Unsupervised Deep Disentangled Representation of Single-Cell Omics

Amir Ali Moinfar[1,2], Fabian J. Theis[1,2,3‡]

1 School of Computing, Information and Technology, Technical University of Munich, Garching, Germany.
2 Institute of Computational Biology, Helmholtz Center Munich, Neuherberg, Germany.
3 TUM School of Life Sciences Weihenstephan, Technical University of Munich, Freising, Germany.

†Correspondence to:
fabian.theis@helmholtz-muenchen.de

Abstract

Single-cell genomics allows for the unbiased exploration of cellular heterogeneity. Representation learning methods summarize high-dimensional single-cell data into a manageable latent space in a typically nonlinear fashion, allowing cross-sample integration or generative modeling. However, these methods often produce entangled representations, limiting interpretability and downstream analyses. Existing disentanglement methods instead either require supervised information or impose sparsity and linearity, which may not capture the complexity of biological data. We, therefore, introduce Disentangled Representation Variational Inference (DRVI), an unsupervised deep generative model that learns nonlinear, disentangled representations of single-cell omics. This is achieved by combining recently introduced additive decoders with nonlinear pooling, for which we theoretically prove disentanglement under reasonable assumptions. We validate DRVI's disentanglement capabilities across diverse relevant biological problems, from development to perturbational studies and cell atlases, decomposing, for example, the Human Lung Cell Atlas into meaningful, interpretable latent dimensions. Moreover, we demonstrate that if applied to batch integration, DRVI's integration quality does not suffer from the disentanglement constraints and instead is on par with entangled integration methods. With its disentangled latent space, DRVI is inherently interpretable and facilitates the identification of rare cell types, provides novel insights into cellular heterogeneity beyond traditional cell types, and highlights developmental stages.

Introduction

Single-cell RNA-seq technologies are widely used for profiling transcriptomes or other omics levels at single-cell resolution in a high-throughput fashion [1, 2]. These powerful readouts empower researchers to explore developmental processes, assess the impact of drugs and perturbations across diverse cell lines, construct reference atlases, and compare organoids with primary tissues [3–10]. Recent advancements have expanded single-cell measurements beyond gene expression to additional modalities, including chromatin accessibility, spatial information, and surface proteins [11–17]. Consequently, many tools and methods have emerged to facilitate the analysis of these datasets [18]. Among these, generative models have effectively summarized data within a latent space, particularly when dealing with multiple batches [19, 20], outperforming more traditional methods in large-scale benchmarks and competitions for integration and cross-modality inference [21]. Subsequently, several extensions have been developed to address various modalities and use cases [22–25].

Despite their widespread use, generative models commonly produce entangled latent representations, where multiple unrelated biological processes are intertwined within a single dimension [26]. This limits the interpretability of the latent space, restricting its direct application for downstream tasks

[27, 28]. As a result, the latent space is often only used as an intermediate, and its utility is limited to neighborhood graphs constructed on top of the cellular representations. Specifically, common techniques like Uniform manifold approximation and projection (UMAP) [29, 30] for visualization and Leiden clustering [31] for cell type identification exclusively rely on these neighborhood graphs. While clustering methods are essential for extracting biological variations from the high-dimensional single-cell omics data [32], they have limitations. They cannot capture continuous processes or identify shared programs among multiple cell types as a single concept and often identify the most dominant signals, such as cell types, which makes it challenging to uncover secondary signals.

Disentangled representation learning addresses these limitations by separating the underlying factors of variation into distinct, semantically meaningful latent variables [33]. However, unsupervised learning of truly disentangled representations presents a fundamental challenge, as demonstrated by theoretical impossibility results, which highlight the necessity of inductive biases on both models and data [34]. This has motivated diverse approaches, from methods tailored for specific data modalities like recommendation systems [35], to more general mathematical frameworks [36, 37].

Several disentangled latent space models have been proposed for single-cell omics data. However, they either rely on linear assumptions [38, 39], auxiliary data, such as variations of interest [40–42], or both [26, 43–45] to enforce disentanglement. In practice, for new discoveries, supervised data is absent or, if present, frequently biased. And, linear models fail to capture the complexity of biological data, particularly in scenarios involving multiple batches. These limitations motivate the development of unsupervised nonlinear disentangled representation learning. While there are a few unsupervised and unconstrained disentanglement models [46, 47], we will experimentally confirm the theoretical impossibility results [34], and show none of these unconstrained models demonstrate satisfactory disentanglement performance, which may explain the fact that they see little practical use. In Supplementary Note 1, we discuss disentangled representation learning in more detail, provide an overview of disentangled latent space models tailored specifically for single-cell omics data (Supplementary Table 1), and highlight the contributions of this work.

We leverage the recently proposed framework of additive decoders [37], which enables latent variable identification in the context of image generation. We present Disentangled Representation Variational Inference (DRVI), an unsupervised generative model capable of learning non-linear interpretable disentangled latent representations from single-cell count data. DRVI addresses the theoretical and practical challenges of unsupervised disentanglement [34] by leveraging additive decoders followed by a novel non-linear pooling function, which serves as a crucial inductive bias tailored for the unique characteristics of single-cell omics. Based on the interaction asymmetry [48] criteria, we theoretically prove that this architecture achieves disentanglement under specific assumptions: additivity of independent underlying processes in count space and the existence of markers for each process. Empirical evaluations across a diverse range of biological applications demonstrate DRVI's capabilities in learning disentangled latent representations, outperforming existing methods in terms of disentanglement without reducing integration quality. DRVI facilitates the identification of biological states by representing cell types across individual latent factors in cell atlases, enables exploration of cellular heterogeneity by identifying biologically meaningful variations beyond cell types, captures developmental stages in developmental datasets, and identifies coherent groups of effective perturbations in perturbational studies. Additionally, DRVI allows users to selectively highlight or exclude latent factors associated with specific known variations, offering flexible post-hoc analysis capabilities.

Results

DRVI learns interpretable disentangled cell representations

DRVI (Disentangled Representation Variational Inference) aims to integrate datasets and learn unsupervised disentangled latent representations of biological processes from single-cell omics data. A representation is disentangled if there is a one-to-one correspondence between its dimensions and components of a ground truth representation (see 'Definitions and Notations' in Methods)[49, 50]. In this context, disentanglement means identifying distinct biological processes and properties, such as cell types, signaling pathways, and developmental stages, and representing them in individual latent dimensions (Fig. 1a). DRVI achieves this by exploiting the covariations present in the data. The model is based on the assumption that independent underlying processes combine additively in gene count space (justified in Supplementary Note 2) or, more generally, some kernel space depending on the choice of pooling function. While DRVI is built within the commonly used conditional variational autoencoders (CVAE) framework [19, 51], its decoder is distinct from existing models. Specifically, it employs additive decoders followed by a pooling function to enforce the disentanglement and represent distinct sources of variations in different latent dimensions.

The encoder g takes the (commonly) normalized gene expression of a cell x and optionally the covariate variable c — representing batch or other technical information to regress out, where both one hot encoding [19] and embedding [52] is supported — as input and maps them into the latent space z. Then, the decoder f constructs the parameters of the generative distribution such as Gaussian or Negative Binomial (Fig. 1b). Unlike the encoder that employs a multi-layer perceptron (MLP) capable of approximating a wide range of functions, the additive decoder architecture [37] is specialized to enforce disentanglement on the latent space dimensions. Specifically, the decoder takes the latent space vector z, of dimension K, and splits it into K/D splits of dimension D, where $D = 1$ by default. Each split, optionally together with the covariate variable c, is decoded separately using a nonlinear subnetwork f_k (where $k \in \{1, \ldots, K/D\}$) back into the expression space $\hat{x}^k$ to later form the parameters of the generative distribution. Finally, the decoded vectors $\hat{x}^k$ are aggregated using a potentially nonlinear pooling function, combining K/D vectors of the same size into one, to form the final generative distribution. The overall model is optimized based on the variational inference framework, with the encoder representing the approximate posterior and the decoder representing the generative model (see 'The Generative model' in Methods).

By default, DRVI uses log-sum-exp (LSE) pooling for aggregation, so altogether, decoding yields $\hat{x} = \log \sum_{k=1}^{K/D} \exp f_k(z_k, c)$. To emphasize the importance of the LSE aggregation, we explore an alternative scenario when we simply average the parameters, referred to as DRVI with average pooling (DRVI-AP). As we prove theoretically (based on the criteria from [37, 48], see 'Additive Decoders, Pooling, and Disentanglement' in Methods) and observe practically, the introduced architecture with the LSE pooling effectively enforces disentanglement and enables the recovery of biological processes beyond cell types. To ensure provable disentanglement, specific conditions must be met, including the presence of at least two marker genes for each process with distinct log-fold-changes, smoothness of the decoder and its inverse, maintaining simplicity in decoder subnetworks (i.e., not representing a logarithm), zero reconstruction error, connected cellular manifold, and an equal number of underlying processes and latent dimensions (see Methods for details). While these ideal conditions are not fully attainable in real-world scenarios, DRVI aims to approximate them as closely as possible. That is the reason DRVI achieves significant improvements in disentanglement, as supported by our empirical results. The disentanglement theorem requires an equal number of underlying processes and latent dimensions. Accordingly, we recommend users provide more latent dimensions than they anticipate, as excess dimensions will vanish, a phenomenon we will discuss in more detail later.

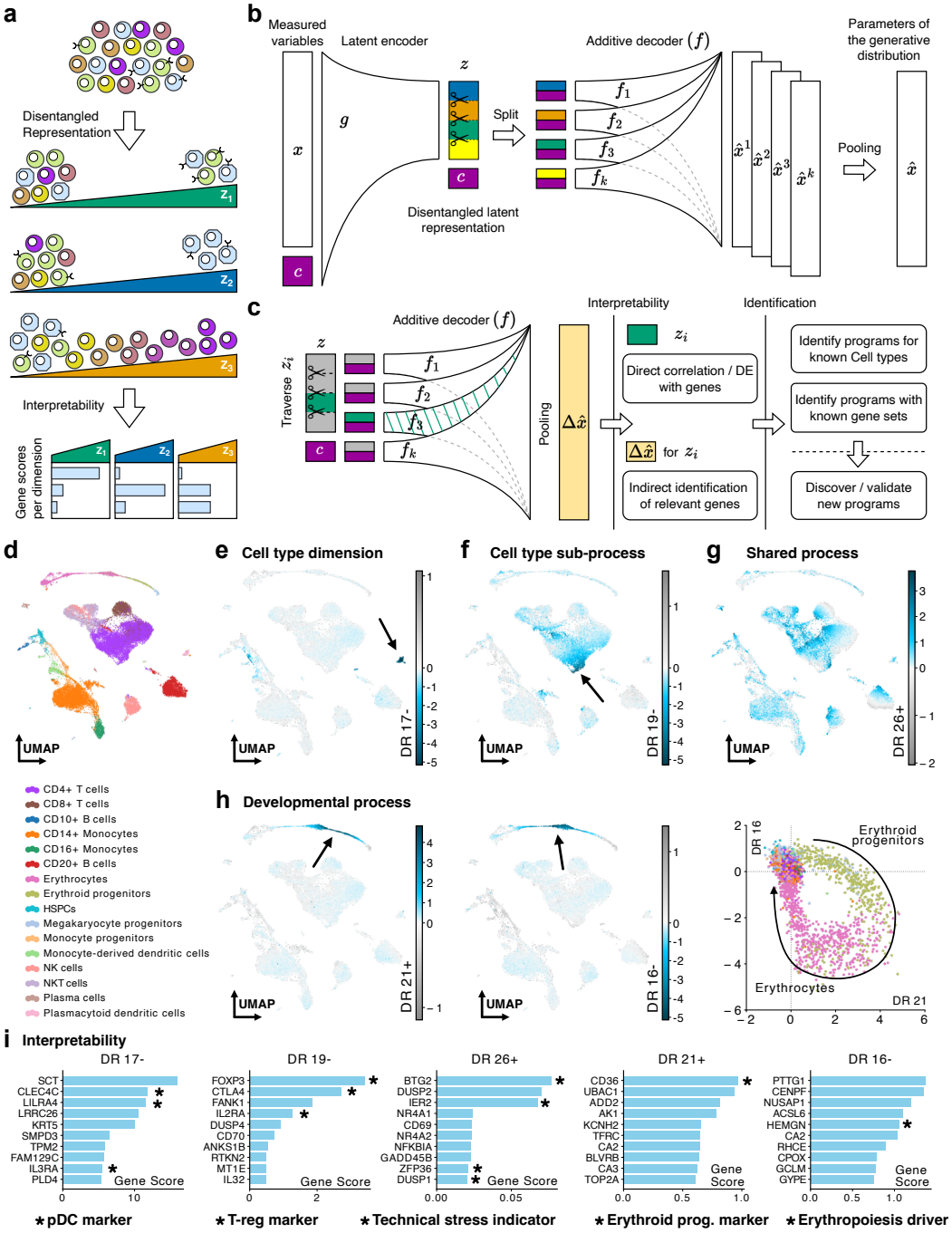


Figure 1: **DRVI enables learning interpretable disentangled latent representations. a,** Disentanglement concept. A disentangled latent model should encode concepts such as cell properties, cell types, and developmental stages in distinct dimensions. Each dimension can then be interpreted as the effect of the dimension on the gene expression. **b,** Schematic of the DRVI model. The model consists of a nonlinear encoder and an additive decoder followed by a pooling function that enforces the disentanglement into the encoder and latent factors. **c,** Interpreting disentangled factors. Relevant genes can be discovered by traversing a latent factor or using DE analysis tools. While some factors can be identified using known gene sets or known cell type annotations, others can be validated as potentially novel programs. **d,** Uniform manifold approximation and projection (UMAP) of the immune dataset integrated by DRVI colored by cell types. **e-h,** Examples of factors identified by DRVI indicating a cell type (e), a cell-type-specific process (f), a shared process (g), and a developmental process (h). For the developmental process, a scatterplot visualizing two consecutive stages is provided. The arrow is for visualization and is not derived from the model. **i,** Top identified genes for each example dimension. DE, differential expression; pDC, plasmacytoid dendritic cells; T-reg, regulatory T-cells; prog., progenitors.

A significant advantage of DRVI is the interpretability of its latent space. DRVI individually decodes each latent factor to the expression space. Consequently, we can interpret the decoded effect of each factor as a nonlinear gene program (Fig. 1c). To identify these gene programs, we traverse each latent dimension and inspect the affected features (e.g., genes in transcriptomics). Since the effect changes nonlinearly as we traverse the dimension, we consider the maximum effect for each dimension (see 'Interpretability' in Methods). For each dimension, we refer to the genes identified by the interpretability pipeline as 'relevant genes' for that dimension. Alternatively, one can perform differential expression (DE) analysis to identify the genes involved in a nonlinear gene program beyond the genes used for model training. Once we obtain the relevant genes, we can identify some known processes through gene-set enrichment analysis (GSEA) or identify cell-type dimensions using available annotations. Unidentified processes are promising candidates for further literature search, additional analysis, and even experimental design. Furthermore, users can selectively regress out unwanted identified biological processes by excluding their corresponding latent dimensions.

DRVI identifies biological signals beyond cell types

To demonstrate DRVI's disentangled and interpretable latent space, we applied DRVI with 32 latent dimensions to an immune dataset composed of 9 batches from four human peripheral blood and bone marrow studies, with 16 annotated cell types [53]. We visualize the integrated cell-embedding space via UMAP projections colored by cell types (Fig. 1d) and then individual dimensions. The UMAPs of DRVI versus other benchmarked methods, colored by batch and cell types, are available in Supplementary Fig. 2a,b. DRVI ranks the latent dimensions based on their impact on reconstruction, labeled as DR 1, DR 2, etc (see 'Ordering the learned factors' in Methods). Like many other dimensionality reduction techniques, DRVI is unable to distinguish between positive and negative directions. Thus, interpretation requires considering both direction and dimension (e.g. DR 1+). To illustrate the disentanglement capabilities, we highlighted example dimensions representing a cell-type identifier, a cell-type sub-process, a shared process, and two consecutive stages in a developmental process. For the example dimensions, we provided UMAPs colored by their values in different cells, referred to as activity (Fig. 1e-h). Additionally, we visualized the geometry of the developmental process in a scatterplot (Fig. 1h). Finally, we identified the top genes related to each dimension (Supplementary Fig. 3) and illustrated them for the example dimensions (Fig. 1i).

DR 17 identifies Plasmacytoid dendritic cells (pDCs) in the negative direction (Fig. 1e). CLEC4C and LILRA4 are marker genes for pDCs, and IL3RA overexpressed in this cell type, is an essential gene for the functioning and development of pDCs (Fig. 1i) [54, 55]. We provided a heatmap of the latent embeddings for DRVI and each benchmarked method to demonstrate DRVI's effectiveness in identifying cell types (Supplementary Fig. 4).

DR 19- highlights a subset of CD4+ T-cells (Fig. 1f). The presence of FOXP3, CTLA4 (CD152), and IL2RA (CD25) in the nonlinear gene program (Fig. 1i) suggests that this dimension is targeting regulatory T-cells (T-regs) [56, 57]. GSEA analysis also supports the association of this dimension with T-cell activation regulation.

A practical application of DRVI in identifying cell types is annotation refinement. We used DR 17, DR 20, and DR 29 to annotate pDC, DC1, and DC2 subtypes of dendritic cells, respectively (Supplementary Fig. 6a-d). We validated this refinement using curated markers of these subtypes (Supplementary Fig. 6j). DRVI also guides users to potential misannotations. For instance, DR 30 revealed an underrepresented population of 12 Fibroblast cells (Supplementary Fig. 6e-h), which were previously misannotated as HSPCs, Monocyte progenitors, or Erythroid progenitors. The validity of this revised annotation was confirmed using COL1A1, a known Fibroblast marker not included in the model's HVG input (Supplementary Fig. 6i).

DR 26+ indicates a shared process among multiple cell types (Fig. 1f). BTG2, DUSP2, and IER2 are the top genes in the nonlinear gene program, overexpressed at higher levels of this dimension (Fig.

1i). BTG2, IER2, DUSP1, FOS, JUN, EGR1, and ZFP36, all within the top 20 relevant genes, have been previously reported as dissociation-induced stress-response genes in single-cell transcriptomic data [58, 59]. We observe that the activity of DR 26+ varies among different cell types, aligning with previous findings [59, 60]. Disentanglement of the dimensions in DRVI allows users to selectively remove irrelevant dimensions from the latent space. For instance, removing DR 26, a technical variation, enhances integration quality (Supplementary Fig. 5a-c).

Dimensions DR 21+ and DR 16- represent two consecutive stages in the developmental lineage of erythrocytes (Fig. 1h). The scatter plot illustrates an increase in DR 21 followed by a delayed decrease (increase in magnitude) of DR 16 in erythrocyte progenitors. As erythrocyte progenitors transition to mature erythrocytes, DR 21 decreases to zero while DR 16 reaches its maximum magnitude. Finally, DR 16 increases back to zero as the cells leave this developmental stage. Looking at the relevant genes, CD36 for DR 21+ is a known marker of erythrocyte progenitors [61]. For DR 16-, HEMGN (EDAG) is a key driver of erythropoiesis [61, 62], and RHCE encodes the RhD antigen in the red blood cell (RBC) membrane [63] (Fig. 1i).

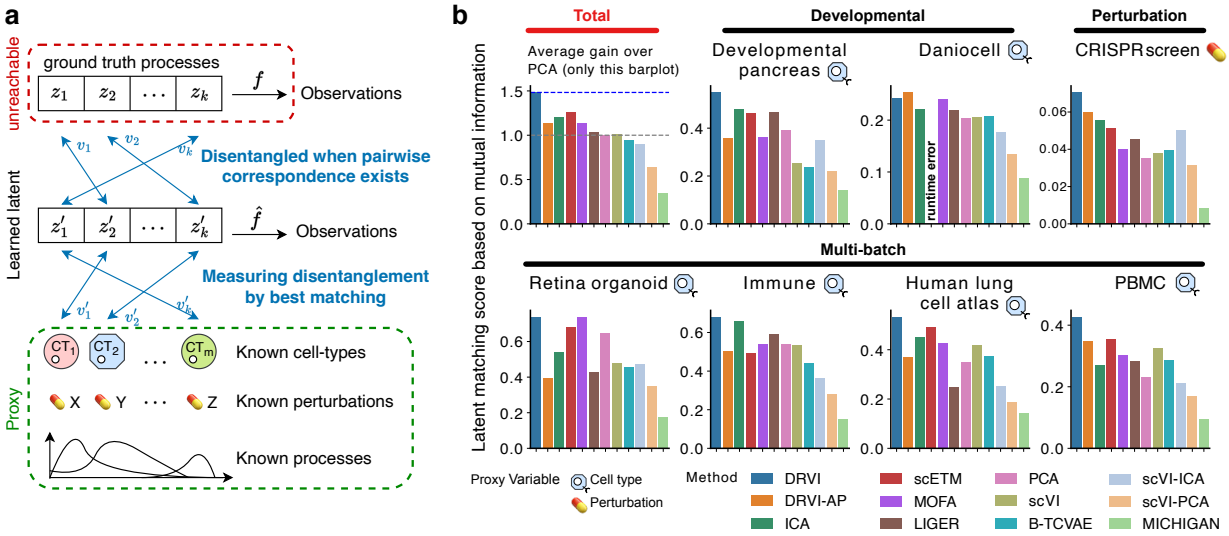
DRVI outperforms previous disentanglement approaches while maintaining integration quality

Disentangled representation learning aims to identify biological processes in different axes of the latent space. We refer to all (often unknown) biological processes that are represented in the data and have contributed to its formation as ground truth (GT) processes. From the GT processes to the observed single-cell omics data, we can consider the biology describing cells as a stochastic GT generative function. We call a latent space to be disentangled when latent space axes correspond to the GT processes (see 'Definitions and Notations' in Methods). Accordingly, we can measure the disentanglement by finding and assessing the best matching between GT processes and latent space dimensions. However, the GT processes and the GT generative function are often unreachable. Here, we propose to address this conundrum by assessing the disentanglement using supervised information such as annotated cell types, applied perturbations, or basic known biological processes such as cell cycle (Fig. 2a). Disentanglement is therefore measured by matching latent variables and proxy variables, technically called latent matching score (LMS) in the disentanglement literature [37] - by default, we measure the matching quality based on the Scaled Mutual Information (SMI, see 'Metrics' in Methods).

To systematically assess how well DRVI learns disentangled latent representations, we benchmarked DRVI against eight unsupervised methods covering linear (PCA, ICA, MOFA, LIGER, and scETM) and nonlinear (scVI, β-TCVAE, and MICHIGAN) representation assumptions. Among these methods, ICA, MOFA, LIGER, scETM, β-TCVAE, and MICHIGAN are claimed to provide meaningful latent representations. We also benchmarked DRVI-AP to specifically assess the effect of LSE aggregation. Furthermore, we applied PCA and ICA to scVI's latent space to demonstrate the inherent complexity of the disentanglement problem in this context. To cover a wide range of scenarios, we evaluated the mentioned methods on two atlas-scale multi-batch data (HLCA [7] and PBMC [64]), a small multi-batch data (immune [53]), a dataset consisting of primary and organoid cells (retina organoid [10]), one small-scale developmental data (developmental pancreas [4]), one large-scale developmental data (Daniocell [65]), and one genetic perturbation dataset (CRISPR screen [5]), summarized in Supplementary Table 2. We used finest-level cell-type annotations as proxy GT processes for all but perturbation data, for which we used perturbation signatures as proxy for underlying GT processes (see 'Datasets' in Methods).

When evaluating DRVI on individual datasets, DRVI consistently outperforms or matches the best benchmarked methods in providing disentangled latent representations (Fig. 2b). Constrained models like ICA, MOFA, LIGER, scETM, and DRVI-AP sometimes achieve acceptable disentanglement, but their performance is inconsistent and generally limited to a few specific datasets. Consistent with the theoretical results [34], unconstrained nonlinear models, including scVI, β-TCVAE, and

MICHIGAN, were unable to achieve significant improvements in disentanglement.



c

Method	Retina organoid			Immune			Human lung cell atlas			PBMC			Average		
	Batch correction	Bio conservation	Total	Batch correction	Bio conservation	Total	Batch correction	Bio conservation	Total	Batch correction	Bio conservation	Total	Batch correction	Bio conservation	Total
DRVI	0.59	0.71	0.66	0.61	0.68	0.65	0.58	0.67	0.63	0.38	0.66	0.55	0.54	0.68*	0.62*
DRVI-AP	0.60	0.67	0.64	0.61	0.70	0.66	0.58	0.66	0.63	0.40	0.61	0.53	0.55	0.66	0.61
LIGER	0.58	0.72	0.67	0.68	0.67	0.67	0.57	0.54	0.55	0.51	0.59	0.56	0.58*	0.63	0.61
scVI	0.62	0.65	0.64	0.65	0.63	0.64	0.60	0.64	0.63	0.39	0.60	0.52	0.56	0.63	0.60
scETM	0.46	0.74	0.63	0.48	0.67	0.59	0.47	0.67	0.59	0.43	0.61	0.54	0.46	0.67	0.59
MOFA	0.44	0.72	0.61	0.39	0.67	0.56	0.50	0.68	0.61	0.44	0.65	0.57	0.44	0.68*	0.59
ICA	0.48	0.66	0.59	0.43	0.66	0.57	0.50	0.64	0.59	0.44	0.58	0.53	0.47	0.64	0.57
PCA	0.43	0.68	0.58	0.37	0.67	0.55	0.39	0.65	0.54	0.38	0.60	0.51	0.39	0.65	0.55
B-TCVAE	0.44	0.69	0.59	0.41	0.64	0.55	0.40	0.60	0.52	0.34	0.58	0.48	0.40	0.63	0.54

Figure 2: **Benchmarking DRVI in terms of disentanglement and integration quality. a,** Evaluation of disentanglement requires access to ground truth biological processes, which are often unreachable. We use cell type annotations, perturbations, and known processes as proxies when available. **b,** Disentanglement was evaluated using latent matching scores, calculated by comparing the alignment of latent dimensions to proxy variables using mutual information. These scores were obtained from benchmarked models across diverse datasets. The proxy variable that serves as the ground truth for each dataset is specified. The total score is determined by averaging the improvements relative to PCA (top left). DRVI achieves the best result. **c,** scIB Integration scores for each individual multi-batch dataset. The last three columns indicate the average scores, with the best method in each category marked with a star. DRVI achieves the best results in unsupervised integration.

To address the inherent variability across different datasets, we normalized each metric by the corresponding PCA score before aggregation. The resulting metric is referred to as the gain over PCA (see 'Metrics' in Methods). The total score was calculated by averaging the gain over PCA across all datasets (Fig. 2b, top left, and Supplementary Fig. 7). We find that also in the overall metric, DRVI outperforms other benchmarked models, achieving a 17% improvement over scETM, the next best-performing model. DRVI's superior performance in disentanglement compared to linear models suggests that the underlying processes in single-cell omics data are inherently nonlinear, and incorporating a nonlinear generative decoder and negative binomial noise model improves the

disentanglement over linear baselines. DRVI and DRVI-AP are optimized using the same objective as scVI. The significant gap between DRVI and scVI demonstrates the effectiveness of the DRVI decoder architecture in enforcing disentanglement, and the gap between DRVI and DRVI-AP demonstrates the significant contribution of the LSE pooling function. The results indicate that conventional methods like PCA and ICA are insufficient for disentangling scVI's latent dimensions, highlighting the necessity for novel architectural approaches, such as DRVI.

To further evaluate the benchmarked methods, we explored two additional similarity functions: same-process neighbors (SPN) and absolute Spearman correlations (ASC) —Although ASC is generally not suitable for discrete targets (see 'Metrics' in Methods). DRVI demonstrated 13% improvement in LMS-SPN and 3% decline in LMS-ASC compared to the closest competitor (Supplementary Fig. 8.). We also present the results from two additional disentanglement metric families (Supplementary Fig. 9): Most Similar Averaging Score (MSAS) and Most Similar Gap Score (MSGS). MSAS measures the maximum similarity between each GT process and latent dimensions, while MSGS penalizes similarity to other dimensions (see 'Metrics' in Methods). Particularly, when using the Scaled Mutual Information (SMI) similarity function, Most Similar Gap Score (MSGS) directly reduces to the Mutual Information Gap (MIG) metric as defined in [66]. Although DRVI consistently outperforms others in MSAS, different individual methods, including DRVI, ICA, LIGER, and scVI-ICA, achieved the best performance in certain datasets and sub-metrics of the MSGS family. The LMS metric family is favored due to its alignment with the formal definition of disentanglement and its ability to penalize overlapping processes within a single dimension, a key obstacle to interoperability. For the sake of completeness, we added unaggregated SMI and SPN metrics of top-picked methods for the Immune and the HLCA datasets (Supplementary Figs. 10-13).

In addition to real-world datasets, we benchmarked models on two simulated datasets (see 'Datasets' in Methods and Supplementary Fig. 14). One dataset featured unique cellular processes per cell, and the other included up to four overlapping processes per cell group. While DRVI, ICA, MOFA, PCA, LIGER, and scVI-ICA performed well in the simplistic non-overlapping scenario, only DRVI and LIGER proved capable of accurately identifying overlapping processes in the second scenario (Supplementary Fig. 15).

Current unsupervised methods such as ICA and MOFA enforce disentanglement through strongly limiting model constraints like linearity and sparsity, restricting their ability to represent rich, multi-batch data. To assess the applicability of disentangled models on complex data, we assess integration quality alongside disentanglement. We consider the datasets containing multiple batches and assessed the biological signal conservation and batch correction quality based on the scIB metrics [53]. Considering the average performance across all datasets, DRVI achieved the highest total integration score among all benchmarked methods (Fig. 2c). This is due to its ability to correct batch effects comparably well (a little worse than scVI) while preserving significantly more biological signals (comparable to the best-performing model in bio-conservation). We can conclude that DRVI can learn disentangled latent representations in complex multi-batch scenarios while integrating the batches comparable to the best unsupervised counterparts.

While the constrained decoder in DRVI leads to a decrease in reconstruction performance (Supplementary Fig. 16), the high biological signal conservation observed in scIB metrics and the strong disentanglement results indicate that DRVI forms rich latent representations in terms of known cell-type variations. To further investigate, we varied the target-KL weight from 0.01 to 10 (default 1.0) for DRVI and CVAE on HLCA and Immune datasets. Compared to the decoder architecture, target-KL was the primary driver of reconstruction performance decline (Supplementary Fig. 17a,e). We show that while disentanglement in CVAE was significantly affected by reducing this parameter, DRVI showed considerable robustness (Supplementary Fig. 17c,d,g,h). Thus, users seeking good reconstruction can safely reduce the target-KL weight without majorly compromising disentanglement.

DRVI allows atlas-level identification of biological processes

To demonstrate the capability of DRVI in disentangling latent factors and identifying biological signals in complex data, we applied DRVI to the Human Lung Cell Atlas (HLCA) [7]. We use the curated core of this atlas consisting of 166 samples from 14 datasets and including a total of 585K cells with curated annotations across 61 distinct cell identities in the finest level annotations. Disentangling 61 cell states requires a minimum of 61 dimensions. Accordingly, we applied DRVI (Fig. 3a) and other benchmarked methods (Supplementary Fig. 18) to form 64-dimensional latent spaces.

To interpret the learned latent factors, we first excluded 12 vanished dimensions, which exhibit near-zero values for all cells. The phenomenon of vanishing dimensions is explained later. We then identified 31 latent dimensions corresponding to cell types in the finest-level annotations. We observed a favorable one-to-one correspondence between latent dimensions and many cell identities (Fig. 3b top), while such correspondences for other benchmarked methods are much weaker (Supplementary Fig. 19). Disentanglement metrics confirm this observation quantitatively. The remaining 21 dimensions are expected to represent disentangled underlying processes that have yet to be identified (Fig. 3b bottom). Some dimensions exhibit two irrelevant processes in their negative and positive directions. This includes 4 cell-type dimensions that represent a biological process in their other direction. Therefore, we analyze the two directions separately.

When examining the cell-type indicator dimensions, quite a few dimensions are exclusively activated in specific cell types. For instance, DR 17+ is uniquely active in AT1 cells, DR 31+ in plasma cells, DR 39- in DC1 cells, and DR 39+ in migratory DCs. A greater number of such dimensions suggests better disentanglement. Some other dimensions indicate broader categories covering multiple cell types, such as a coarse cell type or a shared behavior. For example, DR 4- indicates Macrophages (Mph), DR 5+ indicates ciliated cells covering deuterosomal and multiciliated cells, and DR 32+ indicates AT0 and pre-TB secretory, together known as transitional club-AT2 cells [7] (Fig. 2b). Given that the disentanglement metrics already assess the identification quality of the cell-type indicator dimensions, we will not focus on evaluating this aspect. However, it is noteworthy that DRVI effectively identifies most of the challenging cell types highlighted by the HLCA study, including Migratory DCs (identified by DR 39+), AT0 cells (identified by DR 34+), hillock-like epithelial cells (identified by DR 26-), and pre-TB secretory through transitional club-AT2 cell (identified by DR 32+). Other methods often fall short in identifying these cell types, particularly Migratory DCs, which only DRVI can capture along its latent dimensions (Supplementary Figs. 12 and 13). Interestingly, pre-TB secretory cells are also described in the HLCA study only through marker genes corresponding to transitional club-AT2 cells [7]. This is in full alignment with DRVI's latent geometry where AT0 and pre-TB secretory are together identified by DR 34+. These cell types are only identifiable after multiple iterations of clustering [7] or clustering at very high resolutions with hundreds of clusters (Supplementary Fig. 22). Given that DRVI can capture these cell types in just 52 non-vanished dimensions, we expect it to significantly facilitate the systematic discovery and seed annotation of novel and rare cell types in future cell atlases or related studies.

To demonstrate the validity of the identified variations beyond cell types, we followed the interpretability pipeline and found the non-linear gene programs related to each non-cell-type dimension (Supplementary Fig. 21). Then, we ran a GSEA for all retrieved sets. For gene sets not identified by GSEA, we looked up the top relevant genes in the literature. As a result, we identified relevant biological descriptions for 22 of the 25 dimensions expected to represent biological processes (Fig. 2c). In the following, we detailed five example dimensions and visualized them (Fig. 3d-h). The rest of the dimensions presenting biological processes are identified and briefly explained in Supplementary Note 3.

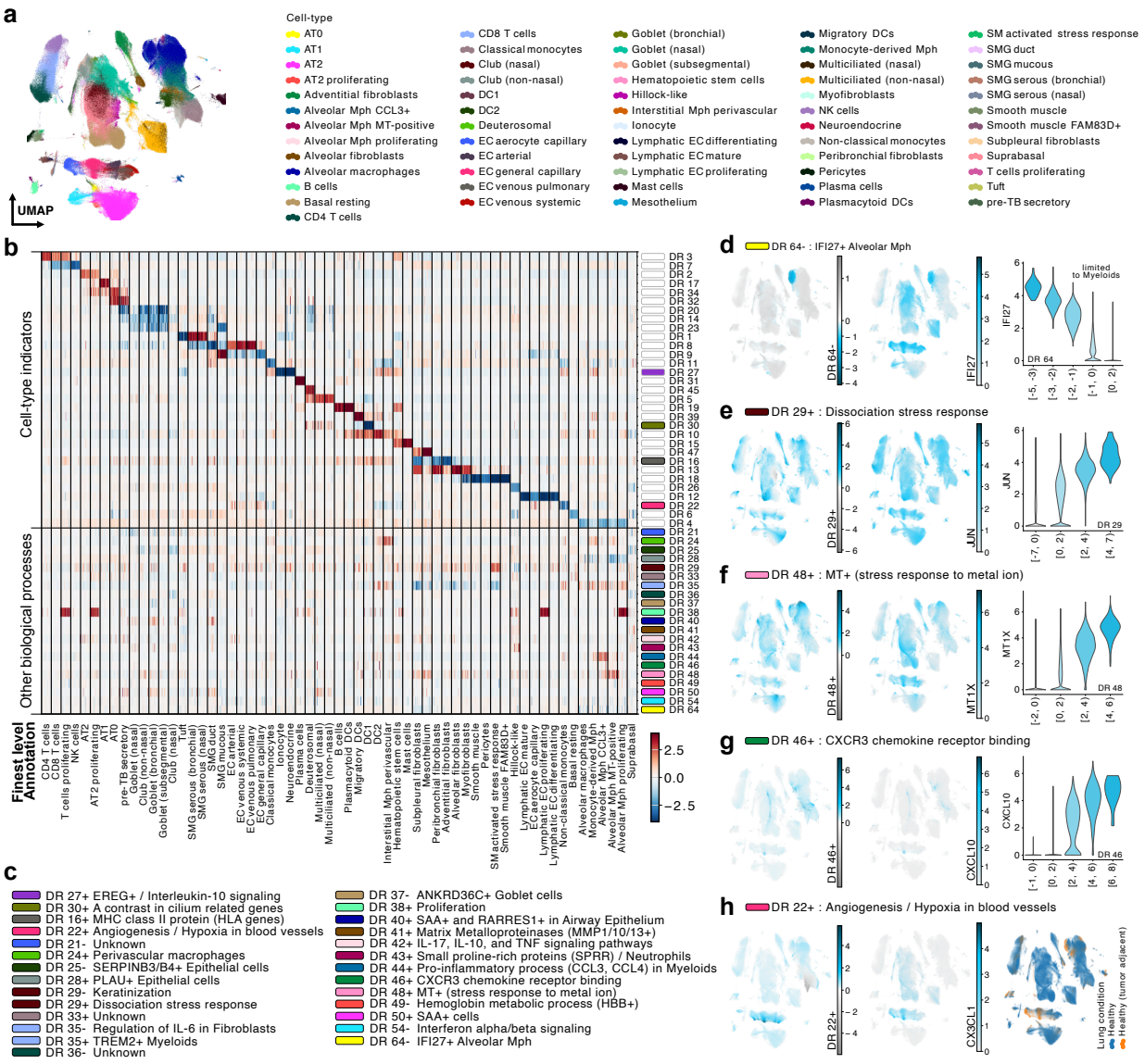


Figure 3: **DRVI uncovers the variations in HLCA beyond cell types. a,** UMAPs of the HLCA integrated by benchmarked methods. Cells are colored by third-level annotations. **b,** DRVI latent dimensions. The heatmap illustrates each dimension's activity in cells vertically grouped by finest level annotations. Cells are subsampled to have an equal number of cells in each cell type. The dimensions are horizontally grouped into two categories: cell-type indicators dimensions (top) and dimensions indicating other biological processes (bottom). Dimensions representing non-cell-type processes are color-coded on the right-hand side of the plot. Vanished dimensions are omitted. For better illustration, some tail values are clipped. **c,** For each dimension indicating a non-cell-type biological process, a short description is provided. **d-g,** Demonstration of five example dimensions. Left: Activity of the example dimensions on the UMAP. Middle: The expression of a relevant gene on the UMAP. Right: Violin plots indicating the expression levels of relevant genes versus the activity of dimensions. **h,** The activity of DR 22+ and one of the identified genes in the nonlinear gene program. This dimension indicates angiogenesis and hypoxia in endothelial cells and is highly active in some tumor-adjacent cells. Mph, Macrophage.

