# Deep identifiable modeling of single-cell atlases enables zero-shot query of cellular states

Mingze Dong[1,2,3*] and Yuval Kluger[1,2,4*]

[1]Interdepartmental Program in Computational Biology & Bioinformatics, Yale University, New Haven, CT, USA.

[2]Department of Pathology, Yale School of Medicine, New Haven, CT, USA.

[3]Department of Biomedical Engineering, Yale University, New Haven, CT, USA.

[4]Applied Mathematics Program, Yale University, New Haven, CT, USA.

*Corresponding author(s). E-mail(s): mingze.dong@yale.edu; yuval.kluger@yale.edu;

**Abstract**

With the emerging single-cell RNA-seq datasets at atlas levels, the potential of a universal model built on existing atlas that can extrapolate to new data remains unclear. A fundamental yet challenging problem for such a model is to identify the underlying biological and batch variations in a zero-shot manner, which is crucial for characterizing scRNA-seq datasets with new biological states. In this work, we present scShift, a mechanistic model that learns batch and biological patterns from atlas-level scRNA-seq data as well as perturbation scRNA-seq data. scShift models genes as functions of latent biological processes, with sparse shifts induced by batch effects and biological perturbations, leveraging recent advances of causal representation learning. Through benchmarking in holdout real datasets, we show scShift reveals unified cell type representations as well as underlying biological variations for query data in zero-shot manners, outperforming widely-used atlas integration, batch correction, and perturbation modeling approaches. scShift enables mapping of gene expression profiles to perturbation labels, and predicts meaningful targets for exhausted T cells as well as a list of diseases in the CellxGene blood atlas.

## Main

With advancements in single-cell technologies, emerging works generate millions of single-cell RNA measurements across different cell types and biological states. The incorporation of large-scale scRNA-seq datasets into computational models is accelerated by consortia such as Tabula Sapiens [1], Human Cell Atlas (HCA) [2], Human Biomolecular Atlas Program (HuBMAP) [3], the Cellular Senescence Network (SenNet) [4] that collects massive sequencing data, and data standardization and curation efforts such as Chan-Zuckberg CellxGene census [5], Azimuth [6], and gget [7].

With the rapid accumulation of large-scale scRNA atlas datasets, a number of computational methods were proposed to address the challenges in analyzing these datasets. These methods considered tasks including label-aware or transfer-learning based atlas integration [8–21], online batch correction [22, 23], and perturbation modeling [24–30]. Several recent works considered the possibility of foundation models trained on atlas datasets [31–37], as an analogy to successful large models in computer vision and natural language processing [38–42].

However, an essential property of an ideal universal model that is missing in existing models is the zero-shot learning ability. In machine learning, zero-shot learning refers to a model's ability to directly return accurate outputs to new datasets with unseen (combinations of) properties [43]. In the single-cell context, given a new dataset of novel diseases, an ideal model should be able to reveal meaningful representations of the disease states as well as the cell types from the gene expression alone. Zero-shot learning is especially useful for characterizing new datasets with diverse single-cell behavior, such as varying immune responses, where defining cell states is challenging. Successful large models in computer vision and natural language processing have shown impressive zero-shot learning abilities by high-quality responses to arbitrary prompts [38, 39, 42]. However, as far as we know, existing models for scRNA-seq atlas analysis either project new datasets to seen cell states [22, 23, 35], or require fine-tuning on the new dataset [20, 21], thus the potential of a real zero-shot model remains to be explored.

A key conceptual barrier preventing single-cell computational models from being zero-shot learners is the non-identifiability between batch effects and latent biological effects. Both batch effects and latent biological effects can drive the difference between datasets. In atlas integration, an expressive model may only learn to preserve cell type information and discard all batch effects as well as latent biological differences across samples. In this case, the model may act perfectly in batch correction and cell type annotation benchmarks, yet cannot provide any information regarding unobserved biological states. Moreover, due to the lack of identifiability, the model may arbitrarily attribute the discrepancy between the query and reference datasets as batch or biological effects. While emerging works consider sophisticated network architectures and training schemes, the problem of non-identifiability is fundamental and cannot be solved by solely modifying the network architecture, scaling up the training dataset, or specific fine-tuning.

In this work, we present scShift, a novel framework that enables zero-shot learning of new scRNA-seq datasets. scShift presents two fundamental advancements compared with prevailing scRNA-seq atlas models. Firstly, in the pretraining phase, scShift utilizes perturbation data which provides true biological effect label to enforce correct attribution of batch and biological effects in single-cell atlas data. Secondly, scShift provides a novel implementation of the sparse mechanism shift regularization adopted from recent advances in nonlinear independent component analysis (ICA) identifiability theory [27, 44]. scShift automatically identifies the latent components corresponding to batch signals, perturbation / biological signals, and cell-type specific signals. This allows the model to directly return a new dataset's cell types, biological states, and sequencing batch in the form of latent space, without the need of its annotation labels. Through benchmarking, we have found scShift learns interpretable latent spaces for cell types and drug perturbation, and is able to generate highly accurate zero-shot inference of the latent space on real hold-out datasets. scShift is able to map cellular states to targets within the training perturbation pool, and offers insightful target predictions for exhausted T cells and a list of diseases covered in the CellxGene blood atlas.

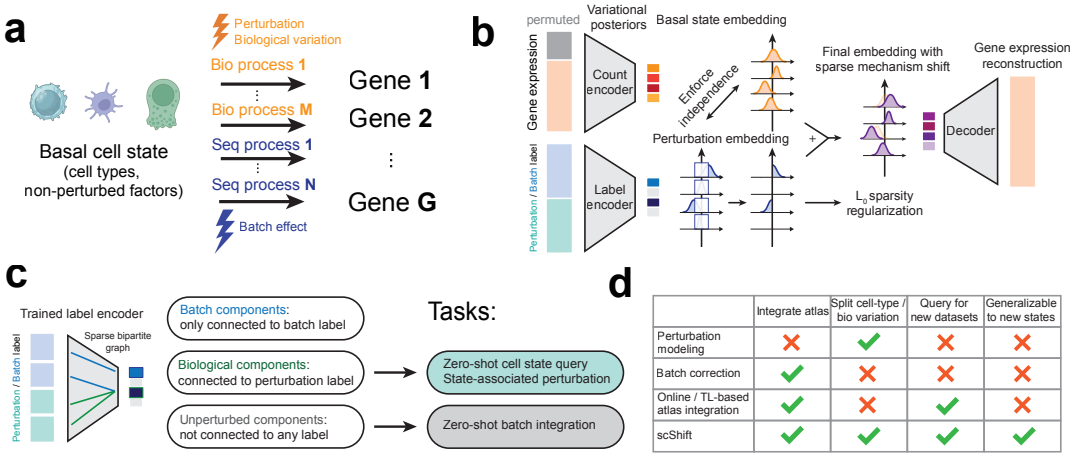



**Fig. 1** Overview of the scShift framework. **a.** The mechanistic generative process considered in the scShift model. Gene expression is the result of basal cell states upon sparse biological and batch perturbation. **b.** Workflow of the scShift model. scShift encodes both gene expression and batch / perturbation labels into the latent space, and models the sparse mechanism shift as well as the independence between basal state and label in the latent space. **c.** Downstream interpretation of trained scShift models and corresponding tasks. **d.** Comparison of the scope of the scShift model with existing models for single-cell atlas data.

# Results

## scShift separates different sources of variation via identifiable mechanistic modeling

scShift considers a mechanistic generative process of cellular gene expression as a function of basal cell state, biological processes, and sequencing processes (Fig. 1a). While the transcriptomic difference across samples can be attributed to both batch effects and biological effects, we additionally include perturbation datasets into the training set, where we assume different perturbations only contribute to biological variation.

We constructed a deep variational inference framework to infer the parameters in the generative process. In order to enforce component-wise (permutation) identifiability, we adopted the sparse mechanism shift framework [27, 44], which assumes different batch or biological factors introduce sparse changes on the latent space. We propose a novel implementation of the sparse mechanism shift by extending the stochastic gate approach to the label variational posterior [45–47] (See methods). Finally, we add (not concatenate) the basal state latent variables and the sparse shift latent variables to create the final embedding, which is used to reconstruct full gene expression. Inspired from modern pretraining strategies, we incorporated noise in the training set by permutation of a small random set (25%) of gene expression over cells within each batch (Fig. 1b).

However, by the sparsity regularization, a model may encode all information in the basal state latent variables resulting in meaningless perturbation latent space. Therefore, we additionally regularize the basal state to enforce its independence with the perturbation space (See methods, Fig. 1b). After pretraining, the model returns a sparse encoding of each batch / perturbation condition in the latent space. A sparse bipartite graph thus can be established between the latent components and batch / perturbation labels. For each component in the latent space, we define batch components as those only connected to the batch label; biological components as those connected to perturbation labels, regardless whether they are connected to batch labels; unperturbed components are neither connected to batch nor perturbation labels. The biological components represent latent biological processes inducible by experimental perturbations, which can be used for querying cell state and furthermore perturbation targets that are able to induce or reverse the state. The unperturbed components correspond to cellular basal states, which can be used for atlas integration (Fig. 1c). Compared with existing deep modeling approaches, scShift simultaneously covers the integration of atlas and the separation of variations, supporting zero-shot queries for new datasets. scShift uniquely enables interpretation of novel biological states using perturbations in training set (Fig. 1d).

## scShift learns interpretable latent space from blood immune atlas

While large scRNA-seq atlas datasets are already available for most tissues or organs, the perturbation datasets are almost all based on cancer cell lines, which has clear biological difference with *in vivo* cells. Therefore in this study, we employed the only large-scale perturbation dataset we were aware of that were conducted on sampled human cells: the open-source drug perturbation dataset from the Kaggle competition "Open Problems in Single Cell Analysis", which covers 144 drug perturbations in human blood cells [48]. Moreover, we adopted the blood immune atlas from CellxGene [5], which contains 8.7 million cells from 2541 human donors in 30 studies. In order to make the dataset sizes comparable between the atlas dataset and the perturbation dataset, we subsampled the atlas dataset to 1 million cells (Fig. 2a). Preliminary analysis using PCA embedding of normalized highly variable genes suggests large batch variation across samples (Fig. 2b). After scShift pretraining, the unperturbed components returned shows clear separation by cell types and reasonable mixing of sequencing assay types and disease states in the UMAP space (Fig. 2c, Extended data fig. 1a). Visualizations of biological components (Fig. 2d, Extended data fig. 1b-c) and unperturbed plus biological components (Extended data fig. 1d-e) shows various disease-specific clusters, while agnostic to or clearly separated by cell types respectively. This indicates that the disease states are automatically characterized by scShift biological components as anticipated. Concatenation of the batch components in the scShift embedding resulted in a number of additional small assay-specific clusters that may correspond to batch effects (Extended data fig. 1f).

