction objectives (Slavutsky et al., 2025). The first reconstructs an observation x from the latent z alone. The second reconstructs x from (z, w), allowing both representations to explain the observation. Each approximate posterior is regularized towards its corresponding prior. With Gaussian observation models of fixed isotropic variance, the reconstruction terms reduce to squared errors. For a mini-batch $\mathcal{B}$, the weighted loss is

$$\mathcal{J}_{\text{var}}(\Omega) = \frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \Big[\frac{\lambda_z}{d} \|x_i - \hat{x}_i\|_2^2 + \frac{\lambda_{zw}}{d} \|x_i - \tilde{x}_i\|_2^2 + \alpha_1 D_{\text{KL}}\big(q_{z|x_i} \,\|\, \mathcal{N}(0, I)\big) + \alpha_2 D_{\text{KL}}\big(q_{w|x_i, y_i} \,\|\, \mathcal{N}(\hat{\mu}_{y_i}, I)\big) \Big], \tag{4}$$

where $\Omega = (\phi, \psi, \rho, \eta)$ denote the encoder and decoder parameters. The weights λ_z and λ_{zw} balance the two reconstruction losses, while α_1 and α_2 set the strength of regularization. Variational bounds, information regarding loss normalization and hyperparameter settings are provided in Appendix A.

Suppressing attribute information. The original loss in Equation 4 does not guarantee that z and y remain disentangled. We therefore add adversarial regularization to discourage attribute information in the shared representation and its reconstruction. We accomplish this in two steps.

Algorithm 1 DISCOLORA Training

Input: Data $\mathcal{D}$; frozen encoder–decoder (f_{ϕ_0}, h_{ψ_0}); rank r; iterations T; learning rate γ; loss weights.
1: Initialize LoRA factors Λ and classifier parameters (β_x, β_z).
2: **for** $t = 1, \ldots, T$ **do**
3: Take a training mini-batch $\mathcal{B} = (x, y)$.
4: Compute $\theta_z = f_\phi(x)$ and $\theta_w = f_\rho(x, y)$.
5: Sample $z \sim q_{\theta_z}$ and $w \sim q_{\theta_w}$ by reparameterization.
6: Compute $\hat{x} = h_\psi(z)$ and $\tilde{x} = h^*_\eta(z, w)$.
7: Estimate class means $\hat{\mu}_k$.
8: Evaluate $C_{\hat{x}}, C_z$ and $\mathcal{J}$ using equation 5 and equation 7.
9: Compute $\nabla_\Lambda \mathcal{J}$, $\nabla_{\beta_x} C_{\hat{x}}$, and $\nabla_{\beta_z} C_z$ at the current parameters.
10: Apply one Adam step to $(\Lambda, \beta_x, \beta_z)$ using these gradients and learning rate γ.
11: **end for**
12: **return** $\Lambda, \beta_x, \beta_z$.

First, we train two classifiers to predict y: $g_{\beta_{\hat{x}}}(y \mid \hat{x})$ from the marginal reconstruction and $g_{\beta_z}(y \mid z)$ from the shared representation. Since $\hat{x} = h_\psi(z)$, the data processing inequality gives $I(y; \hat{x}) \leq I(y; z)$ (Cover & Thomas, 2001). Thus, z may retain attribute information that is absent from $\hat{x}$. The latent classifier supplies adversarial feedback that discourages attribute information remaining in z. The classifiers are fit by minimizing cross-entropy

$$C_{\hat{x}} = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \log g_{\beta_x}(y_i \mid \hat{x}_i), \qquad C_z = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \log g_{\beta_z}(y_i \mid z_i). \tag{5}$$

Second, we train the shared encoder - marginal decoder path to make these classifiers uncertain about y. We measure uncertainty by the average predictive entropy

$$H_u = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \sum_{c \in \mathcal{Y}} g_{\beta_u}(c \mid u_i) \log g_{\beta_u}(c \mid u_i), \tag{6}$$

where $\mathcal{Y}$ denotes the set of attribute classes and $u \in \{z, \hat{x}\}$.

Optimization objective. Combining the variational objective and the attribute suppressing regularization terms, DISCOLORA minimizes the joint objective

$$\mathcal{J}(\Omega; \beta_x, \beta_z) = \mathcal{J}_{\text{var}}(\Omega) - \lambda_{\text{adv}}(H_{\hat{x}} + H_z). \tag{7}$$

Here, $\lambda_{\text{adv}} \geq 0$ controls the strength of adversarial regularization, H_z supplies gradients to the shared encoder, while $H_{\hat{x}}$ supplies gradients to both the shared encoder and marginal decoder. The classifiers are updated only through their cross-entropy losses (Appendix A; Algorithm 1).

Why a predictive entropy regularizer? We seek to make the shared representation z and its reconstruction $\hat{x}$ uninformative about y. A classifier can make incorrect predictions while still extracting attribute information. For a fixed classifier, increasing cross-entropy can reward confidently incorrect predictions. Maximizing predictive entropy instead rewards uncertainty. This motivates our choice of entropy regularization over reversing the classifier's cross-entropy gradients (Ganin & Lempitsky, 2015).

When the classifiers recover the true conditional distributions, this regularizer penalizes attribute information in the shared encoder - marginal decoder path lending the objective an information theoretic interpretation. For fixed model parameters Ω, the data and shared encoder induce

$$q_\Omega(x, y, z) = p_{x,y}(x, y) q_{z|x}(z \mid x), \qquad \hat{x} = h_\psi(z). \tag{8}$$

Suppose each classifier family contains the corresponding conditional distribution and training attains the population cross-entropy minimum. The optimal classifiers then satisfy

$$g_{\beta^\star_{\hat{x}}(\Omega)}(y \mid \hat{x}) = q_\Omega(y \mid \hat{x}), \qquad g_{\beta^\star_z(\Omega)}(y \mid z) = q_\Omega(y \mid z).$$

Their expected predictive entropies equal the conditional entropies,

$$\mathbb{E}[H_u] = H(y \mid u) = H(y) - I(y; u), \qquad u \in \{z, \hat{x}\}. \tag{9}$$

Here, $I(y; u)$ measures how much observing u reduces uncertainty about y. As the label distribution is fixed, increasing $H(y \mid u)$ reduces $I(y; u)$, which corresponds to the attribute information in u (Cover & Thomas, 2001).