As the first example, we showcased a subtle variation within Alveolar Macrophages (AMs - a subset of Myeloid cells), identified by DR 64- (Fig. 3d). IFI27 is the only gene significantly associated with this dimension, and is known to define a subcluster of AMs [67–69]. Despite its expression in other cell types, such as blood vessels, DRVI has nonlinearly localized its expression to AMs due to the

distinct patterns of co-regulated genes. The violin plot of the IFI27 gene with respect to DR 64 in the Myeloid cells confirms the mentioned relationship.

The activity of DR 29+ spans almost all cell types. This dimension is highly associated with the genes ATF3, EGR1, JUN, FOSB, and FOS as top indicators of the nonlinear gene program, suggesting that it represents single-cell dissociation stress response [58, 59]. The violin plot of the JUN gene with respect to DR 29 highlights this relationship (Fig. 3e). Interestingly, we observed the same disentangled process (DR 26) in the immune dataset. Likewise, post-hoc removal of DR 29 enhances integration quality (Supplementary Fig. 5d-f).

The upregulation of Metallothionein (MT) family genes characterizes DR 48+, indicating a response to metal ions. The monotonic relationship between DR 48 and MT1X, visualized in the violin plot, further supports this finding (Fig. 3f). While this dimension is active in MT-positive AM cells, DR 48+ is not exclusive to this cell type. Moreover, MT+ and IFI27+ AMs (DR 48+ and DR 64-) are not disjoint biological subtypes, indicating DRVI's superiority over traditional clustering methods in representing parallel overlapping processes.

A cell subset characterized by the expression of CXCL9, CXCL10, and CXCL11 is highlighted by DR 46+. GSEA analysis suggests an association with CXCR3 chemokine receptor binding [70], which is particularly relevant in the context of lung diseases [71, 72]. This dimension is not specific to any particular cell cluster, indicating a shared response signature.

DR 22+ is primarily active in endothelial cells, with GSEA analysis suggesting an association with angiogenesis. The upregulation of top relevant genes, including CX3CL1, SERPINE1, AKAP12, GRP4, and FSTL3 under hypoxic conditions [73–77], a common phenomenon observed in tumor-adjacent environments [78], indicates a potential role in tumor growth. The increased abundance of tumor-adjacent endothelial cells in the higher end of DR 22+ further supports this interpretation. (Fig. 3h and Supplementary Fig. 23)

Altogether, DRVI enables the disentanglement of biological processes within the HLCA, a large-scale cell atlas. This facilitates the identification of rare cell types and the underlying biological processes. DRVI encodes biological processes such as inflammatory responses, signaling pathways, and processes that introduce variations within cell types. These processes can be shared across related cell types or localized to specific populations, depending on their gene expression signatures.

We note that the cell-type labels are not required to identify variations beyond cell types. One can follow the interpretability pipeline and find relevant gene sets for all dimensions. These genes may indicate different biological signals, including cell types, known pathways, and novel sets. However, as we see in the HLCA example, a good annotation will always facilitate the analysis. In addition, identifying biological programs doesn't depend on GSEA for derived gene sets. Instead, it can involve searching scientific literature, leveraging existing atlases of marker genes and biological programs, using automated tools like language models, or performing expert examination by biologists. It is worth mentioning that since such supervised information is not given to the model, the quality of the labels and supervision does not affect or bias the derived signatures.

DRVI disentangles cell-cycle information

We applied DRVI and other benchmarked methods to single-cell mouse samples of the developmental pancreas at E15.5, consisting of 3,696 cells and 15 states in the finest level annotation. All methods except MICHIGAN, β-TCVAE, and MOFA presented well-separated stages in their UMAPs (Fig. 4a). As a starting point, we considered the cell cycle as one of the well-studied biological processes. While the cell cycle can be a confounding factor in some integration tasks, we expect unsupervised dimension reduction methods to retain this information and disentanglement methods to identify such biological signals. From the practical point of view, disentangling these biological signals allows researchers to analyze the remaining dimensions without the interference of confounding factors.

The cell cycle is a complex, multidimensional process encompassing G1, S, G2, and M phases, frequently modeled as a set of single-dimensional processes corresponding to each phase [79]. Here, we used the cell-cycle G2M and S scores as two ground truth biological processes and measured each method's capability to disentangle these processes. Specifically, for each method, we calculated the mutual information between each dimension and the processes of interest. DRVI, MOFA, LIGER, and ICA were the only models that could separate at least one of the S and G2M scores (Supplementary Fig. 24b,d). DRVI achieved 10% higher sum of mutual information compared to the next best model, and exhibited a smoother representation of cycling cells with clearer state boundaries (Fig. 4d).

DRVI identifies developmental stages

In parallel to cell-cycle in the developmental pancreas example, cells undergo development, namely endocrinogenesis, ultimately reaching alpha, beta, delta, and epsilon as the terminal states. To quantitatively assess the quality of the embeddings in different settings, we applied CellRank [80] to infer macrostates, cell fates, and terminal states. We first applied scVelo [81] to infer cellular dynamics. scVelo determines possible future cell states by considering (I) spliced and unspliced counts and (II) the latent space where cell similarities are calculated. Therefore, we fed embeddings from DRVI and other benchmarked methods into scVelo as different options. Our findings indicate that DRVI improves the quality of velocity vector fields (Supplementary Fig. 25a). This improvement is evident in the Terminal State Identification (TSI) score, which quantifies the overlap between identified macrostates and ground-truth terminal states as the number of macrostates increases. Only DRVI and LIGER achieved a full TSI score (Fig. 4b and 25b-d). Furthermore, only DRVI embedding allowed CellRank to automatically identify and differentiate all four terminal states, distinguishing them from the transitioning and initial states (Fig. 4c Supplementary Fig. 25e). Within its interpretable disentangled latent space, DRVI identifies a set of dimensions and processes leading to these four terminal states, which we selected and described below.

Dimensions DR 23+, DR 1+, and DR 11- describe beta lineage formation, covering the cells from Fev+ beta cells to the beta cells producing insulin (Fig. 4e). Mnx1 and Pdx1 are among the top genes identified for DR 23+, known to promote the differentiation of endocrine progenitors into beta cells [82, 83]. Notably, Pdx1 is also among the top genes for DR 1+, aligning with its role as a maintainer of the beta cell state [84]. DR 1+ highlights the earliest mature state of the developmental pancreas. Sytl4, one of the top identified genes for this dimension, is a known beta marker gene regulating the maturation of beta cells [85]. Additionally, Klein et al. reported other relevant genes for this dimension, such as Sntg1, Mapt, Tenm3, and Plppr1, as beta-specific marker genes [86]. Finally, DR 11- with Ins1 and Ins2 as top identified genes, indicates mature beta cells producing insulin. We illustrated scatterplots highlighting the latent geometry of the endocrine progenitor cells developing towards mature beta cells (Fig. 4e).

DR 9+ and DR 26- model the development towards Delta cells (Fig. 4f). DR 9+ is mainly identified by Hhex, required for delta cell differentiation [87]. Moreover, DR 26- is mainly identified by the Sst gene, secreted by delta cells.

DR 8+ primarily captures alpha and pre-alpha cells, along with a subset of epsilon and pre-beta cells. While Pou6f2 and Pde1a are computationally suggested markers for alpha cells [86], the rest of the genes are not specific to the alpha cells. The inability to fully disentangle the lineage towards alpha cells in DR 8+ may be attributed to the limited number of latent dimensions, insufficient signal in the sequenced data, or modeling limitations. Gcg is the main identifier of DR 4+, suggesting that this dimension highlights alpha cells (Fig. 4g).

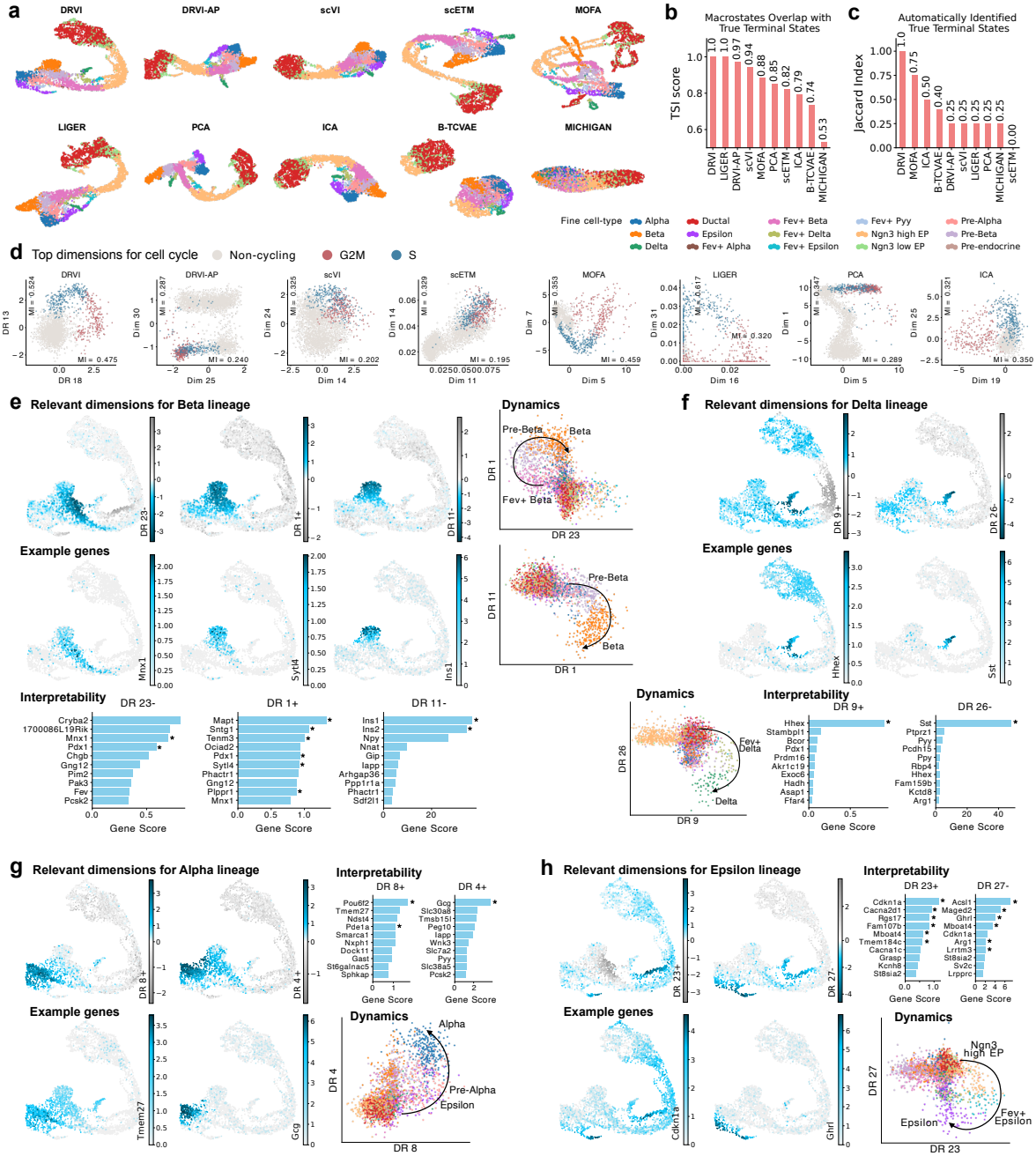


Figure 4: **Identification of cell cycle and developmental stages by DRVI. a,** UMAPs of the developmental pancreas data applied to the latent spaces generated by benchmarked methods. Cells are colored by finest-level annotations. **b,** Terminal State Identification (TSI) scores for DRVI and benchmarked methods. This metric quantifies the overlap between identified stationary states and ground-truth terminal states as the number of macrostates increases. **c,** Jaccard index between the true known terminal states and the automatically identified terminal states using CellRank based on the embeddings from different benchmarked methods. **d,** The learned cell-cycle manifolds. The scatterplots show dimensions maximizing the mutual information with respect to the S-score and G2M-score, with points colored by the dominant cell cycle phase. Cells are assigned to the non-cycling category if both S and G2M scores are below 0.2. Otherwise, they are assigned to the state with the higher score. **e-h,** Relevant processes for beta (e), delta (f), alpha (g), and epsilon (h) differentiation. For each of these terminal states, a group of relevant dimensions is provided. For each dimension, the top relevant genes in the identified gene program are illustrated as a barplot, and the activity of a relevant gene on the UMAP is plotted. In addition, for consecutive developmental stages, a scatterplot illustrates the differentiation process. The arrows are for visualization and are not derived from the model.

The lineage from endocrine progenitors towards epsilon cells is specified by DR 23+ and DR 27-. Among the top identified genes of DR 23+, Cdkn1a, Cacna2d1, Rgs17, Mboat4, Fam107b, and Tmem184c are reported as computational drivers of epsilon progenitors [86]. The presence of Ghrl in the nonlinear gene program of dimension DR 27- indicates that this dimension identifies the epsilon cell type. Additionally, Ascl1, Maged2, Ghrl, Mboat4, Arg1, and Lrrtm3, identified for this dimension, were computationally derived as drivers for the epsilon population [86]. The scatterplot of DR 23 and DR 27, illustrates the differentiation of endocrine progenitors towards epsilon cells (Fig. 4h).

In conclusion, we explored DRVI's latent dimensions that identify processes related to differentiation toward terminal states. We found that DRVI effectively captures most of the developmental stages leading to these terminal states. The interpretability of DRVI allows identifying the relevant genes for each stage. Many of these gene sets align with those previously identified using optimal transport on additional time points [86]. Besides, a quantitative evaluation based on CellRank and scVelo showed that DRVI improved terminal state identification in this dataset.

DRVI preserves geometry of combinatorial effects

Interventions such as drug or genetic perturbations give rise to natural ground truth components to be identified by disentanglement approaches. To test this in a combinatorial setting, we applied DRVI to a CRISPR screen dataset consisting of 104,339 K562 cells perturbed by 106 single-gene and 131 combinatorial perturbations [5]. The disentanglement metric quantifies how well different latent dimensions can isolate the effects of specific perturbations. To gain a deeper understanding of how well different models can identify perturbations, we focus on IRF1 and SET perturbations.

Accordingly, we determined the two dimensions with the highest mutual information with IRF1 and SET perturbations for DRVI and the other benchmarked models (combinatorial perturbations were not considered). Only DRVI and DRVI-AP, and to some extent ICA, scVI-ICA, and LIGER were capable of identifying perturbations in distinct latent dimensions and displaying additive property in their latent spaces (Supplementary Fig. 26), with DRVI outperforming ICA by 12% and 2% in mutual information scores for IRF1 and SET, respectively. This is while SET is one of the perturbations with the highest relevance to a dimension in the scVI model.

Excitingly, disentanglement enables us to visualize the combinatorial effects in the latent space of DRVI, DRVI-AP, and ICA: Considering P1 and P2 as the nonlinear gene programs related to the dimensions with the highest correspondence to IRF1 and SET, we can argue that P1 is activated in IRF1, P2 is activated in SET, and both P1 and P2 are independently activated in IRF1+SET (Supplementary Fig. 26). While this indicates an additive behavior of P1 and P2 programs, we cannot claim the effect beyond P1 and P2 is fully additive. The reason is that there may be other dimensions in the latent space representing a newly activated program when the two perturbations are present. Accordingly, one should also consider the activity of the rest of the dimensions before making claims on the effects of the combinatorial perturbations.

DRVI's latent space effectively models double perturbations. For instance, the CBL+CNN1 perturbation significantly increases DR 24+ (hemoglobin expression, Supplementary Fig. 27a) and DR 5+ (SLC25A37 iron importer, which facilitates heme synthesis, Supplementary Fig. 27b). This suggests hemoglobin overexpression, which was confirmed by flow cytometry [5]. In another example, the effect of ETS2 (DR 15-) is canceled when combined with DUSP9 (Supplementary Fig. 27c). Finally, the mutual inhibition between DUSP9 and MAPK1 can be inferred from the decreased response signatures of both genes (DR 15- and DR 8-) in the combined perturbation (Supplementary Fig. 27d).

DRVI's learned factors saturate when providing more than enough latent dimensions

So far, we have demonstrated DRVI's use as a tool for discovering underlying biological processes. However, the exact number of these processes and, thus, latent dimensions is mostly unknown. Here, we examine the effect of the number of latent dimensions in DRVI.

To start, we applied DRVI to the immune dataset studied in Fig. 1 using various numbers of latent dimensions. Interestingly, we observe that when increasing the number of latent dimensions K, some dimensions start to vanish: To quantify this, we examined the maximum absolute value and standard deviation of each latent dimension for each model (Supplementary Fig. 28a). We clearly observe a group of vanished latent dimensions characterized by a very low maximum absolute value and low standard deviation. Accordingly, we simply define a dimension as vanished if the maximum absolute value in that dimension is less than 1. These dimensions have a negligible impact on the model's latent space variations and reconstruction capabilities. This phenomenon is also observed in other KL-regularized methods [88]. The models trained on the immune dataset use all latent dimensions until $K = 32$ and saturate at $K = 128$, employing 57 non-vanished latent dimensions (Supplementary Fig. 28b). We evaluated the disentanglement by the LMS-SMI metric and integration quality by total scIB score. Providing more latent dimensions does not significantly degrade disentanglement or integration quality (Supplementary Fig. 28c,d).

We repeated this procedure for the remaining datasets. Except for the very complex whole-organism Daniocell example, the number of non-vanished latent dimensions saturates before 512 dimensions (Supplementary Fig. 28d). Daniocell, as an extreme case with 156 cell identities and a more complex network, also saturates, but with a larger number of latent dimensions (Supplementary Fig. 29d,e). The disentanglement metric, LMS-SMI, improves for all datasets while the number of non-vanished dimensions increases. This is while we observe no significant drop in integration quality using wider networks (Supplementary Fig. 28f,g).

The number of required dimensions is influenced by data complexity factors such as the number of cell types, features, and developmental stages. To confirm that this phenomenon persists when using all genes, we trained DRVI on the immune data with all genes. In addition, we increased the number of hidden neurons in the encoder and decoders from 128 to 512 to force no limitation by these networks. The results indicate that while training on the full gene set requires more latent dimensions, the saturation property still holds (Supplementary Fig. 29a-c).

Altogether, the number of utilized latent dimensions saturates, with no significant decline in disentanglement and integration performance, when DRVI is provided with a sufficient number of latent dimensions. While this phenomenon is not specific to DRVI, it facilitates choosing the latent dimensionality, when the true number of biological processes is unknown. Therefore, we recommend starting with a reasonable number of latent dimensions and iteratively increasing this number until most dimensions vanish. However, given that none of the baseline methods were designed to handle a large number of latent dimensions, we used 32 or 64 dimensions in our benchmarks to maintain fairness.

The identified programs are consistent and transfer across datasets

To demonstrate the generalizability of DRVI, we individually trained four DRVI models, each with 128 latent dimensions, on four Human PBMC datasets in CellHint Blood data (Supplementary Fig. 30a). These datasets comprise a total of 336k cells with expert-curated annotations [89]. We observed that the number of non-vanished dimensions ranged from 31 to 43 (Supplementary Fig. 30b,c), with the variability likely stemming from minor biological and technical differences between the datasets. We then directly applied each model trained on one dataset to the other three and measured the disentanglement capabilities. Interestingly, the disentanglement quality remained intact, which suggests the model is generalizable across datasets with similar underlying biology

(Supplementary Figs. 31a, 32). Additionally, we assessed the consistency of the identified programs (the sorted set of relevant genes) across the four models using the Rank-Biased Overlap (RBO) similarity score [90]. The results indicate a favorable one-to-one correspondence for most of the identified programs (Supplementary Figs. 31b). One example is a consistent signature for Tcm/Naive cytotoxic T and Tem/Temra cytotoxic T cell types across these four models (Supplementary Figs. 31c).

DRVI extends to further data modalities

We primarily demonstrate the utility of DRVI in single-cell transcriptomics data. However, DRVI easily extends to other modalities, such as single-cell DNA accessibility (scATAC), to obtain disentangled representations. For this purpose, DRVI supports Negative Binomial, Poisson, and Gaussian (with MSE loss) noise models. As an example, we applied DRVI to the single-cell ATAC modality of the NeurIPS 2021 multimodal data integration challenge [91]. Specifically, we employed the Poisson noise model on ATAC fragments, which has recently been shown to be the proper assumption for this modality [24]. To assess disentanglement and integration, we compared DRVI against DRVI-AP, PoissonVI [24] (a recent best-performing method), and PeakVI [92] (a widely used method in scATAC-seq integration). As in scRNA-seq, we observe that DRVI combines disentaglement with robust integration performance: DRVI improves disentanglement with a 97% improvement in LMS-SMI, a 167% improvement in LMS-SPN, and a 14% improvement in LMS-ASC over the next best-performing model (Supplementary Fig. 33c). Additionally, DRVI significantly outperforms PoissonVI and PeakVI in total scIB integration score (0.63 compared to 0.60), demonstrating its applicability to the ATAC modality (Supplementary Fig. 33d).

Discussion

We have presented DRVI, an unsupervised deep generative model to learn disentangled representations of single-cell omics data. Disentanglement, as the main focus of DRVI, allows the representation of distinct biological processes in different latent dimensions. Nonlinearity improves integration and disentanglement quality and makes the model scalable to cellular atlases. With the presented examples and metrics, we demonstrate the utility and value of disentanglement in single-cell omics analysis. We have shown how disentanglement can describe cell types, capture biological processes beyond cell types, identify developmental stages, and isolate unwanted variations like dissociation stress response and cell cycle in different latent dimensions. Disentangled representations are additionally useful for perturbation modeling, in particular for combinatorial effects. We additionally highlighted the identification of cell-type-specific variations that are challenging to disentangle using linear methods yet captured by DRVI as a nonlinear disentanglement model.

In contrast to clustering in latent space - which provides separate groups based on the dominant signal - disentangled representations allow us to capture dominant and secondary signals in parallel in different latent space dimensions - which enables the identification of variations beyond cell types. Accordingly, considering all latent dimensions along their non-linear gene programs may enable the building of a dictionary of active biological processes in data. DRVI's ability to generate sparse latent variables and remove unnecessary dimensions suggests its potential for scaling to larger, more complex datasets. A future direction may be to expand the model to multiple atlases and create a comprehensive, unified dictionary.

DRVI's additive decoder architecture with LSE pooling allows interpretability and provable disentanglement under specific conditions. It imposes an inductive bias independent of the learning framework and training paradigm, applicable to a wide range of generative models in single-cell biology. By adopting DRVI's decoder, future models may achieve enhanced interpretability and disentanglement across diverse biological applications, regardless of noise model, prior distribution, or

training paradigm choices. An additional feature of this approach is that it enables opening up the black box of fully-connected autoencoders, thereby enabling a higher explainability of the results and errors. Exploring additional nonlinear pooling functions, their theoretical properties, and practical performance may be another line for future research.

Besides the advantages mentioned, DRVI's unique properties make it a promising choice to extend into further modalities and applications. By extending DRVI to multiple modalities, users can interpret each disentangled dimension as a topic and leverage these topics as a bridge to analyze the relationship between modalities, such as the regulation of transcription by chromatin accessibility [93, 94]. We have observed that DRVI's latent dimensions represent consecutive developmental stages as cycles, suggesting its potential for improved modeling of spliced/unspliced counts in velocity data [81]. Moreover, the model's additive property indicates its potential for modeling combinatorial effects, as seen in perturbation and CIM-seq data [5, 95].

Spatial omics technologies have become increasingly popular for analyzing cellular arrangements and interactions within the spatial context [96, 97]. Given DRVI's effectiveness in identifying variations, we anticipate its ability to identify biologically meaningful regions. In addition, in many spatial technologies, counts are accumulated per spot or segment [98]. We believe DRVI's assumption of additivity in count space makes it a promising framework for modeling such data, potentially leading to improved deconvolution and resilience to imperfect segmentation - the underlying framework of additive decoders has indeed been proposed in the imaging domain, which often has such compositionality questions. The model will require further extensions to leverage spatial information for these applications.

Despite promising results, our work, as a step towards unsupervised disentangled representations in single-cell omics, has limitations. DRVI is computationally more expensive than entangled counterparts such as scVI or DCA. The computational cost, in terms of floating-point operations, scales linearly with the number of genes, cells, and latent dimensions. Consequently, it is currently infeasible to simultaneously scale the model to multiple atlases with tens of millions of cells, all genes, and thousands of latent variables. While limiting the decoder is essential for disentanglement [34], it can negatively impact reconstruction performance for use cases such as imputation. Nevertheless, this limitation can be significantly mitigated by using a lower weight for the KL loss. Although we established a systematic framework for evaluating disentanglement, its reliance on supervised proxy variables presents limitations. Cell-type annotations are incomplete, often coarse-grained, and insufficient to evaluate models beyond cell-type variations. Additionally, perturbations are often noisy and redundant and cover multiple processes, making it impossible to achieve high metric scores. The development and standardization of metrics based on well-curated biological processes are crucial for advancing disentangled representation learning.

In conclusion, DRVI enables unsupervised disentangled representation learning in single-cell omics. Researchers can thus explore complex biological processes beyond traditional cell-type annotations and build downstream analyses upon an interpretable latent space.

Code availability

DRVI is available as a package at `http://github.com/theislab/drvi`. The code to reproduce the results and figures is available at `http://github.com/theislab/drvi_reproducibility`.

Data availability

All datasets analyzed in this manuscript are public and have been published in other papers. We have referenced them in the manuscript and provided the pre-processing steps at `http://github.com/theislab/drvi_reproducibility`.

Author Contributions

A.A.M. and F.J.T. conceived the project. A.A.M. designed and implemented the algorithm. A.A.M. performed experiments and analyses. A.A.M. and F.J.T. wrote the manuscript. F.J.T. supervised the work.

Acknowledgments

We thank Malte D. Luecken and Lisa Sikkema for providing constructive feedback on this work and for their expertise in HLCA analysis, Sara Jimenes and Dominik Klein for their valuable insights and discussions on the pancreas data, Leander Dony for a fruitful discussion on the required number of latent dimensions, Lisa Sikkema, Alessandro Palma, Mojtaba Bahrami, and Anastasia Litinetskaya for their constructive criticism and valuable feedback on the manuscript, Mahya Sadat Aramoon for helping in the conceptual visualizations, Mohammad Lotfollahi for discussions on related works, Philipp Weiler for suggestions on the choice of developmental datasets, Daniel Strobl for suggestions on enrichment analysis, and Michaela Müller for a discussion on evaluation framework. Finally, we thank all the members of the Theislab for their insightful discussions and support. This work was supported by the Deutsche Forschungsgemeinschaft (DFG) through the Leibniz Prize and project 458958943 (grant number 5010338). A.A.M. is a member of the ELLIS PhD Program of the European Laboratory for Learning and Intelligent Systems (ELLIS) Society. F.J.T. acknowledges support from the Helmholtz Association's Initiative and Networking Fund through Helmholtz AI (grant number ZT-I-PF-5-01) and from the Leibniz Prize from DFG.

Competing interests

F.J.T. consults for Immunai Inc., CytoReason Ltd, Cellarity, BioTuring Inc., and Genbio.AI Inc., and has an ownership interest in Dermagnostix GmbH and Cellarity.

Methods

DRVI: Disentangled Representation Variational Inference

DRVI is an unsupervised deep generative model that employs log-sum-exp (LSE) pooled additive decoders to disentangle latent factors. Additive decoder architecture has been shown to enforce disentanglement under specific assumptions. The most critical of these are sufficient nonlinearity [37] or, alternatively, within the interaction asymmetry framework [48], interaction asymmetry coupled with sufficient independence.. However, combining current variational autoencoders in single-cell omics with additive decoders without additional modifications, does not satisfy these criteria, and therefore does not result in disentangled representations. To address this, and incorporate multiple modifications that achieve both theoretical and practical disentanglement of latent factors. Most importantly, we introduced LSE aggregation, which, from a mechanistic point of view, assumes that biological processes add up in count space (Supplementary Note 2). To increase the usability and highlight the utility of our architecture, we built our model based on the scvi-tools framework and minimized the deviation from the scVI model [19, 22, 99]. In the following sections, we first present the general unsupervised problem and the generative model. Next, we provide an in-depth explanation of the decoder, which is our primary methodological contribution. Finally, we discuss additional details, including interpretability and the evaluation framework.

Definitions and Notations

Here, we use capital letters to represent random variables (e.g., X), lowercase letters to represent realizations (observations) of random variables (e.g. x is a specific value that X takes), and uppercase bold letters to represent a matrix of observations (e.g. $\boldsymbol{X}$, where rows $X_1, \ldots, X_n$ are independent and identically distributed (IID) observations). We may also represent constants (such as the number of observations) with capital letters (e.g. N).

To define the disentanglement and local disentanglement (a weaker form of disentanglement), we consider a ground truth (GT) generative process that generates the observed samples. Disentanglement implies a universal one-to-one correspondence between learned and GT latent dimensions, while local disentanglement indicates such correspondence exists within each neighborhood of the latent space. The definitions are adapted from additive decoders [37].

Generative process: A generative process is defined based on its latent space $\mathcal{Z} \subset \mathbb{R}^K$, a probability distribution $p_z(z)$ over $\mathcal{Z}$, and a decoder $f : \mathcal{Z} \to \mathbb{R}^M$ that maps latent variable z to the observational space $\mathbb{R}^M$.

Partition: Given a set $\mathcal{A}$, we call a set $\mathcal{P}$ to partition $\mathcal{A}$ when elements of $\mathcal{P}$ are mutually exclusive ($\forall B \neq B' \in \mathcal{P}, B \cap B' = \emptyset$) and their union covers all elements of $\mathcal{A}$ ($\bigcup_{B \in \mathcal{P}} B = \mathcal{A}$).

Projection: Projection of a point $z \in \mathbb{R}^K$ into $B \subseteq \{1, \ldots, K\}$ is defined as $z_B := [z_{d_1} | \ldots | z_{d_l}] \in \mathbf{R}^{|B|}$ where d_i are ordered elements of B ($B = \{d_1 < \cdots < d_l\}$).

Disentanglement: (Adapted from additive decoders [37], Definition 3) Assume that we have a latent space $\mathcal{Z} \subset \mathbb{R}^K$, a partition of latent dimensions $\mathcal{P}$, a GT generative process defined by a decoder f and a learned decoder $\hat{f}$ that is optimized on the samples generated by f. We call $\hat{f}$ to be disentangled with respect to GT decoder f and partition $\mathcal{P}$ if there exist a permutation of latent dimensions π and a function $v : \mathbb{R}^K \to \mathbb{R}^K$ such that for all $z \in \mathcal{Z}$ in the latest space:

- v does not mix different blocks of partition $\mathcal{P}$. Formally, for all distinct blocks $B \neq B' \in \mathcal{P}$ and for all $i \in B, j \in B'$, the (i, j)-th element of the Jacobian of v at point z denoted as $(J_v(z))_{i,j}$ is zero.
- $f(z) = \hat{f}(\pi(v(z)))$, where $\pi(z_1, \ldots, z_K) = (z_{\pi(1)}, \ldots, z_{\pi(K)})$.

In other words, $\hat{f}$ should be identical to f up to blockwise modifications and a permutation of dimensions.

Local disentanglement: (Adapted from additive decoders [37], Definition 4) With the same assumptions as disentanglement definition, we call a function $\hat{f}$ to be locally disentangled with respect to GT decoder f and partition $\mathcal{P}$ if for all z in the latent space, there exists a neighborhood ϵ of z, permutation π, and a function v such that for every z' in ϵ, v does not mix different blocks of partition $\mathcal{P}$ and $f(z') = \hat{f}(\pi(v(z')))$.

Disentanglement is called dimension-wise when it is achieved with respect to the complete partition of the latent space dimensions ($\mathcal{P} = \{\{1\}, \{2\}, \ldots, \{K\}\}$). This is the most powerful and interpretable form of disentanglement, as it implies that the trained decoder should identify each individual latent dimension.

The Generative model

Assume that we have an observed count matrix $\boldsymbol{X} \in \mathbb{N}^{N \times M}$, where N is the number of cells and M is the number of genes. Optionally, we can consider a covariate vector $\boldsymbol{C} \in \mathcal{R}^{N \times R}$, where R is the dimensionality of the covariates for each cell. For example, in batch correction frameworks, $\boldsymbol{C}_i$ could be a one-hot vector representing the batch identity if modeled as in scVI [19], or an embedding vector if modeled as in scPoli [52] (DRVI accepts both conventions). Then, the goal is to discover latent representation $\boldsymbol{Z} \in \mathcal{R}^{N \times K}$ where K is the latent space dimension for the observed data. The

generative process is defined by the pooling of multiple decoders that map latent space blocks to parameters of the noise distribution. Specifically:

$$
\begin{aligned}
&Z \sim Normal(0, I) \\
&\log \mu = f(Z, C) = \sigma \left(\sum_{B \in \mathcal{P}} \psi \circ f^{(B)}(Z_B, C) \right) \\
&X_g \sim Dist(\log \mu_g + \log L, \log \theta_g)
\end{aligned}
\tag{1}
$$

Where f is the decoder, $\mathcal{P}$ is a partition of the latent space dimensions, $f^{(B)}$ denotes the additive decoder subunits defined per block (referred to as basis functions), σ and ψ form pooling function, $Dist$ models the noise distribution, θ_g is a parameter indicating gene-wise dispersion or variance, L corresponds to the random variable indicating library size of the cell, and Z_k denotes kth dimension of Z. DRVI users have an option to alternatively define θ based on batches or optimize it like μ using variational inference. To focus on the architectural contribution, we fixed θ based on genes, a commonly used and simpler approach. Likewise, we deviate from the generative modeling and set L to the observed values of the library size rather than incorporating it into the generative model, which is aligned with standard practices in the field.

DRVI accepts Normal, Poisson, and Negative Binomial distributions as noise models. When using the Normal noise model, the observed data (x) should be log-transformed. The transformations can be as simple as $log(1+x)$, although users are encouraged to apply other normalization techniques that maintain the validity of the normal noise assumption, such as additional library size normalization. For the Poisson noise model, the dispersion parameter is discarded. In Poisson and Negative Binomial noise models, all parameters are transformed using the exponential function. We have:

$$
Dist(\log \mu, \log \theta) = \begin{cases} Normal(\log \mu, \log \theta) & \text{if } Dist = \text{Normal} \\ Poisson(\exp(\log \mu)) & \text{if } Dist = \text{Poisson} \\ NB(\exp(\log \mu), \exp(\log \theta)) & \text{if } Dist = \text{Negative Bonimial} \end{cases}
\tag{2}
$$

We primarily parametrize the distributions in log space to ensure a correct and clear formulation within our framework. This approach also emphasizes that all other methods calculate their parameters in log space and transform them back to count space in the last step.

Fitting the generative process given observed data involves maximizing the log-likelihood function $\log \mathbb{P}_\theta(X|C) = \sum_n \log \mathbb{P}_\theta(\boldsymbol{X}_n|\boldsymbol{C}_n)$ where $\boldsymbol{X}$ is the matrix of observations, C indicates covariates matrix (such as technical batch), and θ represents all parameters in the model. Although the log-likelihood decomposes into sample log-likelihoods, maximizing each term requires integration over all Z values, which is intractable. We employ Variational Bayes, which constructs an approximate posterior probability distribution function $q_\phi(Z|X, C)$ parameterized by ϕ [100]. This distribution is typically formulated as a simple multivariate Gaussian distribution. The optimization objective, known as the evidence lower bound (ELBO) , is formulated as:

$$
\begin{aligned}
&\log \mathbb{P}_\theta(X|C) \\
&\quad = \sum_n \log \mathbb{P}_\theta(\boldsymbol{X}_n|\boldsymbol{C}_n) \\
&\quad \geq \sum_n \mathbb{E}_{Z \sim q_\phi(Z|\boldsymbol{X}_n, \boldsymbol{X}_n)} \Big[\log \mathbb{P}_\theta(\boldsymbol{X}_n|Z, \boldsymbol{C}_n) \Big] - D_{KL} \Big(q_\phi(Z|\boldsymbol{X}_n, \boldsymbol{C}_n) \;\Big|\Big|\; \mathbb{P}(Z) \Big) \\
&\quad = \sum_n -\mathcal{L}(\theta; \boldsymbol{X}_n, \boldsymbol{C}_n) = -ELBO.
\end{aligned}
\tag{3}
$$

Similar to other frameworks, the ELBO is optimized by constructing an autoencoder where $\mathbb{P}_\theta(X|Z, C)$, $q_\phi(Z|X, C)$, and $\mathbb{P}(Z)$ correspond to the decoder, the encoder, and the prior distribution, respec-

tively [19, 52, 101]. Stochastic gradient descent is used to optimize the negative ELBO, and the reparametrization trick is employed to sample from the approximate posterior q [100, 102].

Additive Decoders, Pooling, and Disentanglement

We build on the theoretical foundations of additive decoders [37] and interaction asymmetry [48] to establish disentanglement criteria within DRVI (Supplementary Fig. 34). Our primary methodological contribution is the introduction of an additional pooling function to the often used MLP subnetworks of the additive decoders (Supplementary Fig. 35). This modified architecture is novel, yet it remains within the framework of additive decoders. The nonlinearities within the pooling function promote higher-order interactions, crucial for establishing disentanglement. Adapting the original theorems from additive decoders [37] and interaction asymmetry [48] to our generative process, we explore two pooling function options, examining their theoretical interpretations and practical impact on disentanglement.