Next, we compared scShift with other baseline atlas-integration approaches, including scVI, Harmony, and SCALEX [8, 9, 23]. We also note that scShift has a variational inference backbone similarly to scVI. Therefore, comparing scShift with scVI can also unravel the added value of its additional parts, including sparse encoding of perturbations and enforcement of independence across basal and perturbation latent spaces. Qualitatively, we observed that the scVI latent space is still biased by sequencing assays, while Harmony and SCALEX shows separated clusters within the same cell types (Extended data fig. 1g-i). We quantitatively benchmarked the scShift embeddings with baselines along two axes. First, we evaluated if the scShift unperturbed embedding

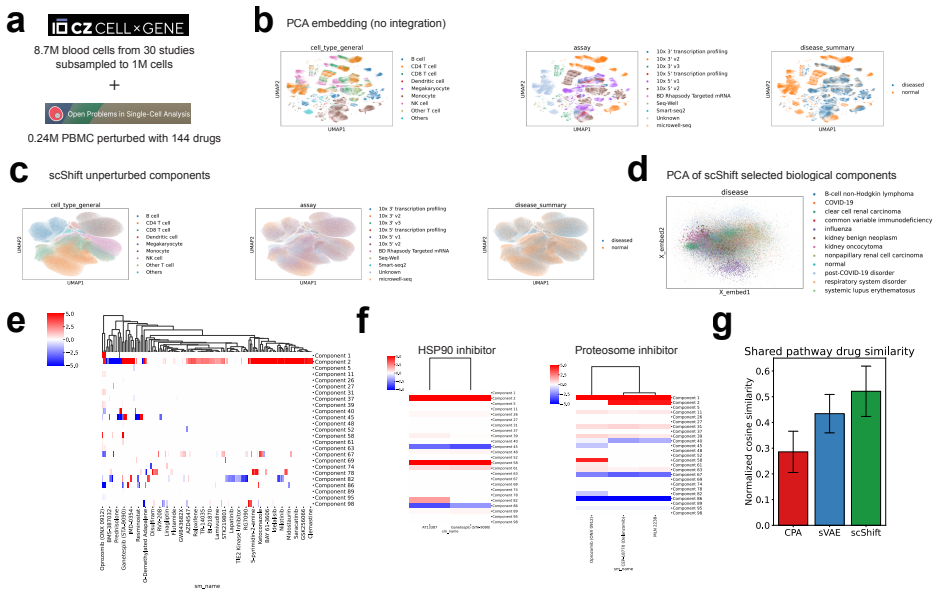



**Fig. 2** Training and validation of scShift on blood atlas data together with drug perturbation. **a.** The training set is composed of subsampled 1 million blood cell data from CellxGene atlas, and the open-source drug perturbation data including 144 drug perturbations in 0.24 million blood cells. **b.** UMAP visualization of the preprocessed PCA embedding, colored by summarized cell type (left), sequencing assay (middle) and disease state (right). **c.** UMAP visualization of the scShift unperturbed embedding, colored by summarized cell type (left), sequencing assay (middle) and disease state (right). **d.** PCA visualization of selected scShift biological components, colored by disease state with rescaled dot sizes. **e.** Heatmap showing the drug perturbation occurrence in scShift biological components. **f.** Heatmap showing perturbation occurrence of representative drugs from shared pathways in scShift biological components. **g.** Bar plot of shared pathway drug similarity across different perturbation modeling approaches. Data are presented as mean values +/- standard error (n=78 denotes the number of drug pairs).

preserved the cell type information and removed batch effects. We anticipate the baseline models to perform reasonably well for this task, as they primarily aim to find "batch-corrected" embeddings that match shared information (such as cell type) across observed datasets. Indeed, we found that baseline methods succeeded in batch removal metrics (SCALEX) or cell type preservation metrics (Harmony), while the scShift unperturbed embedding achieved a balanced performance in batch removal and cell type preservation (Extended data fig. 2a). Next, we evaluated the disentanglement between the scShift unperturbed and biological components via removal of disease / cell type information. We also evaluated their concordance to the respective labels (cell type / disease) via label preservation metrics (See Methods). We observed that scShift achieved the highest disentanglement score and second-highest concordance score compared with other models (Extended data fig. 2b). Together, these results indicate the success of scShift in attributing different sources of variations to respective embeddings.

Furthermore, we validated the quality of the scShift perturbation latent space after pretraining. We observed overall heterogenous sparse patterns for different drugs as anticipated (Fig. 2e). In order to quantitatively benchmark the generated drug embedding, we hypothesized drug pairs with same targets should have similar perturbation embeddings, compared with randomly sampled drug pairs. This is qualitatively supported by the heatmap of example drugs with shared pathways (Fig. 2f). We further quantified the normalized cosine similarity for scShift and compared the performance with that of two deep perturbation modeling approaches, namely compositional autoencoder (CPA) and sparse variational autoencoder (sVAE) [27, 29]. We observed that scShift has a notable advantage as shown by the higher normalized cosine similarity (Fig. 2g). Moreover, with a more detailed investigation, we found that CPA cannot generate sparse patterns (Extended data fig. 3a-c). sVAE performed sparsification of components, yet does not learn a reasonable sparse latent space as shown by highly non-sparse patterns of drug effects in the identified components (Extended data fig. 3d-f). Together, these results suggest scShift reveals highly interpretable latent spaces compared with alternative approaches.

## scShift enables zero-shot query of cell type and biological variation

Next, we performed comprehensive benchmarks on scShift's ability to separate cell type and biological variations in a zero-shot manner. In our benchmarking, we included batch correction methods which require

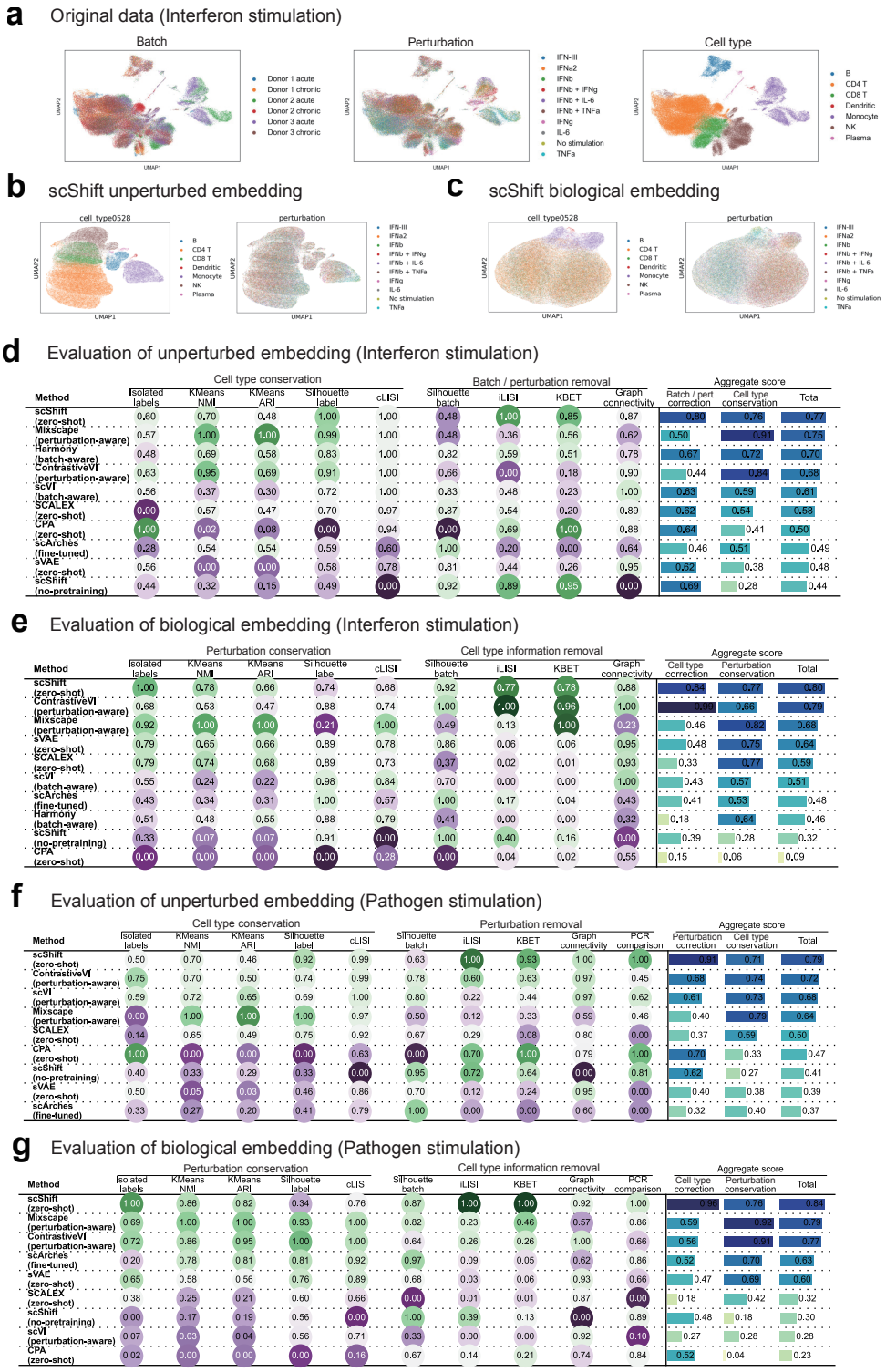



**Fig. 3** scShift uniquely identifies cell type and biological variations in holdout datasets in a zero-shot manner. **a.** UMAP visualization of the PCA embedding of the inteferon stimulation data, colored by batch, stimulation condition, and cell type annotation. **b.** UMAP visualization of the scShift unperturbed embedding for the inteferon stimulation data, colored by batch, stimulation condition, and cell type annotation. **c.** UMAP visualization of the scShift biological embedding for the inteferon stimulation data, colored by batch, stimulation condition, and cell type annotation. **d.** Evaluation of different methods' unperturbed / integrated / background embeddings in the interferon stimulation dataset. **e.** Evaluation of different methods' biological / integrated / salient embeddings in the interferon stimulation dataset. **f.** Evaluation of different methods' unperturbed / integrated / background embeddings in the pathogen stimulation dataset. **g.** Evaluation of different methods' biological / integrated / salient embeddings in the pathogen stimulation dataset. In all evaluations, each column is normalized to have min 0 and max 1.

additional batch label inputs (Harmony, scVI) [8, 9], label-unaware approaches including transfer-learning based scArches with scVI backbone [20] and an online batch integration method SCALEX [23], and four perturbation modeling approaches (Mixscape, contrastiveVI, CPA, sVAE) [24, 25, 27, 29]. Among the perturbation modeling approaches, Mixscape and contrastiveVI were directly trained on the query dataset with the additional perturbation label input, while we trained CPA and sVAE on the reference atlas dataset and tested their zero-shot learning abilities. We did not benchmark recently proposed foundation models (such as Geneformer [31] and scGPT [33]), due to their need of extensive fine-tuning, and recent works showing their limited zero-shot abilities against simpler baselines such as label-aware scVI in atlas integration [49].