Let $\overline{\mathcal{J}}_{\mathrm{var}}(\Omega)$ denote the variational loss averaged over mini-batches and latent samples. Substituting Equation 9 into Equation 7 gives

$$\mathbb{E}\big[\mathcal{J}\big(\Omega; \beta_x^\star(\Omega), \beta_z^\star(\Omega)\big)\big] = \overline{\mathcal{J}}_{\mathrm{var}}(\Omega) + \lambda_{\mathrm{adv}}\big[I(y; \hat{x}) + I(y; z)\big] - 2\lambda_{\mathrm{adv}} H(y). \tag{10}$$

The loss penalizes attribute information in z and $\hat{x}$ while encouraging $\hat{x}$ and $\tilde{x}$ to remain informative about the observations. In practice, classifiers may be uncertain because they fail to detect attribute information that is still present. Entropy regularization therefore provides a practical signal for suppressing attribute information, without a theoretical guarantee of independence from y.

3.4 Interpreting disentangled representations via attribution

We use the learned decomposition to study how variation associated with y appears in the input space. Paired observations at different values of y are often unavailable, especially in biomedical applications, where the same cell or tissue cannot usually be observed under both stimulated and unstimulated conditions. We therefore construct two model-based contrasts. The first varies w while preserving z. The second compares the joint reconstruction $\tilde{x}$ with the marginal reconstruction $\hat{x}$. Integrated gradients attribute these contrasts to latent coordinates and observed features.

Latent directions. For the first contrast, we fix z and interpolate between two attribute-associated representations w_0 and w_1. For binary attributes, the endpoints are class-averaged or matched posterior samples. Letting $v = w_1 - w_0$, the contribution of a latent coordinate k to a decoded feature j is $A_{jk} = v_k \int_0^1 \frac{\partial h^*_{\eta,j}(z, w_0 + tv)}{\partial w_k}\, dt$. The attributions satisfy $\sum_{k=1}^m A_{jk} = h^*_{\eta,j}(z, w_1) - h^*_{\eta,j}(z, w_0)$, provided the decoded path is sufficiently regular (Sundararajan et al., 2017, Proposition 1).

Reconstruction contrasts. For the second contrast, we evaluate the difference between $\tilde{x} = h^*_\eta(z, w)$ and $\hat{x} = h_\psi(z)$. Let $s(x)$ denote the score of an independently trained attribute classifier. The contribution of observed feature j along the path from $\tilde{x}$ to $\hat{x}$ is

$$\mathrm{IG}_j = (\hat{x}_j - \tilde{x}_j) \int_0^1 \partial_j s(\tilde{x} + t(\hat{x} - \tilde{x}))\; dt.$$

These attributions sum to $\sum_{j=1}^d \mathrm{IG}_j = s(\hat{x}) - s(\tilde{x})$ and their signs indicate which features increase or decrease attribute evidence along the reconstruction path.

4 Experiments

We evaluate DisCoLoRA across three distinct settings: synthetic manifolds, natural images, and single-cell transcriptomics. We compare DisCoLoRA against DisCoVR and established methods for learning shared and attribute-associated representations, including CSVAE (Klys et al., 2018), HCSVAE with and without adversarial regularization, DIVA (Ilse et al., 2019), and CCVAE (Joy et al., 2020). We evaluate reconstruction quality using MSE and the separation of shared and attribute-associated information using KNN estimates of mutual information (Kraskov et al. (2003); Büth et al. (2025); Appendix A). When additional ground-truth structure or domain-specific annotations are available, we report complementary dataset-specific metrics within the corresponding experiment.

4.1 Noisy Swiss roll

Dataset and backbone. We use the noisy Swiss roll benchmark of Slavutsky et al. (2025) to test whether low-rank adaptation recovers known shared and attribute-associated factors of variation. The spiral geometry is shared across conditions, while height determines a binary attribute whose labels are independently flipped with probability 0.3 (Figure 2A). We adapt a pretrained VAE using ranks $r \in \{4, 8, 12, 16\}$ and evaluate all methods over five random initializations using fixed training, validation, and test splits. Additional implementation details are provided in Appendix A.2.

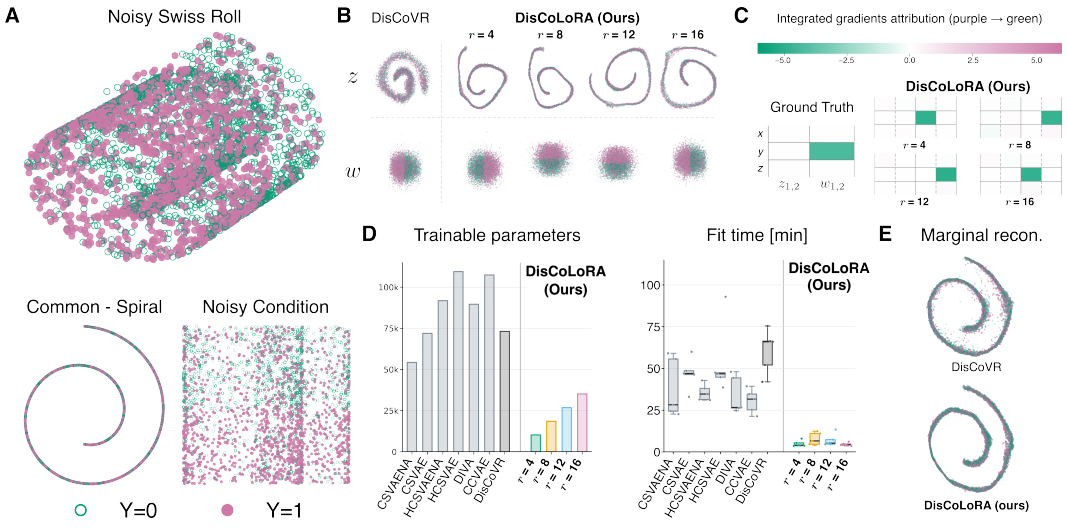


Figure 2: **A:** The Noisy Swiss Roll consists of two sets of points that share a spiral geometry but differ along the y-axis. Their group labels are corrupted with noise level 0.3. **B:** DISCOLORA captures this structure across low-rank approximations. Samples from shared z (top) and attribute-associated w (bottom) are colored by noisy label. **C:** We use integrated gradients to attribute differences between the two groups along joint (z, w) paths. The difference along the y-axis is attributed mainly to w. **D:** All tested ranks use fewer trainable parameters (left) and are faster (right) than fully trained baselines. **E:** Reconstructions from the shared representation retain the spiral geometry.

Evaluation. We measure separation using $I(y; z)$, the attribute information remaining in z, and $I(z; w)$, the dependence between the two representations. Lower values indicate better separation. We also report $\Delta\text{-Bayes}(w) = 100|\text{Acc}_w - 0.70|$. Labels are flipped independently with probability 0.3, so the Bayes-optimal accuracy for predicting the observed label from x is 70%. This score measures the distance from that reference accuracy (Table S1; Figure 2A).