A pooling function defined by $\psi, \sigma : \mathbb{R} \to \mathbb{R}$ aggregates a set of real valued numbers $y_1, \ldots, y_T$ by $\sigma\left(\sum_{1 \leq t \leq T} \psi(y_t)\right)$. Pooling functions can also be applied element-wise to a set of vectors or tensors.

Pooled additive functions: (Based on additive decoders [37], Definition 1) Given a pooling function, a generative function f mapping latent space to the observational space is called to be pooled additive if there exist functions $f^{(B)}$ (for every $B \in \mathcal{P}$) such that for any z

$$f(z) = \sigma\left(\sum_{B \in \mathcal{P}} \psi \circ f^{(B)}(z_B)\right). \tag{4}$$

We call $f^{(B)}$ functions the basis of the pooled additive decoder. Clearly, DRVI's generative function defined in the previous section satisfies the criteria for pooled additiveness. Considering $\psi \circ f^{(B)}$ as basis functions and ignoring σ, every pooled additive function is indeed an additive function. We explicitly extract the shared function ψ to emphasize the mathematical significance of such an architectural decision. In particular, the interaction asymmetry framework proves disentanglement when a few abstract conditions, especially the first-order interaction asymmetry and first-order sufficient independence conditions, are met [48]. These conditions are not generally satisfied and depend on the model architecture and data. For single-cell omics data and $\psi = \exp$, we guarantee first-order interaction asymmetry in the general case (Theorem 1), prove first-order sufficient independence for the one-dimensional case (Theorem 3), and show it is encouraged (Theorem 2) in the general multi-dimensional case.

1st Order Interaction Asymmetry (Assumption 3.5 of [48]): for every $B \neq B' \in \mathcal{P}$ and $i \in B, i' \in B'$ we have $D^2_{i,i'} f(z) = 0$. In addition, for every $B \in \mathcal{P}$, every split of B into two non-empty sets I_1, I_2, and every invertible linear transformation T transforming Z_B and keeping the rest of the dimensions intact, there exist $i_1 \in I_1, i_2 \in I_2$ where $D^2_{i,i'} f(Tz) \neq 0$. As a special case, if $B = \{i\}$ is one-dimensional, $D^2_i f(z) \neq 0$. This intuitively means that interactions between splits are limited to first-order, but within splits, at least second-order interactions are necessary.

For the second criterion, we note that the σ term in pooled additive decoders can be ignored by a proper transformation of outputs (Remark 1 of additive decoders [37]). Using the chain rule and substituting the pooled additive decoder formula (4) into the 1st order sufficient independence criterion, we derive:

1st Order Sufficient Independence (Definition A.9 of [48]) for pooled additive decoders:

A pooled additive decoder meets sufficient independence if, for all z in the latent space:

$$\operatorname{rank}\left(\left[FOI(B)\middle|SOI(B)\right]_{B\in\mathcal{P}}\right) = \sum_{B\in\mathcal{P}} \operatorname{rank}\left(FOI(B)\right) + \operatorname{rank}\left(SOI(B)\right) \tag{5}$$

$$FOI(B) = \left[D_i f^{(B)}(z_B) \times \psi'\left(f^{(B)}(z_B)\right)\right]_{i\in B} \tag{6}$$

$$SOI(B) = \left[D^2_{i,i'} f^{(B)}(z_B) \times \psi'\left(f^{(B)}(z_B)\right) + D_i f^{(B)}(z_B) \times D_{i'} f^{(B)}(z_B) \times \psi''\left(f^{(B)}(z_B)\right)\right]_{i,i'\in B} \tag{7}$$

Where $FOI(B)$ and $SOI(B)$ are the matrices of first order and second order interactions, $D_i f^{(B)}(z_B)$ and $D^2_{i,i'} f^{(B)}(z_B)$ denote first-order and second-order derivatives of $f^{(B)}$ at point z_B, and the functions ψ' and ψ'' are the first-order and second-order derivatives of ψ respectively. Note that $D_i f^{(B)}(z_B)$ and $D^2_{i,i'} f^{(B)}(z_B)$ are $M \times 1$ matrices.

Here we discuss the requirements for achieving disentanglement in our model based on Theorem A.22 of the interaction asymmetry [48] **(I) 1st Order Interaction Asymmetry:** The pooled additive decoders obviously have at most 1st order interactions between splits (Definition 3.3 of [48]). Accordingly, having at least second-order interactions within each split is of interest, which we further discuss based on the different choices of the ψ in the pooling function. **(II) 1st Order Sufficient Independence**: This is the second main criterion for disentanglement. As we will show later, meeting this criterion also depends on the choice of the ψ function. **(III) Diffeomorphism:** The decoder is assumed to be a C^2-diffeomorphism (f and its inverse being C^2). We employ exponential linear units (ELU) [103] activation function in the neural networks defining f, offering a C^1 smoothness alternative to ReLU. However, f still does not satisfy the C^2-diffeomorphism requirement. **(IV) Reconstruction Loss:** Since we estimate the parameters of the noise distribution, solving the reconstruction problem cannot be guaranteed. Nevertheless, achieving an acceptable reconstruction loss is sufficient in practice. **(V) Domain Assumptions:** We can assume that our domain is regularly closed. While path connectivity is typically required to derive disentanglement from local disentanglement, this assumption often falls short in non-developmental data. However, sampling in the latent space promotes greater connectivity and thus encourages global disentanglement. Additionally, the KL term in the loss function helps prevent the coexistence of multiple irrelevant processes within a single dimension, assuming sufficient latent dimensions are available (Supplementary Fig. 36). **(VI) Number of processes:** In disentanglement theorems, it is implicitly assumed that the number of latent dimensions is the same as the number of GT processes. However, this is rarely known. Accordingly, we show that the model will saturate and extra dimensions will vanish when more latent dimensions are provided (see Results).

Having above criteria met, disentanglement of DRVI will be guaranteed based on Theorem A.22 of the interaction asymmetry [48]. Here, we discuss two main criteria (I) and (II) based on different choices of the pooling function when MLP networks are used as building blocks of the decoder.

Average Pooling: Consider the case where $\sigma(x) = x/K$ and ψ is identity function:

$$f(z) = \frac{1}{|\mathcal{P}|} \sum_{B\in\mathcal{P}} f^{(B)}(z_B). \tag{8}$$

Then $\psi'(x) = 1$ and $\psi''(x) = 0$. This will result in the pure definition of additive decoders, which requires additional assumptions on the decoder subnetworks for provable disentanglement [37]. Considering that shallow MLP layers are often used in the representation learning for single-cell omics, we argue that provable disentanglement based on the criteria in additive decoders and interaction asymmetry is impossible.

Specifically, when we model basis functions ($f^{(B)}$) with neural networks employing RELU activation functions, the combination of RELU functions with affine transformations forms piecewise linear functions that behave linearly in most parts of the domain. Consequently, the second-order

derivatives, required to imply second-order interactions within splits, are zero in most parts of the domain. Hence, proving disentanglement within this framework becomes impossible. This limitation is particularly relevant in single-cell omics, where the decoders are shallow fully connected networks with hundreds of hidden nodes. In the extreme case where the basis functions are linear, the decoder reduces to an affine transformation, making disentanglement impossible due to the well-known unidentifiability of linear ICA under independent Gaussian factors. Altogether, the currently used decoders are piecewise linear and have zero second-order derivatives in most parts of the domain. In the interaction asymmetry framework, this directly prevents the first order interaction asymmetry criterion from being met. Similarly, within the additive decoders framework, it's impossible to have full-rank columns in the sufficient nonlinearity matrix, which is the main criterion, because its right-hand side will be zero (more details on the sufficient nonlinearity matrix in [37]). We practically show that although additive decoders with average pooling exhibit slightly better disentanglement than non-additive counterparts, they still fall short compared to LSE pooled additive decoders (see Results).

LSE Pooling: Consider the case where $\sigma = \log$ and $\psi = \exp$:

$$f(z) = \log \sum_{B \in \mathcal{P}} \exp\left(f^{(B)}(z_B)\right). \tag{9}$$

Then, we have $\psi'(x) = \psi''(x) = \exp(x)$ for first-order and second-order derivates. Sufficient independence criterion in this case reduces to:

$$\operatorname{rank}\left(\left[FOI(B)\middle|SOI(B)\right]_{B\in\mathcal{P}}\right) = \sum_{B\in\mathcal{P}} \operatorname{rank}\left(FOI(B)\right) + \operatorname{rank}\left(SOI(B)\right) \tag{10}$$

$$FOI(B) = \left[D_i f^{(B)}(z_B) \times \exp(f^{(B)}(z_B))\right]_{i\in B} \tag{11}$$

$$SOI(B) = \left[\left(D^2_{i,i'} f^{(B)}(z_B) + D_i f^{(B)}(z_B) \times D_{i'} f^{(B)}(z_B)\right) \times \exp\left(f^{(B)}(z_B)\right)\right]_{i,i'\in B} \tag{12}$$

The appearance of $\exp(f^{(B)}(z_B))$ in this case is a favorable property that allows higher-order interactions within splits and facilitates proving sufficient independence.

Theorem 1. Assuming the decoder subnetworks are not logarithmic at any point, A log-sum-exp (LSE) pooled additive decoder with no vanished dimension and maximal splits of latent dimensions satisfies the 1st order interaction asymmetry criteria.

The proof is given in the Supplementary Note 4. Intuitively, the theorem guarantees second-order interactions if decoder subnetworks do not cancel the effect of $\psi = \exp$ in the LSE pooling. Following a logarithmic curve locally is very improbable and is typically not feasible in conventional shallow MLP neural networks, as they behave linearly in almost every neighborhood of their domains.

The sufficient independence condition is typically proven on very small toy examples [37] or optimized heuristically to some extent [48]. Instead, we take two parallel approaches, both assuming marker genes exists for underlying biological processes. We prove 1st order sufficient independence when the splits are one-dimensional. Besides, we show that for the general multi-dimensional case, making assumptions about the ground truth (GT) processes adequately narrows the gap between the two sides of the sufficient independence condition, and thus encourages disentanglement.

Theorem 2. Consider the ground-truth (GT) processes $p1, p2, \ldots, p_k$ underlying single-cell omics data. Assume that each of these processes has a distinct subset of involved genes. Then, for log-sum-exp (LSE) pooled additive decoders:

$$\operatorname{rank}\left(\left[FOI(B)\middle|SOI(B)\right]_{B\in\mathcal{P}}\right) = \sum_{B\in\mathcal{P}} \operatorname{rank}\left([FOI(B)|SOI(B)]\right) \tag{13}$$

This theorem shows that the gap between the two sides of the sufficient independence condition can be adequately narrowed down for single-cell omics data. The next theorem closes the gap when the splits are one-dimensional.

Theorem 3. Assume a generative process is modeled by a log-sum-exp (LSE) pooled additive decoder with one-dimensional splits. Consider the ground-truth (GT) processes $p1, p2, \ldots, p_k$ underlying single-cell omics data. Assume each of these processes has a distinct subset of at least two involved genes with different log-fold-changes. Then, the 1st order sufficient independence is guaranteed.

The proofs for both theorems are in Supplementary Note 4. Altogether, Theorems 1, 2, and 3 imply the main criteria for provable disentanglement outlined in interaction asymmetry framework [48].

Mechanistic interpretation

An important aspect of LSE pooling is its mechanistic interpretation. As described before, the decoder of DRVI (as in most other methods) models the mean parameters of the output distributions in the log-transformed space. Using average pooling means that predicted mean tensors from different factors are added up in log space, equivalent to the product of the tensors in count space, which makes no sense. Formally let $\log S_g^{(k)}$ be the output of kth decoder for gene g then for average pooling we have:

$$\begin{aligned} \log \mu_g &= \sum_k \log S_g^{(k)} \\ \Rightarrow \mu_g &= \exp\left(\sum_k \log S_g^{(k)}\right) = \prod_k S_g^{(k)}. \end{aligned} \tag{14}$$

The same phenomenon can be concluded in current matrix factorization and, to some extent, autoencoder models (Supplementary note 5). While this behavior causes no problem when active genes in different factors do not overlap, it makes no sense when a gene is regulated by multiple factors. On the other hand, using the LSE pooling, as described earlier, corresponds to the aggregation of predicted mean tensors by the LSE function. Since these parameters are defined in log space, their aggregation by LSE corresponds to an addition in the original count space. Formally let $\log S_g^{(k)}$ be the output of kth decoder for gene g, then for LSE pooling we have:

$$\begin{aligned} \log \mu_g &= \log \sum_k \exp\left(\log S_g^{(k)}\right) \\ \Rightarrow \mu_g &= \sum_k \exp\left(\log S_g^{(k)}\right) = \sum_k S_g^{(k)}. \end{aligned} \tag{15}$$

This means using the LSE pooling in the framework of additive decoders corresponds to the assumption that "activities induced by different factors (independent biological processes) sum up in the count space." This assumption aligns with the inherent additivity of transcribed genes and is consistent with additive technical effects such as doublets and background expression commonly observed in single-cell omics. (Supplementary Fig. 1 and Supplementary Note 2).

Modeling Details

We employ an MLP network to model the approximate posterior as an encoder. This encoder accepts log-transformed count data and outputs mean and variance vectors of a multivariate Gaussian distribution with a diagonal covariance matrix. Each layer of the MLP consists of an affine transformation (multiplication by a weight matrix followed by the addition of a bias vector), an activation function, and a normalization layer. Dropout is applied to input gene values and hidden layers. We

employ ELU as the default activation function and layer normalization (without elementwise affine transformation) as the default normalization layer. The final layer of the encoder is a simple affine transformation without any additional units.

We use the same settings to model each basis function $f^{(k)}$. Specifically, we define them as

$$f^{(k)}(z_k, c) = b_d^{(k)} + W_d^{(k)} \Gamma_{d-1}^{(k)} \left(\cdots \Gamma_2^{(k)} \left(b_2^{(k)} + W_2^{(k)} \Gamma_1^{(k)} \left(b_1^{(k)} + W_1^{(k)} [\nu^{(k)}(z_k); c]^T \right) \right) \right) \tag{16}$$

Where d is the depth of the decoder, $W_i^{(k)}, b_i^{(k)}$ indicates the ith affine transformation, $\Gamma_i^{(k)}$ consists of activation function, normalization, and dropout applied at each layer, c is the covariate vector, and $\nu^{(k)}(z_k)$ denotes the vectorization function for dimension k of the latent space.

In the DRVI framework, parameters between different basis functions can be shared in several ways: sharing nothing, the last affine transformation, or all affine transformations. We note when using layer normalization without elementwise affine transformation, the $Gamma_i^{(k)}$ functions do not have any state to share. In practice, even sharing all parameters works perfectly and is the default option.

For simplicity in computations, $\nu^{(k)}(z_k)$ outputs a vector with all elements zero but the kth value, which is z_k. This enables simultaneous matrix operations (multiplication with $W_1^{(1)}, \ldots,$ and, $W_1^{(K)}$) in a single tensor operation. The outputs of the basis functions are finally aggregated to form the parameters of the output distribution. Users can choose between average or LSE pooling. As discussed earlier, we utilize LSE pooling, implemented with the *logsumexp* function, as the default and correct aggregation method.

DRVI can handle categorical covariates with one-hot encoding and vector embedding strategies [19, 52]. For the sake of fair comparison with benchmarked unsupervised methods, we use one-hot encoding in all the analyses. However, we encourage users to take the second approach for studies with a high number of batches as it is shown to perform better [52]. It is worth mentioning that DRVI accepts multiple categorical covariates by concatenating their representations.

Choosing the appropriate noise model depends on the specific characteristics of the data. The Negative Binomial distribution is widely accepted for single-cell RNA sequencing (scRNA-seq) data and is extensively used in applications ranging from differential expression analysis to integration models [19, 51, 104–106]. Recent studies have demonstrated that the Poisson distribution may be a more suitable choice for single-cell ATAC sequencing (scATAC-seq) data [24]. Still, many methods such as scGEN and sysVI use normalized data with Normal distribution while performing well in their tasks [20, 88]. DRVI supports Negative Binomial, Poisson, and Normal noise distributions. For the latter user should provide normalized data rather than raw counts. We recommend using Negative Binomial for scRNA-seq and Poisson for modeling fragments in scATAC-seq data.

In current generative models, parameters are transformed from log space to count space, after which the Negative Binomial distribution's likelihood is computed based on these count-space parameters. To enhance numerical precision, we introduce a novel parametrization of the Negative Binomial likelihood that is defined based on $\log \mu$ instead of μ (see Supplementary Note 6 for mathematical details). Log-space parametrization of the Negative Binomial distribution is the default option to calculate the likelihood in DRVI, but users can still use the count-space parameterizations.

DRVI adjusts the final predicted mean parameter (after aggregation) to account for the library size, similar to scVI's default behavior of ensuring that the sum of mean parameters equals the library size. This default behavior can be disabled.

Finally, DRVI users can adjust the number of splits using a model parameter; we set this by default to equal the number of latent dimensions. It is important to note that while the DRVI framework theoretically and practically supports latent partitions of arbitrary size, our focus here has been on the full partition case where each latent factor is disentangled. The reason is that full partition disentanglement is the most powerful scenario in terms of disentanglement, and interestingly, it also aligns well with the characteristics of single-cell count data. The additive assumption in the full

partition case may seem restrictive. However, the Kolmogorov–Arnold representation theorem states that the sum of one-dimensional functions followed by a trainable pooling function can approximate any continuous multivariate function [107, 108]. This theorem doesn't directly apply to our model due to the employment of a fixed shared pooling function for all output variables in the decoder. Nevertheless, we mentioned this theorem to provide additional context for interested readers.

Interpretability

Learned latent representations may present known or unknown abstract concepts. Interpreting latent space dimensions in terms of the genes involved helps understand unknown abstract concepts and find explanations for known ones. In factorization models, the loadings (weights of the low-rank matrices) are interpreted as the effect of each gene on each factor. However, in nonlinear models, interpretability is challenging since the dependencies are nonlinear, and the effect of each latent space dimension also depends on the values of other latent dimensions. While the nonlinearity challenge still exists in DRVI, the additive architecture prevents interactions across latent dimensions, and allows interpretability as each latent space dimension is decoded separately until the final pooling function. Altogether, unlike autoencoders with MLP decoders, where the effect of one dimension is highly coupled and dependent on all other latent dimensions, DRVI latent factors are decoded separately, making them inherently individually interpretable. Specifically, for each latent factor with value z_i, the ith subnetwork decodes values right into the count space, and $f^{(i)}(z_i)$ is the effect of this dimension in count space.

To interpret the latent space dimensions in terms of the genes involved, we make use of the additive decoder architecture and explain each basis function separately. We start with the average pooling as the simpler case. For DRVI-AP, the generative process is defined as:

$$\log \mu = \frac{1}{K} \sum_{1 \leq k \leq K} f^{(k)}(z_k, c) \tag{17}$$

where μ, z_k, $f^{(k)}$, and c indicate the mean parameter of the generative distribution, kth latent space dimension, kth basis function in the additive decoder, and the covariate vector, respectively. In this case, for gene g and dimension i of the latent space we have:

$$\log \mu_g = \sum_{k \in K \wedge k \neq i} \left[\frac{1}{K} f^{(k)}(z_k, c) \right]_g + \left[\frac{1}{K} f^{(i)}(z_i, c) \right]_g \tag{18}$$

When perturbing z_i, the LFC of μ_g solely depends on $f^{(i)}(z_i, c)$. We take the mean over covariates to consider the average effect over different batches. Since $f^{(i)}$ is a nonlinear function, we are interested in the maximum effect of this latent dimension. Accordingly, we maximize the effect over different choices of z_i. In practice, we observe dimensions encoding two different concepts in their negative and positive directions. So, we split these two directions for better interpretability. Consequently, the effect of dimension Z_i on gene g is defined as:

$$\text{effect}^{\text{AP}}_{\text{lfc}}(Z_i^-, g) = \mathop{\mathbb{E}}_{c \in C} \left[\max_{z \in [\min(Z_i), 0]} \left[\frac{1}{K} f^{(i)}(z, c) \right]_g - \min_{z \in [\min(Z_i), \max(Z_i)]} \left[\frac{1}{K} f^{(i)}(z, c) \right]_g \right] \tag{19}$$

$$\text{effect}^{\text{AP}}_{\text{lfc}}(Z_i^+, g) = \mathop{\mathbb{E}}_{c \in C} \left[\max_{z \in [0, \max(Z_i)]} \left[\frac{1}{K} f^{(i)}(z, c) \right]_g - \min_{z \in [\min(Z_i), \max(Z_i)]} \left[\frac{1}{K} f^{(i)}(z, c) \right]_g \right] \tag{20}$$

These effects can be efficiently approximated by sampling from covariates and traversing (one-dimensional) discrete values over each latent space dimension.

For LSE pooling function, the generative process is:

$$\log \mu = \log \sum_{1 \leq k \leq K} \exp\left(f^{(k)}(z_k, c)\right) \tag{21}$$

where μ, z_k, $f^{(k)}$, and c are defined as in the average pooling case.

Accordingly, for each gene g and dimensions i of the latent space we have:

$$\mu_g = \sum_{k \in K \wedge k \neq i} \left[\exp\left(f^{(k)}(z_k, c)\right)\right]_g + \left[\exp\left(f^{(i)}(z_i, c)\right)\right]_g \tag{22}$$

Here, the effect of perturbing z_i in count space solely depends on $f^{(i)}(z_i, c)$. Formally:

$$\begin{aligned} \text{effect}^{\text{LSE}}_{\text{counts}}(Z_i, g) = \mathop{\mathbb{E}}_{c \in C} \Bigg[& \max_{z \in [\min(Z_i), \max(Z_i)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g - \\ & \min_{z \in [\min(Z_i), \max(Z_i)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g \Bigg] \end{aligned} \tag{23}$$

However, since we are usually interested in the LFC rather than the count space effect, we also have to consider the effect of other latent dimensions. The LFC will be more significant if other dimensions do not affect the target gene and will be smaller otherwise. Here, we make use of the additiveness property to write the minimum/maximum effect of a group of dimensions as the sum of the minimum/maximum effect of each individual dimension. Accordingly, to calculate the maximum possible effect, we have:

$$\text{effect}^{\text{LSE}}_{\text{max-lfc}}(Z_i^-, g) = \mathop{\mathbb{E}}_{c \in C} \left[\log \left(\frac{\Delta_{min} + \max_{z \in [\min(Z_i), 0]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g}{\Delta_{min} + \min_{z \in [\min(Z_i), \max(Z_i)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g + \epsilon} \right) \right] \tag{24}$$

$$\text{effect}^{\text{LSE}}_{\text{max-lfc}}(Z_i^+, g) = \mathop{\mathbb{E}}_{c \in C} \left[\log \left(\frac{\Delta_{min} + \max_{z \in [0, \max(Z_i)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g}{\Delta_{min} + \min_{z \in [\min(Z_i), \max(Z_i)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g + \epsilon} \right) \right] \tag{25}$$

Where Δ_{min} is defined as $\sum_{k \neq i} \min_{z \in [\min(Z_k), \max(Z_k)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g$ and ϵ is a small number defined for numerical stability.

Likewise, for the minimum effect, we define:

$$\text{effect}^{\text{LSE}}_{\text{min-lfc}}(Z_i^-, g) = \mathop{\mathbb{E}}_{c \in C} \left[\log \left(\frac{\Delta_{max} + \max_{z \in [\min(Z_i), 0]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g}{\Delta_{max} + \min_{z \in [\min(Z_i), \max(Z_i)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g + \epsilon} \right) \right] \tag{26}$$

$$\text{effect}^{\text{LSE}}_{\text{min-lfc}}(Z_i^+, g) = \mathop{\mathbb{E}}_{c \in C} \left[\log \left(\frac{\Delta_{max} + \max_{z \in [0, \max(Z_i)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g}{\Delta_{max} + \min_{z \in [\min(Z_i), \max(Z_i)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g + \epsilon} \right) \right] \tag{27}$$

Where Δ_{max} is defined as $\sum_{k \neq i} \max_{z \in [\min(Z_k), \max(Z_k)]} \left[\exp\left(f^{(i)}(z, c)\right)\right]_g$.

All the minimum and maximum values over decoder basis functions can be efficiently approximated by traversing discrete ranges over the latent space dimensions. Averaging over categorical covariates can be done by sampling. The same approach applies to DRVI with any other pooling function.

The minimum and maximum effects measure two different aspects of the latent dimensions.

The $\text{effect}_{\text{max-lfc}}$ measures the maximum effect of a latent dimension regardless of the baseline expression. For example, for two genes with similar average expression, changing a log-mean parameter of a gene from 0.1 to 0.5 has the same score as changing the other from 1 to 5. The $\text{effect}_{\text{min-lfc}}$ score is robust to such cases by taking the maximum expression of other dimensions into

account as a baseline. Still, it drops significantly if a gene participates in multiple latent dimensions. Consequently, we combine the two scores by multiplying them. Formally: $\text{effect}^{\text{LSE}}_{\text{combined}}(Z_i^*, g) = \text{effect}^{\text{LSE}}_{\text{max-lfc}}(Z_i^*, g) * \text{effect}^{\text{LSE}}_{\text{min-lfc}}(Z_i^*, g)$ where Z_i^* can be either Z_i^- or Z_i^+. Additionally, we remove all the insignificant effects meeting any of the following criteria:

$$\text{remove effect}^{\text{LSE}}_{\text{combined}}(Z_i^*, g) \text{ if any } \begin{cases} \text{effect}^{\text{LSE}}_{\text{max-lfc}}(Z_i^*, g) < 1 \\ \text{effect}^{\text{LSE}}_{\text{max-lfc}}(Z_i^*, g) < \max_{g' \in G} \left[\text{effect}^{\text{LSE}}_{\text{max-lfc}}(Z_i^*, g')\right] /2 \\ \text{effect}^{\text{LSE}}_{\text{min-lfc}}(Z_i^*, g) < \max_{g' \in G} \left[\text{effect}^{\text{LSE}}_{\text{min-lfc}}(Z_i^*, g')\right] /10 \end{cases} \tag{28}$$

We found the mentioned interpretability pipeline accurate enough for our analysis. However, further tuning of the pipeline and the combining logic can be a follow-up, which is out of the scope of the current study. A pseudocode for the DRVI pipeline, including training and interpretability, is provided in Supplementary Notes 7.

Since the introduced approach directly reflects the model's understanding of the generative process, it is limited to the set of genes included in the training data. Alternatively, one can perform differential expression (DE) analysis to identify the genes involved in a nonlinear gene program beyond the genes used for model training. To this end, cells can be categorized into high-valued and low-valued groups based on a latent dimension, or latent dimensions can be incorporated into the design matrix of a regression model for DE testing. While DE analysis is a well-established approach, it may highlight genes that are not specific to a process but are simply expressed at higher-than-average levels in the target population. In addition, lowly expressed genes may be missed.

To retrieve the most relevant gene sets (GSEA analysis), we used the Python interface of the gProfiler service [109]. For each dimension, we selected the top 100 genes associated with the nonlinear gene program as the input set. To account for the biological bias introduced by the selection of HVGs, they were given as the reference set to the gProfiler.

Ordering the learned factors

Unlike PCA, where principal components are ordered according to the explained variance, the latent dimensions learned by autoencoders and many other latent space models are inherently unordered. Nevertheless, we take advantage of the additive architecture and apply heuristics to sort the latent space dimensions based on their contribution to the reconstruction.

Heuristically, we remove each dimension and check how much it will affect the reconstruction. Assume we have an observed count matrix $\boldsymbol{X} \in \mathbb{N}^{N \times M}$, a trained DRVI model with latent space $\boldsymbol{Z}$, and the corresponding fitted generative process based on the decoder f. Then for kth latent dimensions we construct matrix $\boldsymbol{S}^{(k)} \in \mathbb{R}^{N \times M}$ where its ith row (corresponding to the sample i) is defined as:

$$\boldsymbol{S}_i^{(k)} = f^{(k)}(\boldsymbol{Z}_{i,k}, \boldsymbol{C}_i). \tag{29}$$

The reconstructed log-mean parameter in DRVI-AP will be $\sum \boldsymbol{S}^{(k)}$. Therefore, removing the ith dimension will change the log-space parameters by $|\boldsymbol{S}^{(k)}|$. So, for the average pooling case, we order latent dimensions based on the L1 norm of the $\boldsymbol{S}^{(k)}$ matrix.

For DRVI with LSE pooling, the effect of removing the ith dimension will be:

$$\log \sum_k \exp(\boldsymbol{S}^{(k)}) - \log \sum_{k \neq i} \exp(\boldsymbol{S}^{(k)}). \tag{30}$$

For stability of the effect, we add $\epsilon = 1$ to the logarithm function. Therefore, the overall impact of dimension i is defined as:

$$\log\left(\epsilon + \sum_k \exp(\boldsymbol{S}^{(k)})\right) - \log\left(\epsilon + \sum_{k\neq i} \exp(\boldsymbol{S}^{(k)})\right) = -\log\left(1 - \frac{\exp(\boldsymbol{S}^{(i)})}{\epsilon + \sum_k \exp(\boldsymbol{S}^{(k)})}\right), \tag{31}$$

where the latter can be efficiently computed by the softmax function.

Hyperparameters

We explored various choices of normalization functions, activation functions, noise models, parameter sharing between decoder subnetworks, and aggregation functions on the immune dataset. Based on our findings, we selected layer normalization (stateless), ELU activation function, parametrized negative binomial noise distribution (on RNA), full parameter sharing between decoder subnetworks, and exponential aggregation function. We then applied these settings uniformly across all other datasets, without extensive hyperparameter tuning. For the Daniocell dataset as an extreme case consisting of 156 cell identities, only the last layer of the decoder subnetworks is shared. The parameters of the deep learning models (DRVI, DRVI-AP, and scVI) were kept consistent where possible. We used the wandb platform to keep track of the experiments [110].

For smaller datasets, we used 32 latent dimensions with one hidden layer in both the encoder and decoder. For larger datasets, we employed 64 latent dimensions and deeper networks for the encoders and decoders. Due to its complexity, the CRISPR screen dataset required an even deeper architecture. Detailed information on the number of latent factors and the hidden layer architecture for autoencoder methods used in each study can be found in Supplementary Table 3.

Metrics

Since DRVI aims to bridge the gap between disentangled linear models and flexible non-disentangled methods, we evaluate its performance from two perspectives: disentanglement and integration. We assess the integration quality of multi-batch datasets based on the scIB metrics framework. Accordingly, the following metrics from the "scib-metrics" package were used:

- Biological conservation metrics
  - Isolated labels: average-width silhouette score (ASW) of isolated labels
  - K-Means NMI: normalized mutual information (NMI) between kmeans clusters and labels
  - K-Means ARI: adjusted rand-index (ARI) between kmeans clusters and labels
  - Silhouette Label: ASW of labels
  - cLISI: cell-type local inverse simpson index (LISI)
- Batch mixing metrics
  - Silhouette Batch: ASW of batches
  - iLISI: batch local inverse simpson index (i stands for integration)
  - KBET: K-nearest-neighbor batch-effect test (citation needed)
  - Graph Connectivity: connectivity of the subgraph per label
  - PCR Comparison: measures explained variance based on principal component regression

Details on these metrics can be found at [53]. As in the original study, the total score is derived by averaging biological conservation and batch integration scores with weights of 0.6 and 0.4, respectively.

Disentanglement refers to the existence of a one-to-one correspondence between the GT processes and the learned latent factors. Given a one-to-one correspondence, one can use an appropriate similarity function to measure how much each ground truth process is similar to the corresponding latent factor. Accordingly, we measure disentanglement by finding the one-to-one correspondence that maximizes the average pairwise similarities. This category of metrics is known as the Latent Matching Score (LMS). Since the number of GT processes is unknown to the models, the number of GT processes may differ from the number of latent dimensions. Therefore, we define the LMS metrics by finding the best non-overlapping matching between the GT processes and the latent space dimensions [37]:

$$\text{LMS-}F_{sim}(P, Z) = \frac{1}{|P|} \max_{\pi \in \Pi(P,Z)} \left[\sum_{p_l, z_k \in \pi} F_{sim}(p_l, z_k) \right], \tag{32}$$

where F_{sim} can be any similarity function comparing learned factors with respect to GT processes, P indicates the set of vectors corresponding to all ground-truth processes, Z is the set of latent dimensions, and Π is the set of all injective correspondences from P to Z.

In the following, we define three similarity functions based on Spearman correlation, Mutual Information, and Nearest Neighbor Consistency.

Scaled Mutual Information (SMI): We use Mutual Information as a nonlinear metric to compare GT processes and latent space dimensions. Since mutual information is bounded by the entropy of the GT process, we scale it so that the maximum value of $+1$ becomes achievable.

$$SMI(p_l, z_k) = \frac{\text{MI}(p_l, z_k)}{\text{H}(p_l)}, \tag{33}$$

where MI is the mutual information function and H is the entropy function. SMI is also known by other names such as proficiency or uncertainty coefficient [111].

Same-Process Neighbors (SPN): Consider a process with only two "on" and "off" states. Then, we would like to see the cells in the "on" state to form a group in a latent space. Accordingly, we measure the non-random probability for which a cell and its right neighbor are in the "on" state at the same time. So we have:

$$SPN(p, z_k) = \max\left(0, \frac{\mathbb{P}\big(p_j = 1 \big| p_i = 1 \wedge i, j \in \text{NG}(z_k)\big) - \mathbb{P}_{p_i \in p}\big(p_i = 1\big)}{1 - \mathbb{P}_{p_i \in p}\big(p_i = 1\big)}\right), \tag{34}$$

where $NG(z_k)$ indicates the set of all neighboring cells (no cell in between, not necessarily nearest neighbors) in dimension k of the latent space. The function $SPN(p_l, z_k)$ measures the frequency of two neighboring cells being active simultaneously minus the probability of such an outcome when cells are shuffled randomly. This metric is also adjusted to reach $+1$ as its maximum. $LSM - SPN$ can be calculated efficiently in $O(|P|Kn\log(n))$ by sorting the cells with respect to each derived latent space dimension. Conceptually, the frequency of two neighboring cells being active simultaneously can be interpreted as an approximation to the precision of a 1NN classifier predicting the activation of a GT process from a given latent space dimension.

Absolute Spearman Correlations (ASC): This metric is defined based on the absolute Spearman correlation between a GT process and a latent space dimension as

$$ASC(p_l, z_k) = \big|Corr_s(p_l, z_k)\big|, \tag{35}$$

where $Corr_s$ indicates the Spearman correlation function. We note that the order of values in the Spearman correlation function will contain many repeated values when the GT process is a discrete biological process or a continuous process with many inactive cells. In such cases, the maximum achievable Spearman correlation with respect to the continuous values in the latent space will be substantially small, and the other two metrics will represent much more meaningful values.

We use $LMS - SMI$ as the default metric to evaluate the disentanglement of a trained model with respect to known biological processes. While the SMI and ASC metrics accept both discrete and continuous processes, the SPN metric is designed specifically for on/off processes.

In addition to the LMS metrics, we use two other secondary metric families to better understand the model's behavior. To determine if a GT process is represented in at least one factor, we find the most similar dimension in the derived latent space for each GT process and then average these similarities across all GT processes. This is called the Most Similar Averaging Score (MSAS). Unlike LMS, MSAS does not penalize the model for presenting multiple irrelevant biological signals within a single dimension or representing a signal in multiple dimensions. Consequently, MSAS serves as an upper bound to LMS. We use the Most Similar Gap Score (MSGS), which builds on MSAS by adding a term that penalizes the presentation of a GT process in multiple dimensions. Specifically, MSGS calculates the average difference between the most similar dimension and the second most similar dimension for all GT processes, yet this score still lacks penalizing the presence of multiple irrelevant biological signals within a single dimension. When using the SMI similarity function, MSGS reduces to the Mutual Information Gap (MIG) metric as defined in Ref [66]. Formally, we have:

$$\text{MSAS-}F_{sim}(P, Z) = \frac{1}{|P|} \sum_{p_l \in P} \max_{z_k \in Z} F_{sim}(p_l, z_k) \tag{36}$$

$$\text{MSGS-}F_{sim}(P, Z) = \frac{1}{|P|} \sum_{p_l \in P} \max_{z_k \in Z} \left[F_{sim}(p_l, z_k) - \max_{z_{k'} \neq z_k \in Z} F_{sim}(p_l, z_k) \right], \tag{37}$$

where F_{sim} can be any of the SMI, SPN, or ASC functions, and P and Z are defined as before.

We primarily use the LMS metric family for evaluation and examine the MSAS and MSGS metric families to further evaluate model behavior. This choice is because MSAS and MSGS metrics do not penalize the presence of multiple irrelevant biological signals within a single dimension. In addition, the MSGS metric family assumes statistical independence between latent variables. However, in biological systems, underlying GT processes often exhibit causal relationships or interdependencies (e.g., cell cycle is not independent of cell type, as not all cell types proliferate). Finally, when using coarse-grained proxy variables for evaluation, the MSGS metric family can heavily penalize models that successfully disentangle finer, biologically meaningful subtypes.

We may know a subset of GT processes in a few developmental datasets. However, GT processes are usually unknown. Therefore, we use perturbations in perturbational data and cell types in data with known cell types as proxies for GT biological processes. Each cell type undergoes specific biological processes, so we expect a disentangled model to represent each cell type in at least one distinct dimension. Similarly, since perturbations activate specific biological processes, a disentangled model should be able to identify such processes and, to some extent, such perturbations in separate dimensions. We emphasize that while there is no one-to-one relationship between biological processes and these proxy variables (cell types and perturbations), we use them in the absence of a comprehensive list of biological processes to measure the effectiveness of the models in terms of disentanglement. This imperfection in the choice of the proxy variables results in limitations such as variability in the range of scores across datasets.

There can be a significant variation in the range of disentanglement scores across different datasets. This variation is influenced by both the complexity of the datasets and the choice of the proxy variable. The disentanglement task becomes more challenging as the number of underlying processes increases. For example, identifying 100 cell types is much harder than identifying 10 cell types. A more significant factor affecting both the range of observed metrics and the maximum achievable values is the use of imperfect proxy variables. For example, perturbations are often redundant, combinatorial, and partially effective. However, metrics expect the models to push all cells affected by each perturbation to an individual axis, which is impossible. The same happens when there are combinatorial effects involving cell types. Another category of imperfect proxy variables is coarse

cell-type annotations. When the cell-type annotation is coarser than the true biological variations, a well-disentangled representation identifies subtypes in different axes. However, disentanglement metrics expect to see all cells in the course grouping in one axis, leading to undesirably low metric values for better models. These are the reasons that the metric range varies between datasets. Although not perfect, using available supervised data, such as annotations and perturbations, is currently the best way to measure disentanglement.