An ideal dataset for our evaluation should contain definitive biological differences across samples. A cohort-based scRNA-seq dataset is thus not appropriate for the evaluation, since the difference across samples may be caused by a mixing of batch and biological effects. To this end, we employed two public datasets that cultured blood cells with 9 interferon stimulation conditions and 6 pathogen conditions respectively [26, 50]. The perturbation of the two datasets were not covered by the training perturbation pool, thus they are appropriate for benchmarking the zero-shot abilities of different models (Fig. 3a, Extended data fig. 6a). In UMAP visualizations, the scShift zero-shot latent space revealed cell type variation in the unperturbed space, while highlighting perturbation information and the differential response in monocytes [26, 50] (Fig. 4b-c, Extended data fig. 6k-l). Moreover, the unperturbed embedding of both datasets and the training datasets automatically showed overlap on the UMAP space (Extended data fig. 5i). We next quantitatively benchmarked different methods' cell type (unperturbed) representation via cell type preservation and removal of batch and perturbation covariates, and their biological representation via preservation of the perturbation information and removal of cell type information. These objectives are well covered by batch integration metrics implemented in the scIB benchmarking framework [19]. In order to account the differential response across cell types to a perturbation, for the evaluation of the biological represenation, we selected cell types with shared response patterns according to the results established in original works (See Methods) [26, 50]. Our results suggest that in both datasets, scShift outperformed all other methods from different categories in identifying accurate cell type representation and biological states (Fig. 3d-g, Extended data fig. 4a-b). Specifically, scShift substantially outperformed other zero-shot methods and TL-based scArches. scShift still maintained a lead (sometimes marginal) over the label-aware methods, which are specifically designed for the batch integration or perturbation modeling tasks and not generalizable. Our quantitative results were also supported by qualitative visualizations, where other zero-shot methods yield suboptimal performance in various aspects. For example, sVAE, scArches and SCALEX failed to remove the perturbation-associated variation in the cell type representation, while CPA showed limited zero-shot perturbation removal ability and did not separate cell types well in the latent space (Fig. 3b-c, extended data fig. 4c-f, 5-6). Together, our results indicate scShift stands out as the sole methodology capable of accurately inferring cell type and biological variations in a zero-shot manner.

## scShift reveals meaningful perturbation-disease mappings

Finally, we show the potential of scShift in finding novel perturbation targets for *in silico* engineering biological states including diseased cellular states and the exhausted T cell state. To minimize possible false positives for both cases, we conducted the analyses at the pseudo-bulk level, i.e., we averaged the biological components for each donor to generate pseudobulk profiles for diseased states, and created donor-cell-type pseudobulks for T cell subtypes in B-cell non-Hodgkin lymphoma patients. In the former case, due to the variety of individual characteristics (age, sex, etc) across different donors and studies, we employed the normalized overlap weight estimator to calculate the disease effects on scShift biological components, adjusting for confounders using propensity scores [51, 52] (See methods). In both cases, we observed that these abnormal cellular states have shared shift patterns in component 5, 27, 49, and 89 respectively (Fig. 4a, b). As each component in the latent space is enforced to be standard normally distributed, we anticipated these shifts represent condition-induced shifts related to pathological states. Next, we compared these components with the drug perturbation space to find drug targets that have opposite effect to reverse the disease states (Fig. 4c). At this stage, we excluded cancer and respiratory system disorder due to their primary source not being immune system, thus the meaning of reversing these disease states in immune cells remains unclear (Extended data fig. 7). Only a small subset of drugs were predicted to have a role in reversing these biological components, due to the sparse perturbation encoding in scShift. For the exhausted T cells, only JTC-801, a selective antagonist for the nociceptin receptor, is able to reverse the shift of component 5 in exhausted T cells. This aligns with previous researches suggesting the opioid receptor-like (ORL1) / nociceptin system down-regulating the immune system function [53]. Furthermore, the drug I-BET151 is predicted by scShift to reverse the cellular states in systemic

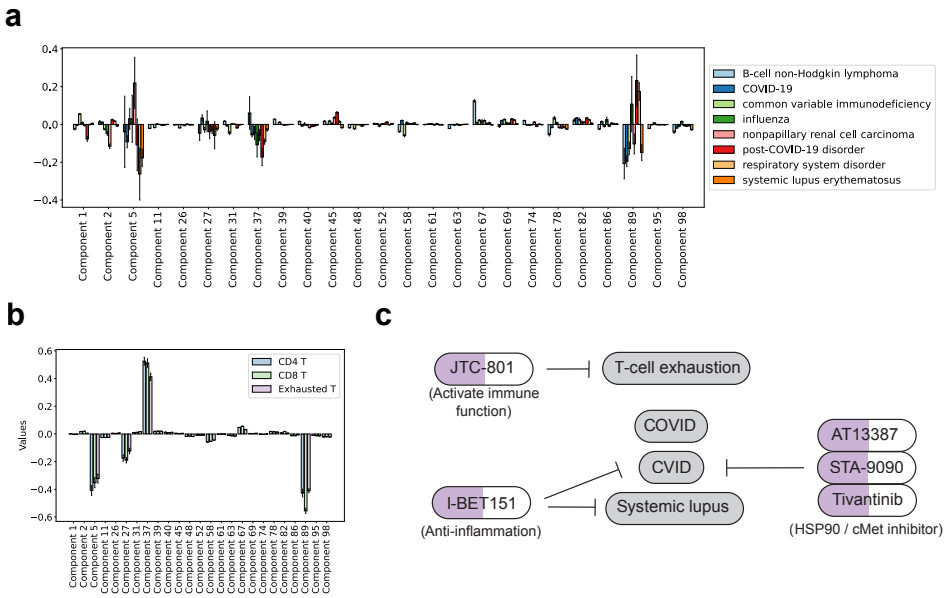



**Fig. 4** scShift identifies meaningful drug targets for exhausted T cells and disease states. **a.** Bar plot of scShift biological component values for diseases. Data are presented as average treatment effect +/- standard error by nonparametric bootstrap (patient number=24, 503, 11, 17, 10, 16, 20, 162). **b.** Bar plot of scShift T cell biological component values. Data are presented as mean values +/- standard error (n=24). **c.** Illustration of identified drug-disease mapping with scShift.

lupus erythematosus and common variable immunodeficiency (CVID) by component 27. This is consistent with the systemic inflammation observed in both diseases and previous studies showing the potential of I-BET151 in anti-inflammation [54–56]. Finally, HSP90 inhibitors AT13387, STA-9090 and cMet inhibitor Tivantinib, all experimental anti-cancer drugs, is predicted by scShift to reverse the disease state in component 89 for diseases including COVID, CVID, and systemic lupus erythematosus. These three drugs are anti-cancer drugs with complex biological roles due to the targets' functions in various pathways [57–61]. The roles of these drugs in these diseases remain unclear. Together, scShift predicts meaningful drug targets for reversing dysfunction / disease cellular states and provide novel insights for further drug repurposing studies.

# Discussion

We presented scShift, an identifiable variational inference framework that enables zero-shot query of cellular states. scShift learns a highly interpretable latent space, uniquely identifies accurate cell types and biological states in a zero-shot manner, and maps between perturbation and cell biological states. As far as we know, scShift is the first deep model showing remarkable zero-shot learning abilities for scRNA-seq datasets, enabling dissection of single-cell "phenoscapes" at the atlas level [62]. Our work also sheds light on predictive modeling of perturbations on unseen cell states or unseen perturbations as future work.

Due to the limited perturbation data for *in vivo* cells, we only trained the scShift model on blood immune cells, where cell culture and drug perturbation were available. A challenge for transfering the knowledge in the cell line data to real tissue cells is the existence of distinct biological processes, which would provide additional complexity for the model to distinguish difference in biological states and the difference of basal cell states. Integrating epigenomic or prioritizing gene regulatory information may help guide the model for better transfer learning [21, 63, 64]. Another possibility to improve the zero-shot ability of the scShift model is the size of the perturbation pool. While the current 144 drug screen already covers a variety of biological processes in immune cells, there may be additional biological processes outside the perturbation pool, which may be instead attributed as batch effect or non-perturbable effect. While our benchmarking results suggest that the current data may be sufficient for the model to reveal interferon and pathogen responses, expanding the drug pool may lead to improved universal zero-learning abilities and eventually systematic data-driven drug repurposing. Additionally, incorporating data during different developmental stages and cellular reprogramming may improve the model's power of *in silico* engineering for cellular states. Finally, while scShift provides a proof-of-concept for zero-shot learning in scRNA-seq data, the potential of identifiable deep models with more sophisticated design remains a promising direction to explore.

# Data availability

All datasets analyzed in this paper from previous publications are publicly available. The CellxGene blood atlas dataset were downloaded from CellxGene census (https://chanzuckerberg.github.io/cellxgene-census/) via gget [7]. The drug perturbation dataset was downloaded from Kaggle (https://www.kaggle.com/competitions/open-problems-single-cell-perturbations). The interferon stimulation dataset was downloaded from dryad (https://datadryad.org/stash/dataset/doi:10.5061/dryad.4xgxd25g1). The pathogen stimulation dataset (EGAS00001005376) was downloaded from https://eqtlgen.org/sc/datasets/1m-scbloodnl.html.

# Code availability

We have made scShift available as a public open-source Python package, which as well as our trained models can be accessed at https://github.com/MingzeDong/scShift. scShift is implemented using scvi-tools [65].

# Acknowledgements

Y.K. disclose support for the research of this work from NIH [R01GM131642, UM1PA051410, U54AG076043, U54AG079759, U01DA053628, P50CA121974, and R33DA047037].

# Author contributions

M.D. conceived the study, developed scShift and performed analyses in the work. Y.K. provided overall supervision of the study. M.D. and Y.K. wrote the manuscript.

# Competing interests

The authors declare no competing interests.

# Methods

## Generative model of scShift

scShift assumes the following generative process for modeling the distribution of entries $x_{ig}$ in the count matrix of the single-cell RNA-sequencing data:

$$\begin{aligned} z_i &\in \mathbb{R}^d, z_i \sim \text{Normal}(0, I) \\ s_i &\in \mathbb{R}^d, z_i \perp s_i, \text{supp}(s_i) = \mathbb{R}^d \\ \rho_i &= f(z_i + s_i, b_i) \\ w_{ig} &\sim \text{Gamma}(\rho_{ig}, \theta_g) \\ y_{ig} &\sim \text{Poisson}(l_i w_{ig}) \\ h_{ig} &\sim \text{Bernoulli}(l_i w_{ig}) \\ x_{ig} &= h_{ig} y_{ig}. \end{aligned} \tag{1}$$

In the generative process, the latent variable $z_i$ represents the basal state of cell $i$, and the latent variable $s_i$ indicate the perturbation received by cell $i$, which includes both biological and technical factors. We here do not specify the marginal distribution of $s_i$. Instead, we derive the distributional regularization for $s_i$ from the identifiability condition requirement (See next section). $b_i$ accounts for other possible experimental covariates such as sequencing assays. $\rho_i \in \mathbb{R}^p$ represents the mean of gene expression output by neural network decoder $f$. The function $f$ enforces $\sum_{g=1}^{p} \rho_{ig} \equiv 1$ for all $i$ through a soft-max layer. $\theta \in \mathbb{R}^p$ defines the gene-specific shape parameter of the Gamma distribution. $l_i$ denotes the library size of cell $i$ and is fixed as $l_i = \sum_g x_{ig}$. Taken together, $y_{ig}$ follows a negative binomial (NB) distribution by its Gamma-Poisson mixture representation. $h_{ig}$ represents the zero inflation level. Together, $x_{ig}$ is modeled as a sample from a zero-inflated negative binomial (ZINB) distribution. The generative process of scShift mostly aligned with the scvi-tools framework [10, 65], with the primary difference being the perturbation latent variable $s_i$.