Results. DISCOLORA recovers the marginal geometry of the data (spiral) in z, with both noisy-label classes following the same geometry for all tested ranks (Figure 2B). Mean signed IG along joint (z, w) paths localizes the feature determining the class to height and attributes this feature primarily to w across all ranks, matching the known decomposition (Figure 2C).

DISCOLORA uses fewer trainable parameters and shorter adaptation times than every baseline across all ranks (Figure 2D). The marginal reconstruction also recovers the marginal spiral geometry (Figure 2E). Among adapter ranks, $r = 4$ gives the lowest mean $I(y; z)$, $r = 12$ the lowest mean $I(z; w)$, and $r = 8$ the lowest mean Δ-Bayes score (Table S1).

4.2 CELEBA-GLASSES

Dataset and backbone. We use CelebA faces annotated for eyeglasses (Liu et al., 2014), following Slavutsky et al. (2025). Within each official split, we retain all images with eyeglasses and sample twice as many without. We adapt the pretrained Stable Diffusion VAE (Rombach et al., 2021) using the same benchmark suite as above. Implementation details are provided in Appendix A.3.

Evaluation. We report validation MSE of the joint and marginal reconstructions against the original images, quantifying fidelity with and without w, and $I(z; w)$ to measure dependence between the two representations.

Results. The joint reconstruction uses (z, w) and the marginal reconstruction uses z alone (Figure 3A). Marginalizing w attenuates the glasses attribute while preserving facial structure: joint reconstructions retain the observed glasses state, whereas marginal reconstructions move both classes toward an intermediate appearance, adding glasses-like structure to faces without glasses and weakening existing glasses (Figure 3B–C). Pose, hairstyle, and other facial features are largely



Figure 3: **Controlled evaluation on CelebA with eyeglasses as the target attribute.** We test whether DISCOLORA can separate variation associated with glasses from other facial variation and identify where these differences appear in the image. **A:** The marginal reconstruction $\hat{x}$ is learned from z alone; the full reconstruction $\tilde{x}$ uses both z and w. **B:** Marginal reconstructions introduce glasses-like features in faces without glasses and weaken them in faces with glasses. **C:** Original images alongside joint and marginal reconstructions for faces with and without glasses. Joint and marginal reconstructions differ mainly around the eyes, as shown by the mean RGB difference.. **D:** Integrated gradients identify image regions that contribute to changes in an independent classifier's prediction from joint to marginal reconstructions. The predicted probability of glasses increases for faces without glasses and decreases for faces with glasses.

unchanged. The RGB differences between the two reconstructions concentrate around the eyes (Figure 3C, bottom), and signed IG through an independent classifier gives the same directional picture: marginalization raises glasses evidence for faces without glasses and lowers it for faces with glasses (Figure 3D).Quantitatively, the rank-16 model attains the lowest joint and marginal reconstruction MSE and the lowest $I(z; w)$ of all methods (Table S2), improving on the baselines by an order of magnitude in reconstruction error.

4.3 IFN-β STIMULATION

Dataset and backbone. We use a single-cell transcriptomic dataset of peripheral blood mononuclear cells (PBMCs) (Kang et al., 2017), comprising control and IFN-β-stimulated cells from eight donors across eight annotated cell types. As the encoder-decoder backbone, we use the pretrained whole-human scGPT model (Cui et al., 2024) and its 512-dimensional CLS representation. Additional preprocessing and implementation details are provided in Appendix A.4.

Evaluation. We evaluate disentanglement with K-means normalized mutual information (NMI), measuring stimulation structure in w and cell-type versus stimulation structure in z. Good disentanglement yields high w–stimulation and z–cell type NMI, and low z–stimulation NMI. We rank genes by their IG attributions to w, and assess the resulting markers using Gene Ontology enrichment and known IFN-β response genes from the literature.

Results. DISCOLORA successfully separates stimulation from cell identity while preserving cell-type structure. In Supplementary Table S3, the DISCOLORA row shows strong stimulation structure in w, the highest cell-type NMI in z among the compared methods, and little residual stimulation information in z. Relative to scGPT with adapters off, adaptation substantially strengthens cell-type

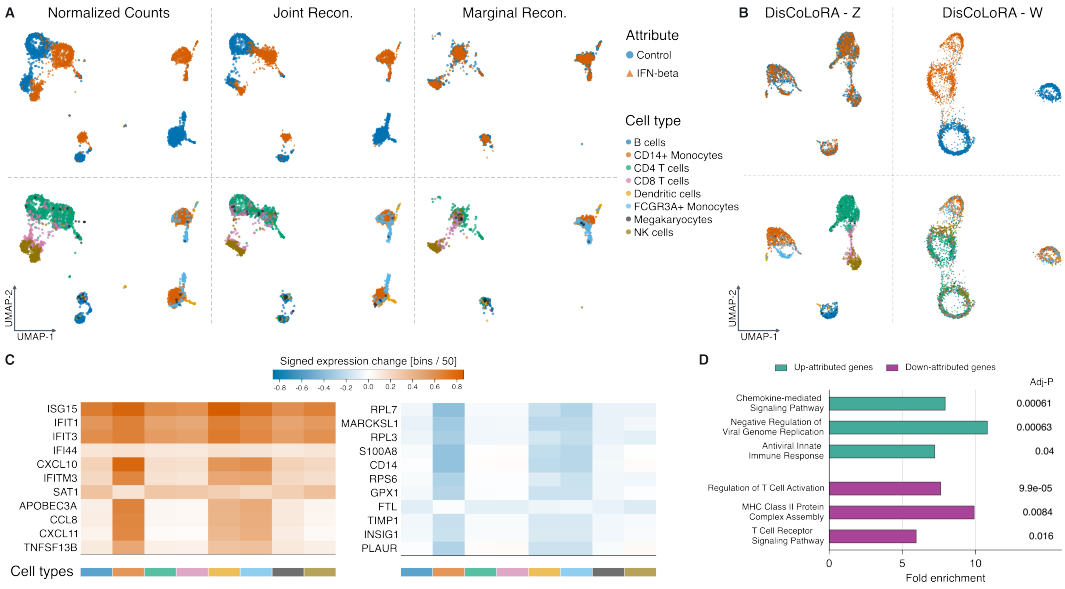


Figure 4: DISCOLORA on IFN-β stimulation in PBMCs. **A:** UMAPs of normalized expression, joint reconstructions, and marginal reconstructions; colored by stimulation (top) and cell type (bottom). **B:** UMAPs of z and w, colored as in A. **C:** Signed IG attributions through w for top changing genes across cell types. **D:** Gene Ontology enrichment of positively and i attributed genes. Labels report adjusted P-values.

structure while further removing stimulation information from the shared representation. Consistently, marginal reconstructions reduce stimulation separation while retaining cell-type structure (Figure 4A), and the learned z and w embeddings organize primarily by cell type and stimulation, respectively (Figure 4B).