In addition to disentanglement metrics, we used Rank Biased Overlap (RBO) similarity score to demonstrate the consistency of interpreted signatures across datasets. For two sorted sets S and T, RBO is defined as:

$$\mathrm{RBO}(S,T) = (1-p)\sum_{d=1}^{\infty}\frac{\mathrm{overlap}(T_{1\ldots d}, S_{1\ldots d})}{d}p^{d-1} \tag{38}$$

Where $p \in (0,1)$ is parameter controlling the importance of top elements and $\mathrm{overlap}(T_{1\ldots d}, S_{1\ldots d})$ is the number of shared items between top d elements of S and T. We use $p = 0.9$ by default.

Baseline models

All baseline models are trained with the same number of latent dimensions as used for DRVI. For baselines not accepting count data, we used the log-transformed normalized gene expression provided by the dataset when available. Otherwise, we normalized the total counts per cell and log-transformed the data using scanpy [112, 113].

- **PCA** from scikit-learn [114], was applied to the log-transformed and normalized data. Prior to the transformation, the data was centered without scaling.
- For **ICA**, we applied the FastICA [115] from scikit-learn [114]. Since the number of samples exceeded the number of features across all datasets, we chose the whiten_solver="eigh" option. The data was centered before applying ICA, without any scaling.
- **scVI** is trained with the same number of epochs and hidden layers as DRVI (Supplementary Table 3). To allow for learning a more structured latent space and fair comparison, early stopping is not used. For **scVI-PCA** and **scVI-ICA**, we applied PCA and ICA to the latent spaces learned from the scVI model. PCA from the scikit-learn package with default parameters and FastICA from the scikit-learn package [114] with whiten_solver="eigh" are used.
- For **β-TCVAE** and **MICHIGAN**, we used the implementation provided by the MICHIGAN package [47]. Since default parameters yielded suboptimal performance, we optimized the key scalar hyperparameters (TotalCorrelation_lamb, GradientPenaly_lambda, MutualInformation_lamb) by evaluating values in the range $\{0.1, 1, 10, 100\}$. Consequently, we adopted the default settings with GradientPenaly_lambda=100, which we subsequently confirmed to perform better than the default parameters in almost all benchmarked scenarios.
- We performed **MOFA** on the log-transformed normalized data using the Python implementation in the mofapy2 library [38]. To induce sparsity in the factor loadings and enhance the interpretability of the latent factors, we enabled the spikeslab_weights option. Additionally, the convergence_mode was set to "fast".
- **scETM** [116] was trained on raw count data for the same number of epochs as scVI and DRVI (Supplementary Table 3), with batch information provided and default hyperparameters. The resulting topic mixture (θ) was used to define the latent space.
- For **LIGER**, we utilized the Python implementation in the pyLIGER package [117]. The model was trained on the count matrices, and the resulting normalized cell latent factors (H_norm) were used to define the latent space.

Ablation Study

To further investigate the impact of modeling choices, we trained several DRVI variants through ablation studies. Details of these models are provided below:

- **DRVI-AP:** We trained this variant with average pooling instead of LSE pooling to highlight the practical importance of LSE pooling.
- **DRVI-APnoEXP:** Similar to DRVI-AP, this variant averages the decoder outputs to generate predictions. However, unlike scVI, DRVI, and DRVI-AP, which apply the softmax function (exp followed by $l1$ normalization) at the end – a step we hypothesize transfers parameters from log space to count space – we used the smooth softplus function to ensure the positivity of the negative binomial's mean parameter. This variant was trained to demonstrate that the limited effectiveness of average pooling is not due to the assumption that decoder outputs reside in log space.
- **DRVI-noShare:** The default configuration shares weights across all but the first layer of the decoder subnetworks. In this variant, we trained the model without any weight sharing between these subnetworks.
- **DRVI-2D:** This variant employs two-dimensional splits of the latent space. While less powerful than the default DRVI, it highlights DRVI's applicability in multidimensional split scenarios.
- **CVAE:** This variant shares the architectural settings of scVI but uses smooth ELU activations and the log-space parameterized negative binomial loss from DRVI (defined in Supplementary Note 6). Consequently, we can view this as a standard conditional variational autoencoder (CVAE) implemented within the DRVI framework, employing MLP decoders instead of DRVI's additive decoders. We trained this variant to enable a fair comparison of reconstruction performance.

The ablation results indicate that the nonlinear shared function prior to summation, implemented by $\psi = \exp$ in the LSE pooling, is the main critical point for achieving disentanglement in our settings. Furthermore, sharing weights within decoder subnetworks slightly enhances disentanglement capabilities (Supplementary Figs. 37 and 38).

Datasets

In the following, we provide information on the simulated data and datasets used in this work (summarized in Supplementary Table 2. The pre-processing steps for each dataset are described. Additionally, the code to reproduce simulated data as well as the pre-processing steps are available at https://github.com/theislab/drvi_reproducibility.

Simulated Data

We simulated two single-cell gene expression datasets to evaluate representation learning methods under controlled conditions. Each dataset consists of $G = 1000$ genes and $N = 6000$ cells, each assigned to either a specific condition or a control. For each condition $c \in \{1, \ldots, C\}$, we sample a binary vector $\mathbf{y}_c \in \{0, 1\}^K$, where $K = 20$ is the number of latent biological processes and $y_{c,k} = 1$ indicates that process k is active in condition c. Each cell under condition c inherits the same binary process activity vector. Latent representations $\mathbf{z}_i \in \mathbb{R}^K$ are drawn independently for each cell i, where $z_{i,k} \sim \text{Beta}(10, 2)$ if $y_{c,k} = 1$, and $z_{i,k} = 0$ otherwise. Control cells are assigned zero latent activity ($\mathbf{z}_i = \mathbf{0}$). We ensure that all processes are represented across the dataset and control the degree of overlap across conditions by varying $D = \max \|\mathbf{y}_c\|_0$, the maximum number of active processes per condition. In one dataset, each condition is linked to a single active biological

process ($D = 1$). In the other, four distinct processes ($D = 4$) may simultaneously be activated per condition, creating controlled overlap and compositional complexity.

Gene expression values are generated additively by mapping each active process to a sparse (and potentially overlapping) subset of genes using nonlinear monotonic cubic spline functions [118]. Specifically, the expression of each relevant target gene is computed individually using a smooth monotonic transformation of the latent dimension, simulating process-specific gene regulation. To account for biological variability consistent with the negative binomial characteristics of single-cell data, we model each process-specific contribution with a Gamma distribution, $\text{Gamma}(\alpha = s, \theta = \mu/s)$, where μ is the derived process-wise mean and s is a shape parameter per gene sampled from $(1, 5)$. In addition to ground-truth-related variation, we introduce background noise by injecting gene-wise Gamma-distributed noise with shape and scale parameters randomly sampled from $(0.05, 0.1)$ and $(0.1, 10)$, respectively. The final expression profile for each cell is obtained by summing the noise vector and process contributions, followed by Poisson sampling. The code to generate simulated data is available in the reproducibility repository.

Immune Dataset

The immune dataset consists of 32,484 cells collected from 4 human PBMC studies, including 9 batches and 16 distinct cell types after pre-processing. The data is obtained from Luecken et al. (2022) [53, 119]. The "Villani" study was excluded due to its non-integer values. The dataset was subset to 2,000 highly-variable genes (HVGs) in a batch-aware manner [53].

The HLCA

The HLCA consists of a core dataset and an extended version [7]. We obtained the curated core part from cellxgene (https://cellxgene.cziscience.com/collections/6f6d381a-7701-4781-935c-db10d30de293), which includes 584,944 cells across 166 samples from 14 datasets. This dataset contains annotations at different levels. We used samples as the batch covariate and the finest-level annotations comprising 61 cell types to benchmark the methods. The dataset was subset to 1,996 HVGs originally used to construct the HLCA reference model available on Zenodo (https://doi.org/10.5281/zenodo.7599104).

CRISPR Screen Dataset

The CRISPR screen dataset, also known as the Norman perturb-seq dataset, comprises 104,339 single-cells of K562 cell line (including control cells) perturbed by 106 single-gene perturbations and 131 combinatorial perturbations [5]. The 238 unique perturbation signatures were used to benchmark methods. We downloaded the pre-processed data using the pertpy package [120]. The data was limited to the 5,000 HVGs originally provided in the pre-processed data.

PBMC Dataset

The PBMC dataset comprises 647,366 cells across 143 samples from 130 healthy and COVID-19 patients [64]. The data was collected at three sites, and we used the site as the batch covariate to account for the highest technical variation. We used the finest-level annotation containing 51 unique cell types for benchmarking the methods. The data was obtained from https://www.covid19cellatlas.org/ and was subset to 2,000 HVGs in a batch-aware manner.

Retina Organoid Dataset

The retina organoid data consists of 19,768 cells from fovea, 34,723 cells from periphery, and 43,857 retina organoid cells (total 98,348 cells) [10]. The cells come from 41 samples and are annotated into 21 unique cell types. Considering that here, we have two very different biological systems, we take the intersection between 3000 HVGs of the two systems, which results in 2001 HVGs shared between primary and organoid data [88]. The data is obtained from https://cellxgene.cziscience.com/collections/2f4c738f-e2f3-4553-9db2-0582a38ea4dc.

Developmental Pancreas Dataset

The developmental pancreas dataset consists of 3,696 pancreatic mouse cells during endocrinogenesis at embryonic day 15.5 [4]. The data was obtained from the scVelo package [81]. We used scVelo to pre-process and subset the data to the top 2,000 HVGs. The CellRank package provides a subset of this data with finer annotations [80, 121]. We transferred these finer annotations to the complete dataset and used the finest annotation comprising 15 unique cell states for benchmarking.

Daniocell Dataset

The Daniocell dataset is a comprehensive transcriptional atlas of early zebrafish development, comprising 489,686 cells from 62 stages during zebrafish embryogenesis [65]. The data covers 20 tissues and 156 identity clusters, which we used to benchmark the data. We obtained the h5ad object from https://zenodo.org/records/8133569 [122]. The pre-processing of this data involved marking data with the annotations provided in the original publication, removing author-identified doublets, and subsetting data to 2,000 HVGs.

NeurIPS 2021 Dataset

The NeurIPS 2021 Dataset contains nearly 120K single-cell paired RNA-ATAC and RNA-ADT (antibody-derived tags) measurements from the human bone marrow of 10 donors [91]. We used the ATAC modality of this dataset consisting of 69,247 cells from 13 batches covering 22 cell types. The dataset is retrieved from GSE194122 and subset to 14,865 highly variable peaks.

Blood Dataset

Blood data consists of 336k cells collected from 4 human PBMC studies, including 65 donors and 27 cell types in its expert-curated annotation [89]. The data is obtained from https://cellxgene.cziscience.com/e/d86edd6a-4b5d-437a-ad80-1fd976a5e23a.cxg/. The dataset was subset to 2,000 highly-variable genes (HVGs) in a batch-aware manner [53].

Runtime Analysis

We benchmarked DRVI against scVI (for scRNA-seq) and peakVI (for scATAC-seq) as widely-used efficient models for dataset integration. We recorded the total runtime and maximum GPU memory allocation. We note that DRVI's runtime scales linearly with the number of cells. Accordingly, we plotted the runtime divided by the number of cells and the number of epochs (runtime per cell per epoch). Our results indicate that despite DRVI's higher computational cost in terms of floating-point operations, its parallel and efficient implementation maintains runtime on the same order as scVI or peakVI up to a small scaling factor (Supplementary Fig. 39). Furthermore, for each dataset, the runtime for both the interpretability pipeline (gene scores per latent dimension)

and the latent dimension sorting algorithm was measured, with a worst-case execution time of under 80 seconds (Supplementary Fig. 40). The variability between runs arises from the fact that sorting latent dimensions requires a full iteration of model evaluation over all cells, whereas the interpretability runtime depends solely on the number of latent dimensions, the number of features, and the complexity of the decoder. It is worth mentioning that DRVI is based on the scvi-tools framework and requires the same amount of CPU memory for training. The runtime experiments were performed on a GPU server equipped with two Intel(R) Xeon(R) Platinum 8480+ processors (totaling 112 cores and 224 threads at 2.0 GHz), 2 TB of memory, and eight NVIDIA H100 (80GB) GPUs. Each program execution was constrained to **a single GPU** and a maximum of **20 CPU threads**.

References

[1] Svensson, V., Vento-Tormo, R. & Teichmann, S. A. Exponential scaling of single-cell RNA-seq in the past decade. Nat. Protoc. **13**, 599–604 (2018).

[2] Angerer, P. et al. Single cells make big data: New challenges and opportunities in transcriptomics. Curr. Opin. Syst. Biol. **4**, 85–91 (2017).

[3] Farrell, J. A. et al. Single-cell reconstruction of developmental trajectories during zebrafish embryogenesis. Science **360** (2018).

[4] Bastidas-Ponce, A. et al. Comprehensive single cell mRNA profiling reveals a detailed roadmap for pancreatic endocrinogenesis. Development **146** (2019).

[5] Norman, T. M. et al. Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. Science **365**, 786–793 (2019).

[6] Srivatsan, S. R. et al. Massively multiplex chemical transcriptomics at single-cell resolution. Science **367**, 45–51 (2020).

[7] Sikkema, L. et al. An integrated cell atlas of the lung in health and disease. Nat. Med. **29**, 1563–1577 (2023).

[8] Rood, J. E., Maartens, A., Hupalowska, A., Teichmann, S. A. & Regev, A. Impact of the human cell atlas on medicine. Nat. Med. **28**, 2486–2496 (2022).

[9] Kanton, S. et al. Organoid single-cell genomic atlas uncovers human-specific features of brain development. Nature **574**, 418–422 (2019).

[10] Cowan, C. S. et al. Cell types of the human retina and its organoids at Single-Cell resolution. Cell **182**, 1623–1640.e34 (2020).

[11] Buenrostro, J. D. et al. Single-cell chromatin accessibility reveals principles of regulatory variation. Nature **523**, 486–490 (2015).

[12] Zhang, K. et al. A single-cell atlas of chromatin accessibility in the human genome. Cell **184**, 5985–6001.e19 (2021).

[13] Thornton, C. A. et al. Spatially mapped single-cell chromatin accessibility. Nat. Commun. **12**, 1274 (2021).

[14] Moffitt, J. R., Lundberg, E. & Heyn, H. The emerging landscape of spatial profiling technologies. Nat. Rev. Genet. **23**, 741–759 (2022).

[15] Fawkner-Corbett, D. et al. Spatiotemporal analysis of human intestinal development at single-cell resolution. Cell **184**, 810–826.e23 (2021).

[16] Peterson, V. M. et al. Multiplexed quantification of proteins and transcripts in single cells. Nat. Biotechnol. **35**, 936–939 (2017).

[17] Stoeckius, M. et al. Simultaneous epitope and transcriptome measurement in single cells. Nat. Methods **14**, 865–868 (2017).

[18] Zappia, L. & Theis, F. J. Over 1000 tools reveal trends in the single-cell RNA-seq analysis landscape. Genome Biol. **22**, 301 (2021).

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

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

[21] Lance, C. et al. Multimodal single cell data integration challenge: Results and lessons learned. In NeurIPS 2021 Competitions and Demonstrations Track, 162–176 (PMLR, 2022).

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

[23] Gayoso, A. et al. Joint probabilistic modeling of single-cell multi-omic data with totalVI. Nat. Methods **18**, 272–282 (2021).

[24] Martens, L. D., Fischer, D. S., Yépez, V. A., Theis, F. J. & Gagneur, J. Modeling fragment counts improves single-cell ATAC-seq analysis. Nat. Methods **21**, 28–31 (2024).

[25] Cao, Z.-J. & Gao, G. Multi-omics single-cell data integration and regulatory inference with graph-linked embedding. Nat. Biotechnol. **40**, 1458–1466 (2022).

[26] Kunes, R. Z., Walle, T., Land, M., Nawy, T. & Pe'er, D. Supervised discovery of interpretable gene programs from single-cell data. Nat. Biotechnol. (2023).

[27] Svensson, V., Gayoso, A., Yosef, N. & Pachter, L. Interpretable factor models of single-cell RNA-seq via variational autoencoders. Bioinformatics **36**, 3418–3421 (2020).

[28] DeTomaso, D. & Yosef, N. Hotspot identifies informative gene modules across modalities of single-cell genomics. Cell Syst. **12**, 446–456.e9 (2021).

[29] McInnes, L., Healy, J. & Melville, J. UMAP: Uniform manifold approximation and projection for dimension reduction. arXiv [stat.ML] (2018).

[30] Becht, E. et al. Dimensionality reduction for visualizing single-cell data using UMAP. Nat. Biotechnol. **37**, 38–44 (2018).

[31] Traag, V. A., Waltman, L. & van Eck, N. J. From louvain to leiden: guaranteeing well-connected communities. Sci. Rep. **9**, 5233 (2019).

[32] Kiselev, V. Y., Andrews, T. S. & Hemberg, M. Challenges in unsupervised clustering of single-cell RNA-seq data. Nat. Rev. Genet. **20**, 273–282 (2019).

[33] Wang, X., Chen, H., Tang, S., Wu, Z. & Zhu, W. Disentangled representation learning. IEEE Trans. Pattern Anal. Mach. Intell. **46**, 9677–9696 (2024).

[34] Locatello, F. et al. Challenging common assumptions in the unsupervised learning of disentangled representations. ICML **97**, 4114–4124 (2018).

[35] Ma, J., Zhou, C., Cui, P., Yang, H. & Zhu, W. Learning disentangled representations for recommendation. Advances in Neural Information Processing Systems **32** (2019).

[36] Lachapelle, S. et al. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ICA. arXiv [stat.ML] (2021).

[37] Lachapelle, S., Mahajan, D., Mitliagkas, I. & Lacoste-Julien, S. Additive decoders for latent variables identification and cartesian-product extrapolation (2023).

[38] Argelaguet, R. et al. MOFA+: a statistical framework for comprehensive integration of multi-modal single-cell data. Genome Biol. **21**, 111 (2020).

[39] Hackenberg, M., Brunn, N., Vogel, T. & Binder, H. Infusing structural assumptions into dimension reduction for single-cell RNA sequencing data to identify small gene sets (2024).

[40] Lopez, R. et al. Learning causal representations of single cells via sparse mechanism shift modeling (2022).

[41] Bereket, M. & Karaletsos, T. Modelling cellular perturbations with the sparse additive mechanism shift variational autoencoder (2023).

[42] Hediyeh-zadeh, S., Fischer, T. & Theis, F. J. Disentanglement via mechanism sparsity by replaying realizations of the past (2024).

[43] Lotfollahi, M. et al. Biologically informed deep learning to query gene programs in single-cell atlases. Nat. Cell Biol. **25**, 337–350 (2023).

[44] Gut, G., Stark, S. G., Rätsch, G. & Davidson, N. R. pmVAE: Learning interpretable Single-Cell representations with pathway modules (2021).

[45] Nazaret, A. et al. Deep generative model deciphers derailed trajectories in acute myeloid leukemia (2023).

[46] Lotfollahi, M., Dony, L., Agarwala, H. & Theis, F. Out-of-distribution prediction with disentangled representations for single-cell RNA sequencing data. bioRxiv 2021.09.01.458535 (2021).

[47] Yu, H. & Welch, J. D. MichiGAN: sampling from disentangled representations of single-cell data using generative adversarial networks. Genome Biol. **22**, 158 (2021).

[48] Brady, J. et al. Interaction asymmetry: A general principle for learning composable abstractions. arXiv [cs.LG] (2024).

[49] Khemakhem, I., Kingma, D., Monti, R. & Hyvarinen, A. Variational autoencoders and nonlinear ICA: A unifying framework. In International Conference on Artificial Intelligence and Statistics, 2207–2217 (PMLR, 2020).

[50] Lachapelle, S. et al. Synergies between disentanglement and sparsity: Generalization and identifiability in multi-task learning. ICML **202**, 18171–18206 (2022).

[51] Eraslan, G., Simon, L. M., Mircea, M., Mueller, N. S. & Theis, F. J. Single-cell RNA-seq denoising using a deep count autoencoder. Nat. Commun. **10**, 390 (2019).

[52] De Donno, C. et al. Population-level integration of single-cell datasets enables multi-scale analysis across samples. Nat. Methods **20**, 1683–1692 (2023).

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

[54] Worah, K. et al. Proteomics of human dendritic cell subsets reveals Subset-Specific surface markers and differential inflammasome function. Cell Rep. **16**, 2953–2966 (2016).

[55] Roussel, X., Garnache Ottou, F. & Renosi, F. Plasmacytoid dendritic cells, a novel target in myeloid neoplasms. Cancers **14** (2022).

[56] Zheng, Y. & Rudensky, A. Y. Foxp3 in control of the regulatory T cell lineage. Nat. Immunol. **8**, 457–462 (2007).

[57] Kolar, P. et al. CTLA-4 (CD152) controls homeostasis and suppressive capacity of regulatory T cells in mice. Arthritis Rheum. **60**, 123–132 (2009).

[58] van den Brink, S. C. et al. Single-cell sequencing reveals dissociation-induced gene expression in tissue subpopulations. Nat. Methods **14**, 935–936 (2017).

[59] Denisenko, E. et al. Systematic assessment of tissue dissociation and storage biases in single-cell and single-nucleus RNA-seq workflows. Genome Biol. **21**, 130 (2020).

[60] Adam, M., Potter, A. S. & Potter, S. S. Psychrophilic proteases dramatically reduce single-cell RNA-seq artifacts: a molecular atlas of kidney development. Development **144**, 3625–3632 (2017).

[61] Gautier, E.-F. et al. Comprehensive proteomic analysis of human erythropoiesis. Cell Rep. **16**, 1470–1484 (2016).

[62] Guo, X., Zhao, Y., Kim, J. & Dean, A. Hemogen/BRG1 cooperativity modulates promoter and enhancer activation during erythropoiesis. Blood **139**, 3532–3545 (2022).

[63] Dean, L. Blood Groups and Red Cell Antigens (National Center for Biotechnology Information (US), 2005).

[64] Stephenson, E. et al. Single-cell multi-omics analysis of the immune response in COVID-19. Nat. Med. **27**, 904–916 (2021).

[65] Sur, A. et al. Single-cell analysis of shared signatures and transcriptional diversity during zebrafish development. Dev. Cell **58**, 3028–3047.e12 (2023).

[66] Chen, T., Li, X., Grosse, R. & Duvenaud, D. Isolating sources of disentanglement in variational autoencoders. Adv. Neural Inf. Process. Syst. **abs/1802.04942** (2018).

[67] Li, X. et al. ScRNA-seq expression of IFI27 and APOC2 identifies four alveolar macrophage superclusters in healthy BALF. Life Sci Alliance **5** (2022).

[68] Aegerter, H., Lambrecht, B. N. & Jakubzick, C. V. Biology of lung macrophages in health and disease. Immunity **55**, 1564–1580 (2022).

[69] Bailey, J. I. et al. Expansion of profibrotic monocyte-derived alveolar macrophages in patients with persistent respiratory symptoms and radiographic abnormalities after COVID-19 (2023).

[70] Tokunaga, R. et al. CXCL9, CXCL10, CXCL11/CXCR3 axis for immune activation - a target for novel cancer therapy. Cancer Treat. Rev. **63**, 40–47 (2018).

[71] Kameda, M. et al. CXCL9, CXCL10, and CXCL11; biomarkers of pulmonary inflammation associated with autoimmunity in patients with collagen vascular diseases-associated interstitial lung disease and interstitial pneumonia with autoimmune features. PLoS One **15**, e0241719 (2020).

[72] Callahan, V. et al. The Pro-Inflammatory chemokines CXCL9, CXCL10 and CXCL11 are up-regulated following SARS-CoV-2 infection in an AKT-Dependent manner. Viruses **13** (2021).

[73] Korbecki, J. et al. Fractalkine/CX3CL1 in neoplastic processes. Int. J. Mol. Sci. **21**, 3723 (2020).

[74] Azimi, I., Petersen, R. M., Thompson, E. W., Roberts-Thomson, S. J. & Monteith, G. R. Hypoxia-induced reactive oxygen species mediate N-cadherin and SERPINE1 expression, EGFR signalling and motility in MDA-MB-468 breast cancer cells. Sci. Rep. **7**, 15140 (2017).

[75] Finger, E. C. et al. Hypoxic induction of AKAP12 variant 2 shifts PKA-mediated protein phosphorylation to enhance migration and metastasis of melanoma cells. Proc. Natl. Acad. Sci. U. S. A. **112**, 4441–4446 (2015).

[76] Qi, H., Yao, C., Xing, J. & Qin, Y. Hypoxia-induced GPR4 suppresses trophoblast cell migration and proliferation through the MAPK signaling pathway. Reprod. Toxicol. **99**, 1–8 (2021).

[77] Biron-Shental, T. et al. Hypoxia enhances the expression of follistatin-like 3 in term human trophoblasts. Placenta **29**, 51–57 (2008).

[78] Chen, Z., Han, F., Du, Y., Shi, H. & Zhou, W. Hypoxic microenvironment in cancer: molecular mechanisms and therapeutic interventions. Signal Transduct. Target. Ther. **8**, 70 (2023).

[79] Guo, X. & Chen, L. From G1 to M: a comparative study of methods for identifying cell cycle phases. Brief. Bioinform. **25**, bbad517 (2024).

[80] Weiler, P., Lange, M., Klein, M., Pe'er, D. & Theis, F. CellRank 2: unified fate mapping in multiview single-cell data. Nat. Methods (2024).

[81] Bergen, V., Lange, M., Peidli, S., Wolf, F. A. & Theis, F. J. Generalizing RNA velocity to transient cell states through dynamical modeling. Nat. Biotechnol. **38**, 1408–1414 (2020).

[82] Pan, F. C., Brissova, M., Powers, A. C., Pfaff, S. & Wright, C. V. E. Inactivating the permanent neonatal diabetes gene Mnx1 switches insulin-producing beta-cells to a delta-like fate and reveals a facultative proliferative capacity in aged beta-cells. Development **142**, 3637–3648 (2015).

[83] Kulkarni, R. N. et al. PDX-1 haploinsufficiency limits the compensatory islet hyperplasia that occurs in response to insulin resistance. J. Clin. Invest. **114**, 828–836 (2004).

[84] Gao, T. et al. Pdx1 maintains beta cell identity and function by repressing an alpha cell program. Cell Metab. **19**, 259–271 (2014).

[85] Huang, C. et al. Synaptotagmin 4 regulates pancreatic beta cell maturation by modulating the Ca2+ sensitivity of insulin secretion vesicles. Dev. Cell **45**, 347–361.e5 (2018).

[86] Klein, D. et al. Mapping cells through time and space with moscot. bioRxiv 2023.05.11.540374 (2023).

[87] Zhang, J., McKenna, L. B., Bogue, C. W. & Kaestner, K. H. The diabetes gene hhex maintains delta-cell differentiation and islet function. Genes Dev. **28**, 829–834 (2014).

[88] Hrovatin, K. et al. Integrating single-cell RNA-seq datasets with substantial batch effects. bioRxiv (2024).

[89] Xu, C. et al. Automatic cell-type harmonization and integration across human cell atlas datasets. Cell **186**, 5876–5891.e20 (2023).

[90] Webber, W., Moffat, A. & Zobel, J. A similarity measure for indefinite rankings. ACM Trans. Inf. Syst. **28**, 1–38 (2010).

[91] Luecken, M. D. et al. A sandbox for prediction and integration of DNA, RNA, and proteins in single cells (2021).

[92] Ashuach, T., Reidenbach, D. A., Gayoso, A. & Yosef, N. PeakVI: A deep generative model for single-cell chromatin accessibility analysis. Cell Rep Methods **2**, 100182 (2022).

[93] Bravo González-Blas, C. et al. cisTopic: cis-regulatory topic modeling on single-cell ATAC-seq data. Nat. Methods **16**, 397–400 (2019).

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

[95] Andrews, N. et al. An unsupervised method for physical cell interaction profiling of complex tissues. Nat. Methods **18**, 912–920 (2021).

[96] Asp, M., Bergenstråhle, J. & Lundeberg, J. Spatially resolved transcriptomes-next generation tools for tissue exploration. Bioessays **42**, e1900221 (2020).

[97] Rao, A., Barkley, D., França, G. S. & Yanai, I. Exploring tissue architecture using spatial transcriptomics. Nature **596**, 211–220 (2021).

[98] Heumos, L. et al. Best practices for single-cell analysis across modalities. Nat. Rev. Genet. **24**, 550–572 (2023).

[99] Virshup, I. et al. The scverse project provides a computational ecosystem for single-cell omics data analysis. Nat. Biotechnol. **41**, 604–606 (2023).

[100] Kingma, D. P. & Welling, M. Auto-Encoding variational bayes (2013).

[101] Lotfollahi, M., Naghipourfar, M., Theis, F. J. & Wolf, F. A. Conditional out-of-distribution generation for unpaired data using transfer VAE. Bioinformatics **36**, i610–i617 (2020).

[102] Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. arXiv [cs.LG] (2014).

[103] Clevert, D.-A., Unterthiner, T. & Hochreiter, S. Fast and accurate deep network learning by exponential linear units (ELUs) (2015).

[104] Svensson, V. Droplet scRNA-seq is not zero-inflated. Nat. Biotechnol. **38**, 147–150 (2020).

[105] Cao, Y., Kitanovski, S., Küppers, R. & Hoffmann, D. UMI or not UMI, that is the question for scRNA-seq zero-inflation. Nat. Biotechnol. **39**, 158–159 (2021).

[106] Grün, D., Kester, L. & van Oudenaarden, A. Validation of noise models for single-cell transcriptomics. Nat. Methods **11**, 637–640 (2014).

[107] Braun, J. & Griebel, M. On a constructive proof of kolmogorov's superposition theorem. Constr. Approx. **30**, 653–675 (2009).

[108] Diaconis, P. & Shahshahani, M. On nonlinear functions of linear combinations. SIAM J. Sci. Stat. Comput. **5**, 175–191 (1984).

[109] Kolberg, L. et al. g:profiler-interoperable web service for functional enrichment analysis and gene identifier mapping (2023 update). Nucleic Acids Res. **51**, W207–W212 (2023).

[110] Biewald, L. Experiment tracking with weights and biases (2020). Software available from wandb.com.

[111] White, J. V., Steingold, S. & Fournelle, C. Performance metrics for group-detection algorithms. Proceedings of Interface (2004).

[112] Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. Genome Biol. **19**, 15 (2018).

[113] Ahlmann-Eltze, C. & Huber, W. Comparison of transformations for single-cell RNA-seq data. Nat. Methods **20**, 665–672 (2023).

[114] Pedregosa, F. et al. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011).

[115] Hyvärinen, A. & Oja, E. Independent component analysis: algorithms and applications. Neural Netw. **13**, 411–430 (2000).

[116] Zhao, Y., Cai, H., Zhang, Z., Tang, J. & Li, Y. Learning interpretable cellular and gene signature embeddings from single-cell transcriptomic data. Nat. Commun. **12**, 5261 (2021).

[117] Liu, J. et al. Jointly defining cell types from multiple single-cell datasets using LIGER. Nat. Protoc. **15**, 3632–3662 (2020).

[118] Fritsch, F. N. & Butland, J. A method for constructing local monotone piecewise cubic interpolants. SIAM J. Sci. Stat. Comput. **5**, 300–304 (1984).

[119] Luecken, M. et al. Benchmarking atlas-level data integration in single-cell genomics - integration task datasets (2022).

[120] Heumos, L. et al. Pertpy: an end-to-end framework for perturbation analysis. bioRxiv 2024.08.04.606516 (2024).

[121] Lange, M. et al. CellRank for directed single-cell fate mapping. Nat. Methods **19**, 159–170 (2022).

[122] Creators Kristian K Ullrich1 Show affiliations 1. Max Planck Institute for Evolutionary Biology. zebrafish GSE223922 scRNA data set objects.

[123] Locatello, F. et al. Disentangling factors of variation using few labels. arXiv [cs.LG] (2019).

[124] He, Z. et al. Mosaic integration and knowledge transfer of single-cell multimodal data with MIDAS. Nat. Biotechnol. (2024).

[125] Aliee, H., Kapl, F., Hediyeh-Zadeh, S. & Theis, F. J. Conditionally invariant representation learning for disentangling cellular heterogeneity (2023).

[126] Weinberger, E., Lin, C. & Lee, S.-I. Isolating salient variations of interest in single-cell data with contrastiveVI. Nat. Methods **20**, 1336–1345 (2023).

[127] Tu, X. et al. A supervised contrastive framework for learning disentangled representations of cell perturbation data (2024).

[128] Lopez, R., Huetter, J.-C., Hajiramezanali, E., Pritchard, J. K. & Regev, A. Toward the identifiability of comparative deep generative models. In Causal Learning and Reasoning, 868–912 (PMLR, 2024).

[129] Zhang, Z., Zhao, X., Qiu, P. & Zhang, X. scDisInFact: disentangled learning for integration and prediction of multi-batch multi-condition single-cell RNA-sequencing data (2023).

[130] Piran, Z., Cohen, N., Hoshen, Y. & Nitzan, M. Disentanglement of single-cell data with biolord. Nat. Biotechnol. (2024).

[131] Paige, B. et al. Learning disentangled representations with semi-supervised deep generative models. Advances in neural information processing systems **30** (2017).

[132] Eastwood, C. et al. Self-supervised disentanglement by leveraging structure in data augmentations. arXiv [cs.LG] (2023).

[133] Xu, D. et al. A sparsity principle for partially observable causal representation learning. arXiv [cs.LG] (2024).

[134] Theis, F. J. Towards a general independent subspace analysis. In Advances in Neural Information Processing Systems 19, 1361–1368 (The MIT Press, 2007).

[135] Choi, Y., Li, R. & Quon, G. siVAE: interpretable deep generative models for single-cell transcriptomes. Genome Biol. **24**, 29 (2023).

[136] Nalio Ramos, R. et al. Tissue-resident FOLR2+ macrophages associate with CD8+ T cell infiltration in human breast cancer. Cell **185**, 1189–1207.e25 (2022).

[137] Sun, Y., Sheshadri, N. & Zong, W.-X. SERPINB3 and B4: From biochemistry to biology. Semin. Cell Dev. Biol. **62**, 170–177 (2017).

[138] Lee, P. H. et al. Alteration in claudin-4 contributes to airway inflammation and responsiveness in asthma. Allergy Asthma Immunol. Res. **10**, 25–33 (2018).

[139] Martínez, I., Lombardía, L., García-Barreno, B., Domínguez, O. & Melero, J. A. Distinct gene subsets are induced at different time points after human respiratory syncytial virus infection of A549 cells. J. Gen. Virol. **88**, 570–581 (2007).

[140] Ohta, Y. et al. Claudin-4 as a marker for distinguishing malignant mesothelioma from lung carcinoma and serous adenocarcinoma. Int. J. Surg. Pathol. **21**, 493–501 (2013).

[141] Ruiz García, S. et al. Novel dynamics of human mucociliary differentiation revealed by single-cell RNA sequencing of nasal epithelial cultures. Development **146**, dev.177428 (2019).

[142] Yoshida, M. et al. Local and systemic responses to SARS-CoV-2 infection in children and adults. Nature **602**, 321–327 (2022).

[143] Mimata, Y. et al. Interleukin-6 upregulates expression of ADAMTS-4 in fibroblast-like synoviocytes from patients with rheumatoid arthritis: IL-6 upregulates ADAMTS-4 in RA-FLS. Int. J. Rheum. Dis. **15**, 36–44 (2012).

[144] Zhang, H. et al. CRISPLD2 (LGL1) inhibits proinflammatory mediators in human fetal, adult, and COPD lung fibroblasts and epithelial cells. Physiol Rep **4** (2016).

[145] Himes, B. E. et al. RNA-Seq transcriptome profiling identifies CRISPLD2 as a glucocorticoid responsive gene that modulates cytokine function in airway smooth muscle cells. PLoS One **9**, e99625 (2014).

[146] Xiong, D., Wang, Y. & You, M. A gene expression signature of TREM2hi macrophages and $\gamma\delta$ T cells predicts immunotherapy response. Nat. Commun. **11**, 5084 (2020).

[147] Wu, K. et al. TREM-2 promotes macrophage survival and lung disease after respiratory viral infection. J. Exp. Med. **212**, 681–697 (2015).

[148] Xiong, D., Wang, Y. & You, M. A gene expression signature of TREM2hi macrophages and $\gamma\delta$ T cells predicts immunotherapy response. Nat. Commun. **11**, 5084 (2020).

[149] Houghton, A. M. Matrix metalloproteinases in destructive lung disease. Matrix Biol. **44-46**, 167–174 (2015).

[150] Bui, T. M., Wiesolek, H. L. & Sumagin, R. ICAM-1: A master regulator of cellular responses in inflammation, injury resolution, and tumorigenesis. J. Leukoc. Biol. **108**, 787–799 (2020).

[151] Liu, W. & Rodgers, G. P. Olfactomedin 4 expression and functions in innate immunity, inflammation, and cancer. Cancer Metastasis Rev. **35**, 201–212 (2016).

[152] Jarczak, J. et al. Defensins: natural component of human innate immunity. Hum. Immunol. **74**, 1069–1079 (2013).

[153] Xu, H. et al. New genetic and epigenetic insights into the chemokine system: the latest discoveries aiding progression toward precision medicine. Cell. Mol. Immunol. **20**, 739–776 (2023).

[154] Sokol, C. L. & Luster, A. D. The chemokine system in innate immunity. Cold Spring Harb. Perspect. Biol. **7** (2015).

[155] Zhou, Z. et al. Single-cell profiling identifies IL1Bhi macrophages associated with inflammation in PD-1 inhibitor-induced inflammatory arthritis. Nat. Commun. **15**, 2107 (2024).

[156] Yang, C. et al. Heterogeneity of human bone marrow and blood natural killer cells defined by single-cell transcriptome. Nat. Commun. **10**, 3931 (2019).

[157] Marozkina, N. et al. Somatic cell hemoglobin modulates nitrogen oxide metabolism in the human airway epithelium. Sci. Rep. **11**, 15498 (2021).