## Sparse mechanism shift

Without additional constraints, it is known that the generative process is not identifiable [66, 67], meaning apart from the ground truth $z_i, s_i$, a family of latent variables $\tilde{z}_i, \tilde{s}_i$ may give rise to the same gene expression $x_i$ through different functions $f$. Each component in $\tilde{z}_i + \tilde{s}_i$ can be generated by a mixing of all components in the ground truth latent variables, leading to poor interpretability. In order to learn interpretable components, we should minimize the entanglement across components from different sources. In order to do that, we developed a novel implementation of the sparse mechanism shift framework [44]. The framework was originally established for time-series data [44], and was extended to scRNA-seq data in sVAE [27]. Specifically, denote the batch / perturbation label for cell $i$ as $\delta_i$, $p(s_i|\delta_i)$ defines a $d$-dimensional distribution. We assume the ground truth generative process $p$ maximizes the probability of each component being zero:

$$p \in \mathcal{P} = \{\text{argmax}_{\hat{p}} \mathbb{E}_{\delta_i} \sum_{j=1}^{d} \hat{p}(s_{ij} = 0|\delta_i) \text{ and satisfies (1)}\}. \tag{2}$$

While sVAE and our work both adopted the sparse mechanism shift framework in scRNA-seq data, the implementations were fundamentally different. In scShift, we formulate the sparse mechanism shift as a probabilistic $l_0$ loss by the stochastic gate approach [26, 45, 46] (See next section). On the other hand, sVAE adopted the sparse autoencoder design from [68], which defines the following generative process:

$$\begin{aligned} \pi_{ij}|\delta_i &\sim \text{Beta}(1, K) \\ \gamma_{ij}|\delta_i &\sim \text{Bernoulli}(\pi_{ij}) \\ z_{ij}|\delta_i &\sim \gamma_{ij} N(\mu_j(\delta_i), 1) + (1 - \gamma_{ij}) N(0, 1). \end{aligned} \tag{3}$$

sVAE does not formulate a sparsity regularization term. Instead the Beta prior for $\pi_{ij}$ in sVAE encourages sparsity therefore the generative process reflects the sparse mechanism shift assumption. However, through our experiments, we have found that while sVAE identifies a sparse set of components, the links between the perturbation labels and the components were dense (Extended data fig. 3d). This is also observed in the original publication where the effects of drugs on the learned latent space were mostly dense [27]. This means in practice, the prior helps identify the correct number of components (nodes in the bipartite graph), but not necessarily the graph structure (edges in the graph), thus inconsistent with the definition of sparse mechanism shift [44].

## Approximate posterior inference of scShift

In order to infer the parameters in the scShift model, we approximate the posterior distribution via variational inference, a standard framework for inferring parameters for deep probabilistic models. The posterior distribution is approximated by the following factorization:

$$q_\Phi(z_i, s_i|x_i, \delta_i, b_i) = q_{\Phi_z}(z_i|x_i, b_i) q_{\Phi_s}(s_i|\delta_i). \tag{4}$$

Here $\Phi$ denotes neural network weights that determine the variational posterior. After the factorization, the evidence lower bound (ELBO) can be derived via straightforward computation.

$$
\begin{aligned}
\log p(x_i|\delta_i, b_i) &= \log \int p(x_i, z_i, s_i|\delta_i, b_i) dz_i ds_i \\
&= \log \int \frac{p(x_i, z_i, s_i|\delta_i, b_i) q(z_i, s_i|x_i, \delta_i, b_i)}{q(z_i, s_i|x_i, \delta_i, b_i)} dz_i ds_i \\
&\geq \int q(z_i, s_i|x_i, \delta_i, b_i) \log \frac{p(x_i, z_i, s_i|\delta_i, b_i)}{q(z_i, s_i|x_i, \delta_i, b_i)} dz_i ds_i \\
&= \int q(z_i, s_i|x_i, \delta_i, b_i) \log \frac{p(x_i|z_i, s_i, b_i) p(z_i, s_i|\delta_i, b_i)}{q(z_i, s_i|x_i, \delta_i, b_i)} dz_i ds_i \\
&= \mathbb{E}_{q(z_i, s_i|x_i, \delta_i, b_i)} \log p(x_i|z_i, s_i, b_i) \\
&\quad - D_{\mathrm{KL}}(q(z_i, s_i|x_i, \delta_i, b_i)||p(z_i, s_i|\delta_i, b_i)) \\
&= \mathbb{E}_{q(z_i, s_i|x_i, \delta_i, b_i)} \log p(x_i|z_i, s_i, b_i) - D_{\mathrm{KL}}(q(z_i|x_i, b_i)||p(z_i|b_i)) \\
&\quad - D_{\mathrm{KL}}(q(s_i|\delta_i))||p(s_i|\delta_i)). \quad \text{(ELBO)}
\end{aligned} \tag{5}
$$

The first two terms coincide with the reconstruction and the KL divergence terms in vanilla variational autoencoders. For the the third term, while its value is not analytically tractable, we have

$$
\min_p D_{\mathrm{KL}}(q(s_i|\delta_i)||p(s_i|\delta_i)) \begin{cases} = 0 & \text{iff. } \exists \hat{p} \text{ s.t. } q \overset{\text{a.s.}}{=} \hat{p} \in \mathcal{P} \\ > 0 & \text{Else.} \end{cases} \tag{6}
$$

Therefore, if we assume the ground truth generative process $p$ is within the variational posterior family and the variational posterior family comprises only continuous distributions (so that $q$ and $p$ cannot differ on a measure of zero), then optimizing the ELBO term is equivalent to:

$$
\max \mathbb{E}_{q(z_i, s_i|x_i, \delta_i, b_i)} \log p(x_i|z_i, s_i, b_i) - D_{\mathrm{KL}}(q(z_i|x_i, b_i)||p(z_i|b_i)), \text{ s.t. } q \in \mathcal{P}. \tag{7}
$$

We are able to approximate the constrained optimization problem above by the following unconstrained optimization, where we denote $\lambda$ as the regularization parameter:

$$
\max \mathbb{E}_{q(z_i, s_i|x_i, \delta_i, b_i)} \log p(x_i|z_i, s_i, b_i) - D_{\mathrm{KL}}(q(z_i|x_i, b_i)||p(z_i|b_i)) + \lambda \mathbb{E}_{\delta_i} \sum_{j=1}^{d} q(s_{ij} = 0|\delta_i) \tag{8}
$$

The third term above now is a probabilistic $l_0$ loss term which can be explicitly derived using the stochastic gate approach [26, 45, 46]. We use the vanilla variational encoder design that yields $\hat{s}_i \sim N(\mu(\delta_i), \mathrm{diag}(\sigma^2(\delta_i)))$. Then $s_i$ is defined as the following entry-wise truncation of $\hat{s}_i$, where $(x)_+ = \max(x, 0)$ and $(x)_- = \min(x, 0)$.

$$
\forall j, s_{ij} = \begin{cases} (\hat{s}_{ij} - 0.1)_+ & \mu_j(\delta_i) \geq 0 \\ (\hat{s}_{ij} + 0.1)_- & \mu_j(\delta_i) < 0. \end{cases} \tag{9}
$$

As we anticipate each different perturbation or batch have effects on the gene expression, we relax the sparse mechanism penalty so that the strongest mechanism will not be penalized. In practice, this helps the model's robustness to the hyperparameter selection of $\lambda$, as the effect size of $\delta_i$ greatly varies across samples or perturbations. Denoting the Gaussian error function as $\mathrm{erf}()$, Then the non-zero probability of $s_{ij}|\delta_i$ can be expressed as follows:

$$
\begin{aligned}
q(s_{ij} \neq 0|\delta_i) &= \begin{cases} q(\hat{s}_{ij} > 0.1|\delta_i) & \mu_j(\delta_i) \geq 0 \\ q(\hat{s}_{ij} < -0.1|\delta_i) & \mu_j(\delta_i) < 0 \end{cases} \\
&= \frac{1}{2} - \frac{1}{2} \mathrm{erf}\left(\frac{|\mu_j(\delta_i)| - 0.1}{\sqrt{2}\sigma_j(\delta_i)}\right).
\end{aligned} \tag{10}
$$

Therefore we have

$$
\begin{aligned}
&\mathbb{E}_{\delta_i}[\sum_{j=1}^{d} q(s_{ij}=0|\delta_i) - \max_j q(s_{ij}=0|\delta_i)] \\
&= \mathbb{E}_{\delta_i} \sum_{j=1}^{d} (\frac{1}{2} + \frac{1}{2}\mathrm{erf}(\frac{|\mu_j(\delta_i)| - 0.1}{\sqrt{2}\sigma_j(\delta_i)}) - \max_i (\frac{1}{2} + \frac{1}{2}\mathrm{erf}(\frac{|\mu_j(\delta_i)| - 0.1}{\sqrt{2}\sigma_j(\delta_i)})).
\end{aligned} \tag{11}
$$

The formula above is the final probabilistic $l_0$ loss term used in our optimization. Apart from the $l_0$ regularization, an explicit regularization for the independence between $z_i$ and $s_i$ is needed for scShift model. In practice, the perturbation information is already largely included in the gene expression information space. Without additional regularizations, the probabilistic $l_0$ loss would enforce encoding all information into $z_i$, in which case the perturbation latent space $s_i$ is of constant zero thus meaningless.

We resolve the issue in our model using a carefully designed regularization term. We note that when $z_i$ and $s_i$ are independent, the distribution of $z_i + s_i|\delta_i$ can be approximated by the following distribution:

$$
p(z_i + s_i|\delta_i) \sim N(E(s_i|\delta_i), I + cov(s_i|\delta_i)). \tag{12}
$$

As a result, the discrepency between $N(E(s_i|\delta_i), I + cov(s_i|\delta_i))$ and $q(z_i + s_i|\delta_i)$ can serve as a indicator for the dependence level between $z$ and $s$. We quantify the discrepancy using the KL divergence, and only update $z$ based on the loss term. This term is incorporated into the ELBO term as an regularization with hyperparameter $\alpha$.

## Model interpretation

After the scShift model is trained, it returns both latent variables $z$ and $s$. $z$ represents the basal state latent space, and $s$ represents the effect of the input batch or perturbation label on the latent space. By averaging $s$ over cells with same labels, we obtain a matrix representing the weighted bipartite graph between the batch / perturbation labels and the latent components. Then we attribute each latent component as: 1. batch component, if it is only connected to batch labels; 2. biological component, if it is connected to perturbation labels; 3. unperturbed component, if it is not connected to any label. For a query dataset, we perform preprocessing to get the count matrix with matched gene names. Then we input the count matrix into the model to get the latent space $z$, and extract the batch, biological, and unperturbed components out of $z$ for downstream analyses.

## Model training

We trained the scShift model on subsampled 1 million CellxGenes blood atlas data and the Open problem in single-cell analysis data of 240090 cells from three donors with 144 drug perturbations. We selected highly variable genes (n_top_gene=5000) from the atlas dataset, using the Scanpy pipeline (library size normalization, log1p transformation, and highly variable gene selection) with default settings [69]. Then we concatenated the drug perturbation dataset with the blood atlas dataset. The final dataset comprises 1240090 cells and 5036 genes. The scShift model was initialized with 100 latent components and 1000 hidden components. The parameter number of the model is 28M (27,938,546). The model was trained on 90% of the dataset for 100 epoches on the dataset, with batch size = 400. We used the balanced batch sampler so that 2 cells from each perturbation / batch condition is included in each batch. The trained model is available in our Github repository.