Gene attributions obtained through IG on w recover increased expression of *ISG15*, *IFIT1*, *CXCL10*, *CXCL11*, *CCL8*, and *IFITM3* as being the drivers of condition specific differences (Figure 4C), all previously associated with IFN-β responses in human PBMCs (Satoh et al., 2006). Positive attributions are enriched for chemokine and antiviral programs, whereas negative attributions are enriched for T-cell activation, MHC class II complex assembly, and T-cell receptor signaling (Figure 4D), consistent with reported effects of IFN-β on T-cell activation and MHC class II expression (Noronha et al., 1993; Lu et al., 1995). Together, these results indicate that w captures stimulation-associated variation consistent with previously reported IFN-β responses, while z retains shared cell identity.

5 CONCLUSION

We present DISCOLORA, a parameter-efficient way to adapt pretrained encoder–decoders when labels for new attributes become available. The method separates shared and attribute-associated variation through low-rank updates while preserving the pretrained model's reconstruction capabilities. Across synthetic, image, and single-cell data, the learned representations organize variation according to the intended shared and attribute-associated roles.

The resulting latent paths and reconstruction contrasts give practitioners ways to inspect how a model represents an attribute. When combined with interpretability techniques like integrated gradients, they connect changes in learned latent representations to observed features, helping assess whether these differences agrees with domain knowledge and exposing associations that may otherwise remain hidden. This makes DISCOLORA useful for probing pretrained generative models before they are used in scientific analysis. Future work will extend the method to multimodal data, multiple interacting attributes, and data with spatio-temporal structure.

Reproducibility Statement

Implementation of the methods discussed in the study are provided as a supplementary file. We further describe all technical details required to run the experiments in Appendix A. The repository and the code will be made publicly available upon acceptance.

Ethics Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

AI Use Statement

Generative AI tools were used for editing the manuscript to improve writing prose. The authors take full responsibility for the final content of this work.

References

Alexander A. Alemi, Ian Fischer, and Joshua V. Dillon. Deep variational information bottleneck. In *International Conference on Learning Representations*, 2017. doi: 10.48550/arxiv.1612.00410. URL https://arxiv.org/abs/1612.00410.

Ozgur Beker, Simon Van Deursen, Michael Tarnow, Dreyton Amador, J. Cheong, Jose Francisco Pomarino Nima, Mark D. Robinson, Yvon Woappi, and Bianca Dumitrascu. Patches: A representation learning framework for decoding shared and condition-specific transcriptional programs in wound healing, 2026. URL http://dx.doi.org/10.1101/2024.12.23.630186.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

C. Büth, Kishor Acharya, and M. Zanin. infomeasure: a comprehensive python package for information theory measures and estimators. *Scientific Reports*, 15(1):29323, 2025. ISSN 2045-2322. doi: 10.1038/s41598-025-14053-5. URL http://dx.doi.org/10.1038/s41598-025-14053-5.

Edward Y. Chen, Christopher M. Tan, Y. Kou, Qiaonan Duan, Zichen Wang, G. Meirelles, N. R. Clark, and Avi Ma'ayan. Enrichr: Interactive and collaborative html5 gene list enrichment analysis tool. *BMC Bioinformatics*, 14(1):128, 2013. doi: 10.1186/1471-2105-14-128.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2 edition, 2001. ISBN 9780471241959. doi: 10.1002/9781118032770.ch6. URL http://dx.doi.org/10.1002/047174882X.

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. ISSN 1548-7091. doi: 10.1038/s41592-024-02201-0. URL http://dx.doi.org/10.1038/s41592-024-02201-0.

Suzi A James P Seth J Michael Dustin Marc Pascale Nomi Aleksander Balhoff Carbon Cherry Ebert Feuermann G, Suzi A Aleksander, J. Balhoff, S. Carbon, J. Cherry, Dustin Ebert, M. Feuermann, Pascale Gaudet, N. Harris, David P. Hill, et al. The gene ontology knowledgebase in 2026. *Nucleic Acids Research*, 54(D1):D1779–D1792, 2025. doi: 10.1093/nar/gkaf1292.

Glenn A Gaesser, David C Poole, and Siddhartha S Angadi. Quantifying the benefits of inefficient walking: Monty python inspired laboratory based experimental study. *British medical journal*, 379:e072833, 2022. ISSN 0959-8138. doi: 10.1136/bmj-2022-072833. URL http://dx.doi.org/10.1136/bmj-2022-072833.

Rohit Gandikota, Joanna Materzyńska, Tingrui Zhou, Antonio Torralba, and David Bau. Concept sliders: Lora adaptors for precise control in diffusion models. In *Lecture Notes in Computer Science*, Lecture Notes in Computer Science, pp. 172–188. Springer Nature Switzerland, 2025a. ISBN 9783031736605. doi: 10.1007/978-3-031-73661-2_10. URL http://dx.doi.org/10.1007/978-3-031-73661-2_10.

Rohit Gandikota, Zongze Wu, Richard Zhang, David Bau, Eli Shechtman, and Nick Kolkin. Sliderspace: Decomposing the visual capabilities of diffusion models. In *IEEE International Conference on Computer Vision*, pp. 15994–16003. IEEE, 2025b. doi: 10.1109/iccv51701.2025.01484. URL http://dx.doi.org/10.1109/ICCV51701.2025.01484.

Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation, 2015. URL https://arxiv.org/abs/1409.7495.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Thomas Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karén Simonyan, Erich Elsen, Oriol Vinyals, Jack Rae, and Laurent Sifre. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, volume 35, pp. 30016–30030. Curran Associates, Inc., 2022. doi: 10.52202/068431-2176. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/c1e2faff6f588870935f114ebe04a3e5-Abstract-Conference.html.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp, 2019. URL https://arxiv.org/abs/1902.00751.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models, 2021. URL https://arxiv.org/abs/2106.09685.

Maximilian Ilse, Jakub M. Tomczak, Christos Louizos, and Max Welling. Diva: Domain invariant variational autoencoders, 2019. URL https://arxiv.org/abs/1905.10427.

Tom Joy, Sebastian M. Schmon, Philip H. S. Torr, N. Siddharth, and Tom Rainforth. Capturing label characteristics in vaes, 2020. URL https://arxiv.org/abs/2006.10102.