Supplemental Material

Contents

1 Related works and paper contributions **III**

2 On the additivity of underlying biological processes in count space **VI**

3 Identification of the remaining non-cell-type dimensions for the HLCA **VII**

4 Theorems and Lemmas Required for Disentanglement **IX**

5 On multiplicative behavior of current autoencoders and factorization models **XII**

6 Log-space Parametrization of the Negative Binomial log-likelihood **XIII**

7 DRVI's code functionality **XIV**

Supplemental Tables **XV**

Related Methods. XV

Summary of Datasets. XV

Summary of Model Parameters. XVI

Supplemental Figures **XVII**

Plausibility of DRVI's additive assumption in scRNA-seq data. XVII

UMAPs and integration results for benchmarked methods on the immune dataset. XVII

Interpretability of DRVI latent factors in immune dataset. XVIII

Latent space visualization for benchmarked methods in the immune dataset. XIX

The effect of removing technical stress response dimensions. XX

Annotation refinement and correction using DRVI in the immune dataset. XXII

Summary of all metrics. XXII

Results of the primary disentanglement metric family. XXIII

Results of the secondary disentanglement metrics. XXIV

Unaggregated Scaled Mutual Information (SMI) scores for the Immune dataset. XXV

Unaggregated Same-Process Neighbors (SPN) scores for the Immune dataset. XXVI

Unaggregated Scaled Mutual Information (SMI) scores for the HLCA. XXVII

Unaggregated Same-Process Neighbors (SPN) scores for the HLCA. XXVIII

Heatmap of simulated datasets with known ground truth. XXIX

Evaluation results for simulated datasets with known ground truth. XXX

The optimization of training objectives. XXXI

The effect of KL loss weight on reconstruction and disentanglement performance. XXXI

UMAPs and integration results for benchmarked methods on the HLCA dataset. XXXIII

Latent space visualization for benchmarked methods on the HLCA. XXXIII

Activity of DRVI factors representing non-cell-type processes in the HLCA on UMAP . . XXXIV

Interpretability of DRVI latent factors corresponding to non-cell-type processes in the HLCA XXXV

Rare cell-type identification performance of Leiden clustering in the HLCA. XXXVI

Indication of tumor-adjacent blood vessels in the HLCA. XXXVIII

DRVI disentangles cell cycle stages. XXXVIII

Developmental fate prediction based on model embeddings. XXXIX

DRVI identifies and preserves the geometry of combinatorial perturbations. XL

DRVI's latent space effectively models interacting double perturbations. XLI

The saturation of the number of employed latent dimensions by DRVI. XLII

Further analysis on the saturation of employed latent dimension. XLIII

Training DRVI on four datasets of the Blood data. XLIV
Assessing the generalization and disentanglement of latent programs in unseen Blood data XLV
Full disentanglement metrics on generalization to unseen datasets in the Blood data. . . . XLVI
DRVI is applicable to scATAC-seq data. XLVIII
DRVI leverages additive decoders and interaction asymmetry to prove disentanglement. . XLVIII
The effective and ignorable parts of the pooling function XLIX
How latent space sampling and KL loss promote disentanglement. L
Summary of all metrics for the ablation study. LI
Details of disentanglement metrics for the ablation study. LII
Runtime analysis of DRVI. LIII
Runtime for interoperability and sorting latent dimensions. LV

Supplemental Note 1: Related works and paper contributions

To position DRVI within the field, we begin by discussing its relation to the broader disentangled representation learning literature. Then, we review existing disentanglement methods applied to single-cell omics and finally highlight the specific contributions of this paper.

Disentangled representation learning

Disentangled representation learning methods can be categorized according to architecture, representation structure, and supervision signal [33].

Common disentangled representation learning methods utilize generative models such as variational autoencoders (VAEs), generative adversarial networks (GANs), and recently, diffusion models [33]. These models are primarily applied in computer vision with various regularization losses to encourage disentanglement. Nevertheless, according to the well-established limitation that unsupervised disentanglement is theoretically impossible without inductive bias on methods and datasets [34], the possibility of disentanglement in these architectures can be attributed to their inherent inductive biases, such as convolutional neural networks (CNNs). However, given that generative models in single-cell representation learning are typically unconstrained and lack such inherent inductive biases, the direct applicability of these paradigms to single-cell omics is questionable. Consequently, DRVI addresses this limitation by incorporating specific architectural designs, such as additive decoders [37] with log-sum-exp pooling, which serve as inductive biases to enforce disentanglement. This means while DRVI is built within the CVAE framework, its inductive bias can be adapted and incorporated within the diverse generative model settings outlined in [33].

Disentangled latent space models can be categorized based on their approach to building conceptual disentangled blocks. Some models partition the latent space into independent blocks (block-based disentanglement), each representing a high-level process. However, dimensions within these blocks may remain entangled. Other disentangled latent space models take a more granular approach, disentangling every dimension in the latent space (dimension-wise disentanglement) to represent underlying processes. DRVI falls into this second category, while also offering users the flexibility to perform block-based disentanglement (though without theoretical guarantees) by adjusting default parameters.

The impossibility of general disentanglement in unsupervised settings motivated the incorporation of supervised and weakly supervised information [33, 123]. In accordance with this, many existing single-cell disentanglement methods rely on auxiliary data or supervised information to enforce disentanglement. In contrast, employing a suitable inductive bias allows DRVI to learn disentangled representations without requiring labeled information about the underlying biological processes.

Disentanglement in single-cell omics

Here, we describe relevant works on disentangled representation learning of single-cell omics, focused on the assumptions and the structure of their latent space representations.

In block-based disentanglement, methods such as MIDAS [124] and inVAE [125] disentangle biological and technical effects into two distinct blocks within the latent space. Similarly, ContrastiveVI [126] and SC-VAE [127, 128] construct two latent space vectors based on the given perturbational signatures to isolate salient variations. Finally, scDisInFact [129] and BIOLORD [130] construct multiple conceptual blocks given variations of interest.

The impossibility of achieving complete dimension-wise disentanglement in the general case of latent space models is a well-established theorem [34]. Researchers have come up with various strategies to address this challenge, including incorporating auxiliary data (supervision) or constraining the

model search space [34, 37, 131–134]. In single-cell omics literature, sVAE [40], SAMS-VAE [41], and sVAE-ligr [42] use perturbations as supervision and sparsely represent groups of similar perturbations in each dimension. Models like expiMap [43] and pmVAE [44] incorporate pathway knowledge into their architectures to achieve disentanglement. Decipher [45] aligns each of the two primary latent dimensions considering given variations of interest. BAE [39] and MOFA+ [38] try to disentangle each latent dimension by enforcing sparsity on their linear models. Finally, Spectra [26] disentangles the factors by using cell-type information and gene-gene knowledge graph as supervision, enforcing sparsity on a non-negative matrix factorization model. In addition to the mentioned models, there are a few other approaches seeking to achieve nonlinear unconstrained disentanglement, such as β-TCVAE and MICHIGAN [46, 47]. As confirmed through practical experiments and theoretical analysis, using fully connected layers and lacking an inductive bias, none of these two models significantly improve disentanglement [34]. We summarized disentangled latent space models tailored specifically for single-cell omics data in Supplementary Table 1.

Interpretability is one of the main advantages of dimension-wise disentanglement methods. Topic modeling and factorization models are two relevant lines of research in single-cell representation learning, providing interpretable latent representations [116, 117, 135]. Topic modeling methods often assume that cells are sampled from disjoint topics with probabilities that sum to one (usually modeled by the Dirichlet distribution) [116]. Consequently, while topic modeling approaches can identify dominant competing signals (interpretable as soft clustering), they may be unable to identifying secondary or shared signals. In contrast, disentangled representation learning allows multiple parallel processes to coexist within a cell. Factorization models represent the observational matrix as the product of two low-dimensional matrices [38, 116, 117, 135]. While the linearity constraint alone isn't enough to mitigate the well-known unidentifiability issue [123], additional assumptions such as sparsity [38] or non-negativity [117] of loadings can potentially enforce disentanglement and facilitate interpretation.

Paper contributions

- **Highlighting the Importance of Disentanglement in single-cell omics**: We demonstrated the significance of unsupervised dimension-wise disentanglement as an important yet understudied problem in single-cell omics. We explained its advantage compared to clustering, its ability to identify rare cell types and variations beyond cell types, and its capacity to represent continuous developmental processes.
- **Integration of Additive Decoders and scVI for Disentangled Representation learning**: We successfully integrated scVI [19] with additive decoders [37] to achieve disentangled representations. We demonstrated that a naive combination of these approaches failed to produce disentangled latent spaces for single-cell omics data, showing the need for the introduction of novel architectural modifications, which we will outline next.
- **Novel Architecture with Shared Activation**: We introduced a neural network architecture where each decoder subnetwork was followed by a shared activation function ψ, and we formulated a more practical implementation as a pooling function. This architecture (that still remains within the additive decoders framework) is novel and not discussed in [37]. In addition, we highlighted the mechanistic interpretation of this decision in single-cell omics: "Independent biological processes add up in count space," and provided a rationale based on biological and technical perspectives.
- **Theoretical Guarantee of Disentanglement**: We demonstrated that for $\psi = \exp$ – formulated as log-sum-exp pooling – disentanglement could be theoretically proven under reasonable assumptions specific to single-cell omics. This proof leverages the interaction asymmetry framework [48].

- **Efficient and Effective Implementation**: Beyond the core architecture, we introduced several practical modifications necessary for achieving optimal results. These include the parametrization of the Negative Binomial distribution in log-space (Supplementary Note 6), the use of ELU as a smooth alternative to ReLU, and, most importantly, the implementation of efficient deep learning modules to enable parallel processing of MLPs and layer normalization.
- **Efficient Interpretability Pipeline**: We developed an efficient interpretability pipeline that allows for the interpretation of latent factors in terms of gene scores. The runtime of this algorithm is independent of the number of cells.
- **Comprehensive Evaluation Framework**: We proposed a comprehensive framework for evaluating disentanglement in single-cell omics. This framework comprises both existing and novel metrics, and we provided a justification for our choices within different metric families.

Supplemental Note 2: On the additivity of underlying biological processes in count space

In DRVI, we proposed the additivity in count space as an alternative to the current trend in white-box models, where data is modeled in log space, and effects are added up. Log-space additivity leads to the assumption that when two underlying processes affect the same gene, their effects multiply in count space (Supplementary Note 5).

We offer a mechanistic justification for this assumption. Considering the decay of expressed RNAs within each process (as a modeling choice, RNA degradation is not modeled explicitly due to the requirement for more complex models and data limitations), if two biological processes affect the same gene in different time frames, the newly transcribed genes will result in an additive effect on the total RNA count and therefore add up in count space when sequenced (Supplementary Fig. 1 top). This aligns with biological scenarios such as developmental bifurcations, induced perturbations, and stress responses, where newly transcribed RNAs contribute additively. Furthermore, this assumption also aligns with many technical processes and artifacts during the generation of scRNA-seq data. For instance, the combined expression of doublet cells, background contamination, hemoglobin (HBB) gene contamination, and even residual counts from neighboring cells due to imperfect segmentation in spatial omics all contribute additively to the observed count space (Supplementary Fig. 1 bottom). Thus, in contrast to prior models, our additive assumption provides a more plausible framework for modeling single-cell count data.

We would like to clarify that here the task of interest in disentanglement is to identify the underlying biological processes, without inferring causal relationships between them. For instance, consider two processes, P1 and P2, where P1 activates P2, and P2 suppresses P1. Each process has its own set of genes that are transcribed when it is active: a subset of genes expressed by P1 may activate P2, while the genes expressed by P2 may inhibit P1. Although the activation of P2 may influence the expression of genes associated with P1, we do not consider these indirectly affected genes as part of the direct transcriptional output of P2. Accordingly, while we are interested in identifying P1 and P2, we are not interested in identifying their causal relationships, although disentanglement may help in this downstream task. In our framework, we define the output of a process solely based on its direct transcriptional effects, disregarding downstream or propagated causal effects. This assumption allows us to model the activity of individual processes additively. And, inferring relationships between identified processes, such as potential causal links between P1 and P2, is beyond the scope of this study.

Finally, when the gene expression from two distinct processes (P1 and P2) creates a novel transcriptional activity, such as the synergistic regulation of transcription factors, their combined effect can be effectively modeled as an entirely new process within our framework.

Supplemental Note 3: Identification of the remaining non-cell-type dimensions for the HLCA

Here, we try to identify and explain the rest of the non-cell-type dimensions not described in the results section. This is done based on the activity of the dimensions (Supplementary Fig. 20) and corresponding identified gene programs (Supplementary Fig. 21). Dimensions are described with the same order as Fig. 3b.

DR 27+: This dimension indicates an inflammatory process in myeloid cells. The primary gene associated with this dimension is EREG, with genes such as IL1B, IL1A, CXCL8, and CXCL3 upregulated in this subpopulation. GSEA analysis suggests an association with the regulation of apoptotic signaling and interleukin-10 signaling. As mentioned earlier, there are other dimensions representing inflammation, each with its own specific gene signature.

DR 30+: This dimension highlights a gradient in ciliated cells. DNAH11, DNAAF1, DNAH12, and DNAH6 are among the top identifiers of this dimension, expressed across ciliated cells, with significantly higher expression observed in the upper gradient of DR 30+. GSEA also suggests cilium for this dimension.

DR 16+: All the genes identified for DR16+ are human leukocyte antigen gene complex (HLA) genes, including HLA-DRA, HLA-DRB1, and HLA-DPA. GSEA suggests that this dimension corresponds to the MHC class II protein complex. Notably, while HLA genes are generally more highly expressed in certain cell types like myeloid cells, this dimension provides a per-cell-type normalized representation, covering similar ranges across relevant cell types, such as myeloid cells and blood vessels.

DR 21-: This dimension highlights a subset of epithelial cells mostly active in nasal goblet cells, identified by KLK7, TMPRSS11B, SPINK7, ECM1, SCEL, and KLK6 as their top relevant genes. We could not identify a biologically meaningful explanation for this dimension.

DR 24+: FOLR2, F13A1, STAB1, CCL13, and SELENOP are the main identifiers of the gene program associated with DR 24+. These genes indicate a perivascular population of macrophages [69, 136]. High activity of this dimension in Interstitial Mph perivascular cells confirms this relevance.

DR 25-: SERPINB3 and SERPINB4 are the top identified gene for DR 25-, active in epithelial cells. Although the coexpression of these two genes is biologically relevant [137], we could not find a specific biological explanation for the subpopulation of cells active in this dimension.

DR 28+: The activity of DR 28+ indicates the expression of PLAU, its receptor PLAUR, and an elevated expression of CLDN4 (Claudin-4) in airway epithelium. While PLAU and CLDN4 are separately suggested as markers of lung injury or infection, we could not find a comprehensive description for this dimension [138–140].

DR 29-: This dimension highlights a subset of epithelial cells, including Hillock-like cells. Our interpretability pipeline identified KRT6B, KRT13, KRT16, and KRT6A as key markers, suggesting a keratization process within this dimension. Previous studies ([141, 142]) have employed a diverse set of KRT genes to characterize epithelial subpopulations. In particular, KRT16 and KRT6A are among the markers of certain Hillock-like cell populations.

DR 33+: EDN1, ERRFI1, DST, and TNFRSF12A are the main indicators of DR 33+. This dimension is more active in basal cells, with a significant upregulation of cells from the Nawijn_2021 dataset. However, a clear technical or biological explanation for this dimension remains open.

DR 35-: This dimension indicates a subset of fibroblast lineage and smooth muscle cells identified by a gene program composed of genes such as CRISPLD2, IL6, and ADAMTS4. IL-6 is a pro-inflammatory cytokine, reported to induce expression of ADAMTS4 in rheumatoid arthritis samples [143]. Conversely, CRISPLD2 is a known inhibitory regulator of IL-6 [144, 145]. The coexpression of these genes suggests a dynamic of IL-6 regulation. The exact role of this process needs further

experiments.

DR 35+: C1QA, C1QB, C1QC, and TREM2 are the top indicators of the DR 35+, Suggesting that this dimension represents TREM2 and the C1Q complex in myeloid cells, including TREM2+ macrophages [146–148].

DR 36-: LYPD3, CALML3, AQP3, TASCTD2, and LY6D are the genes identifiers of the nonlinear gene program for DR 36-. The role of this dimension is not identified.

DR 37-: ANKRD36C gene is the main driver of the DR 37- expressed only in a subset of goblet cells. Comparison with cell types suggests that this dimension is closely related to the Bronchial subsets of the Goblet cells. However, we could not find any evidence of the validity of such a subcluster in the literature, and this dimension seems to be dataset-specific.

DR 38+: DR 38+, characterized by MKI67, TYMS, and PCLAF expression, is associated with cell-cycle processes as identified by GSEA. This aligns with the cell-type annotations (Fig. 3b).

DR 40+: This dimension highlights a subset of airway epithelial cells, with RARRES1, CXCL6, SERPINB4, and IL17C as the key drivers of its gene expression program. While GSEA suggested an IL-17 signaling pathway, we found the relationship between these genes and the pathway to be weak. As a result, we were unable to identify a meaningful explanation for this dimension.

DR 41+: This dimension identifies a subset of epithelial cells, with MMP1, STC1, MMP13, and MMP10 as their top relevant genes. Accordingly, this dimension highlights the activity of certain proteins in the Matrix Metalloproteinase family, suggesting an ongoing destructive process [149].

DR 42+: With genes such as CCL20, CXCL2, CXCL3, CSF2, CSF3, and ICAM1 among the top associated genes, GSEA identifies IL-17, IL-10, and TNF signaling pathways for DR 42+ (Supplementary Fig. 21). This suggests an ongoing inflammation process [150].

DR 43+: This dimension highlights two subsets of cells (Supplementary Fig. 20). The tiny subset in Myeloid cells is associated with genes such as OLFM4, S100A7, and DEFA3, suggesting a small population of neutrophils [151, 152]. The other subset in airway epithelium is targeted by S100A7, SPRR2D, SPRR1B, SPRR2E, SPRR2F, and SPRR3 in the top relevant genes. GSEA suggests keratinization and formation of the cornified envelope for this dimension. The coexistence of two unrelated cell subsets within this dimension suggests a potential failure in disentanglement, possibly due to insufficient latent dimensions.

DR 44+: This dimension indicates a subset of myeloid cells characterized by but not limited to CCL3, CCL4, CCL3L1, CCL4L2, CXCL3, IL1A, and IL1B genes. CCL3 (MIP-1α) and CCL4 (MIP-1β) and their paralogs CCL3L1 and CCL4L2, are proinflammatory chemokines that attract immune cells [153]. IL-1β is an inflammatory cytokine that activates cells to produce more CCL3 and CCL4 chemokines [154]. GSEA describes this dimension by pro-inflammatory process and cytokine activity. This subpopulation of myeloid cells is of biological and clinical interest [155]. It is worth mentioning that DR 27+ complements this by highlighting another population of myeloid cells undergoing interleukin-1 signaling pathway.

DR 49-: With HBA2, HBA1, and HBB as its top identifiers, GSEA analysis suggests that DR 49- represents the hemoglobin metabolic process. While this process is expected to be most active in erythroid cells that are not present in the HLCA, we observed non-negligible expression of hemoglobin genes in other cell types. The expression of hemoglobin genes in the HLCA could be attributed to technical or biological factors [156, 157].

Dim DR 50+: SAA1 and SAA2 are the main identifiers of the dim DR 50+. So, we can conclude that this dimension presents the activity of SAA as an inflammatory marker.

DR 54-: The nonlinear gene program associated with DR 54- includes multiple genes such as ISG15, IFIT1, IFI44L, and IFI6. GSEA describes this dimension by interferon alpha/beta signaling and response to virus pathways.

Supplemental Note 4: Theorems and Lemmas Required for Disentanglement

Lemma 1. A function f with $D^2 f(z) + (Df(z))^2 = 0$ is either constant or of the form $f(z) = \log|z - C_1| + C_2$ where C_1 and C_2 are constants.

Proof. Let $p(z) = f'(z)$ then $\frac{dp}{dz} + p^2 = 0$. Obviously, $p = 0$ (constant f) is a solution. For the rest, we assume $p \neq 0$.

This is a separable first-order differential equation for p.

$$\frac{dp}{dz} = -p^2$$
$$\frac{dp}{p^2} = -dz$$

Integrating both sides, we have:

$$\int \frac{1}{p^2} dp = \int -1 dz$$
$$-\frac{1}{p} = -z + C_1$$
$$\frac{1}{p} = z - C_1$$
$$p = \frac{1}{z - C_1}$$

Now substitute back $p(z) = f'(z)$:

$$\frac{df}{dz} = \frac{1}{z - C_1}$$

Integrate both sides with respect to z:

$$\int df = \int \frac{1}{z - C_1} dz$$
$$f(z) = \log|z - C_1| + C_2$$

So, $f(z) = \log|z - C_1| + C_2$, where C_1 and C_2 are constants. □

Lemma 2. Let C be a constant, A function f with $D^2 f(z) + C \times Df(z) = 0$ is of the form $C_1 \exp(-Cz) + C_2$.

Proof. The condition is a homogeneous linear differential equation with characteristic polynomial $r^2 + Cr$ and roots $-C$ and 0. So the solutions are of the form $C_1 \exp(-Cz) + C_2 \exp(0 \times z) = C_1 \exp(-Cz) + C_2$, where C_1 and C_2 are constants. This concludes the proof. □

Theorem 1. Assuming the decoder subnetworks are not logarithmic at any point, A log-sum-exp (LSE) pooled additive decoder with no vanished dimension and maximal splits of latent dimensions satisfies the 1st order interaction asymmetry criteria.

Proof. As previously discussed, the σ term in pooled additive decoders can be disregarded through a suitable transformation of the outputs (Remark 1 of additive decoders [37]). If we ignore the σ term, pooled additive decoders inherently exhibit at most first-order interactions between the splits. For the main part (second part) of interaction asymmetry, we employ a proof by contradiction to demonstrate that the introduced $\psi = \exp$ implies second-order interactions within the splits. Let's assume the contrary.

For a one-dimensional split ($B = \{i\}$), we observe that $D_i^2(\psi \circ f^{(B)}(z_i)) = (D_i^2 f(z_i) + (D_i f(z_i))^2) \times \exp(f^{(B)}(z_i))$. If there are no second-order interactions, then $(D_i^2 f(z_i) + (D_i f(z_i))^2) \times \exp(f^{(B)}(z_i)) = 0$ must hold for a neighborhood around z. Due to smoothness, one of the factors must be zero over this neighborhood. If $\exp(f^{(B)}(z_i)) = 0$ over the neighborhood, this contradicts the assumption of a non-vanished dimension. Alternatively, if $D_i^2 f(z_i) + (D_i f(z_i))^2 = 0$, then by Lemma 1 $f(z_i) = \log|z_i - C_1| + C_2$, where C_1 and C_2 are multi-dimensional constant vectors. This contradicts our first assumption.

For a multi-dimensional split, consider a linear transformation T and a partition of B into I_1, I_2, as defined in the definition of 1st order interaction asymmetry. Now, let's assume, for contradiction, that for all $i_1 \in I_1, i_2 \in I_2$, we have $D_{i,i'}^2(\psi \circ f)(Tz) = 0$. By Proposition 7 from [37], this would imply that the limitation of $\psi \circ f \circ T$ to I_1 and I_2 forms two additive functions, which contradicts the maximality of splits. □

The proof above demonstrates that the $\psi = \exp$ function, being relatively complicated and having infinitely many derivatives, couples latent variables within each split. The interactions introduced by this coupling are preserved unless a complex basis function, resembling the inverse of ψ, cancels this coupling. By properly extending Lemma 1, this proof can be generalized to other ψ functions.

Theorem 2. Consider the ground-truth (GT) processes $p1, p2, \ldots, p_k$ underlying single-cell omics data. Assume that each of these processes has a distinct subset of involved genes. Then, for log-sum-exp (LSE) pooled additive decoders:

$$\operatorname{rank}\left(\left[FOI(B)\middle|SOI(B)\right]_{B\in\mathcal{P}}\right) = \sum_{B\in\mathcal{P}} \operatorname{rank}\left([FOI(B)|SOI(B)]\right) \tag{39}$$

Proof. Consider the GT processes $p1, p2, \ldots, p_k$ underlying single-cell omics data, although we have no access to them. We note that there may be genes involved in multiple processes. Without loss of generality, we drop all genes not specific to one process. Now, remember the 1st order sufficient independence criterion for LSE additive decoders:

$$\operatorname{rank}\left(\left[FOI(B)\middle|SOI(B)\right]_{B\in\mathcal{P}}\right) = \sum_{B\in\mathcal{P}} \operatorname{rank}\left(FOI(B)\right) + \operatorname{rank}\left(SOI(B)\right) \tag{40}$$

$$FOI(B) = \left[D_i f^{(B)}(z_B) \times \exp(f^{(B)}(z_B))\right]_{i\in B} \tag{41}$$

$$SOI(B) = \left[\left(D_{i,i'}^2 f^{(B)}(z_B) + D_i f^{(B)}(z_B) \times D_{i'} f^{(B)}(z_B)\right) \times \exp\left(f^{(B)}(z_B)\right)\right]_{i,i'\in B} \tag{42}$$

Here, because of using LSE pooling, the term $\exp(f^{(B)}(Z_B))$ appears in all elements of both the first-order interaction ($FOI(B)$) and second-order interaction ($SOI(B)$) matrices. Furthermore, the

aggregation of these values ($\sum_{B\in\mathcal{P}} \exp(f^{(B)}(Z_B))$) will reconstruct the expression matrix. Considering that the outputs of each split of the generative process ($\exp(f^{(B)}(Z_B))$) are non-negative, they are nonzero if and only if the gene is a marker of the split. This means that for each split B, only the rows corresponding to the marker genes of that split are nonzero. Therefore, $[FOI(B)|SOI(B)]$ matrices have no common nonzero rows between splits. This concludes the proof. □

Theorem 2 significantly brings the two sides of the 1st order sufficient independence criterion together. We close this gap for the one-dimensional case in Theorem 3. It is worth mentioning that in Theorem 2, having more and more markers for multi-dimensional processes further encourages satisfaction of 1st order sufficient independence criterion.

Theorem 3. Assume a generative process is modeled by a log-sum-exp (LSE) pooled additive decoder with one-dimensional splits. Consider the ground-truth (GT) processes $p1, p2, \ldots, p_k$ underlying single-cell omics data. Assume each of these processes has a distinct subset of at least two involved genes with different log-fold-changes. Then, the 1st order sufficient independence is guaranteed.

Proof. Considering Theorem 2, it is enough to show that for each split B, $\text{rank}\,([FOI(B)|SOI(B)]) = \text{rank}\,(FOI(B)) + \text{rank}\,(SOI(B))$. To this end, we represent the nonzero rows of these matrices by their values:

$$FOI(B=\{i\}) = \begin{bmatrix} D_i(\exp\circ f_{g_1}^{(B)})(z_i) \\ \vdots \\ D_i(\exp\circ f_{g_t}^{(B)})(z_i) \end{bmatrix} \tag{43}$$

$$SOI(B=\{i\}) = \begin{bmatrix} D_i^2(\exp\circ f_{g_1}^{(B)})(z_i) \\ \vdots \\ D_i^2(\exp\circ f_{g_t}^{(B)})(z_i) \end{bmatrix} \tag{44}$$

Where $g_1, \ldots, g_t$ are the markers for the process corresponding to split B. We prove the sufficient independence by contradiction. Let's assume that $FOI(B)$ and $SOI(B)$ are dependent on a neighborhood of z. Then, there exists C where $SOI(B) + C \times FOI(B) = 0$. Then, by Lemma 2 we can conclude that all of the functions $\exp\circ f_{g_1}^{(B)}, \ldots, \exp\circ f_{g_t}^{(B)}$ are of the form $C_{B,1}\exp(-Cz) + C_{B,2}$, where C is defined before, and $C_{B,1}, C_{B,2}$ are split-wise constants. This means that all subnetworks $f_{g_1}, \ldots, f_{g_t}$ are linear with equal slope (C), which contradicts the assumption of having at least two markers with different log-fold changes. This completes the proof. □

In Theorem 3, it's important to note that establishing first-order sufficient independence doesn't solely rely on observing differing log-fold changes in two marker genes. The proof by contradiction could also be concluded if we had anything other than having a set of linear subnetworks $f_{g_1}, \ldots, f_{g_t}$ with equal slopes.

Supplemental Note 5: On multiplicative behavior of current autoencoders and factorization models

Let $\boldsymbol{Z} \in \mathbb{N}^{N \times K}$, $\boldsymbol{W} \in \mathbb{N}^{K \times G}$ be the left-hand-side and right-hand-side matrices in a factorization model that models the data in log-space. Generally, log-transformed counts are modeled as the multiplication of these two matrices. So, formally, we have

$$log(\boldsymbol{X} + 1) = \boldsymbol{Z}\boldsymbol{W}. \tag{45}$$

Generally, $\boldsymbol{Z}_{n,k}$ and $\boldsymbol{W}_{k,g}$ are interpreted as the activity of factor k in cell n and the activity of gene g in factor k, respectively. Likewise, $\boldsymbol{Z}_{n,k}\boldsymbol{W}_{k,g}$ shows if gene g should be active in cell n because of the activation of the process k. Let's map the factors in log-space to the original count space:

$$\begin{aligned} log(\boldsymbol{X}_{n,g} + 1) &= \sum_k \boldsymbol{Z}_{n,k}\boldsymbol{W}_{k,g} \\ \Rightarrow \boldsymbol{X}_{n,g} + 1 &= \exp\left(\sum_k \boldsymbol{Z}_{n,k}\boldsymbol{W}_{k,g}\right) \\ \Rightarrow \boldsymbol{X}_{n,g} + 1 &= \prod_k \exp(\boldsymbol{Z}_{n,k}\boldsymbol{W}_{k,g}) \end{aligned} \tag{46}$$

As we see, the values $\boldsymbol{Z}_{n,k}\boldsymbol{W}_{k,g}$ multiply in count space. This leads to the fact that modeling relying on additivity in log space implies that if gene g in cell n is active because of the activation of multiple independent processes, their effect is multiplied in count space.

We can conclude the same in autoencoder models. Let $\boldsymbol{Z}_{n,k}$ be the latent factors in an autoencoder model with a linear decoder. As in the factorization case, let $\boldsymbol{W}$ be the weight matrix in the last layer of the decoder. Interpreting these $\boldsymbol{Z}_{n,k}$ as activity of some factors for cell n and $\boldsymbol{W}_{k,g}$ as the signature of each factor, we can conclude the same.

In the case of deeper decoders, one can define $\boldsymbol{Z}_{n,k}$ as the value for cell n in node k in the last hidden layer of the decoder (instead of latent variables). Considering these values as "hypothetical" factors indicating activity of hypothetical factor k in cell n, we can conclude that autoencoders with MLP decoders also rely on the assumption that some activities (not latent activities but activities in the last layer) aggregate through multiplication in count space.

Supplemental Note 6: Log-space Parametrization of the Negative Binomial log-likelihood

In current generative models that use Negative Binomial distribution, the decoder calculates the parameters in log-space, and then these log-space parameters are transferred to count space by applying the exponential function. The count-space likelihood of the Negative Binomial distribution is then calculated based on the following expression:

$$\begin{aligned}
&\log\mathcal{L}(\mu, \theta \ ; \ x) = \\
&\quad \theta\left(\log(\theta+\epsilon) - \log(\theta+\mu+\epsilon)\right) + \\
&\quad x\left(\log(\mu+\epsilon) - \log(\theta+\mu+\epsilon)\right) + \\
&\quad \log\Gamma(x+\theta) - \log\Gamma(\theta) - \log\Gamma(x+1)
\end{aligned} \tag{47}$$

Where μ and θ represent mean and dispersion in Negative Binomial distribution, x is the observed expression in count space, Γ is the gamma function used as a smoothed version of the factorial function ($\Gamma(n) = (n-1)!$), and ϵ is a small value to increase numerical stability.

As we see above, this approach involves canceling exponential and logarithm functions that may reduce the numerical precision (i.e., $\log(\theta)$ since θ itself is an output of the exponential function applied to the decoder output). This is our motivation to define the likelihood of the Negative Binomial distribution directly based on the log-space parameters as much as possible (especially the mean parameter). Here we need an intermediate trick to simplify $\log\left(\frac{a}{a+b}\right)$:

$$\begin{aligned}
\log\left(\frac{a}{a+b}\right) &= -\log\left(1+\frac{b}{a}\right) \\
&= -\log\left(1+\exp\left(\log b - \log a\right)\right) \\
&= -\text{SoftPlus}\left(\log b - \log a\right)
\end{aligned} \tag{48}$$

Now we have:

$$\begin{aligned}
&\log\mathcal{L}(\log\mu, \log\theta \ ; \ x) \\
&= \log\left(\binom{x+\theta-1}{x}\left(1-\frac{r}{\mu+r}\right)^{x}\left(\frac{r}{\mu+r}\right)^{\theta}\right) \\
&= \log\binom{x+\theta-1}{x} + \log\left(\left(\frac{\mu}{\mu+r}\right)^{x}\right) + \log\left(\left(\frac{r}{\mu+r}\right)^{\theta}\right) \\
&= \log\binom{x+\theta-1}{x} + x\log\left(\frac{\mu}{\mu+r}\right) + \theta\log\left(\frac{r}{\mu+r}\right) \\
&= \log\Gamma(x+\theta) - \log\Gamma(x+1) - \log\Gamma(\theta) \\
&\quad - x\ \text{SoftPlus}(\log\theta - \log\mu) - \theta\ \text{SoftPlus}(\log\mu - \log\theta)
\end{aligned} \tag{49}$$

Finally, we add some small values to make the expression numerically stable:

$$\begin{aligned}
&\log\mathcal{L}(\log\mu, \log\theta \ ; \ x) \\
&= \log\Gamma(x+\theta+\epsilon) - \log\Gamma(x+1+\epsilon) - \log\Gamma(\theta+\epsilon) \\
&\quad - x\ \text{SoftPlus}(\log\theta - \log\mu + \epsilon) - \theta\ \text{SoftPlus}(\log\mu - \log\theta + \epsilon)
\end{aligned} \tag{50}$$

Where $\log\mu$ and $\log\theta$ represent mean and dispersion in Negative Binomial distribution, x is the observed expression in count space, Γ is the gamma function used as a smoothed version of the factorial function ($\Gamma(n) = (n-1)!$), and ϵ is a small value to increase numerical stability. Please note that although θ and $\log\theta$ both appear in the expression, we were able to parametrize the mean parameter (μ) solely based on the log value. This increases the precision in our model since $\log\mu$ is the parameter composed of multiple values aggregated together, and the gradients for this parameter are of great interest.

Supplemental Note 7: DRVI's code functionality

Below is a pseudocode demonstrating how to apply DRVI for training, latent space visualization, and interoperability. The parts unrelated to DRVI are provided with abstract functions starting with `abs_`.

```
# Load Dataset
adata = abs_read_data()

# Highly-Variable Gene Selection
hvg_genes = abs_hvg_calculation(adata)
adata = adata[:, hvg_genes]

# Train DRVI
DRVI.setup_anndata(adata, layer="counts", categorical_covariate_keys=["batch"])
model = DRVI(adata, n_latent=64, encoder_dims=[256, 256], decoder_dims=[256, 256])
model.train(max_epochs=400)                    # Train Model for 400 Epochs

# Latent Space Analysis
embed = model.get_latent_representation()      # Obtain Latent Representation
embed = abs_preprocess_latent_space(embed)     # Neighbors and UMAP Embedding

# Latent Dimension Statistics (e.g., Find Vanished Dimensions)
drvi.utils.tl.set_latent_dimension_stats(model, embed)
drvi.utils.pl.plot_latent_dimension_stats(embed)

# Visualize Latent Space and Activity of Individual Dimensions
abs_visualize_umap(embed, color=["batch", "annotation"])
drvi.utils.pl.plot_latent_dims_in_umap(embed)  # Plot Latent Activities on UMAP
# Make a Heatmap of Latent Dimensions Grouped by Cell Types (e.g., Fig. 3b)
drvi.utils.pl.plot_latent_dims_in_heatmap(embed, "annotation", title_col="title")

# Traverse Latent Space and Find Top Relevant Genes
traverse_adata = drvi.utils.tl.traverse_latent(model, embed, n_samples=20)
drvi.utils.tl.calculate_differential_vars(traverse_adata)
drvi.utils.pl.show_top_differential_vars(traverse_adata, key="combined_score")

# Explore Individual Dimension (e.g., DR 8+) to Get Insights
# Details of Differential Variables
drvi.utils.pl.show_differential_vars_scatter_plot(
    traverse_adata,
    key_x="max_possible",
    key_y="min_possible",
    key_combined="combined_score",
    dim_subset=["DR 8+"],
)
# Plot Top Relevant Genes on UMAP
drvi.utils.pl.plot_relevant_genes_on_umap(embed, traverse_adata, dims=["DR 8+"])
```

Detailed code and usage are available in the project github repository.

Supplemental Tables

Method	Disentangled Entities	Supervised input	Representational limitations
inVAE [125]	Two groups of latent factors	Batch / [biological and technical factors]	-
MIDAS [124]		Batch	
ContrastiveVI [126], SC-VAE [127]		Perturbations	
BIOLORD [130]	Multiple groups of latent factors	Variations of interest for each group	
scDisInFact [129]		Variations of interest for each group / Perturbations	
sVAE / sVAE+ [40]	Each latent factor	Perturbations	-
SAMS-VAE [41]		Perturbations	
sVAE-ligr [42]		Perturbations / Variations of interest	
Decipher [45]	Two level-2 latent factors	Variables of interest	-
pmVAE [44]	Each latent factor	Pathways	Nonlinear with pathway masks
expiMap [43]		Pathways	Linear decoder with pathway masks and sparsity
Spectra [26]		Gene-gene knowledge graph and Cell-types	Linear decomposition and sparsity
MOFA / MOFA+ [38]	Each latent factor	No	Linear decomposition and sparsity
BAE [39]			Sparse linear encoder
β-TCVAE, MICHIGAN [46, 47]			Unconstrained
DRVI (Ours)			Additive decoders

Supplemental Table 1 | Comparison of various methods for disentangling latent factors in single-cell omics.