## Benchmarking

For benchmarking the integrated representation on the original dataset, we compared the unperturbed embedding generated by scShift with the batch integrated embedding generated by Harmony [9], scVI [8] and SCALEX [23]. Our benchmarking was based on scIB [19], which quantifies the batch removal metrics and biological preservation metrics for a given embedding of a dataset. scIB requires batch key, label key and baseline pre-integrated embedding to be specified. We used the default settings in Harmony and SCALEX, and used scVI model settings compatible with scShift, with 100 latent compoents, 1000 hidden components, and trained for 100 epoches on the dataset. We used the default settings in scib-metrics package for benchmarking the scShift unperturbed embedding and batch integrated embeddings generated by Harmony, SCALEX and

scVI, with the batch key as the sequencing assay, and the label key as the coarse-grained cell type. For benchmarking the variation disentanglement and concordance, we computed the averaged batch integration score with batch key as cell type / disease state, and the label key as disease state / cell type. The results for both evaluations were averaged to generate the final table. For benchmarking efficiency, our benchmarking results were established on a subsampled 100,000 cell subset. Harmony and SCALEX was directly run on the subset, while scShift and scVI are trained on the full dataset.

For benchmarking the perturbation space, we compared the biological embedding generated by scShift with the perturbation space generated by CPA [29] and sVAE [27]. We trained CPA using the default setting for 100 epoches and sVAE (SpikeSlabVAE) using 100 latent components for 50 epochs. For computing the similarity measure for drugs with shared pathways, we first annotated the drugs based on their targets, then for each drug pair with shared pathways, we computed their cosine similarity. Null distribution of the cosine similarity was computed based on randomly sampled 1000 drug pairs. The cosine similarity for each drug pair with shared pathway is then normalized by substracting the mean of the null distribution then dividing by the standard deviation of the null distribution.

For benchmarking the zero-shot learning performance of scShift, we covered a list batch integration (Harmony [9], scVI [8]), transfer learning (scArches [20]), online batch integration (SCALEX [23]) and perturbation modeling approaches (the perturbation signature in Mixscape [24], ContrastiveVI [25], CPA [29], sVAE [27]). For the batch-aware methods including Harmony, and scVI, the batch label was input into the models. We implemented Harmony using default settings in Harmonypy. The scVI model was implemented with default settings. Mixscape and ContrastiveVI received perturbation labels as inputs. We implemented the perturbation signature calculation in Mixscape approach on the PCA components with default neighbor numbers to define its corresponding cell type representation and perturbation representation. The contrastiveVI model was implemented with suggested parameters in its Alzheimers tutorial and trained for 100 epoches with batch size 500. For scArches, we ran the reference query method based on the scVI model trained on the atlas plus perturbation dataset previously described. We used the SCALEX, CPA and sVAE models trained earlier on the atlas-level dataset to test their zero-shot abilities. To make the results directly comparable, we used the 5036 genes, derived from the highly variable gene of the CellxGene atlas dataset, in the benchmarking datasets as input for all methods.

We used two real holdout datasets for benchmarking the zero-shot learning performance, namely the interferon stimulation dataset [26] and the pathogen stimulation dataset [50]. For the interferon stimulation dataset, we used the full dataset from [26] containing 103518 cells. We adopted the PCA embedding from the original work as the baseline pre-integrated embedding [26]. We ran two evaluations on the unperturbed (cell type) representation across different methods. For both evaluations, we used the cell type annotation as the label key, while using batch / perturbation as the batch key respectively. The computed scIB metrics from the two evaluations were averaged to generate the final table. For the evaluation of the biological embedding, we quantified scIB metrics in CD8, CD4 and NK cells, using perturbation as the label key and cell type as the batch key. We found that for the dataset, most approaches generate negative PCR score, therefore this metric was excluded in the comparison. This metric was also noted to be excluded from a recent scRNA-seq atlas curation work [70]. For the pathogen stimulation dataset, we used the v3 dataset in the data repository containing 302598 cells. The baseline pre-integrated embedding was the PCA embedding computed from the top 2000 highly variable genes with library normalization and log1p transformation implemented with Scanpy default settings [69]. In this dataset, as there is no batch labels, we did not run Harmony, and the scVI model was trained in a perturbation-label aware manner. In the evaluation of unperturbed (cell type) representation, we used the cell type annotation as the label key, and perturbation as the batch key. In the evaluation of the biological embedding, we computed the scIB metrics in CD4 and CD8 T cells, using perturbation as the label key and cell type as the batch key.

In all evaluations, the label preservation score / batch correction score was computed by the average of the corresponding metrics. The final total score was 0.6 times label preservation score plus 0.4 times batch correction score, as proposed in the scIB work [19].

## Perturbation-disease mapping

For computing the disease-associated latent space shifts, we first extracted the disease cell's biological components in the latent embedding and averaged them for each donor as the donor pseudobulks. Three diseases (clear

cell renal carcinoma, kidney benign neoplasm, kidney oncocytoma) were excluded due to the low patient number ($< 5$). Next we extracted pseudobulk subsets for each disease and the normal condition. For each subset, we estimated the propensity score via logistic regression with covariates as sex, age, and cell type abundance, and the prediction variable to be the disease condition. Then we used the normalized overlap weight (OW) estimator based on the propensity score to estimate the average treatment effect and derived the standard error via nonparametric bootstrap. We used the nonparametric bootstrap 95% confidence interval to infer signficant biological components for each disease. For computing the T-cell-exhaustion-associated latent space shifts, we averaged the latent component value within each cell type for each donor to formulate the donor-cell-type pseudobulks. To identify the corresponding targets for the exhausted T cells, we performed paired Wilcoxon test on the difference of exhausted T cell latent components and the CD8 T cell latent components from same patients, and used 0.05 as the significance threshold for adjusted p-values. The drug targets were identified by comparing each strongly shifted component (component 5, 27, 37, 89) with the drug perturbation space. These four components were also the selected biological components shown in Fig. 2 and Extended data fig. 1.

# References


[1] TabulaSapiensConsortium* *et al.* The tabula sapiens: A multiple-organ, single-cell transcriptomic atlas of humans. *Science* **376**, eabl4896 (2022).

[2] Regev, A. *et al.* The human cell atlas. *elife* **6**, e27041 (2017).

[3] HubMAPConsortium. The human body at cellular resolution: the nih human biomolecular atlas program. *Nature* **574**, 187–192 (2019).

[4] SenNetConsortium. Nih sennet consortium to map senescent cells throughout the human lifespan to understand physiological health. *Nature aging* **2**, 1090–1100 (2022).

[5] ChanZuckerbergInitiative. Cz cellxgene discover. *https://cellxgene.cziscience.com/* (Accessed Oct 13 2023).

[6] Hao, Y. *et al.* Integrated analysis of multimodal single-cell data. *Cell* **184**, 3573–3587 (2021).

[7] Luebbert, L. & Pachter, L. Efficient querying of genomic databases for single-cell rna-seq with gget. *bioRxiv, 2022.05. 17.492392* (2022).

[8] Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nature methods* **15**, 1053–1058 (2018).

[9] Korsunsky, I. *et al.* Fast, sensitive and accurate integration of single-cell data with harmony. *Nature methods* **16**, 1289–1296 (2019).

[10] Xu, C. *et al.* Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Molecular systems biology* **17**, e9620 (2021).

[11] Büttner, M., Miao, Z., Wolf, F. A., Teichmann, S. A. & Theis, F. J. A test metric for assessing single-cell rna-seq batch correction. *Nature methods* **16**, 43–49 (2019).

[12] Haghverdi, L., Lun, A. T., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell rna-sequencing data are corrected by matching mutual nearest neighbors. *Nature biotechnology* **36**, 421–427 (2018).

[13] Shaham, U. *et al.* Removal of batch effects using distribution-matching residual networks. *Bioinformatics* **33**, 2539–2546 (2017).

[14] Stuart, T. *et al.* Comprehensive integration of single-cell data. *Cell* **177**, 1888–1902 (2019).

[15] Hie, B., Bryson, B. & Berger, B. Efficient integration of heterogeneous single-cell transcriptomes using scanorama. *Nature biotechnology* **37**, 685–691 (2019).

[16] Welch, J. D. *et al.* Single-cell multi-omic integration compares and contrasts features of brain cell identity. *Cell* **177**, 1873–1887 (2019).

[17] Barkas, N. *et al.* Joint analysis of heterogeneous single-cell rna-seq dataset collections. *Nature methods* **16**, 695–698 (2019).

[18] Amodio, M. *et al.* Exploring single-cell data with deep multitasking neural networks. *Nature methods* **16**, 1139–1145 (2019).

[19] Luecken, M. D. *et al.* Benchmarking atlas-level data integration in single-cell genomics. *Nature methods* **19**, 41–50 (2022).

[20] Lotfollahi, M. *et al.* Mapping single-cell data to reference atlases by transfer learning. *Nature biotechnology* **40**, 121–130 (2022).

[21] De Donno, C. *et al.* Population-level integration of single-cell datasets enables multi-scale analysis across samples. *Nature Methods* 1–10 (2023).

[22] Gao, C. *et al.* Iterative single-cell multi-omic integration using online learning. *Nature biotechnology* **39**, 1000–1007 (2021).

[23] Xiong, L. *et al.* Online single-cell data integration through projecting heterogeneous datasets into a common cell-embedding space. *Nature Communications* **13**, 6118 (2022).

[24] Papalexi, E. *et al.* Characterizing the molecular regulation of inhibitory immune checkpoints with multimodal single-cell screens. *Nature genetics* **53**, 322–331 (2021).

[25] Weinberger, E., Lin, C. & Lee, S.-I. Isolating salient variations of interest in single-cell data with contrastivevi. *Nature Methods* 1–10 (2023).

[26] Dong, M. *et al.* Causal identification of single-cell experimental perturbation effects with cinema-ot. *bioRxiv* 2022–07 (2022).

[27] Lopez, R. *et al.* Learning causal representations of single cells via sparse mechanism shift modeling. *Conference on Causal Learning and Reasoning* 662–691 (2023).

[28] Bunne, C. *et al.* Learning single-cell perturbation responses using neural optimal transport. *Nature Methods* 1–10 (2023).

[29] Lotfollahi, M. *et al.* Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular Systems Biology* e11517 (2023).

[30] Lotfollahi, M., Wolf, F. A. & Theis, F. J. scgen predicts single-cell perturbation responses. *Nature methods* **16**, 715–721 (2019).

[31] Theodoris, C. V. *et al.* Transfer learning enables predictions in network biology. *Nature* 1–9 (2023).

[32] Hao, M. *et al.* Large scale foundation model on single-cell transcriptomics. *bioRxiv* 2023–05 (2023).

[33] Cui, H. *et al.* scgpt: Towards building a foundation model for single-cell multi-omics using generative ai. *bioRxiv* 2023–04 (2023).

[34] Zhao, S., Zhang, J. & Nie, Z. Large-scale cell representation learning via divide-and-conquer contrastive learning. *arXiv preprint arXiv:2306.04371* (2023).

[35] Heimberg, G. *et al.* Scalable querying of human cell atlases via a foundational model reveals commonalities across fibrosis-associated macrophages. *bioRxiv* 2023–07 (2023).

[36] Levine, D. *et al.* Cell2sentence: Teaching large language models the language of biology. *bioRxiv* 2023–09 (2023).

[37] Chen, Y. T. & Zou, J. Genept: A simple but hard-to-beat foundation model for genes and cells built from chatgpt. *bioRxiv* 2023–10 (2023).

[38] Brown, T. *et al.* Language models are few-shot learners. *Advances in neural information processing systems* **33**, 1877–1901 (2020).

[39] OpenAI. Gpt-4 technical report (2023).