Hyun Min Kang, Meena Subramaniam, Sasha Targ, Michelle Nguyen, Lenka Maliskova, Elizabeth McCarthy, Eunice Wan, Simon Wong, Lauren Byrnes, Cristina M Lanata, et al. Multiplexed droplet single-cell rna-sequencing using natural genetic variation. *Nature Biotechnology*, 36(1): 89–94, Jan 2017. ISSN 1087-0156. doi: 10.1038/nbt.4042. URL http://dx.doi.org/10.1038/nbt.4042.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2668–2677. PMLR, 2017. doi: 10.48550/arxiv.1711.11279. URL https://proceedings.mlr.press/v80/kim18d.html.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. URL https://arxiv.org/abs/1312.6114.

Jack Klys, Jake Snell, and Richard Zemel. Learning latent subspaces in variational autoencoders. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. doi: 10.48550/arxiv.1812.06190. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/73e5080f0f3804cb9cf470a8ce895dac-Paper.pdf.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5338–5348. PMLR, 2020. doi: 10.48550/arxiv.2007.04612. URL https://proceedings.mlr.press/v119/koh20a.html.

Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 69(6):066138, 2003. ISSN 1539-3755. doi: 10.1103/physreve.69.066138. URL http://dx.doi.org/10.1103/PhysRevE.69.066138.

Maxim V. Kuleshov, Matthew R. Jones, Andrew D. Rouillard, Nicolas F. Fernandez, Qiaonan Duan, Zichen Wang, Simon Koplev, Sherry L. Jenkins, Kathleen M. Jagodnik, Alexander Lachmann, et al. Enrichr: A comprehensive gene set enrichment analysis web server 2016 update. *Nucleic Acids Res.*, 44(W1):W90–W97, 2016. doi: 10.1093/nar/gkw377.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *IEEE International Conference on Computer Vision*, pp. 3730–3738. IEEE, 2014. doi: 10.1109/iccv.2015.425. URL http://dx.doi.org/10.1109/ICCV.2015.425.

H. T. Lu, J. L. Riley, G. T. Babcock, M. Huston, G. R. Stark, J. M. Boss, and R. M. Ransohoff. Interferon (ifn) beta acts downstream of ifn-gamma-induced class ii transactivator messenger rna accumulation to block major histocompatibility complex class ii gene expression and requires the 48-kd dna-binding protein, isgf3-gamma. *Journal of Experimental Medicine*, 182(5):1517–1525, 1995. doi: 10.1084/jem.182.5.1517.

A. Noronha, A. Toscas, and M. A. Jensen. Interferon beta decreases t cell activation and interferon gamma production in multiple sclerosis. *Journal of Neuroimmunology*, 46(1-2):145–153, 1993. doi: 10.1016/0165-5728(93)90244-S.

Fabian Pedregosa, G. Varoquaux, Alexandre Gramfort, V. Michel, B. Thirion, O. Grisel, Mathieu Blondel, Gilles Louppe, P. Prettenhofer, Ron Weiss, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. doi: 10.5555/1953048.2078195.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, October 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Peebles_Scalable_Diffusion_Models_with_Transformers_ICCV_2023_paper.html.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjorn Ommer. High-resolution image synthesis with latent diffusion models. In *Computer Vision and Pattern Recognition*, pp. 10674–10685. IEEE, 2021. doi: 10.1109/cvpr52688.2022.01042. URL http://dx.doi.org/10.1109/cvpr52688.2022.01042.

Jun-ichi Satoh, Yusuke Nanri, Hiroko Tabunoki, and Takashi Yamamura. Microarray analysis identifies a set of cxcr3 and ccr2 ligand chemokines as early ifnbeta-responsive genes in peripheral blood lymphocytes in vitro: an implication for ifnbeta-related adverse effects in multiple sclerosis. *BMC Neurology*, 6(1):18, 2006. doi: 10.1186/1471-2377-6-18.

Yichun Shi, Anil K. Jain, and Nathan D. Kalka. Probabilistic face embeddings. In *IEEE International Conference on Computer Vision*, pp. 6901–6910. IEEE, 2019. ISBN 9781728148038. doi: 10.1109/iccv.2019.00700. URL http://dx.doi.org/10.1109/ICCV.2019.00700.

Yuli Slavutsky, Ozgur Beker, David Blei, and Bianca Dumitrascu. Variational Learning of Disentangled Representations, 2025. URL https://arxiv.org/abs/2506.17182.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic Attribution for Deep Networks, 2017. URL https://arxiv.org/abs/1703.01365.

Sana Tonekaboni, Viktoria Schuster, and Caroline Uhler. Multiloreft: Decoupling shared and modality-specific subspaces in multimodal learning via low-rank representation fine-tuning, 2026. URL https://arxiv.org/abs/2607.16789.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher Manning, and Christopher Potts. Reft: Representation finetuning for language models. In *Advances in Neural Information Processing Systems 37*, NeurIPS 2024, pp. 63908–63962. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2024. doi: 10.52202/079017-2041. URL http://dx.doi.org/10.52202/079017-2041.

Zhuorui Xie, Allison Bailey, Maxim V. Kuleshov, Daniel J. B. Clarke, J. E. Evangelista, Sherry L. Jenkins, Alexander Lachmann, Megan L. Wojciechowicz, Eryk Kropiwnicki, Kathleen M. Jagodnik, et al. Gene set knowledge discovery with enrichr. *Current Protocols*, 1(3):e90, 2021. doi: 10.1002/cpz1.90.

A Technical Details

A.1 Model parameterization and training

Variational bounds. The two reconstruction terms in equation 4 arise from variational bounds following DisCoVR (Slavutsky et al., 2025, Section 2.3). For the shared encoder - marginal decoder path, we introduce the auxiliary observation model $p_\psi(x \mid z)$ with prior p_z. Its variational bound is

$$\mathcal{L}_z(x) = \mathbb{E}_{q_{z|x}}\left[\log p_\psi^z(x \mid z)\right] - D_{\mathrm{KL}}\big(q_{z|x} \,\|\, p_z\big) \le \log p_\psi(x). \tag{11}$$

For the joint path, the factorization in equation 1 gives, for fixed class-prior means,

$$\begin{aligned}\mathcal{L}_w(x,y) = \mathbb{E}_{q_{z|x}q_{w|x,y}}\left[\log p(x \mid z,w)\right] \\ - D_{\mathrm{KL}}\big(q_{z|x} \,\|\, p_z\big) - D_{\mathrm{KL}}\big(q_{w|x,y} \,\|\, p_{w|y}\big) \le \log p(x \mid y).\end{aligned} \tag{12}$$

The objective in equation 4 uses the reconstruction and KL terms from these bounds with separate weights. During training, the conditional prior is centered at the mini-batch estimate $\hat{\mu}_k$.