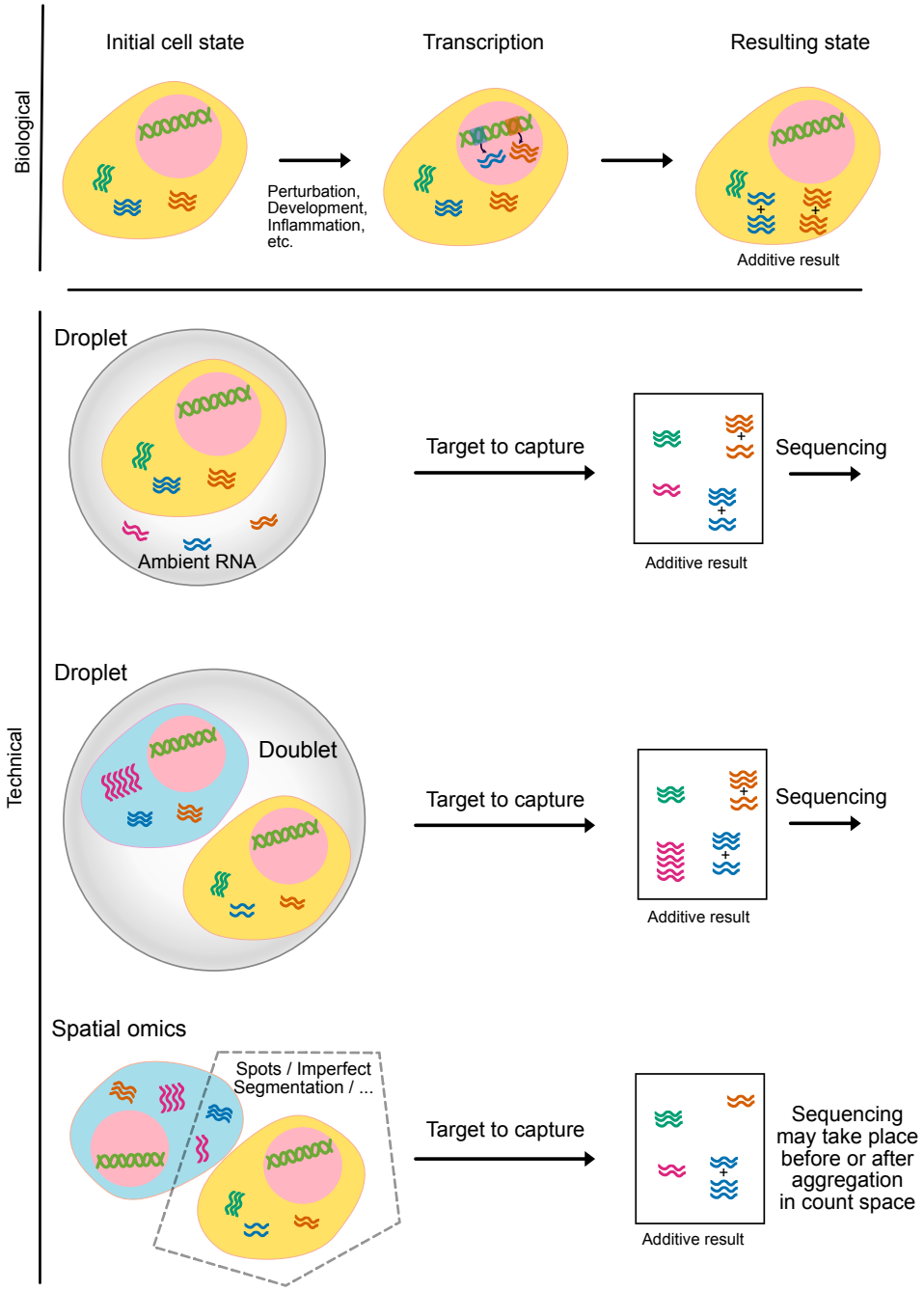
Dataset Name	# Cells	# Batches	# Cell Types	# HV features
Developmental Pancreas [4]	3,696	-	15	2,000
Immune PBMC [53]	32,484	9	16	2,000
Retina Organoid [10]	98,348	41	21	2,001
CRISPR screen [5]	104,339	-	-	5,000
Daniocell [65]	489,686	-	156	2,000
HLCA-core [7]	584,944	166	61	1,996
PBMC [64]	647,366	3	51	2,000
NeurIPS 2021 - ATAC [91]	69,247	13	22	14,865

Supplemental Table 2 | Summary of Datasets.

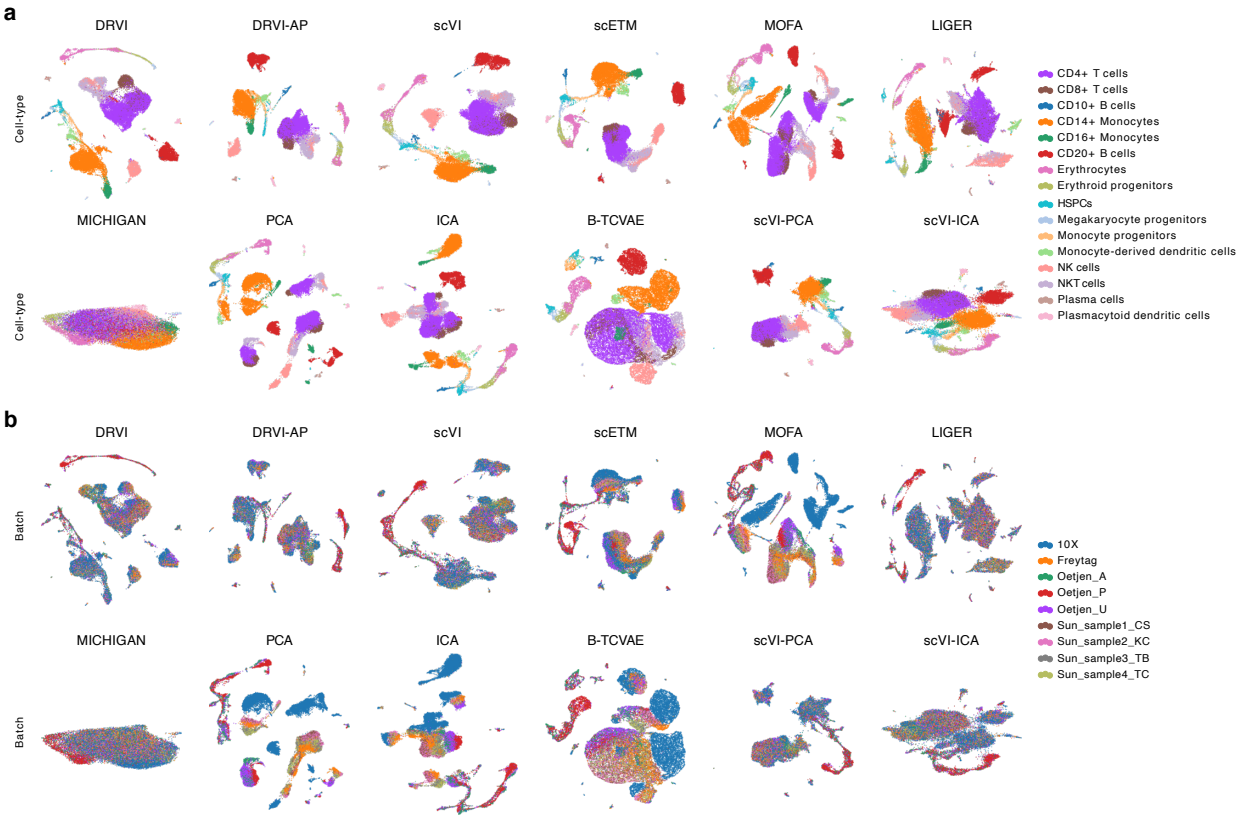
Dataset Name	# Latent for all methods	Hidden Layers for DRVI / scVI	Sharing decoder subnetworks	Training epochs
Developmental Pancreas	32	128	all layers	1,000
Immune PBMC	32	128	all layers	400
Retina Organoid	32	128	all layers	400
CRISPR screen	64	512, 512, 512	all layers	400
Daniocell	64	256, 256	last layer	400
HLCA-core	64	256, 256	all layers	400
PBMC	64	256, 256	all layers	400
NeurIPS 2021 - ATAC	64	256, 256	all layers	400

Supplemental Table 3 | Summary of Model Parameters.

Supplemental Figures



Supplemental Figure 1: **Plausibility of DRVI's additive assumption in scRNA-seq data.** The additivity of independent biological processes makes sense from both biological and technical perspectives. Newly transcribed genes (for example, during developmental bifurcations, induced perturbations, stress, etc.) add up within the cell and therefore add up in count space when sequenced. In addition, technical effects such as doublet cells, ambient background counts, and residual counts from neighboring cells due to imperfect segmentation in spatial omics show additive behaviors.

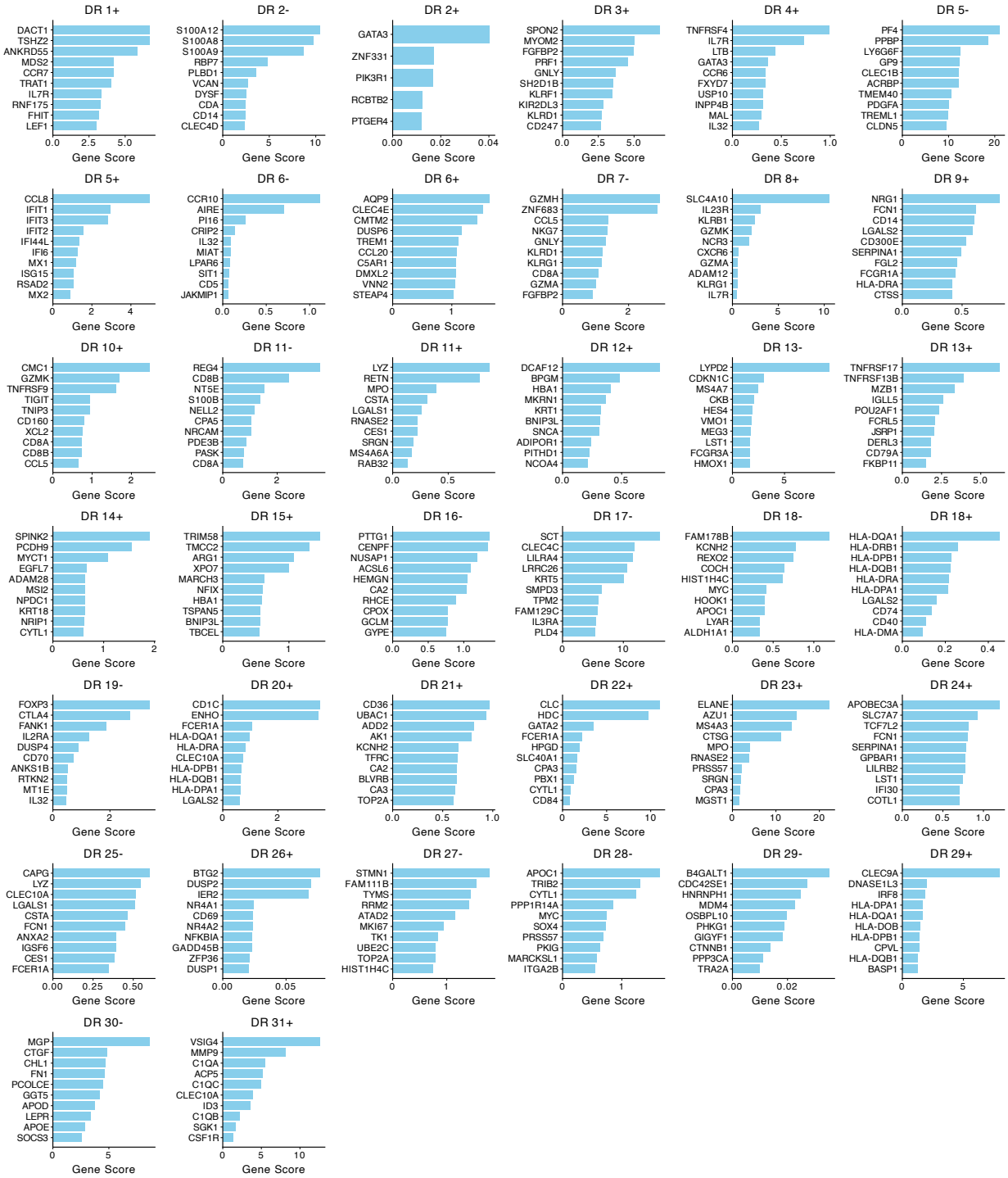


c

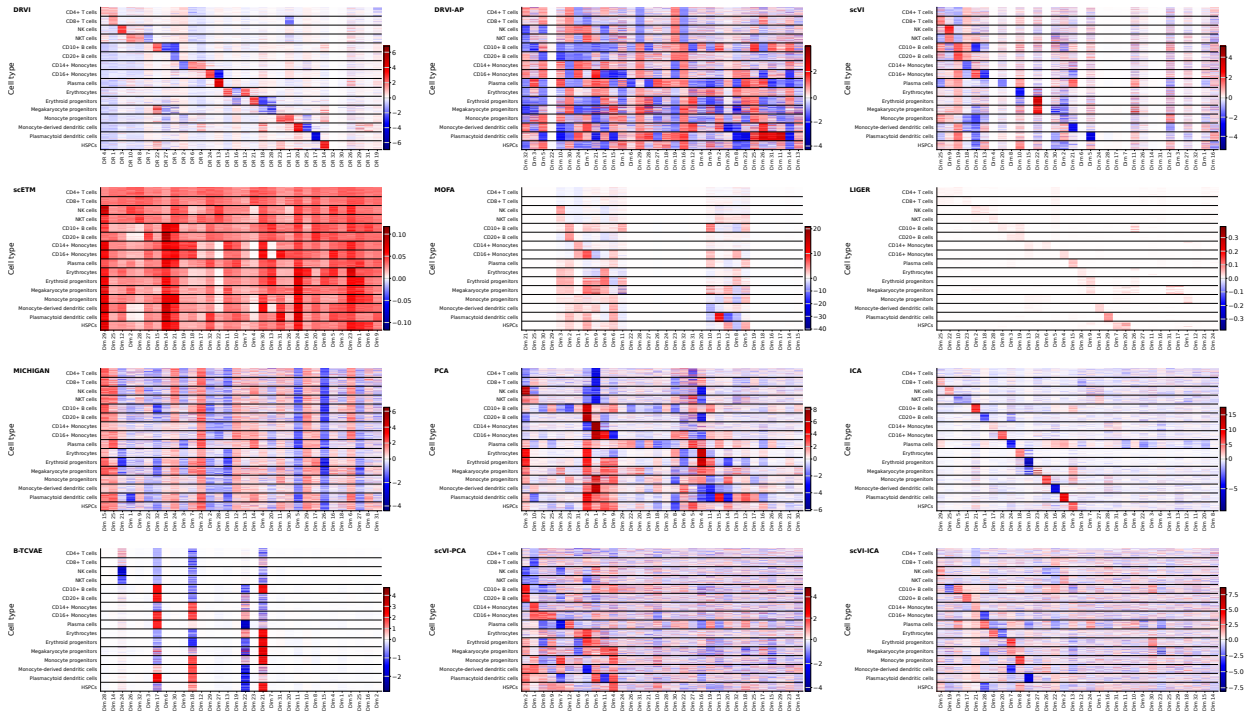
Method	Bio conservation					Batch correction					Aggregate score		
	Isolated labels	KMeans NMI	KMeans ARI	Silhouette label	cLISI	Silhouette batch	iLISI	KBET	Graph connectivity	PCR comparison	Batch correction	Bio conservation	Total
LIGER	0.55	0.67	0.52	0.59	1.00	0.86	0.33	0.59	0.78	0.83	0.68	0.67	0.67
DRVI-AP	0.63	0.74	0.53	0.58	1.00	0.91	0.26	0.23	0.87	0.78	0.61	0.70	0.66
DRVI	0.59	0.71	0.50	0.61	1.00	0.88	0.22	0.32	0.84	0.81	0.61	0.68	0.65
scVI-PCA	0.54	0.67	0.42	0.52	0.99	0.95	0.35	0.30	0.87	0.87	0.67	0.63	0.64
scVI	0.59	0.66	0.38	0.54	1.00	0.92	0.30	0.31	0.91	0.81	0.65	0.63	0.64
scVI-ICA	0.53	0.60	0.38	0.51	0.98	0.95	0.39	0.34	0.84	0.92	0.69	0.60	0.64
scETM	0.64	0.69	0.42	0.60	1.00	0.89	0.10	0.19	0.88	0.31	0.48	0.67	0.59
ICA	0.66	0.70	0.42	0.54	1.00	0.89	0.07	0.11	0.90	0.20	0.43	0.66	0.57
MOFA	0.64	0.67	0.45	0.59	1.00	0.82	0.08	0.18	0.87	0.00	0.39	0.67	0.56
PCA	0.66	0.67	0.42	0.60	1.00	0.86	0.04	0.10	0.85	0.00	0.37	0.67	0.55
B-TCVAE	0.68	0.63	0.32	0.58	1.00	0.84	0.11	0.20	0.92	0.00	0.41	0.64	0.55
MICHIGAN	0.52	0.28	0.17	0.46	0.91	0.95	0.30	0.68	0.49	0.85	0.65	0.47	0.54

Supplemental Figure 2: **UMAPs and integration results for benchmarked methods on the immune dataset. a,b,** UMAPs of the immune dataset integrated by DRVI and benchmarked methods colored by cell types and batches. **c,** Integratin results based on the scIB metrics for DRVI and benchmarked methods.

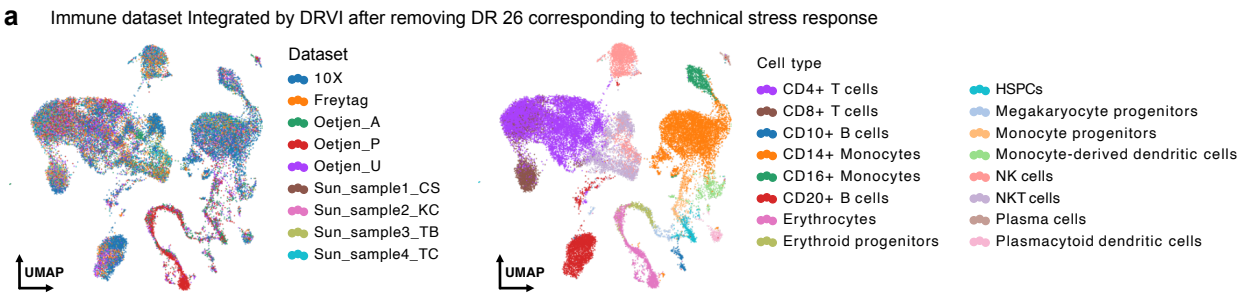
2155



Supplemental Figure 3: **Interpretability of DRVI latent factors in immune dataset.** The top relevant genes and their scores are plotted for each latent factor of DRVI trained on the immune dataset.

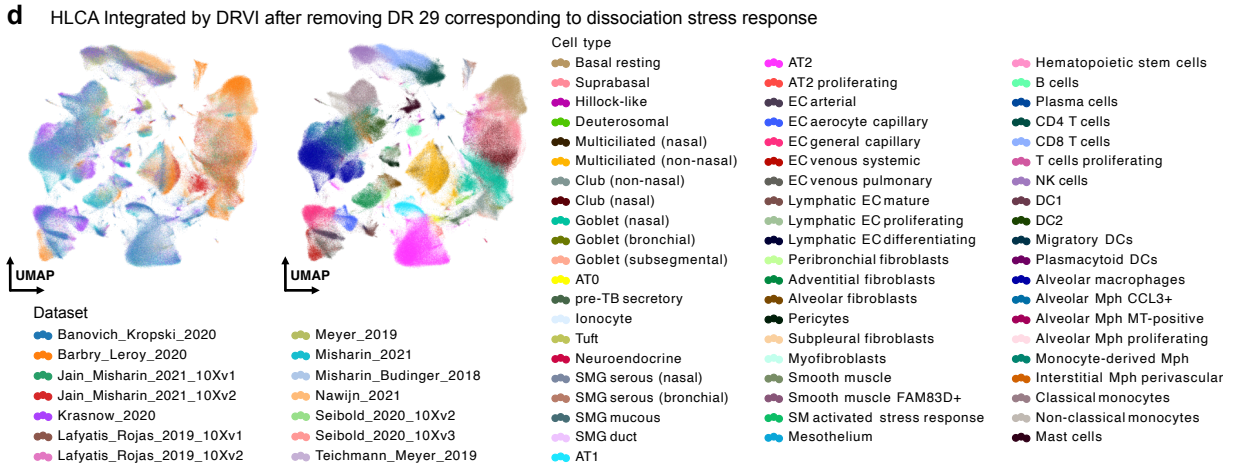


Supplemental Figure 4: **Latent space visualization for benchmarked methods in the immune dataset.** For each method, a heatmap visualizing latent representation activity across different cell types is provided. Cells were subsampled to ensure a balanced number of cells from each cell type. Dimensions are reordered using a shared algorithm to maximize the diagonal appearance.



c

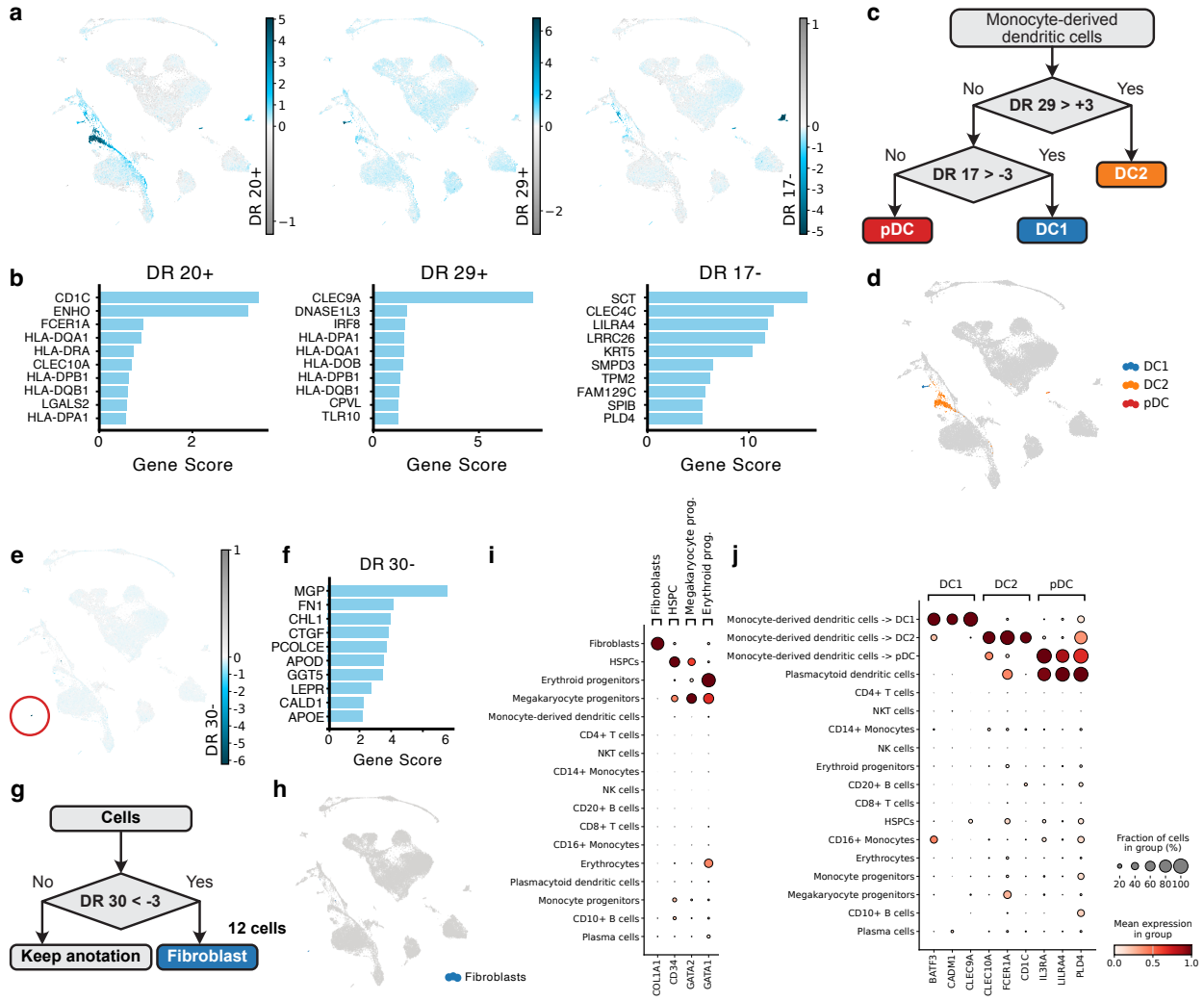
Method	Bio conservation					Batch correction					Aggregate score		
	Isolated labels	KMeans NMI	KMeans ARI	Silhouette label	cLISI	Silhouette batch	iLISI	KBET	Graph connectivity	PCR comparison	Batch correction	Bio conservation	Total
DRVI-AP	0.63	0.74	0.53	0.58	1.00	0.91	0.26	0.23	0.87	0.78	0.61	0.70	0.66
DRVI without DR 26	0.60	0.70	0.49	0.61	1.00	0.88	0.23	0.35	0.85	0.82	0.63	0.68	0.66
DRVI	0.59	0.71	0.50	0.61	1.00	0.88	0.22	0.32	0.84	0.81	0.61 (+2)	0.68	0.65 (+1)
scVI	0.59	0.66	0.38	0.54	1.00	0.92	0.30	0.31	0.91	0.81	0.65	0.63	0.64



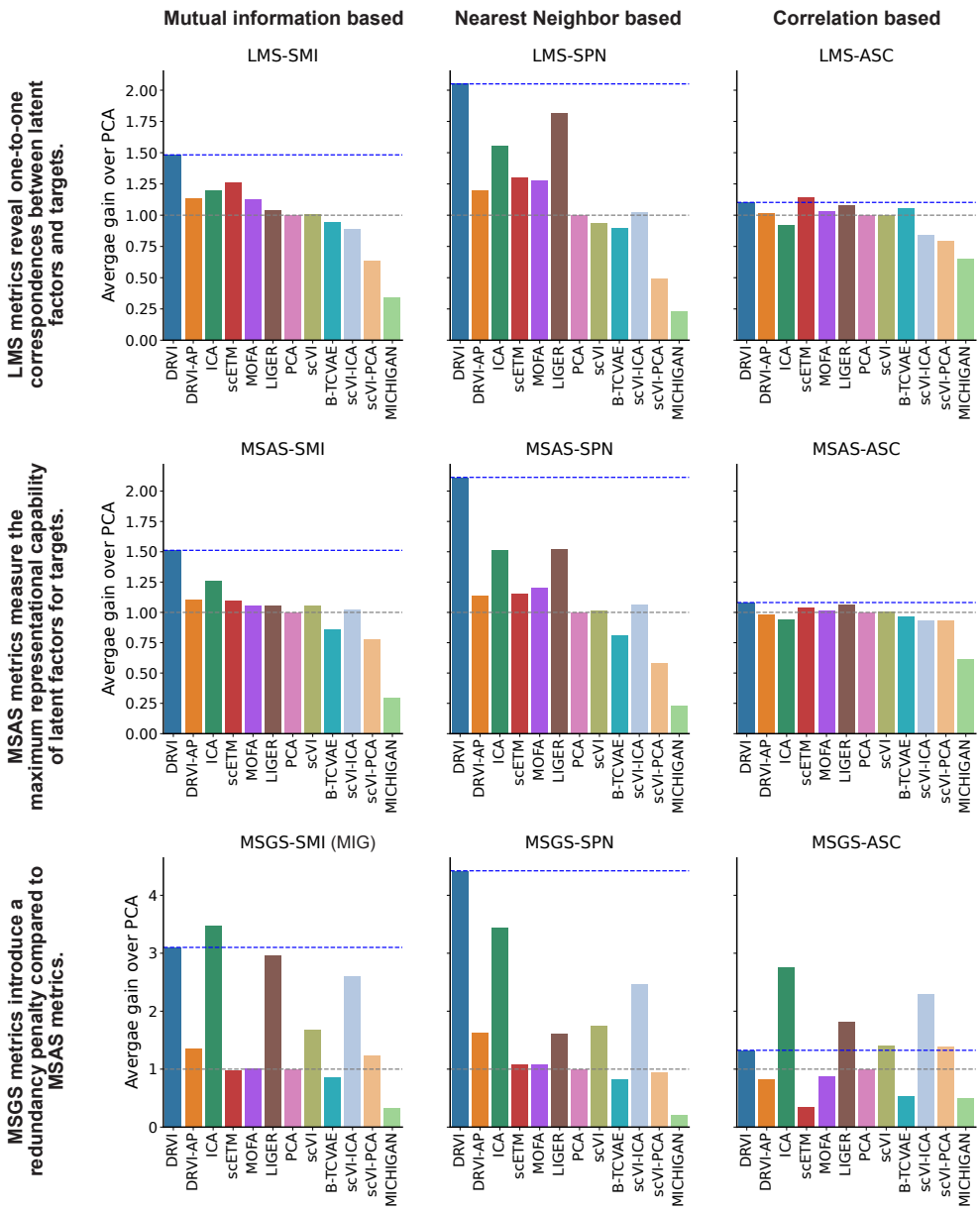
f

Method	Bio conservation					Batch 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
DRVI without DR 29	0.69	0.71	0.41	0.56	1.00	0.90	0.06	0.32	0.85	0.80	0.59	0.68	0.64
DRVI	0.69	0.71	0.39	0.56	1.00	0.90	0.05	0.29	0.86	0.80	0.58 (+1)	0.67 (+1)	0.63 (+1)
scANVI (HLCA original)	0.68	0.74	0.42	0.55	1.00	0.90	0.02	0.23	0.95	0.66	0.55	0.68	0.63
DRVI-AP	0.62	0.73	0.42	0.54	1.00	0.92	0.04	0.24	0.90	0.79	0.58	0.66	0.63
scVI	0.64	0.68	0.36	0.53	1.00	0.91	0.06	0.28	0.91	0.83	0.60	0.64	0.63

Supplemental Figure 5: **The effect of removing technical stress response dimensions. a,b,** UMAP representations of the integrated immune dataset, colored by dataset and cell type, following DRVI integration and removal of stress response dimension (DR 26). **c,** Integration quality assessment using scIB metrics for DRVI before and after DR 26 removal. Results are compared to DRVI-AP and scVI, two other top-performing models. **d,e,** UMAP representations of the integrated HLCA, colored by dataset and finest-level annotations, following DRVI integration after removal of stress response dimension (DR 29). **f,** Integration quality assessment using scIB metrics for DRVI before and after DR 29 removal. Results are compared to DRVI-AP and scVI, two other top-performing models, and scANVI provided by the HLCA. scANVI is the best-performing integration method provided by the HLCA paper, included for readers' information. However, it is not fairly comparable due to its use of supervised information and a different number of latent dimensions.

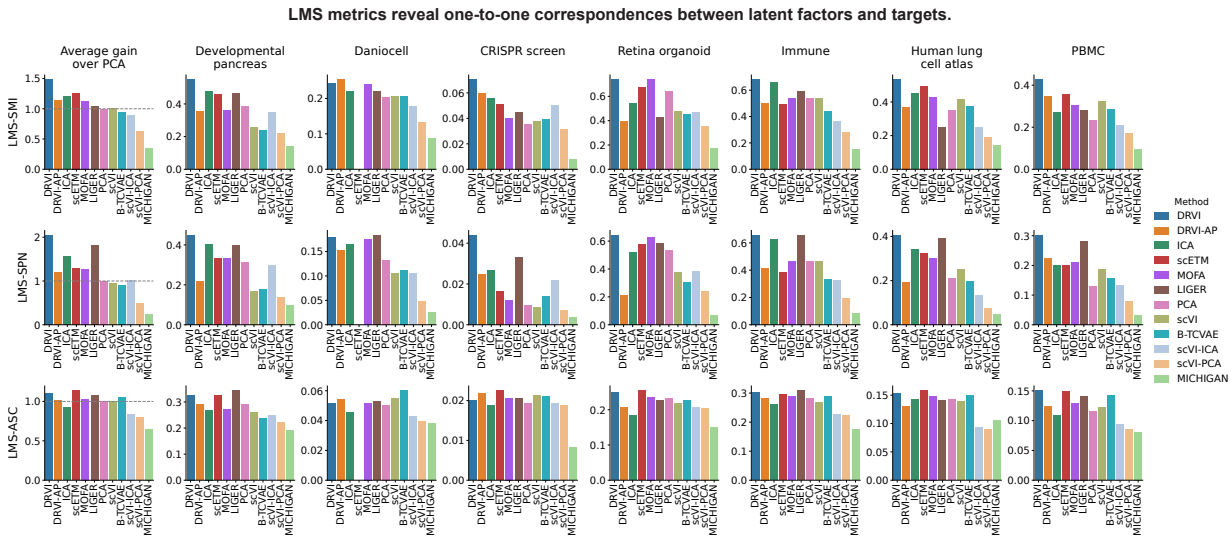


Supplemental Figure 6: **Annotation refinement and correction using DRVI in the immune dataset.** **a,** UMAP visualization of latent dimensions identifying DC subtypes. **b,** Top genes associated with DC subtype dimension. **c,** Flowchart for refining DC subtype annotations. **d,** UMAP visualization of refined DC subtype annotations. **e,** UMAP visualization highlighting the latent dimension identifying Fibroblast cells. The population is encircled in red. **f,** Top genes associated with the Fibroblast dimension. **g,** Flowchart for annotating Fibroblast cells. **h,** UMAP visualization of annotated Fibroblast cells. **i,** Dotplot validating new annotations for Fibroblasts, HSPCs, Megakaryocyte progenitors, and Erythroid progenitors using a representative curated marker gene for each. **j,** Dotplot of marker genes for DC subtypes for validation of DC subtype improvement. For validation, pDCs are intentionally labeled differently from the original 'plasmacytoid dendritic cells' annotations. DC, dendritic cells; HSPC, Hematopoietic Stem and Progenitor Cells; pDCs, plasmacytoid dendritic cells.

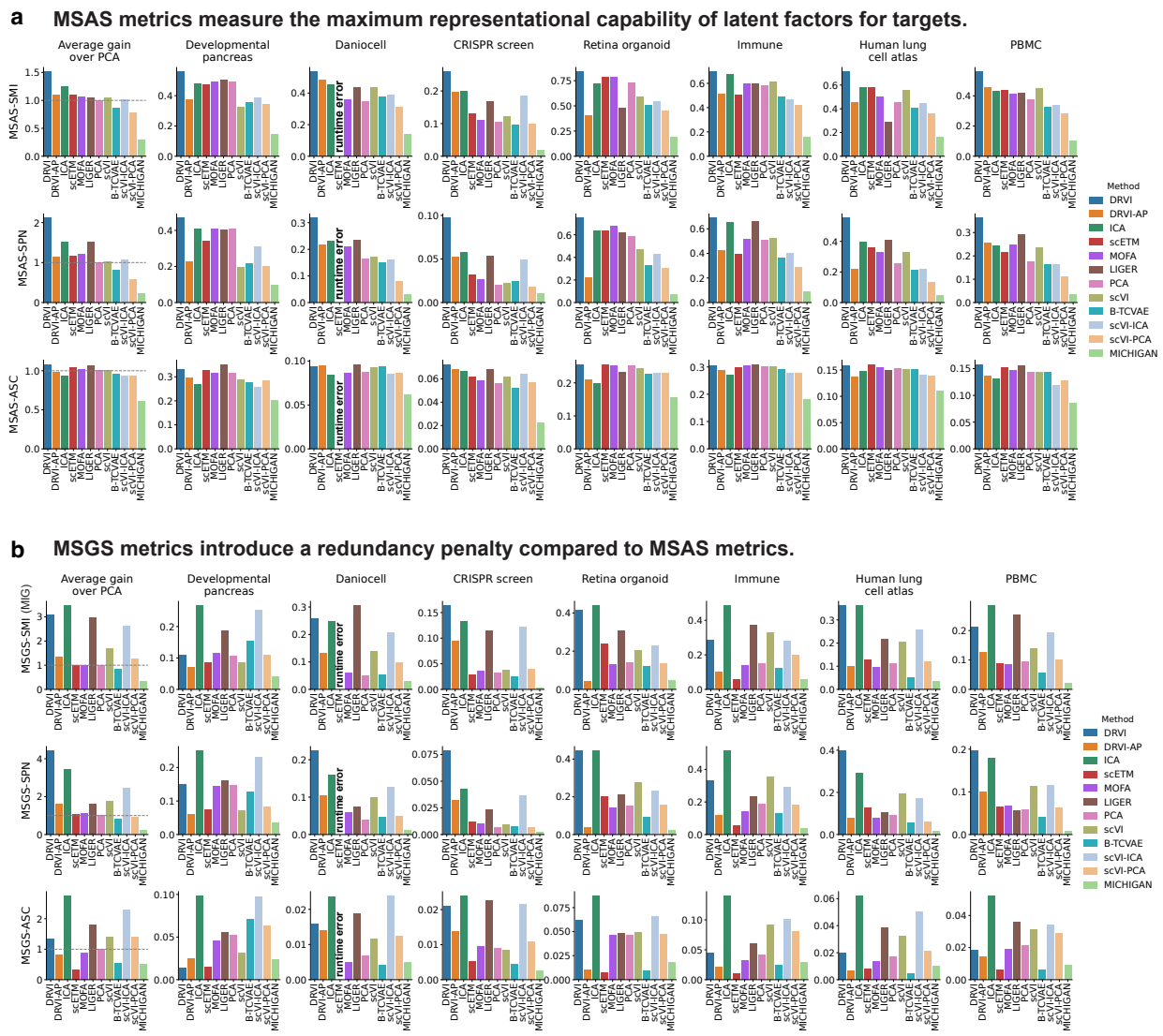


Supplemental Figure 7: **Summary of all metrics.** This plot displays the average gain achieved by each of the benchmarked methods over PCA. The gains are shown for every combination of the SMI, SPN, and ASC similarity functions with the LMS, MSAS, and MSGS aggregation functions. ASC, absolute Spearman correlations; LMS, latent matching score; MSAS, Most Similar Averaging Score; MSGS, Most Similar Gap Score; SMI, scaled mutual information; SPN, same-process neighbors.