[40] Touvron, H. *et al.* Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* (2023).

[41] Rombach, R., Blattmann, A., Lorenz, D., Esser, P. & Ommer, B. High-resolution image synthesis with latent diffusion models. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* 10684–10695 (2022).

[42] Ramesh, A. *et al.* Zero-shot text-to-image generation. *International Conference on Machine Learning* 8821–8831 (2021).

[43] Wang, W., Zheng, V. W., Yu, H. & Miao, C. A survey of zero-shot learning: Settings, methods, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* **10**, 1–37 (2019).

[44] Lachapelle, S. *et al.* Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ica. *Conference on Causal Learning and Reasoning* 428–484 (2022).

[45] Yamada, Y., Lindenbaum, O., Negahban, S. & Kluger, Y. Feature selection using stochastic gates. *International Conference on Machine Learning* 10648–10659 (2020).

[46] Lindenbaum, O., Salhov, M., Averbuch, A. & Kluger, Y. L0-sparse canonical correlation analysis. *International Conference on Learning Representations* (2021).

[47] Dong, M. & Kluger, Y. Geass: Neural causal feature selection for high-dimensional biological data. *The Eleventh International Conference on Learning Representations* (2022).

[48] Burkhardt, D. *et al.* Open problems – single-cell perturbations. *Kaggle* https://kaggle.com/competitions/open–problems–single–cell–perturbations (2023).

[49] Kedzierska, K. Z., Crawford, L., Amini, A. P. & Lu, A. X. Assessing the limits of zero-shot foundation models in single-cell biology. *bioRxiv* (2023).

[50] Oelen, R. *et al.* Single-cell rna-sequencing of peripheral blood mononuclear cells reveals widespread, context-specific gene expression regulation upon pathogenic exposure. *Nature Communications* **13**, 3267 (2022).

[51] Imbens, G. W. & Rubin, D. B. *Causal inference in statistics, social, and biomedical sciences* (Cambridge University Press, 2015).

[52] Li, F., Morgan, K. L. & Zaslavsky, A. M. Balancing covariates via propensity score weighting. *Journal of the American Statistical Association* **113**, 390–400 (2018).

[53] Finley, M., Happel, C., Kaminsky, D. & Rogers, T. Opioid and nociceptin receptors regulate cytokine and cytokine receptor expression. *Cellular immunology* **252**, 146–154 (2008).

[54] Barrett, E., Brothers, S., Wahlestedt, C. & Beurel, E. I-bet151 selectively regulates il-6 production. *Biochimica et Biophysica Acta (BBA)-Molecular Basis of Disease* **1842**, 1549–1555 (2014).

[55] Klein, K. *et al.* The bromodomain protein inhibitor i-bet151 suppresses expression of inflammatory genes and matrix degrading enzymes in rheumatoid arthritis synovial fibroblasts. *Annals of the rheumatic diseases* **75**, 422–429 (2016).

[56] Seal, J. *et al.* Identification of a novel series of bet family bromodomain inhibitors: binding mode and profile of i-bet151 (gsk1210151a). *Bioorganic & medicinal chemistry letters* **22**, 2968–2972 (2012).

[57] Woodhead, A. J. *et al.* Discovery of (2, 4-dihydroxy-5-isopropylphenyl)-[5-(4-methylpiperazin-1-ylmethyl)-1, 3-dihydroisoindol-2-yl] methanone (at13387), a novel inhibitor of the molecular chaperone hsp90 by fragment based drug design. *Journal of medicinal chemistry* **53**, 5956–5969 (2010).

[58] Chan, K. C. *et al.* A novel hsp90 inhibitor at13387 induces senescence in ebv-positive nasopharyngeal carcinoma cells and suppresses tumor formation. *Molecular cancer* **12**, 1–15 (2013).

[59] Wang, Y., Trepel, J. B., Neckers, L. M. & Giaccone, G. Sta-9090, a small-molecule hsp90 inhibitor for the potential treatment of cancer. *Current Opinion in Investigational Drugs (London, England: 2000)* **11**, 1466–1476 (2010).

[60] Santoro, A. *et al.* Tivantinib for second-line treatment of advanced hepatocellular carcinoma: a randomised, placebo-controlled phase 2 study. *The lancet oncology* **14**, 55–63 (2013).

[61] Katayama, R. *et al.* Cytotoxic activity of tivantinib (arq 197) is not due solely to c-met inhibition. *Cancer research* **73**, 3087–3096 (2013).

[62] Fleck, J. S., Camp, J. G. & Treutlein, B. What is a cell type? *Science* **381**, 733–734 (2023).

[63] Bravo González-Blas, C. *et al.* Scenic+: single-cell multiomic inference of enhancers and gene regulatory networks. *Nature Methods* **20**, 1355–1367 (2023).

[64] Kamimoto, K. *et al.* Dissecting cell identity via network inference and in silico gene perturbation. *Nature* **614**, 742–751 (2023).

[65] Gayoso, A. *et al.* A python library for probabilistic analysis of single-cell omics data. *Nature biotechnology* **40**, 163–166 (2022).

[66] Hyvärinen, A. & Pajunen, P. Nonlinear independent component analysis: Existence and uniqueness results. *Neural networks* **12**, 429–439 (1999).

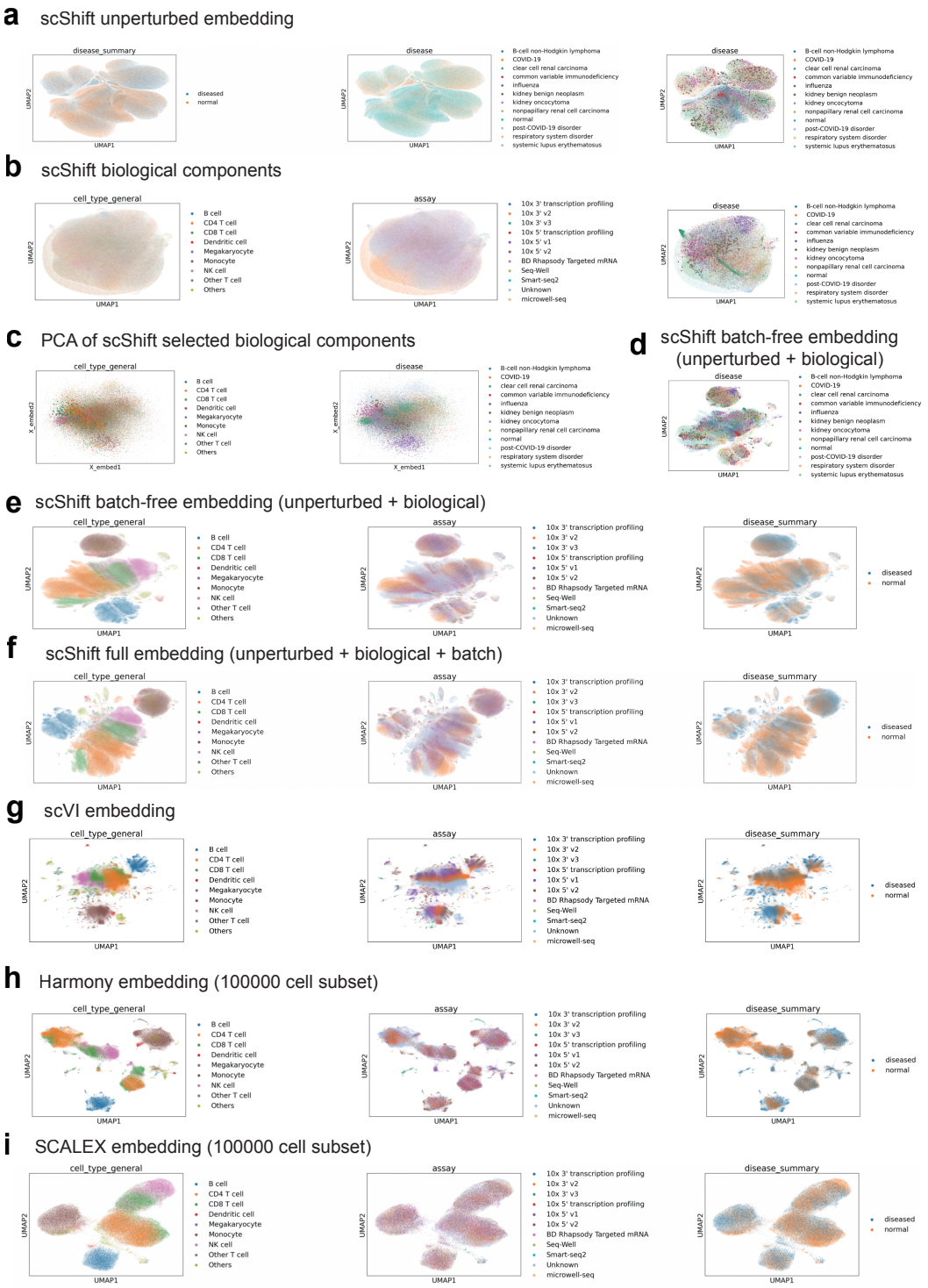
[67] Khemakhem, I., Kingma, D., Monti, R. & Hyvarinen, A. Variational autoencoders and nonlinear ica: A unifying framework 2207–2217 (2020).

[68] Moran, G., Sridhar, D., Wang, Y. & Blei, D. Identifiable deep generative models via sparse decoding. *Transactions on machine learning research* (2022).

[69] Wolf, F. A., Angerer, P. & Theis, F. J. Scanpy: large-scale single-cell gene expression data analysis. *Genome biology* **19**, 1–5 (2018).

[70] He, Z. *et al.* An integrated transcriptomic cell atlas of human neural organoids. *bioRxiv* 2023–10 (2023).

# Extended data figures




**Extended Data Fig. 1** Additional visualizations for scShift embeddings compared with other methods. **a.** UMAP visualization of the scShift unperturbed embedding (unperturbed and biological), colored by disease state, disease type, and disease type with rescaled dot sizes. **b.** UMAP visualization of the scShift biological embedding, colored by cell types, sequencing assays, and cell disease state (dot size rescaled). **c.** PCA visualization of selected scShift biological components, colored by cell types and cell disease state (dot size rescaled). **d.** UMAP visualization of the scShift non-batch embedding (unperturbed and biological), colored by cell disease state (dot size rescaled). **e-i.** UMAP visualization of the scShift non-batch embedding (unperturbed and biological), the scShift full embedding, the scVI integrated embedding, the Harmony embedding, and the SCALEX embedding. Colored by cell types, sequencing assays, and cell disease state.