Zero-extended base parameters. The attribute-associated encoder receives $[x, y]$, where y is represented as a K-dimensional one-hot vector, while the joint decoder receives $[z, w]$. To accommodate these additional inputs without changing the pretrained mapping at initialization, we append zero columns to the first weight matrix of the corresponding frozen encoder and decoder. Let

$$W_{\phi_0}^{(1)} \in \mathbb{R}^{a_\phi \times d}, \qquad W_{\psi_0}^{(1)} \in \mathbb{R}^{a_\psi \times m}$$

denote the first-layer weight matrices of the pretrained encoder and decoder. We define

$$\overline{W}_{\phi_0}^{(1)} = \begin{bmatrix} W_{\phi_0}^{(1)} & \mathbf{0}_{a_\phi \times K} \end{bmatrix} \in \mathbb{R}^{a_\phi \times (d+K)}, \qquad \overline{W}_{\psi_0}^{(1)} = \begin{bmatrix} W_{\psi_0}^{(1)} & \mathbf{0}_{a_\psi \times m} \end{bmatrix} \in \mathbb{R}^{a_\psi \times 2m}. \tag{13}$$

Consequently,

$$\overline{W}_{\phi_0}^{(1)} \begin{pmatrix} x \\ y \end{pmatrix} = W_{\phi_0}^{(1)} x, \qquad \overline{W}_{\psi_0}^{(1)} \begin{pmatrix} z \\ w \end{pmatrix} = W_{\psi_0}^{(1)} z,$$

so the added inputs have no effect before adaptation. We write $\bar{\phi}_0$ and $\bar{\psi}_0$ for the corresponding zero-extended frozen base parameters. The attribute-associated paths are then parameterized as

$$\rho = \bar{\phi}_0 + \Delta\rho, \qquad \eta = \bar{\psi}_0 + \Delta\eta, \tag{14}$$

with $\Delta\rho$ and $\Delta\eta$ given by the learned LoRA updates.

Gaussian posterior heads. Backbones with Gaussian encoders retain their existing posterior parameterization. For deterministic encoders, we place a diagonal Gaussian over the continuous features. Given $h \in \mathbb{R}^m$, an affine head predicts the mean μ and log-variance ℓ:

$$\begin{pmatrix} \mu \\ \ell \end{pmatrix} = \left[\begin{pmatrix} I_m \\ \mathbf{0}_{m \times m} \end{pmatrix} + s_{\mathrm{LoRA}} AB \right] h + \begin{pmatrix} \mathbf{0}_m \\ 2\log(0.1)\,\mathbf{1}_m \end{pmatrix}, \qquad q(\cdot \mid h) = \mathcal{N}\big(\mu, \mathrm{diag}(e^\ell)\big). \tag{15}$$

Since $AB = 0$ at initialization, the posterior is $\mathcal{N}(h, 0.01I)$.

Adapter implementation. We initialize the random LoRA factor B with `kaiming_uniform_` using $a = \sqrt{5}$. Only the LoRA factors and classifier parameters are optimized; base weights, biases, and normalization parameters remain fixed. Frozen encoder and decoder backbones are kept in evaluation mode.

Adversarial updates. For each classifier input $u \in \{z, \hat{x}\}$, let sg denote stop-gradient and let $\bar{u} = u$ in the forward pass, with gradient reversal multiplying derivatives with respect to u by $-\lambda_{\text{adv}}$. We define

$$\begin{aligned} c_u &= \text{sg}[\nabla_{\bar{u}}(H_u - C_u)]\,, \\ \mathcal{S}_u &= C_u - \lambda_{\text{adv}}\,\text{sg}(H_u) + \langle \bar{u} - \text{sg}(\bar{u}), c_u \rangle\,. \end{aligned} \tag{16}$$

The resulting gradients are

$$\nabla_{\beta_u}\mathcal{S}_u = \nabla_{\beta_u}C_u, \qquad \nabla_u\mathcal{S}_u = -\lambda_{\text{adv}}\nabla_{\bar{u}}H_u. \tag{17}$$

We therefore backpropagate through $\mathcal{J}_{\text{var}} + \mathcal{S}_{\hat{x}} + \mathcal{S}_z$, while monitoring the model objective in equation 7.

A.2 Noisy Swiss roll

Data. We generate independent training, validation, and test sets of 10,000, 2,000, and 10,000 observations using scikit-learn's `make_swiss_roll` without coordinate noise (Pedregosa et al., 2011). We set the label to 1 when the unshifted height is below 10 and to 0 otherwise, then flip each label independently with probability 0.3. We translate the height by 5 and each spiral-plane coordinate by 5 minus its training-set minimum, using the same translations for all splits.

Models. We use $z, w \in \mathbb{R}^2$. Encoders and decoders are MLPs with two hidden layers of width 128, batch normalization, and ReLU activations. Gaussian encoders use linear mean outputs and standard deviations $\text{softplus}(s) + 10^{-6}$; decoder outputs are linear. The two DisCoLoRA adversaries each have one hidden layer of width 128. For CSVAE and HCSVAE, the suffix NA denotes removal of the adversary.

Pretraining. For each initialization, we pretrain a two-dimensional VAE with AdamW and zero weight decay. The reconstruction weight is 20, and the initial learning rate is selected from $3 \times 10^{-3}, 10^{-2}$; all selected models use 10^{-2}. The learning rate follows a cosine schedule to 10^{-5} over 15,000 epochs. The KL weight decreases geometrically from 0.1 to 10^{-4} over the first 7,500 epochs and remains fixed thereafter. We select checkpoints by pooled validation RMSE over four posterior draws and reuse the selected VAE across all adapter ranks.

Training and hyperparameters. We jointly optimize the adapters and adversaries with Adam at learning rate 10^{-3} and batch size 64. Baselines use AdamW with the same learning rate and batch size and the loss weights of Slavutsky et al. (2025, Supplementary Table 20); we retain their lowest-validation-loss checkpoint and stop after 100 epochs without improvement. For DisCoLoRA, we set $\lambda_{zw} = 0.9$ and $\alpha_1 = \alpha_2 = 0.2$ and evaluate

$$(\lambda_{\text{adv}}, \lambda_z) \in \{(64, 0.2), (256, 1), (1024, 1), (1024, 4)\}.$$

Evaluation. We evaluate one posterior sample per observation. For mutual information, we use $k = 4$, the maximum norm, and natural logarithms implemented in `infomeasure` (Kraskov et al., 2003; Büth et al., 2025). For Δ-Bayes(w), we fit a Gaussian naive Bayes classifier to the training embeddings of w and evaluate ordinary accuracy on the test set.