2160

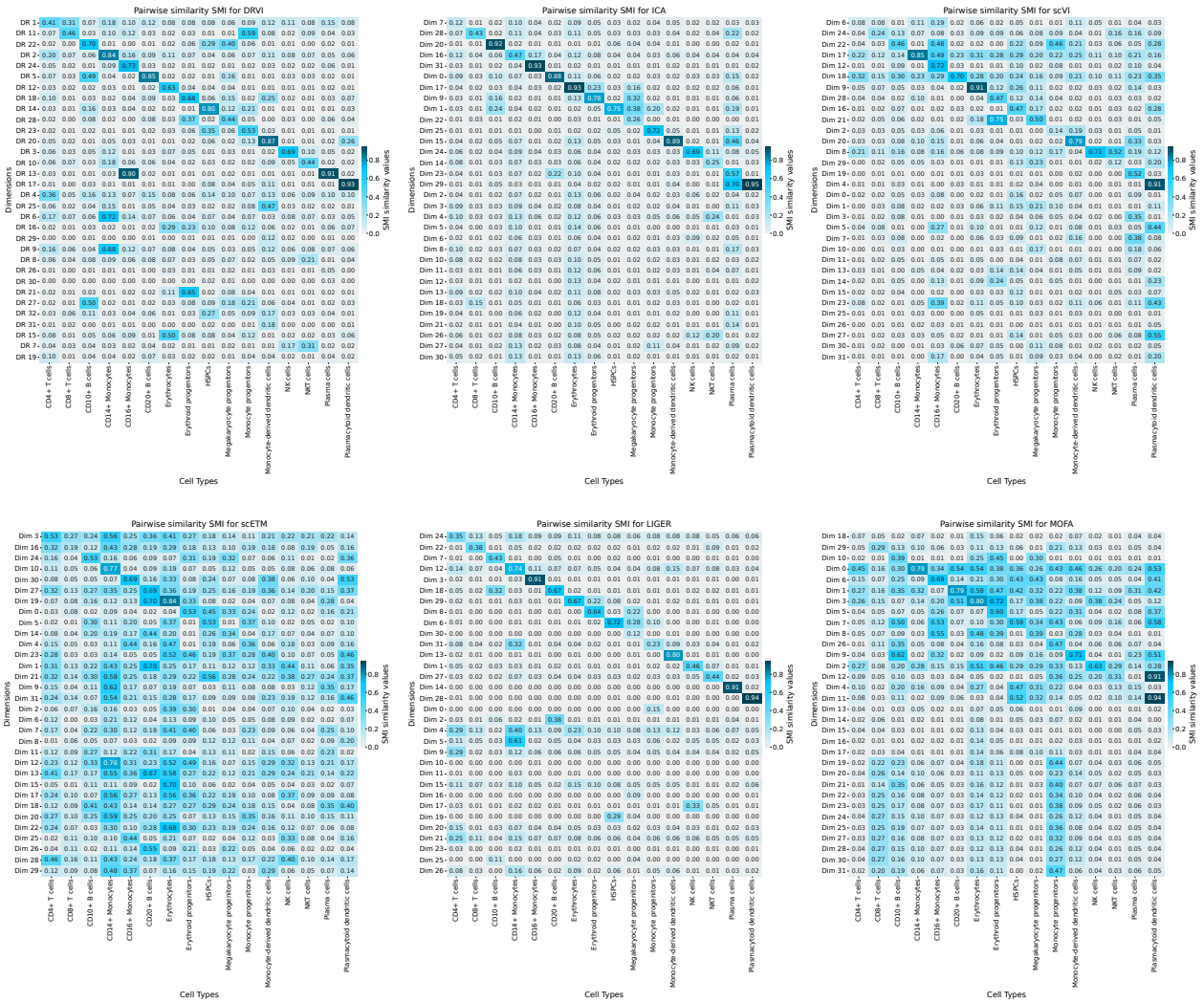


Supplemental Figure 8: **Results of the primary disentanglement metric family.** The results of the LMS metrics, as primary metric family, over different datasets are provided. LMS measures the ability of the model to capture biological processes in individual latent dimensions by matching latent dimensions and ground-truth biological processes. ASC, absolute Spearman correlations; LMS, latent matching score; SMI, scaled mutual information; SPN, same-process neighbors.



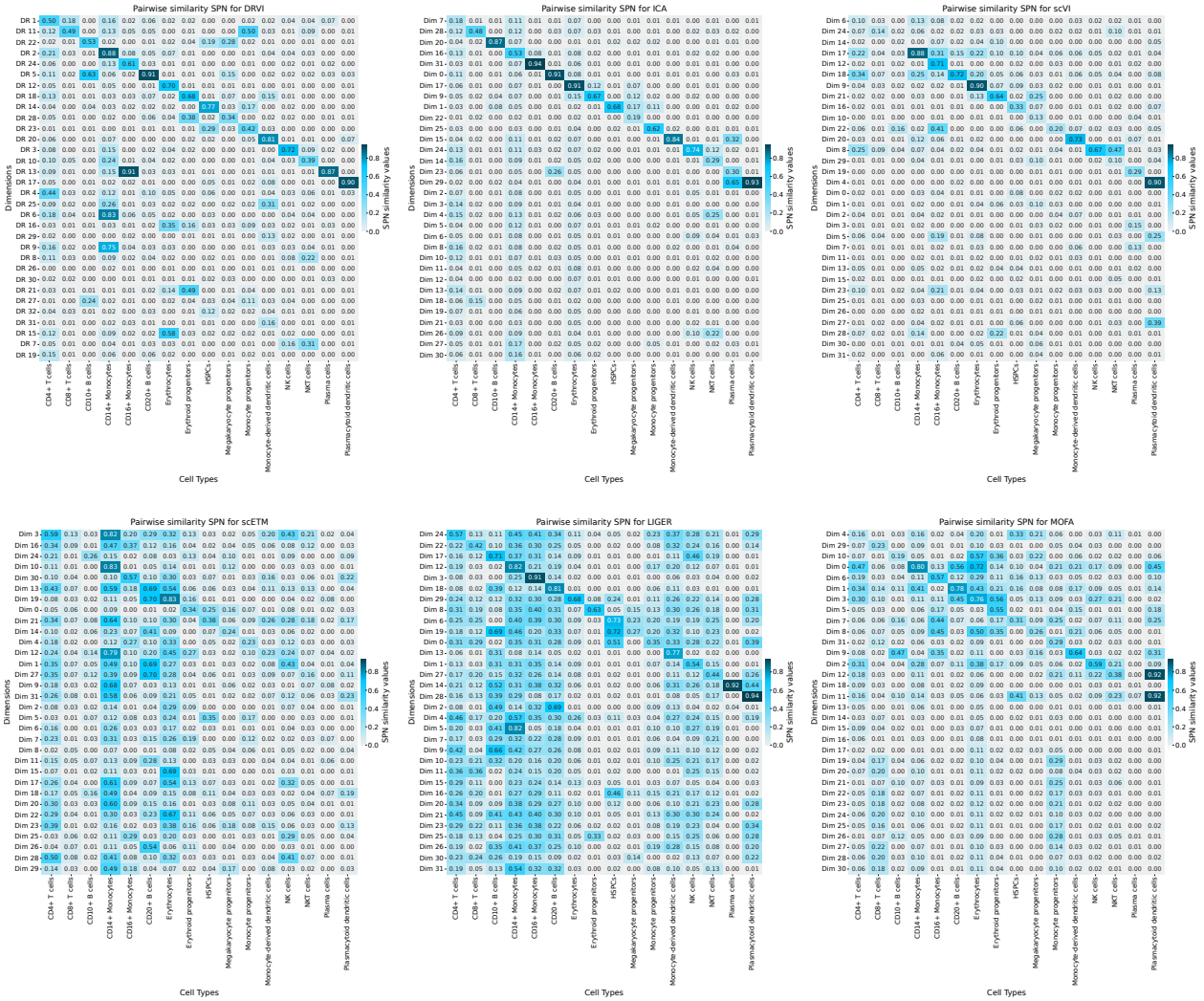
Supplemental Figure 9: **Results of the secondary disentanglement metrics. a,b,** The results of the MSAS (a) and MSGS (b) as secondary metrics over different datasets are provided. MSAS shows the capability of the model to capture biological processes in individual latent dimensions regardless of the redundancy. MSGS measures the amount of captured biological information that is specific to a single (probably overlapping) dimension by penalizing redundancy. Unlike LMS, as the main metric, neither MSAS nor MSGS penalizes overlapping processes in a single dimension. The MSGS-SMI metric is known as the Mutual Information Gap (MIG) in the field of disentangled representation learning [66]. ASC, absolute Spearman correlations; LMS, latent matching score; MSAS, Most Similar Averaging Score; MSGS, Most Similar Gap Score; SMI, scaled mutual information; SPN, same-process neighbors; MIG, mutual information gap.

2162



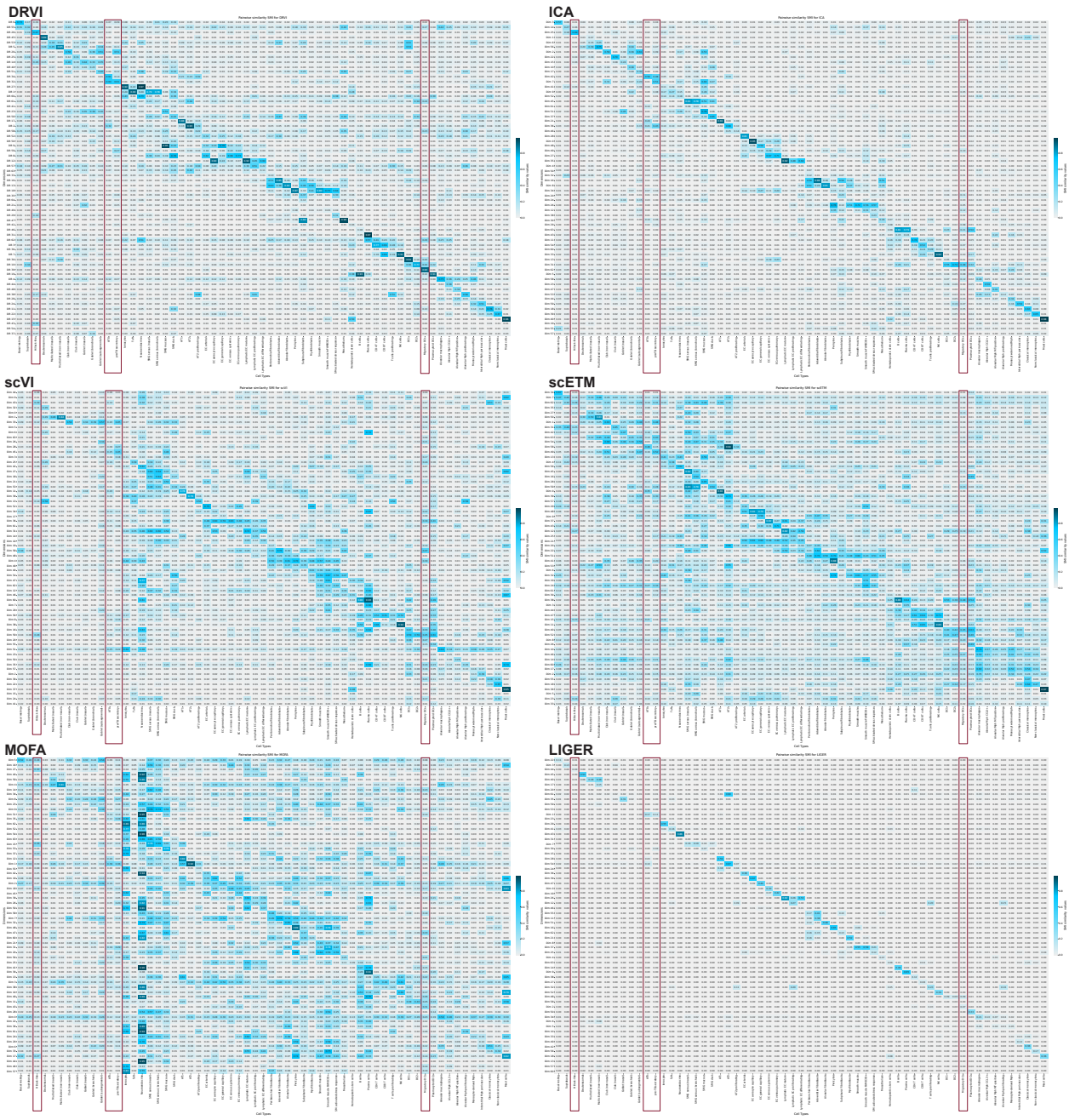
Supplemental Figure 10: **Unaggregated Scaled Mutual Information (SMI) scores for the Immune dataset.** For the top-picked methods, a heatmap visualizing SMI scores between cell types and latent space dimensions is provided. Latent space dimensions are reordered using a shared algorithm to maximize a diagonal appearance.

2163

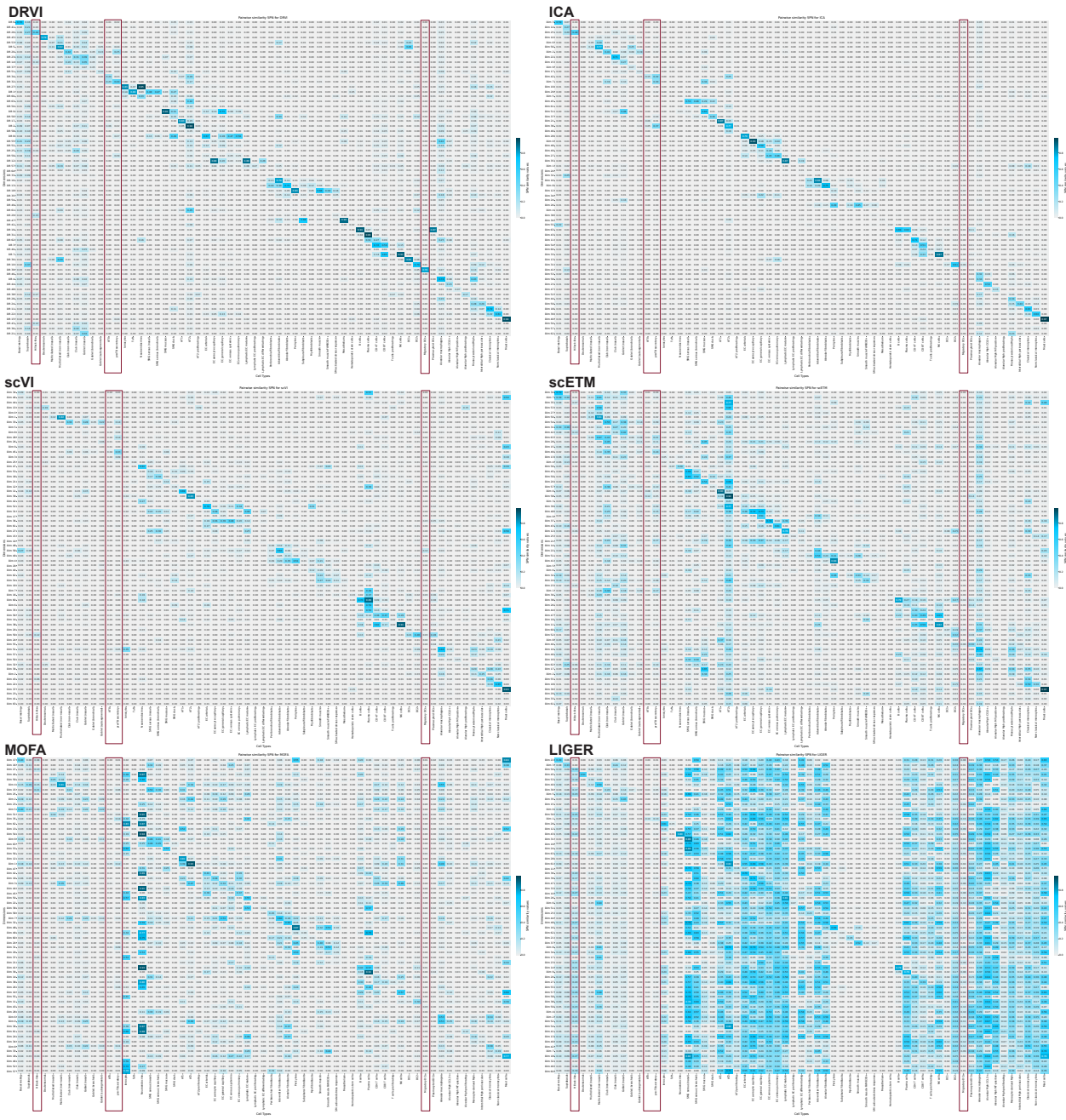


Supplemental Figure 11: **Unaggregated Same-Process Neighbors (SPN) scores for the Immune dataset.** For the top-picked methods, a heatmap visualizing SPN scores between cell types and latent space dimensions is provided. Latent space dimensions are reordered using a shared algorithm to maximize a diagonal appearance.

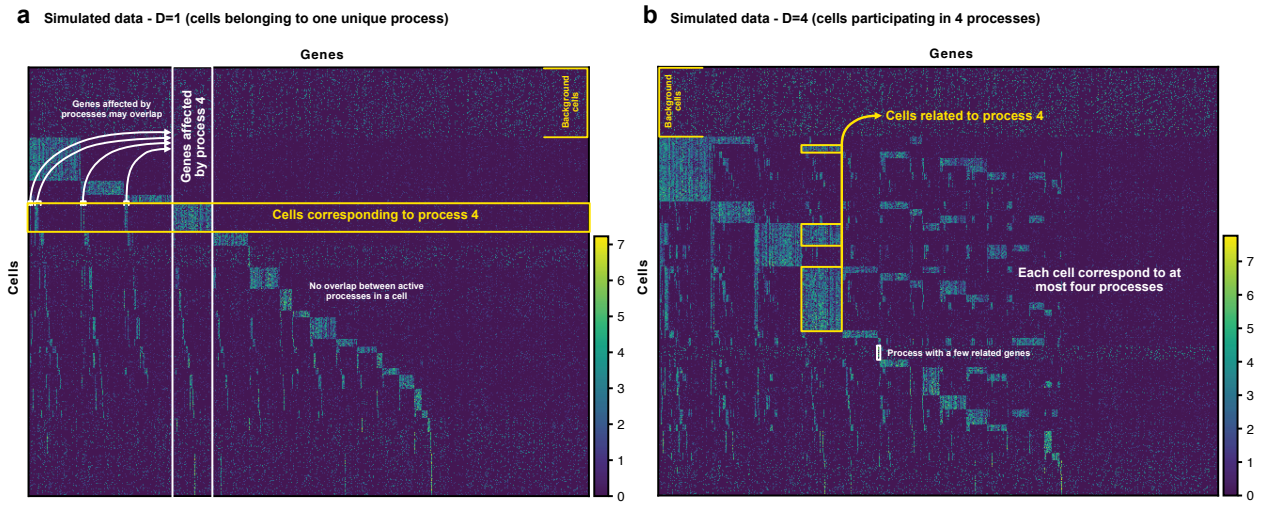
2164



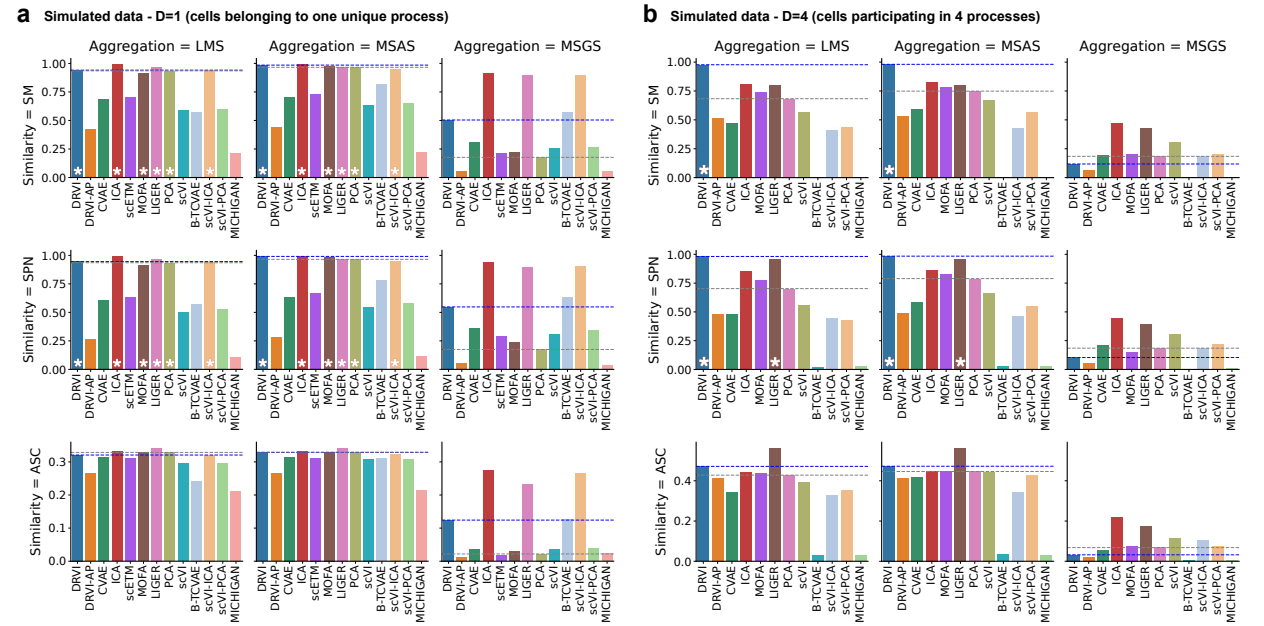
Supplemental Figure 12: **Unaggregated Scaled Mutual Information (SMI) scores for the HLCA.** For the top-picked methods, a heatmap visualizing SMI scores between cell types and latent space dimensions is provided. Latent space dimensions are reordered using a shared algorithm to maximize a diagonal appearance. Rare cell types mentioned in the paper are highlighted with red rectangles.



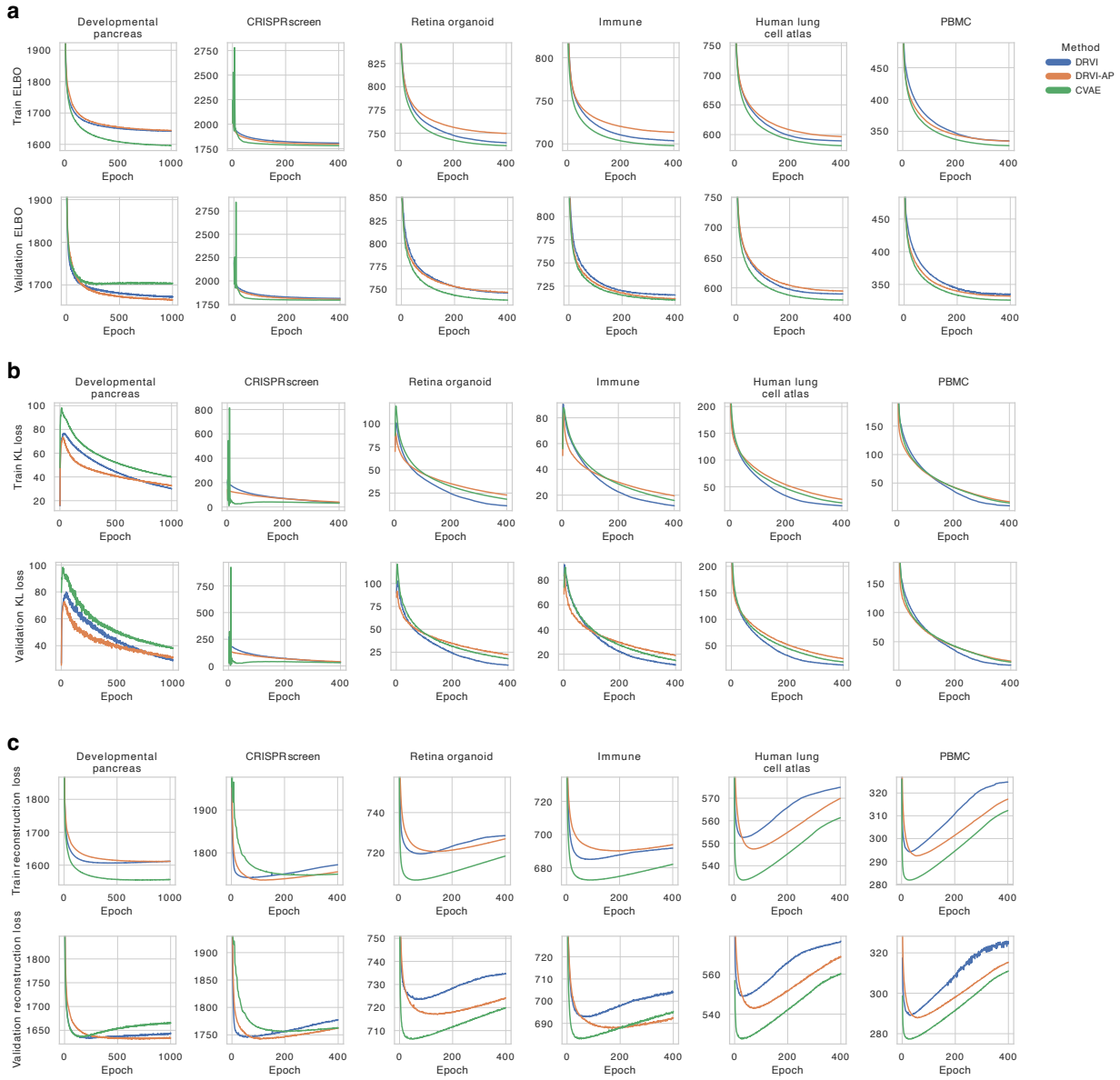
Supplemental Figure 13: **Unaggregated Same-Process Neighbors (SPN) scores for the HLCA.** For the top-picked methods, a heatmap visualizing SPN scores between cell types and latent space dimensions is provided. Latent space dimensions are reordered using a shared algorithm to maximize a diagonal appearance. Rare cell types mentioned in the paper are highlighted with red rectangles.



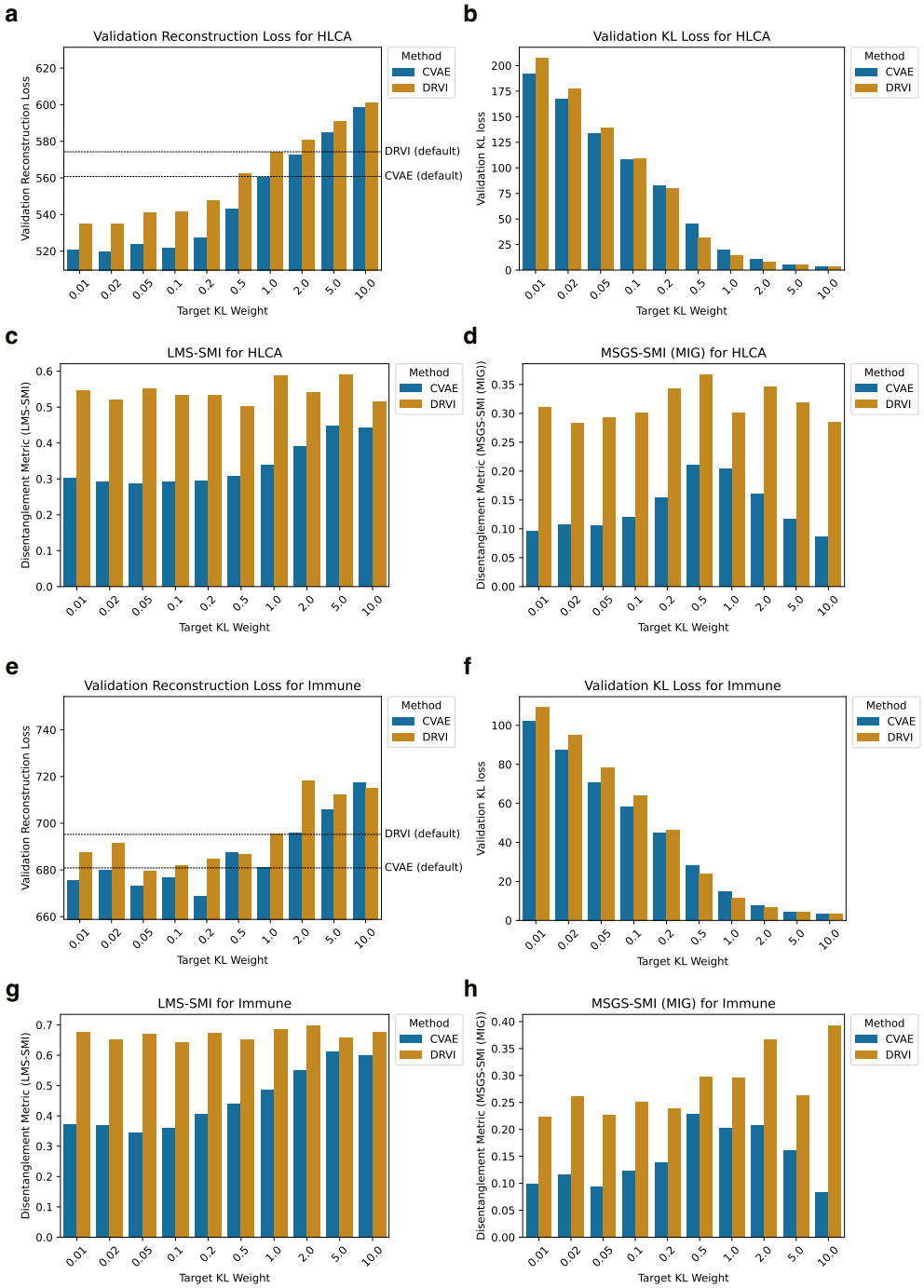
Supplemental Figure 14: **Heatmap of simulated datasets with known ground truth.** The first simulated dataset (**a**) assigns non-background cells to a single process, while genes can be influenced by multiple processes. The second simulated dataset (**b**) allows cells to participate in up to four processes, introducing controlled overlap and compositional complexity. Cells with similar ground truth are grouped, as are genes affected by the same programs. Both cell and gene groups are then arranged to maximize their diagonal presentation.



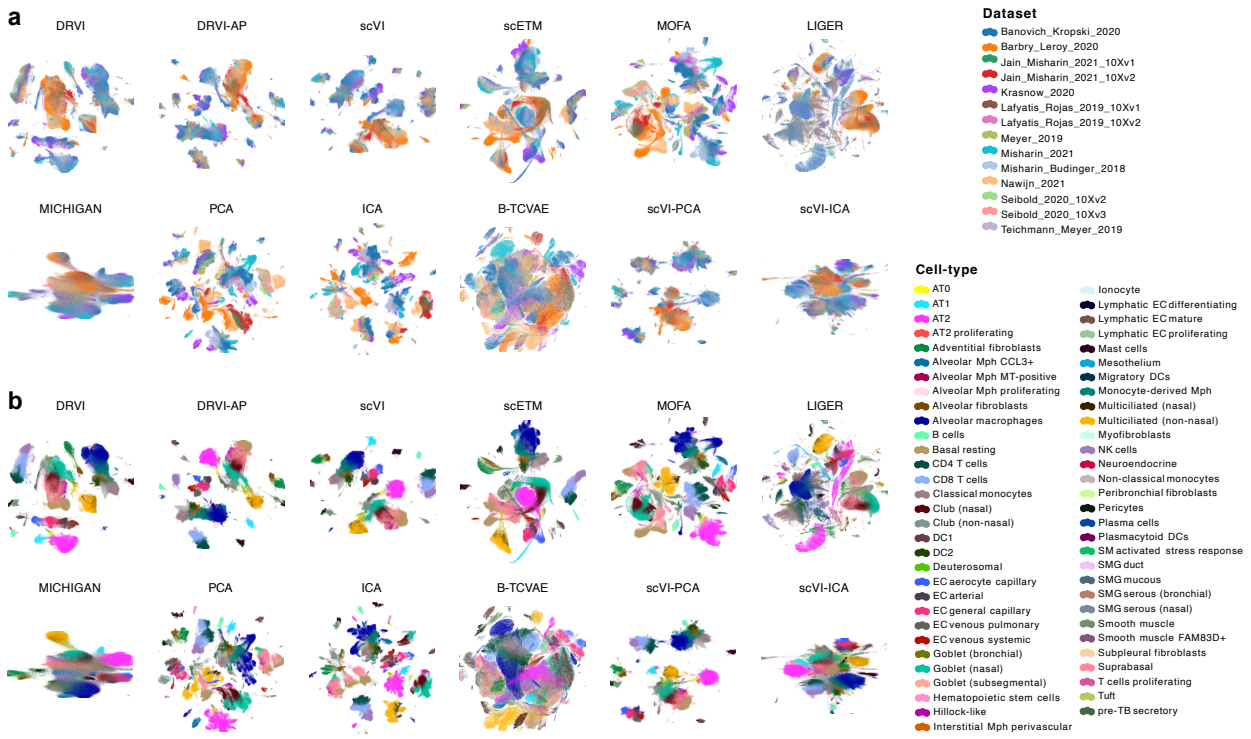
Supplemental Figure 15: **Evaluation results for simulated datasets with known ground truth. a,b,** We calculated metric values for each benchmarked model across the simulated datasets. These metrics are subsequently presented for all possible combinations formed by the SMI, SPN, and ASC similarity functions and the LMS, MSAS, and MSGS aggregation functions. In cases where cellular activities are mutually exclusive (a), DRVI, ICA, MOFA, PCA, LIGER, and scVI-ICA all perform effectively. However, when processes overlap (b), only DRVI and LIGER are capable of accurately identifying ground-truth processes. ASC, absolute Spearman correlations; LMS, latent matching score; MSAS, Most Similar Averaging Score; MSGS, Most Similar Gap Score; SMI, scaled mutual information; SPN, same-process neighbors.



Supplemental Figure 16: **The optimization of training objectives. a-c** During training on various datasets, we monitored the Evidence Lower Bound (ELBO) (a), latent KL loss (b), and negative binomial reconstruction loss (c). We compared DRVI, DRVI-AP, and a CVAE trained specifically for comparison (without latent space splitting), all trained with the same objectives; ELBO as the sum of the KL loss and log-space parametrization of negative binomial for reconstruction. In general, the constrained models (DRVI, DRVI-AP) exhibited worse reconstruction performance but achieved higher KL loss compared to the CVAE on both training and in-distribution validation sets.



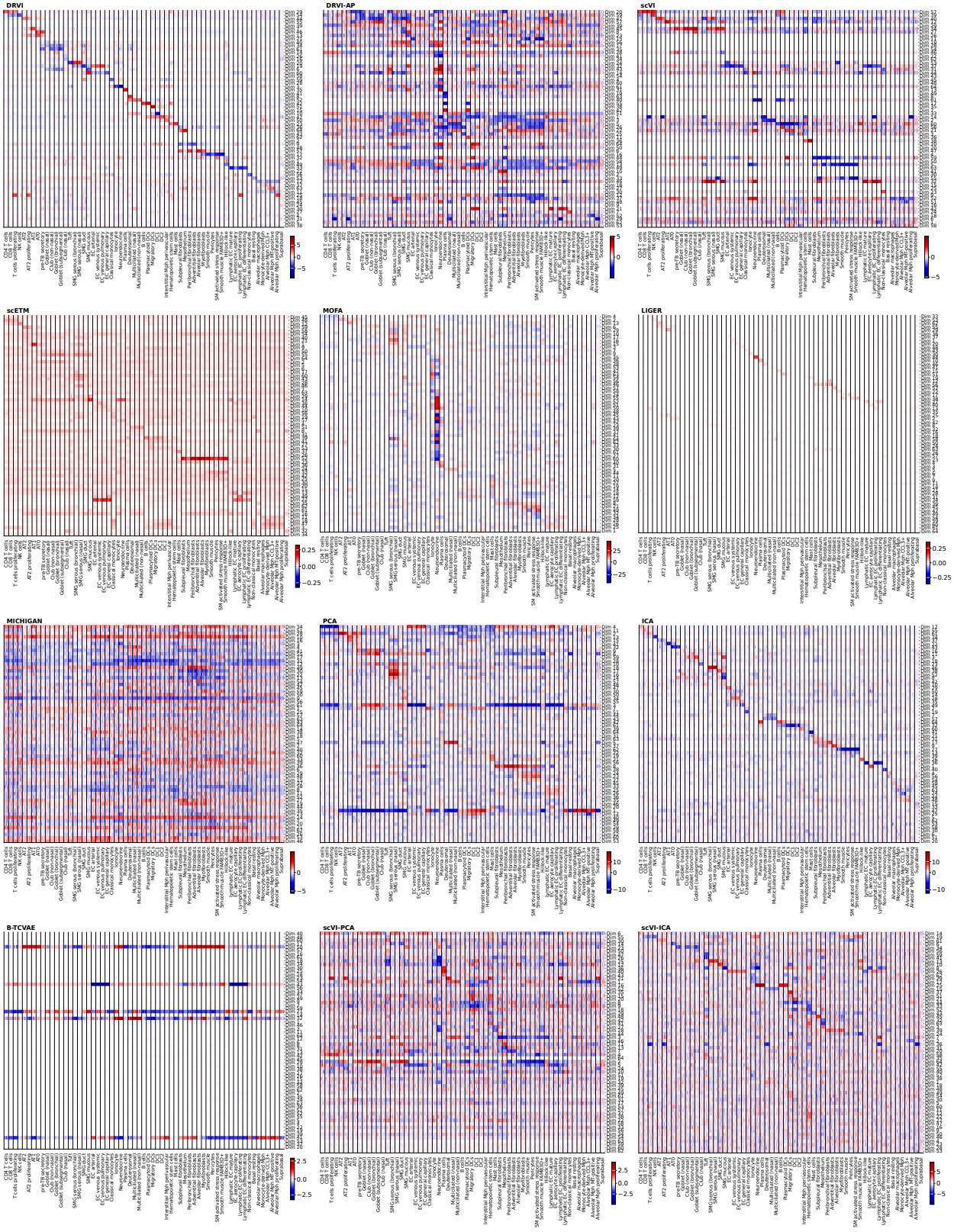
Supplemental Figure 17: **The effect of KL loss weight on reconstruction and disentanglement performance.** To investigate the impact of KL loss on reconstruction and disentanglement, we varied the target-KL weight from 0.01 to 10 (defaulting to 1.0). DRVI models and CVAE baselines (with identical likelihood objectives) were trained on both the HLCA (a-d) and Immune (e-h) datasets. As expected, increasing the target-KL weight significantly increased validation reconstruction loss (a,d) while decreasing the validation KL objective (b,f). The effect of target-KL weight on disentanglement, however, varied by model. For CVAE, this parameter significantly impacted performance: decreasing the target-KL weight results in a decline in disentanglement capability (c,g), and deviating from the default (both higher and lower) decreased independence (d,h). In contrast, DRVI showed robustness; decreasing the target-KL weight led to a very slight decline in identification capability (c,g) and a slight overall decline in independence (d,h). This shows that users requiring good reconstruction performance can safely reduce the target-KL weight without much affecting disentanglement capabilities. LMS, latent matching score; MSGS, Most Similar Gap Score; SMI, scaled mutual information; MIG, mutual information gap.