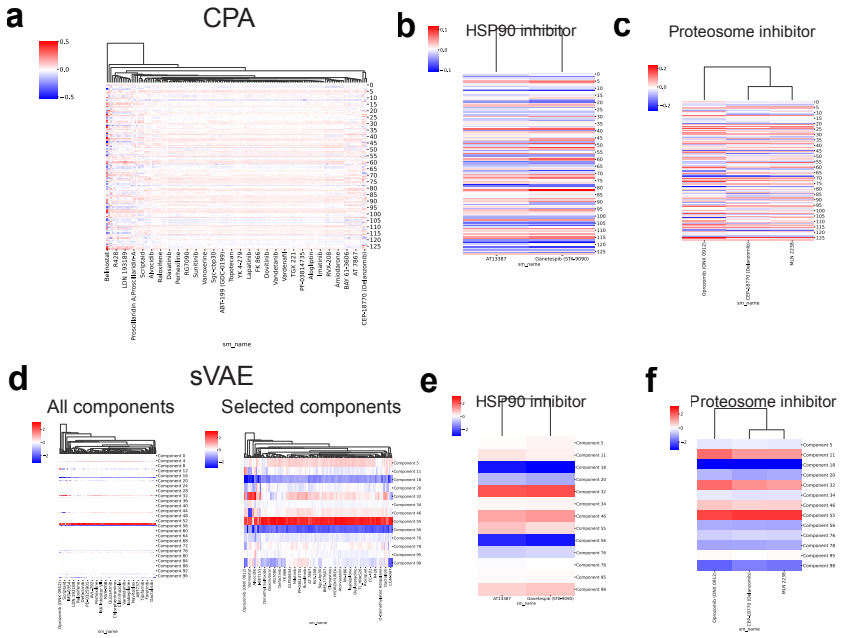
**a**

| Method | Cell type conservation | | | | | Assay correction | | | | | Aggregate score | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Isolated labels | KMeans NMI | KMeans ARI | Silhouette label | cLISI | Silhouette batch | iLISI | KBET | Graph connectivity | PCR comparison | Batch correction | Bio conservation | Total |
| **SCALEX** | 0.00 | 0.51 | 0.39 | 0.66 | 1.00 | 0.91 | 1.00 | 1.00 | 1.00 | 1.00 | 0.98 | 0.51 | 0.70 |
| **Harmony** | 0.98 | 1.00 | 1.00 | 1.00 | 0.80 | 0.51 | 0.00 | 1.00 | 0.00 | 0.00 | 0.30 | 0.95 | 0.69 |
| **scShift (unperturbed)** | 0.43 | 0.58 | 0.49 | 0.83 | 0.73 | 0.00 | 0.53 | 0.07 | 0.44 | 0.78 | 0.36 | 0.61 | 0.51 |
| **scVI** | 0.32 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | 0.17 | 0.55 | 0.30 | 0.62 | 0.53 | 0.06 | 0.25 |

**b**

| Method | Concordance (cell type / disease conservation) | | | | | Disentanglement (disease / cell type removal) | | | | | Aggregate score | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Isolated labels | KMeans NMI | KMeans ARI | Silhouette label | cLISI | Silhouette batch | iLISI | KBET | Graph connectivity | PCR comparison | Disentanglement | Concordance | Total |
| **scShift** | 0.94 | 0.50 | 0.39 | 0.82 | 0.51 | 0.56 | 1.00 | 1.00 | 0.93 | 1.00 | 0.90 | 0.63 | 0.74 |
| **Harmony** | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.00 | 0.00 | 0.16 | 0.00 | 0.00 | 0.03 | 1.00 | 0.61 |
| **SCALEX** | 0.51 | 0.84 | 0.85 | 0.53 | 0.00 | 0.20 | 0.54 | 0.00 | 1.00 | 0.33 | 0.41 | 0.55 | 0.49 |
| **scVI** | 0.00 | 0.00 | 0.00 | 0.00 | 0.29 | 1.00 | 0.39 | 0.08 | 0.40 | 0.95 | 0.56 | 0.06 | 0.26 |

**Extended Data Fig. 2** Quantitative evaluations of scShift latent embeddings. **a.** scIB metrics evaluating the assay information removal and cell type preservation across methods. **b.** averaged scIB metrics evaluating disentanglement and concordance of scShift embeddings compared with other approaches. In all evaluations, each column is normalized to have min 0 and max 1.



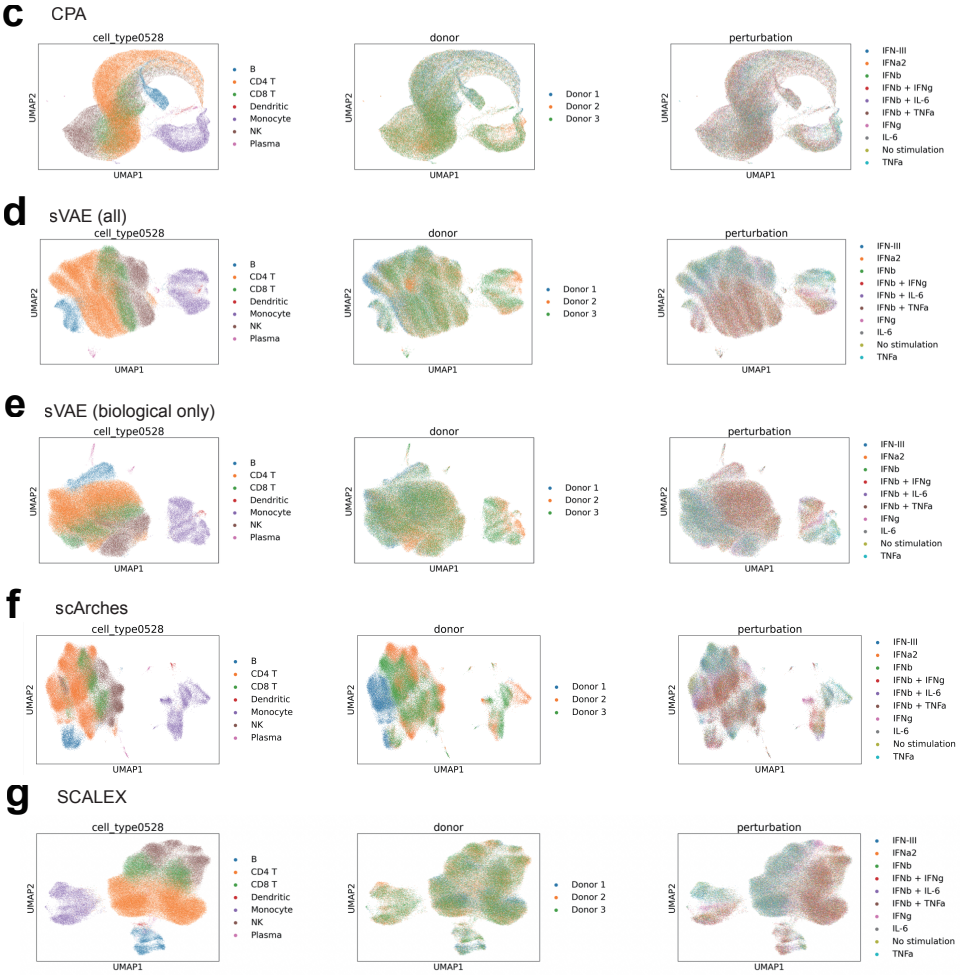

**Extended Data Fig. 3** Perturbation space heatmaps for CPA and sVAE. **a.** Overall heatmap for CPA perturbation space. **b.** Heatmap for HSP90 inhibitors in CPA perturbation space. **c.** Heatmap for proteosome inhibitors in CPA perturbation space. **d.** Overall heatmap for sVAE perturbation space, showing all components (Left) and selected components (Right). **e.** Heatmap for HSP90 inhibitors in sVAE perturbation space. **f.** Heatmap for proteosome inhibitors in sVAE perturbation space.

**a**

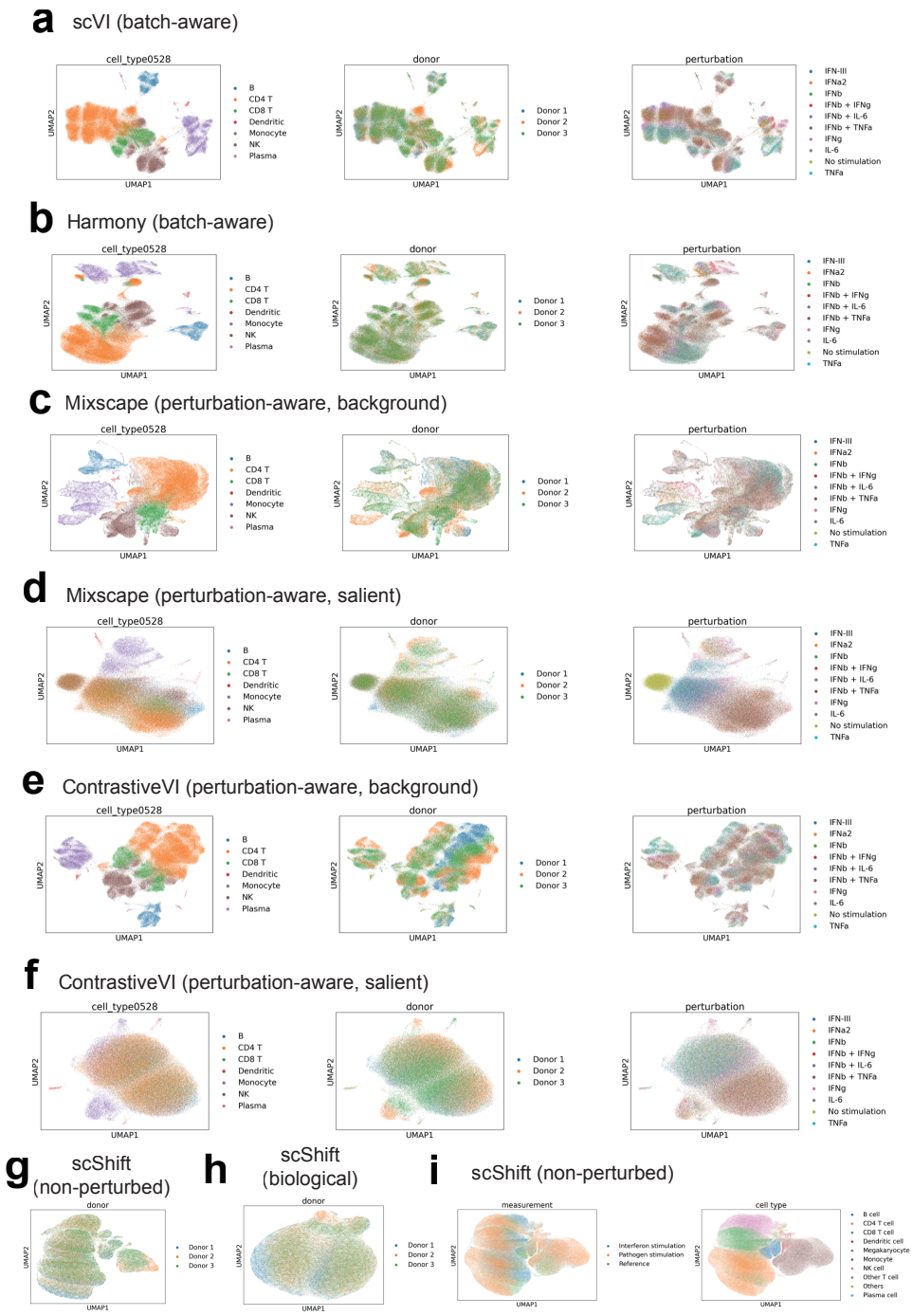
| Method | Bio conservation | | | | | Batch correction | | | | Aggregate score | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Isolated labels | KMeans NMI | KMeans ARI | Silhouette label | cLISI | Silhouette batch | iLISI | KBET | Graph connectivity | Batch correction | Bio conservation | Total |
| Harmony (batch-aware) | 0.41 | 0.69 | 0.58 | 0.83 | 1.00 | 0.86 | 0.90 | 1.00 | 0.78 | 0.89 | 0.70 | 0.78 |
| scShift (zero-shot) | 0.53 | 0.70 | 0.48 | 1.00 | 1.00 | 0.48 | 1.00 | 0.62 | 0.87 | 0.74 | 0.74 | 0.74 |
| Mixscape (perturbation-aware) | 0.46 | 1.00 | 1.00 | 0.99 | 1.00 | 0.54 | 0.44 | 0.36 | 0.62 | 0.49 | 0.89 | 0.73 |
| ContrastiveVI (perturbation-aware) | 0.58 | 0.95 | 0.69 | 0.91 | 1.00 | 0.64 | 0.00 | 0.15 | 0.90 | 0.42 | 0.83 | 0.67 |
| scVI (batch-aware) | 0.56 | 0.37 | 0.30 | 0.72 | 1.00 | 0.84 | 0.77 | 0.46 | 1.00 | 0.77 | 0.59 | 0.68 |
| SCALEX (zero-shot) | 0.00 | 0.57 | 0.47 | 0.70 | 0.97 | 0.95 | 0.78 | 0.44 | 0.89 | 0.77 | 0.54 | 0.63 |
| sVAE (zero-shot) | 0.57 | 0.00 | 0.00 | 0.58 | 0.78 | 0.83 | 0.65 | 0.44 | 0.95 | 0.72 | 0.39 | 0.52 |
| scArches (fine-tuned) | 0.36 | 0.54 | 0.54 | 0.59 | 0.60 | 1.00 | 0.22 | 0.00 | 0.64 | 0.47 | 0.52 | 0.50 |
| CPA (zero-shot) | 1.00 | 0.02 | 0.08 | 0.00 | 0.94 | 0.00 | 0.75 | 0.66 | 0.88 | 0.57 | 0.41 | 0.47 |
| scShift (no-pretraining) | 0.48 | 0.32 | 0.15 | 0.49 | 0.00 | 0.94 | 0.90 | 0.77 | 0.00 | 0.65 | 0.29 | 0.43 |