Attribution. We sample 1,024 test observations without replacement from each noisy-label class and pair them in sampled order. For each pair, let $u_0 = (z_0, w_0)$ and $u_1 = (z_1, w_1)$ denote the joint posterior means for classes 0 and 1. We compute integrated gradients through the joint decoder along the straight path $u(t) = u_0 + t(u_1 - u_0)$, attributing the decoded change to coordinates in both z and w. Integrals use 256-point Gauss–Legendre quadrature in double precision, and signed attributions are averaged across pairs.

A.3 CelebA-Glasses

Data. The resulting subset contains $31{,}563$ training and $4{,}149$ validation images from the official CelebA splits (Liu et al., 2014). We crop aligned RGB images to 148×148 pixels from row 40 and column 15, apply channelwise Gaussian smoothing with standard deviation $\sqrt{148/64}/2$, and resize to 64×64 using cubic interpolation. Pixel values are quantized to eight bits and scaled to $[0, 1]$; inputs to the pretrained encoder are rescaled to $[-1, 1]$.

Models and training. We adapt the `stabilityai/sd-vae-ft-mse` checkpoint of the latent diffusion VAE (Rombach et al., 2021). Both latent paths retain the native $4 \times 8 \times 8$ Gaussian representation. We apply LoRA to linear and convolutional weights with $r = 16$ and $\alpha_{\mathrm{LoRA}} = 32$. The conditional encoder receives the label as an additional image channel, and the joint decoder receives z and w by channel concatenation. Each adversary has one hidden layer of width 128 with batch normalization.

We jointly optimize the adapters and adversaries with Adam at learning rate 10^{-4}, batch size 128, and `bfloat16` mixed precision. We set $\lambda_{zw} = 0.9$, $\lambda_z = 0.1$, $\alpha_1 = \alpha_2 = 10^{-6}$, and $\lambda_{\mathrm{adv}} = 10^{-3}$. We retain the checkpoint with the lowest validation model objective and stop after 100 epochs without improvement.

Baselines are benchmarked with their encoder–decoder convolutional widths chosen to yield approximately 89 million trainable parameters, and are optimized with AdamW at learning rate 10^{-4}.

Evaluation. We evaluate all validation images using sampled latents. For $I(z; w)$, we use all native latent coordinates with $k = 20$, the maximum norm, and natural logarithms, implemented in `infomeasure` (Kraskov et al., 2003; Büth et al., 2025). Trainable parameter counts include the adversaries.

Attribution. For CelebA, we define the classifier score $s(x)$ as the glasses logit minus the no-glasses logit of the independent classifier.

A.4 IFN-β Stimulation

Data and preprocessing. We retain cells with at least 200 detected genes and genes detected in at least three cells. Counts are normalized to the median library size and log-transformed. We select $2{,}000$ highly variable genes using the Seurat procedure and match gene symbols exactly to the scGPT vocabulary, leaving $1{,}803$ genes. Following the scGPT input representation, nonzero expression values are discretized within each cell into 50 quantile bins, with zeros retained at zero. Ties are randomized with a fixed seed, and the resulting values are divided by 50.

We use the existing cell-level split of $17{,}192$ training, $3{,}684$ validation, and $3{,}686$ test cells.

Backbone and decoder. The scGPT encoder contains 12 Transformer layers with eight attention heads and hidden dimension 512. Each input contains a CLS token and the $1{,}803$ retained gene tokens. Gaussian posteriors are parameterized from the CLS representation as described above. For the attribute-associated path, the stimulation label is appended as an additional expression channel.

We construct the observation decoder from the pretrained scGPT MVC gene-query representation. For latent representation c, expression for gene g is decoded as

$$D(c)_g = \frac{\langle W_c c,\, \sigma(W_q e_g + b_q) \rangle}{50}, \tag{18}$$

where e_g denotes the pretrained gene embedding. The gene queries are fixed, so the decoder is linear in c. We use $c = z$ for the marginal decoder and $c = [z, w]$ for the joint decoder.

Before fitting DisCoLoRA, we calibrate the decoder to reconstruct training expression from frozen scGPT CLS representations. We select the checkpoint by validation reconstruction MSE using learning rate 10^{-3}, batch size 512, and early-stopping patience 50.

Training. We use rank $r = 64$ with $\alpha_{\text{LoRA}} = 32$. LoRA is applied to the Transformer linear layers, including attention query–key–value and output projections, and to the added Gaussian posterior projection, implemented as a kernel-size-one convolution on the CLS representation.

We jointly optimize the adapters and adversaries with Adam at learning rate 10^{-5} and batch size 64, using `bfloat16` mixed precision without a learning-rate schedule. We set $\lambda_{zw} = 0.9$, $\lambda_z = 0.5$, $\alpha_1 = \alpha_2 = 10^{-6}$, and $\lambda_{\text{adv}} = 0.3$.

The reconstruction and latent adversaries are MLPs $1803 \to 16 \to 2$ and $512 \to 16 \to 2$, respectively, with batch normalization and ReLU activation in the hidden layer. We use one posterior sample per cell for both training and validation objectives. We retain the checkpoint with the lowest validation model objective and stop after 100 epochs without improvement.

Embedding analysis. We compute K-means NMI from the complete sampled z and w representations, using stimulation or cell-type labels as appropriate. Comparison values for the variational baselines are taken from Slavutsky et al. (2025).

For visualization, we fit separate UMAP projections to the 512-dimensional sampled z and w representations of the training cells and transform the validation cells with the corresponding projection. For expression-space panels, we fit a single UMAP to normalized training expression and use it to transform the original, joint, and marginal reconstructions of the validation cells. Thus, all three expression panels share the same projection.