**b**

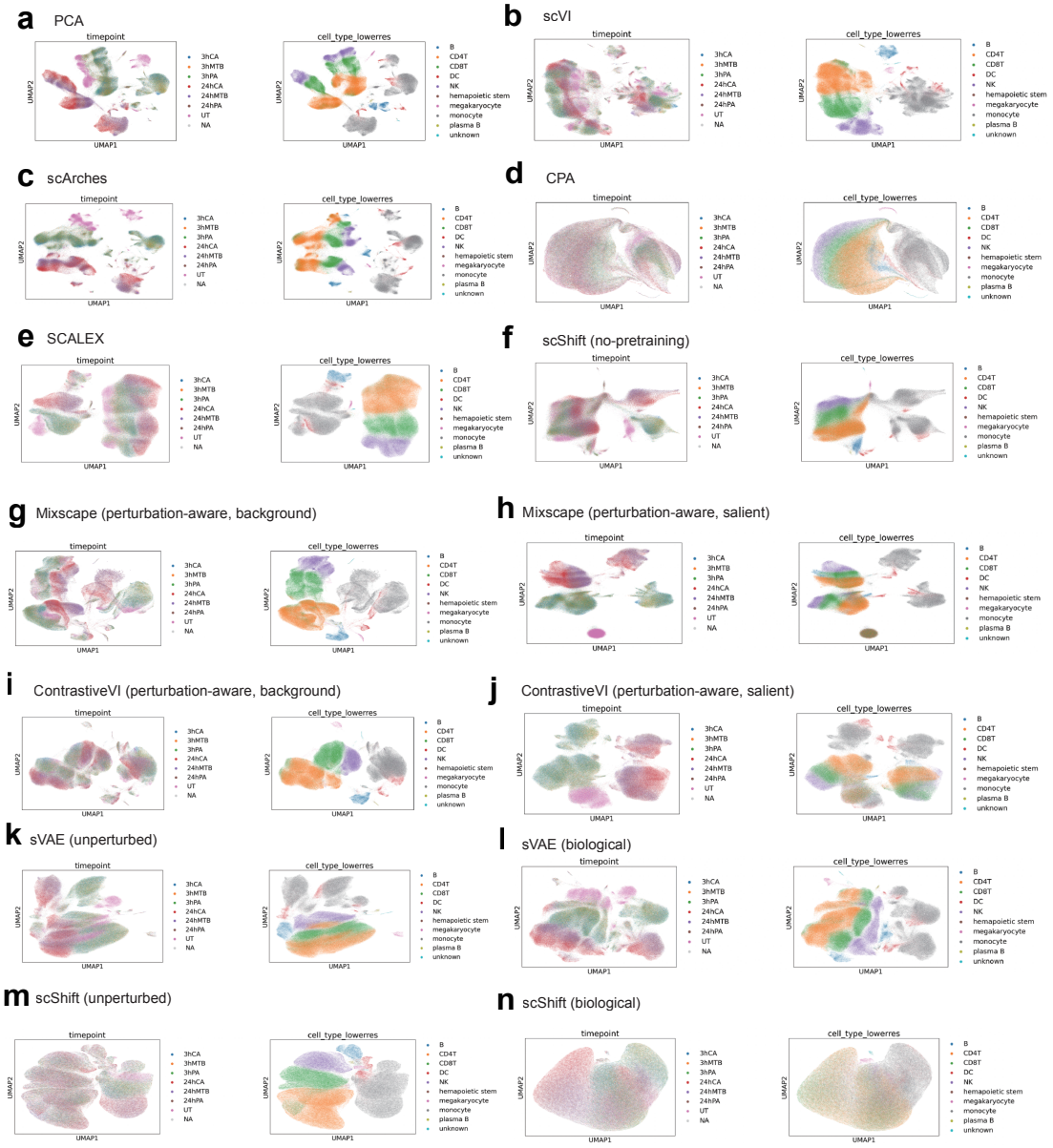
| Method | Bio conservation | | | | | Perturbation removal | | | | Aggregate score | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Isolated labels | KMeans NMI | KMeans ARI | Silhouette label | cLISI | Silhouette batch | iLISI | KBET | Graph connectivity | Batch correction | Bio conservation | Total |
| scShift (zero-shot) | 0.86 | 0.70 | 0.48 | 1.00 | 1.00 | 0.48 | 1.00 | 0.81 | 0.87 | 0.79 | 0.81 | 0.80 |
| Mixscape (perturbation-aware) | 1.00 | 1.00 | 1.00 | 0.99 | 1.00 | 0.40 | 0.31 | 0.57 | 0.62 | 0.47 | 1.00 | 0.79 |
| ContrastiveVI (perturbation-aware) | 0.81 | 0.95 | 0.69 | 0.91 | 1.00 | 0.68 | 0.17 | 0.17 | 0.90 | 0.48 | 0.87 | 0.72 |
| Harmony (batch-aware) | 0.77 | 0.69 | 0.58 | 0.83 | 1.00 | 0.77 | 0.06 | 0.07 | 0.78 | 0.42 | 0.77 | 0.63 |
| scVI (batch-aware) | 0.54 | 0.37 | 0.30 | 0.72 | 1.00 | 0.82 | 0.00 | 0.03 | 1.00 | 0.46 | 0.59 | 0.54 |
| SCALEX (zero-shot) | 0.09 | 0.57 | 0.47 | 0.70 | 0.97 | 0.76 | 0.15 | 0.00 | 0.89 | 0.45 | 0.56 | 0.52 |
| CPA (zero-shot) | 0.89 | 0.02 | 0.08 | 0.00 | 0.94 | 0.00 | 0.62 | 1.00 | 0.88 | 0.63 | 0.39 | 0.48 |
| scArches (fine-tuned) | 0.00 | 0.54 | 0.54 | 0.59 | 0.60 | 1.00 | 0.28 | 0.01 | 0.64 | 0.48 | 0.45 | 0.47 |
| sVAE (zero-shot) | 0.49 | 0.00 | 0.00 | 0.58 | 0.78 | 0.78 | 0.12 | 0.10 | 0.95 | 0.49 | 0.37 | 0.42 |
| scShift (no-pretraining) | 0.28 | 0.32 | 0.15 | 0.49 | 0.00 | 0.90 | 0.90 | 0.85 | 0.00 | 0.66 | 0.25 | 0.41 |



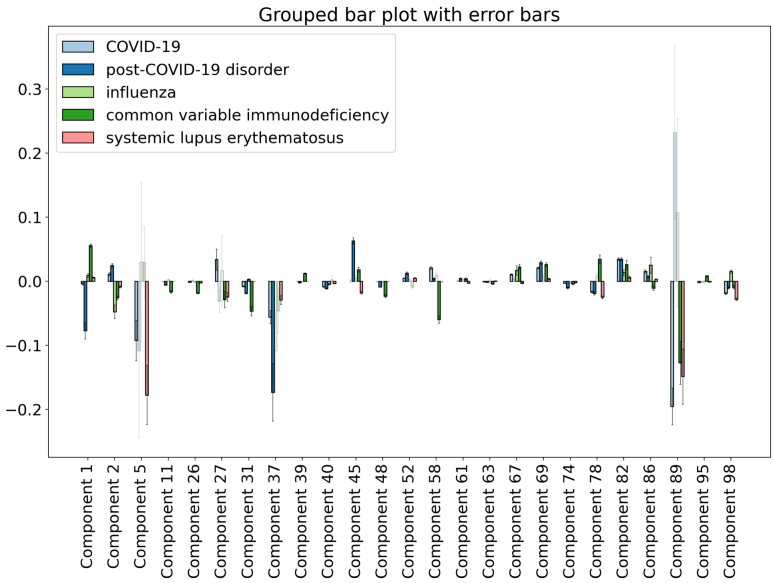

**Extended Data Fig. 4** Additional evaluations for learning cell type and biological representations in interferon stimulation data. **a.** Evaluation of different methods' unperturbed / integrated / background embeddings in batch removal and cell type preservation. **b.** Evaluation of different methods' unperturbed / integrated / background embeddings in perturbation removal and cell type preservation. **c-g.** UMAP visualization of CPA integrated embeddings, sVAE full embeddings, sVAE biological embeddings, scArches integrated embeddings, and SCALEX integrated embeddings, colored by cell type, donor batch, and perturbation.




**Extended Data Fig. 5** Continued UMAP visualizations colored by perturbation, donor, and cell types of each embedding for the interferon stimulation dataset. **a.** scVI; **b.** Harmony; **c.** Mixscape cell type representation; **d.** Mixscape perturbation representation; **e.** ContrastiveVI cell type representation; **f.** ContrastiveVI perturbation representation; **g.** scShift unperturbed embedding colored by donor, following fig. 3b. **h.** scShift biological embedding colored by donor, following fig. 3c. **i.** scShift unperturbed embedding of all three datasets, colored by dataset and cell type.




**Extended Data Fig. 6** UMAP visualizations colored by perturbation and cell types of each embedding for the pathogen stimulation dataset. **a.** PCA; **b.** scVI; **c.** scArches; **d.** CPA; **e.** SCALEX; **f.** scShift (no-pretraining); **g.** Mixscape cell type representation; **h.** Mixscape perturbation representation; **i.** ContrastiveVI cell type representation; **j.** ContrastiveVI perturbation representation; **k.** sVAE unperturbed representation; **l.** sVAE biological representation; **m.** scShift unperturbed representation. **n.** scShift biological representation.




**Extended Data Fig. 7** Bar plot of scShift biological component values for selected diseases. Insignificant components (95% confidence interval covering 0) are depicted in a semi-transparent manner.