Gene attribution. For each validation cell, we select an opposite-condition training cell from the same donor and cell type and orient the pair from control to IFN-β stimulation. Holding the recipient's sampled z fixed, let w_{ctrl} and w_{stim} denote the sampled attribute-associated representations of the two endpoints. We apply IG with $v = w_{\text{stim}} - w_{\text{ctrl}}$ and sum over the coordinates of w. For gene g,

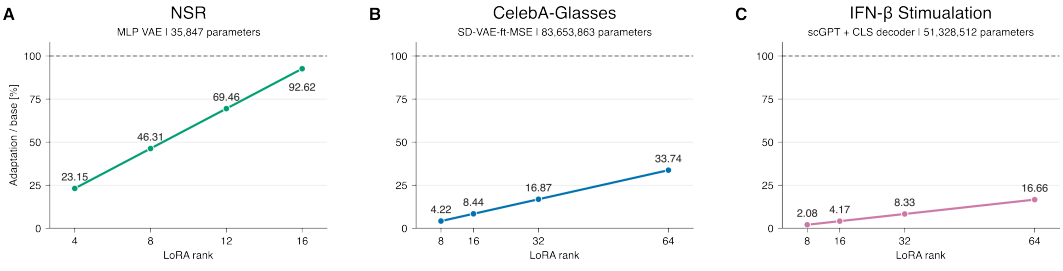
$$\text{IG}_g^w = D_{zw,g}(z, w_{\text{stim}}) - D_{zw,g}(z, w_{\text{ctrl}}). \tag{19}$$

This equality is exact because the scGPT observation decoder is linear in w for fixed z.

We first average attributions over recipient cells within each donor and then equally across donors, separately for each cell type. The resulting values are signed reconstructed-expression changes on the normalized 50-bin expression scale.

Gene Ontology enrichment. For each gene, we average cell-level signed IG expression attributions within donor and then equally across donors. We take the upper and lower 5% of these donor-balanced scores (91 genes per tail) and perform separate Gene Ontology Biological Process enrichment analyses with Enrichr (Chen et al., 2013; Kuleshov et al., 2016; Xie et al., 2021) using the `GO_Biological_Process_2026` library (G et al., 2025) and all $1{,}803$ modeled genes as the background. We retain terms with at least two overlapping genes, fold enrichment greater than one, and Benjamini–Hochberg adjusted $P < 0.05$.

B Supplementary Materials



Supplementary Figure S1: Parameter scaling across the three experiments. **A:** Noisy Swiss Roll with VAE backbone. **B:** CelebA-Glasses with Stable Diffusion VAE backbone. **C:** IFN-β Stimulation with scGPT.

Supplementary Table S1: Information measures on the Noisy Swiss Roll dataset. Among adapter ranks, $r = 4$ has the lowest mean $I(y; z)$, $r = 12$ the lowest mean $I(z; w)$, and $r = 8$ the lowest mean Δ-Bayes. Values are mean $\pm$ standard deviation across five initializations; underline marks the lowest mean within each group of models.

Model	$I(y; z) \downarrow$	$I(z; w) \downarrow$	Δ-Bayes$(w) \downarrow$ [pp]
CSVAENA	0.077 ± 0.010	1.038 ± 0.271	29.908 ± 0.029
CSVAE	0.040 ± 0.037	0.560 ± 0.632	29.858 ± 0.067
HCSVAENA	0.069 ± 0.009	0.094 ± 0.016	29.994 ± 0.009
HCSVAE	0.019 ± 0.011	$\underline{0.020 \pm 0.009}$	29.992 ± 0.008
DIVA	0.063 ± 0.021	3.366 ± 0.709	10.520 ± 6.891
CCVAE	0.062 ± 0.013	3.488 ± 0.519	10.076 ± 6.194
DisCoVR	$\underline{0.009 \pm 0.004}$	0.072 ± 0.020	$\underline{0.544 \pm 0.271}$
DisCoLoRA ($r = 4$)	$\underline{0.021 \pm 0.006}$	0.134 ± 0.082	0.846 ± 0.539
DisCoLoRA ($r = 8$)	0.028 ± 0.011	0.105 ± 0.053	$\underline{0.356 \pm 0.191}$
DisCoLoRA ($r = 12$)	0.027 ± 0.008	$\underline{0.077 \pm 0.017}$	0.496 ± 0.122
DisCoLoRA ($r = 16$)	0.031 ± 0.013	0.307 ± 0.392	0.506 ± 0.434

Supplementary Table S2: Reconstruction and latent dependence on CelebA-Glasses. DisCoLoRA uses the selected rank $r = 16$. Marginal reconstruction MSE is reported only for methods with an explicit marginal reconstruction. Bold denotes the lowest mean value in each column.

Model	Joint MSE $\downarrow$	Marginal MSE $\downarrow$	$I(z; w) \downarrow$
CSVAENA	0.031 ± 0.004	N/A	0.053 ± 0.001
CSVAE	0.009 ± 0.001	N/A	0.068 ± 0.021
HCSVAENA	0.267 ± 0.024	N/A	0.035 ± 0.018
HCSVAE	0.267 ± 0.025	N/A	0.035 ± 0.011
DIVA	0.009 ± 0.001	N/A	0.0341 ± 0.008
CCVAE	0.008 ± 0.001	N/A	0.029 ± 0.005
DisCoVR	0.008 ± 0.001	0.018 ± 0.001	0.015 ± 0.002
Stable Diffusion VAE (Unadapted)	0.086	N/A	N/A
DisCoLoRA ($r = 16$)	$\mathbf{0.002 \pm 0.0001}$	$\mathbf{0.004 \pm 0.0001}$	$\mathbf{0.007 \pm 0.004}$

Supplementary Table S3: K-means NMI for embeddings across stimulation (y) and cell type (shared structure) on the Kang dataset. Higher w–stimulation and z–cell type NMI indicate stronger recovery of attribute-associated and shared structure, respectively, while lower z–stimulation NMI indicates less residual stimulation information in the shared representation. The base model scGPT has a single representation, which is indicated under z. Bold denotes the best value in each column. *Values obtained from (Slavutsky et al., 2025).

Model	w–Stimulation $\uparrow$	z–Cell Type $\uparrow$	z–Stimulation $\downarrow$
CSVAE – No Adv.*	$\mathbf{0.949 \pm 0.003}$	0.702 ± 0.015	0.187 ± 0.000
CSVAE*	0.939 ± 0.002	0.406 ± 0.001	$\mathbf{0.002 \pm 0.000}$
HCSVAE – No Adv.*	0.933 ± 0.006	0.628 ± 0.016	0.091 ± 0.002
HCSVAE*	0.931 ± 0.005	0.433 ± 0.001	0.003 ± 0.000
DIVA*	0.801 ± 0.000	0.628 ± 0.011	0.056 ± 0.000
CCVAE*	0.604 ± 0.000	0.683 ± 0.016	0.103 ± 0.000
DisCoVR*	0.946 ± 0.002	0.688 ± 0.000	$\mathbf{0.002 \pm 0.000}$
scGPT (adapters off)	N/A	0.210	0.044
DisCoLoRA ($r = 64$)	0.757 ± 0.025	$\mathbf{0.723 \pm 0.063}$	0.005 ± 0.001