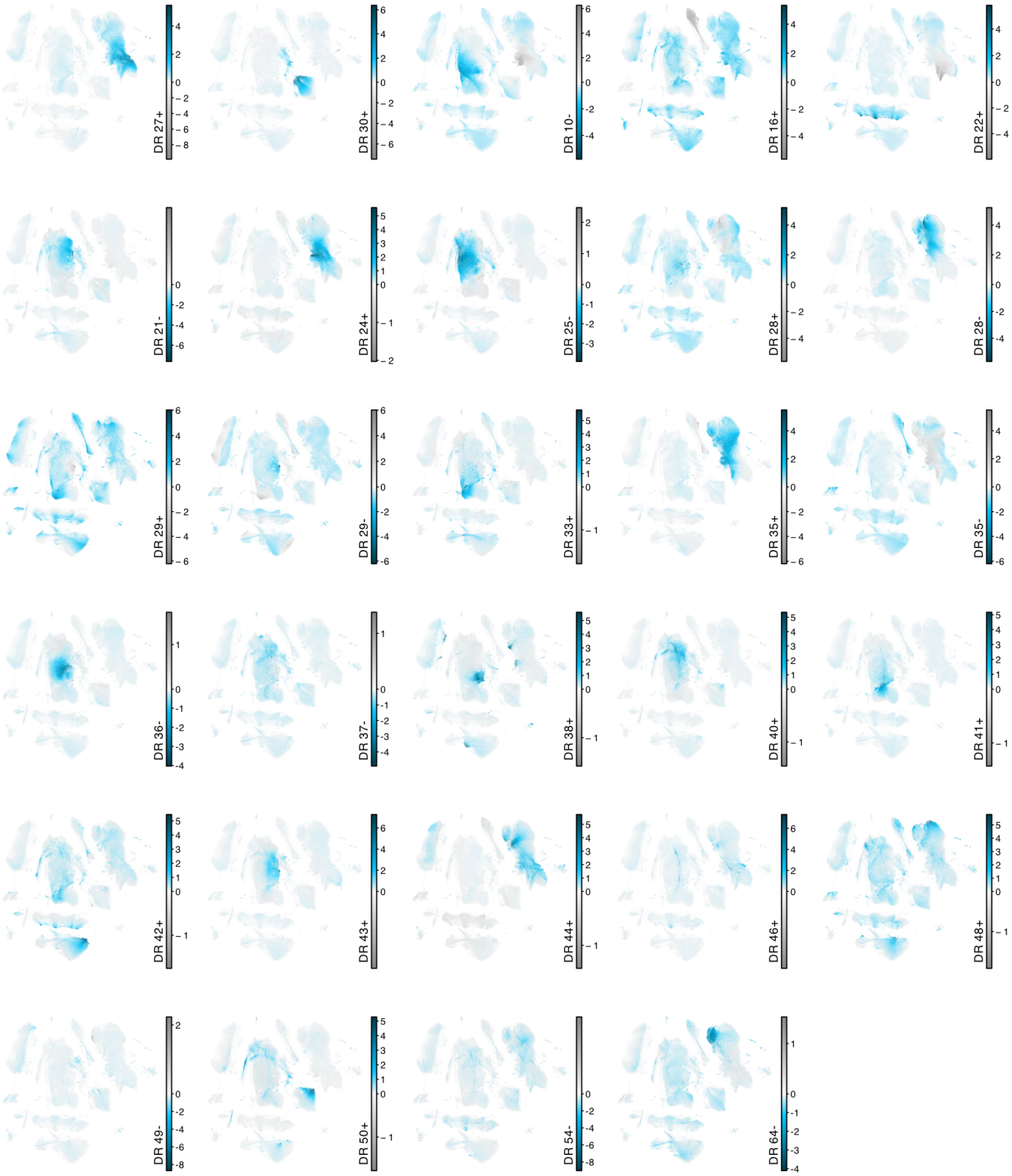
c

Method	Bio conservation: Isolated labels	KMeans NMI	KMeans ARI	Silhouette label	cLISI	Batch correction: Silhouette batch	iLISI	KBET	Graph connectivity	PCR comparison	Aggregate score: Batch correction	Bio conservation	Total
DRVI	0.69	0.71	0.39	0.56	1.00	0.90	0.05	0.29	0.86	0.80	0.58	0.67	0.63
scANVI (HLCA original)	0.68	0.74	0.42	0.55	1.00	0.90	0.02	0.23	0.95	0.66	0.55	0.68	0.63
DRVI-AP	0.62	0.73	0.42	0.54	1.00	0.92	0.04	0.24	0.90	0.79	0.58	0.66	0.63
scVI	0.64	0.68	0.36	0.53	1.00	0.91	0.06	0.28	0.91	0.83	0.60	0.64	0.63
MOFA	0.63	0.74	0.47	0.56	1.00	0.88	0.00	0.21	0.91	0.52	0.50	0.68	0.61
scETM	0.61	0.73	0.45	0.55	1.00	0.86	0.01	0.27	0.86	0.33	0.47	0.67	0.59
ICA	0.61	0.72	0.37	0.51	1.00	0.91	0.00	0.15	0.84	0.62	0.50	0.64	0.59
LIGER	0.42	0.54	0.30	0.45	0.99	0.82	0.14	0.51	0.56	0.84	0.57	0.54	0.55
PCA	0.60	0.73	0.39	0.53	1.00	0.88	0.00	0.15	0.87	0.03	0.39	0.65	0.54
MICHIGAN	0.46	0.52	0.28	0.46	0.99	0.92	0.12	0.43	0.42	0.80	0.54	0.54	0.54
B-TCVAE	0.55	0.67	0.28	0.49	1.00	0.80	0.01	0.21	0.88	0.11	0.40	0.60	0.52

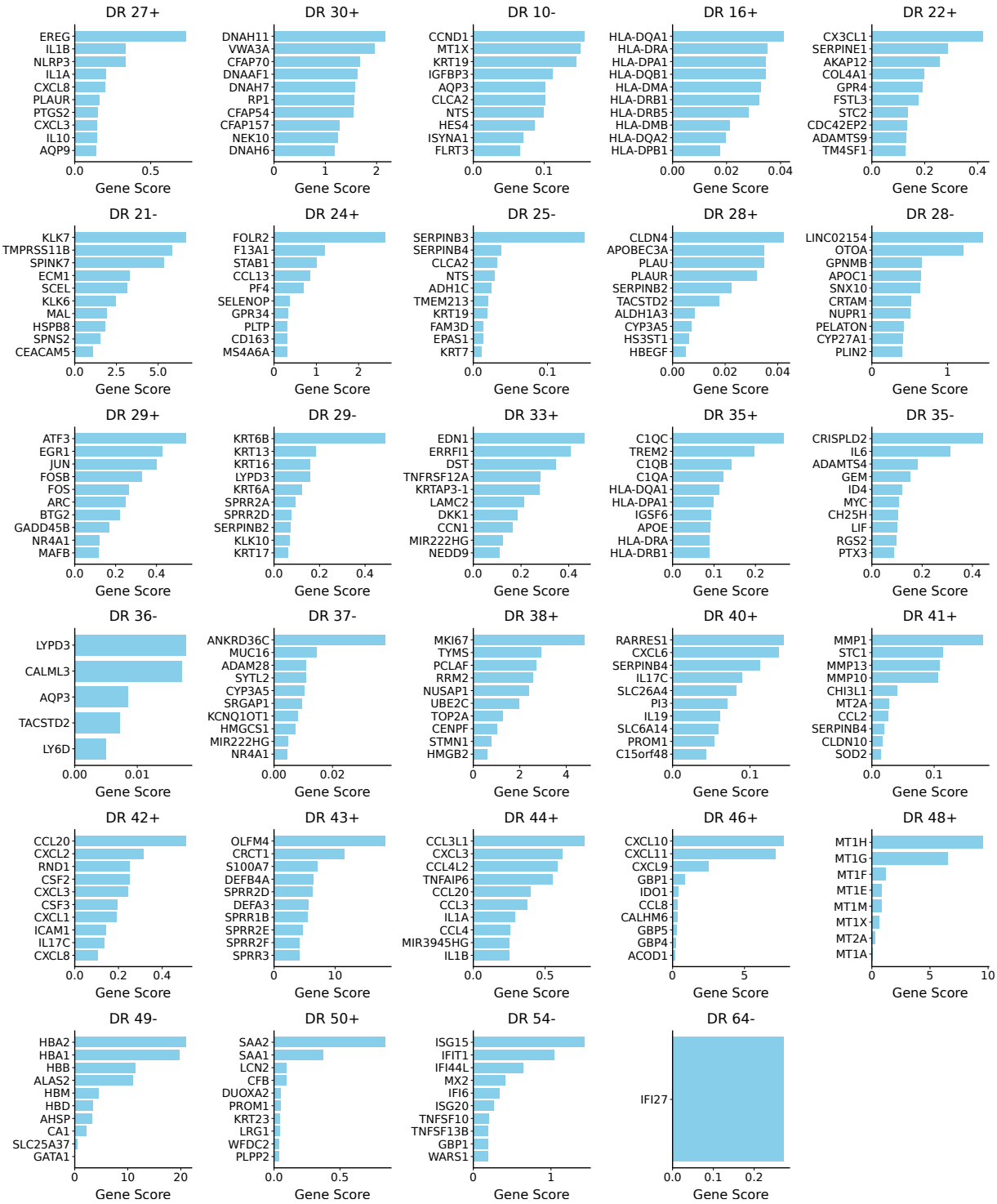
Supplemental Figure 18: **UMAPs and integration results for benchmarked methods on the HLCA dataset. a,b,** UMAPs of the HLCA integrated by DRVI and benchmarked methods colored by datasets and finest-level annotations. **c,** Integratin results based on the scIB metrics for DRVI and benchmarked methods. scANVI was originally chosen as the best-performing integration method provided by the HLCA paper, included for readers' information. However, it is not fairly comparable due to its use of supervised information and a different number of latent dimensions.



Supplemental Figure 19: **Latent space visualization for benchmarked methods on the HLCA.** For each method, a heatmap visualizing latent representation activity across different cell types is provided. Cells were subsampled to ensure a balanced number of cells from each cell type. Dimensions are reordered using a shared algorithm to maximize the diagonal appearance.

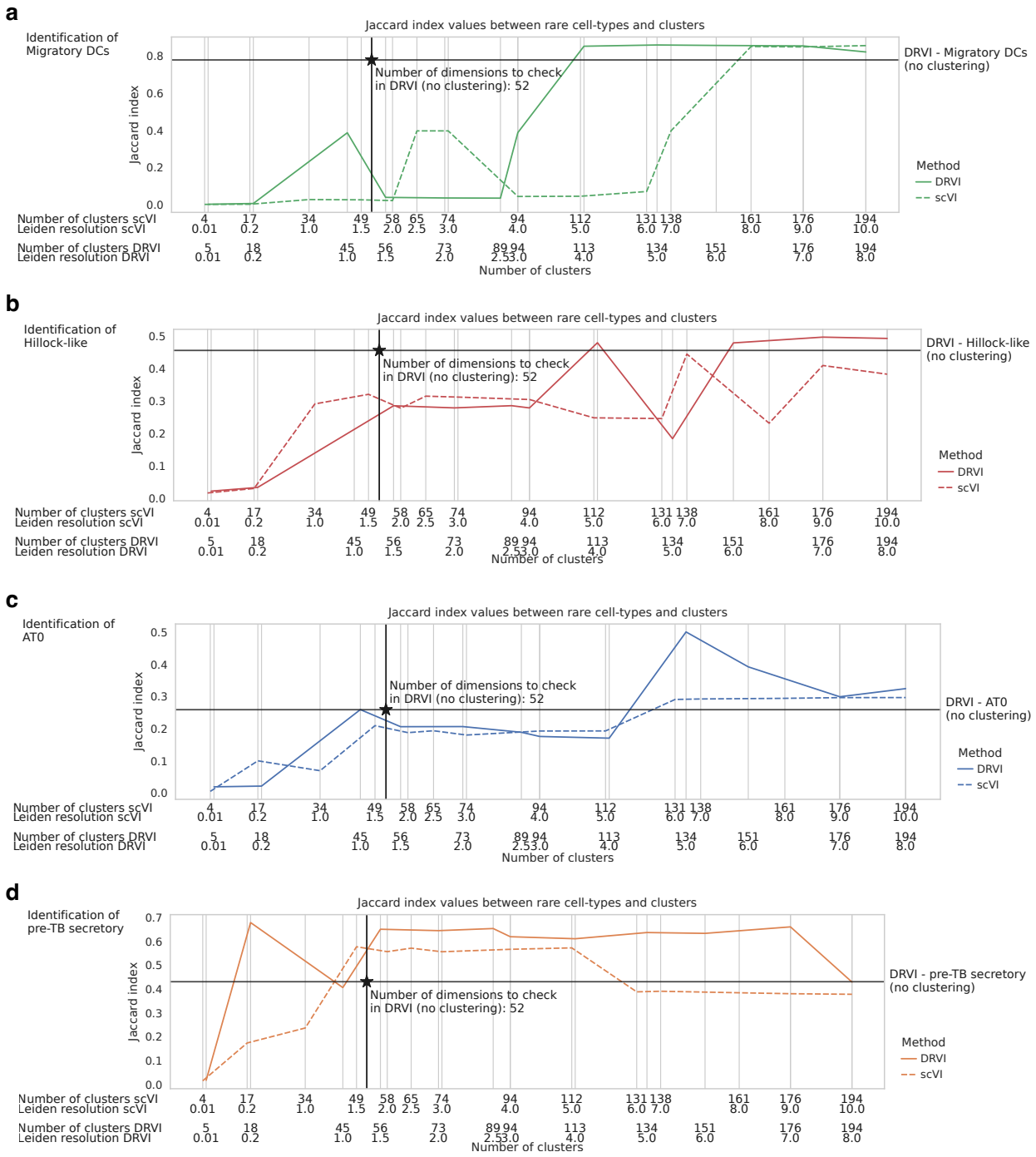


Supplemental Figure 20: **Activity of DRVI factors representing non-cell-type processes in the HLCA on UMAP** For dimensions representing non-cell-type processes, the activity on the UMAP is provided.

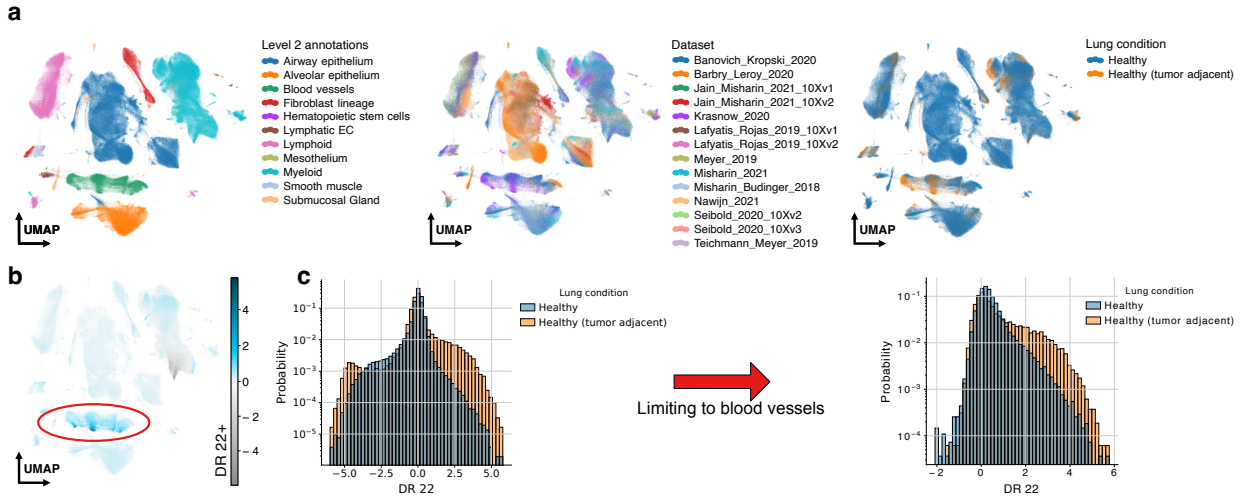


Supplemental Figure 21: **Interpretability of DRVI latent factors corresponding to non-cell-type processes in the HLCA.** The top relevant genes and their scores are plotted for each non-cell-type latent factor of DRVI trained on the HLCA.

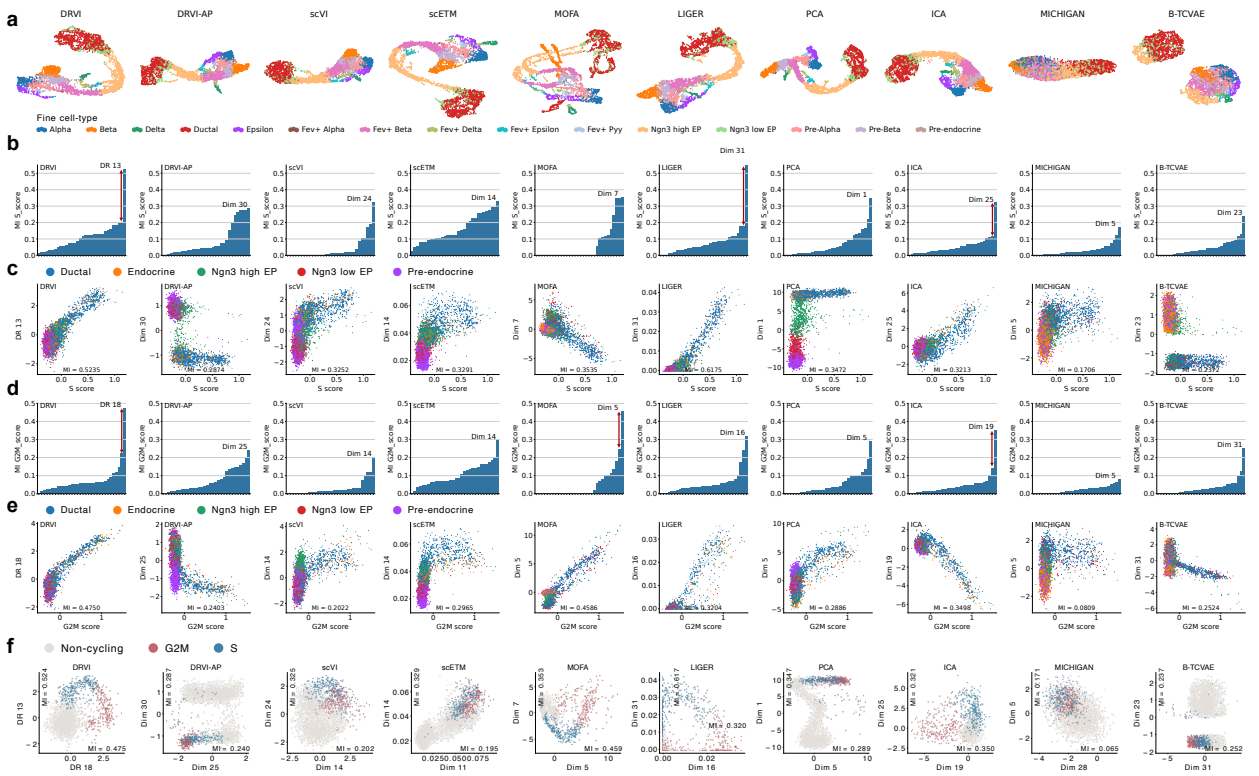
2174



Supplemental Figure 22: **Rare cell-type identification performance of Leiden clustering in the HLCA.** Leiden clustering was applied to DRVI and scVI embeddings at varying resolutions. For each cell type, we calculated the Jaccard index between the cell type and its most relevant cluster at different resolutions. Additionally, we evaluated the performance of DRVI by calculating the Jaccard index between the cell type and the separation achieved using a simple threshold on the most relevant DRVI dimension (marked by a star). Migratory DCs were accurately captured by the Leiden algorithm only at the resolution of 8.0 with 161 clusters. AT0 cells were identified with comparable quality to DRVI starting from the resolution of 6.0 with 131 clusters. Hillock-like cells were captured as well as DRVI at the resolution of 7.0 with 138 clusters. Pre-TB secretory cells were captured well at a resolution of 1.5 with 49 clusters. These results demonstrate that identifying at least three rare cell types requires the creation of approximately 130 to 160 clusters using Leiden clustering. Given that DRVI can capture these cell types within its 52 non-vanished dimensions, we suggest DRVI as an efficient valuable tool for seed annotation and discovery of rare cell populations.

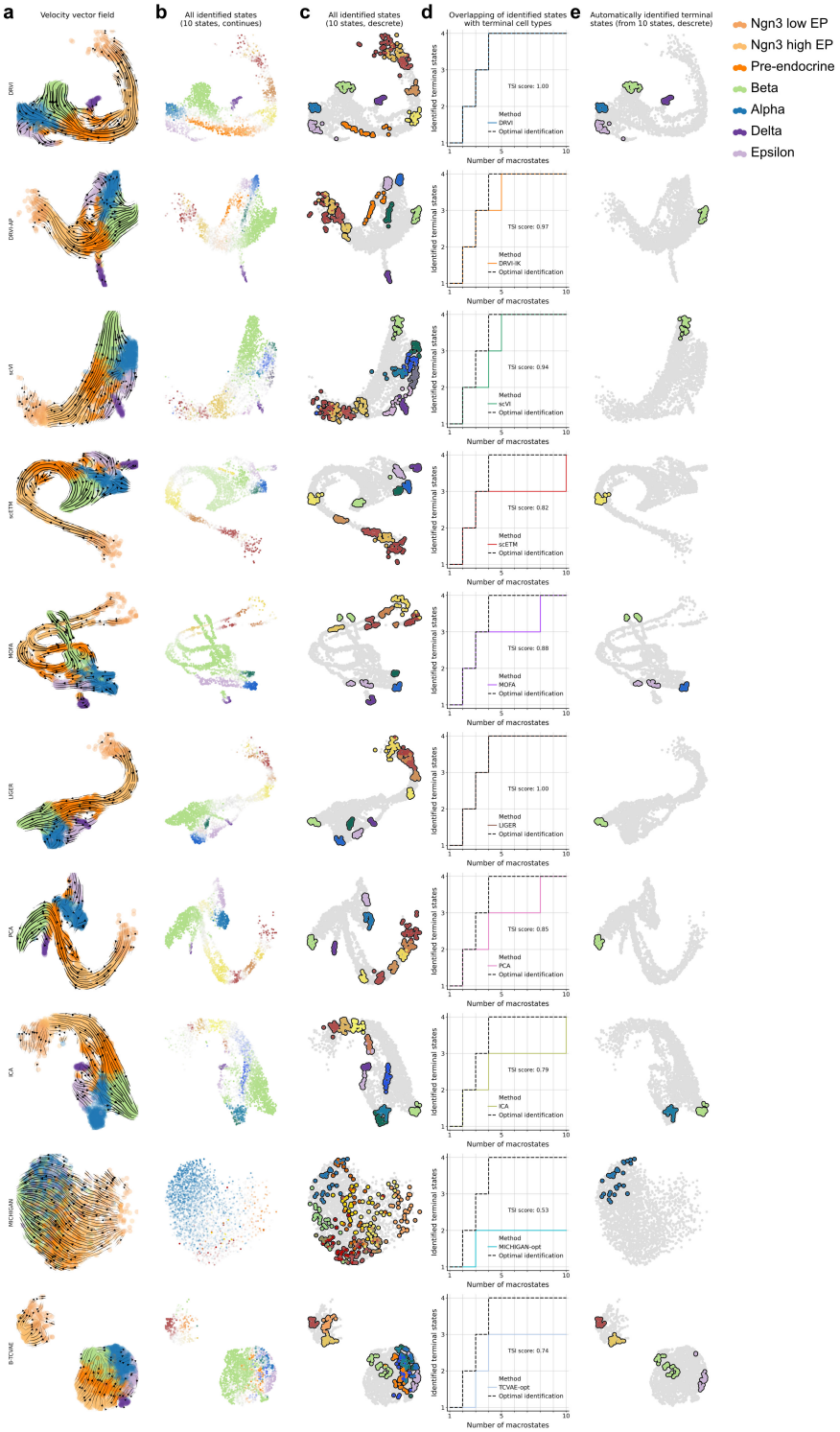


Supplemental Figure 23: **Indication of tumor-adjacent blood vessels in the HLCA. a**, UMAPs of the HLCA integrated by DRVI, colored by second-level annotations, datasets, and tumor adjacency. Tumor-adjacent samples originate from the Krasnow_2020 dataset. By observing the blood vessels in the UMAPs, we find two groups of tumor-adjacent cells. Some integrate seamlessly with normal blood vessels, while others exhibit a degree of separation, suggesting the presence of underlying biological variation. **b**, Activity of DR 22+ on the UMAP. This dimension highlights a subset of tumor-adjacent blood vessels that are distinctly separated in the UMAP, potentially representing a unique biological process or state within this cell population. **c**, Histogram of cells distributed across DR 22+, colored by tumor adjacency status before and after filtering for blood vessels. A significant proportion of cells active in the higher end of DR 22+ correspond to blood vessels located in tumor-adjacent regions, indicating a biological process or state specific to this subset of cells.



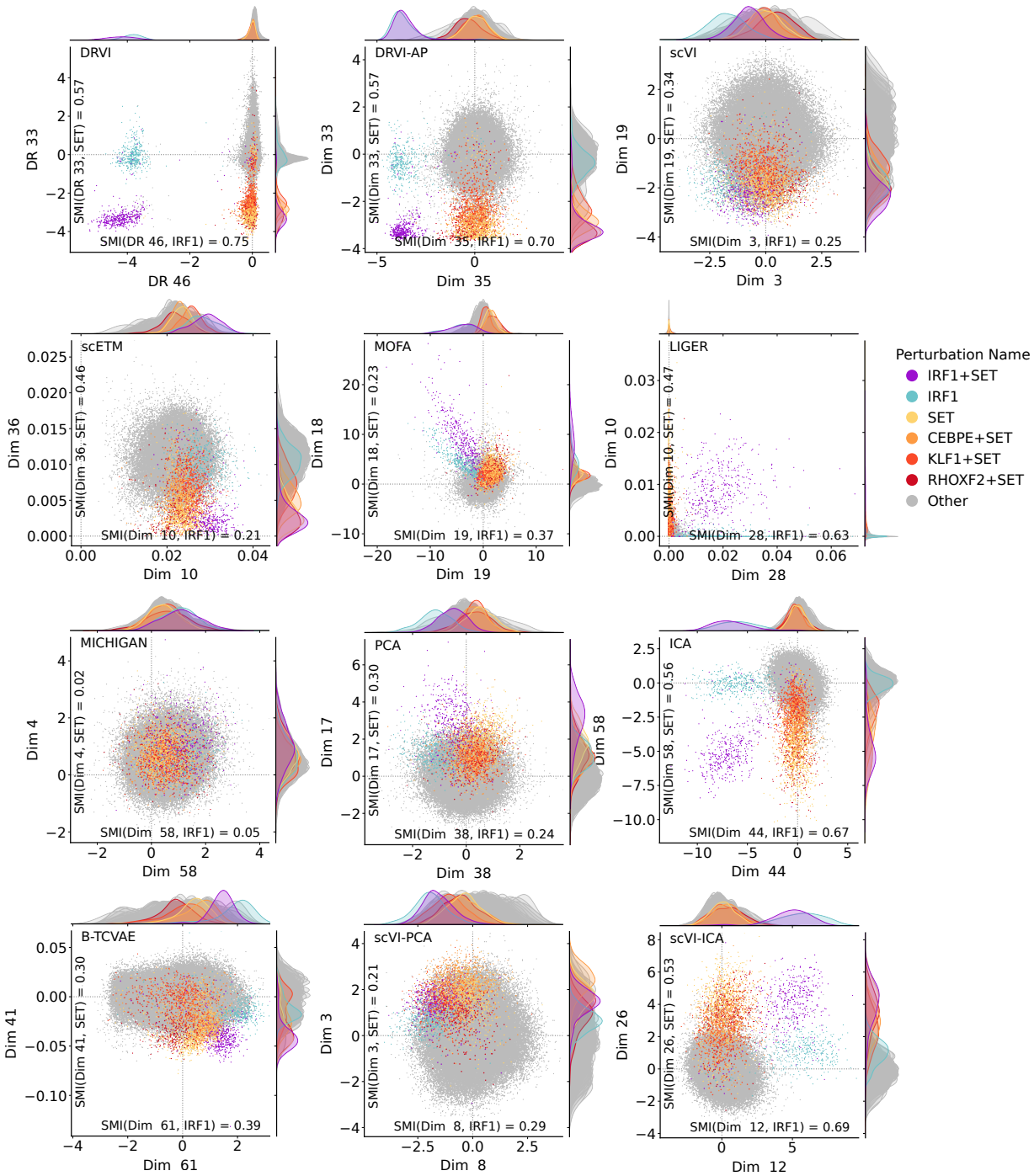
Supplemental Figure 24: **DRVI disentangles cell cycle stages. a,** UMAPs of the developmental pancreas data applied to the latent spaces generated by benchmarked methods. Cells are colored by finest-level annotations. **b-e,** Barplots indicating Mutual information of each dimension with cell-cycle S-score (b) and G2M-score (d). The dimension maximizing the mutual information with the S-score(c) and G2M-score(d) are plotted versus the S-score (c) and G2M-score (e) for each method. Cells are colored by coarse annotations.**f,** The learned cell-cycle manifolds. The scatterplots of dimensions maximizing the mutual information with respect to the S-score and G2M-score are plotted. Cells are colored by the dominant cell cycle phase. MI, mutual information.

2177



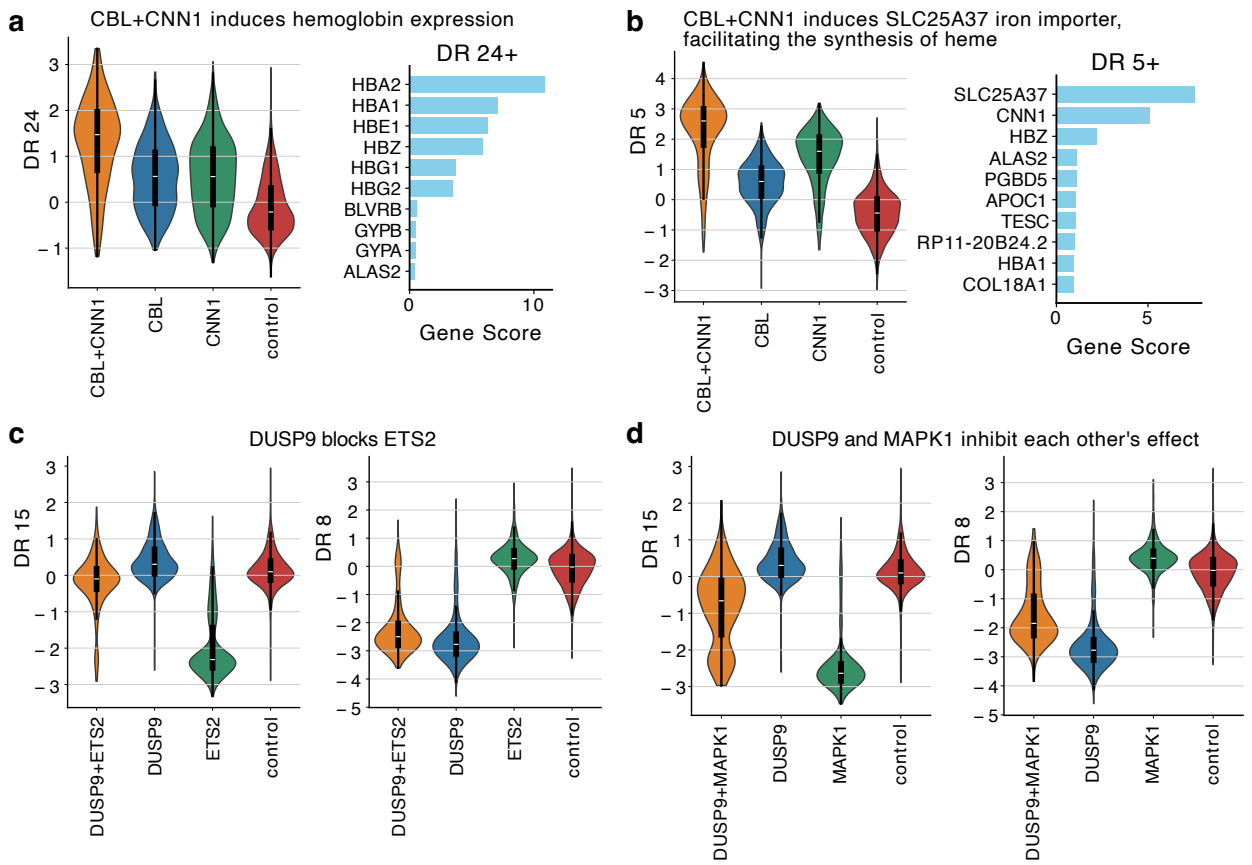
Supplemental Figure 25: **Developmental fate prediction based on model embeddings. a,** UMAP visualizations of scVelo RNA velocity vector fields, generated based on embeddings derived from various benchmarked models. **b,c,** UMAP visualizations of stationary states identified by CellRank, based on the velocity vector fields shown in (a). **d,** Terminal State Identification (TSI) scores for different benchmarked methods. This metric quantifies the overlap between identified stationary states and ground-truth terminal states as the number of macrostates increases. **e,** UMAP visualization of automatically identified terminal states from ten macrostates using CellRank. TSI: Terminal State Identification.

2178



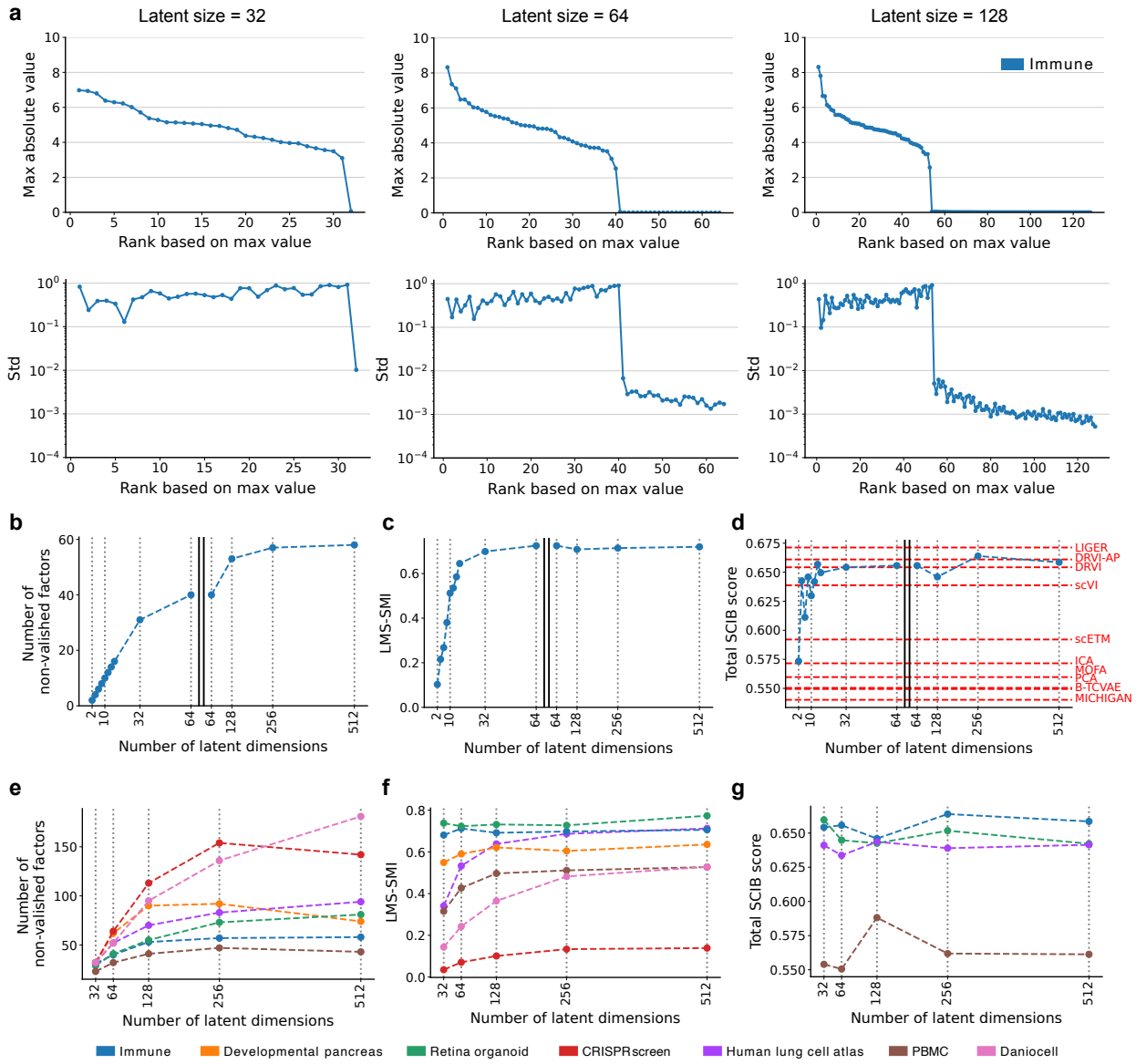
Supplemental Figure 26: **DRVI identifies and preserves the geometry of combinatorial perturbations.** Each scatterplot illustrates the activity of the top two dimensions having the highest mutual information with respect to IRF1 and SET perturbations for a benchmarked method. The cells perturbed by IRF1 and SET and all combinatorial perturbations containing them are highlighted with distinct colors. Any other perturbation signature is colored grey. SMI, scaled mutual information.

2179



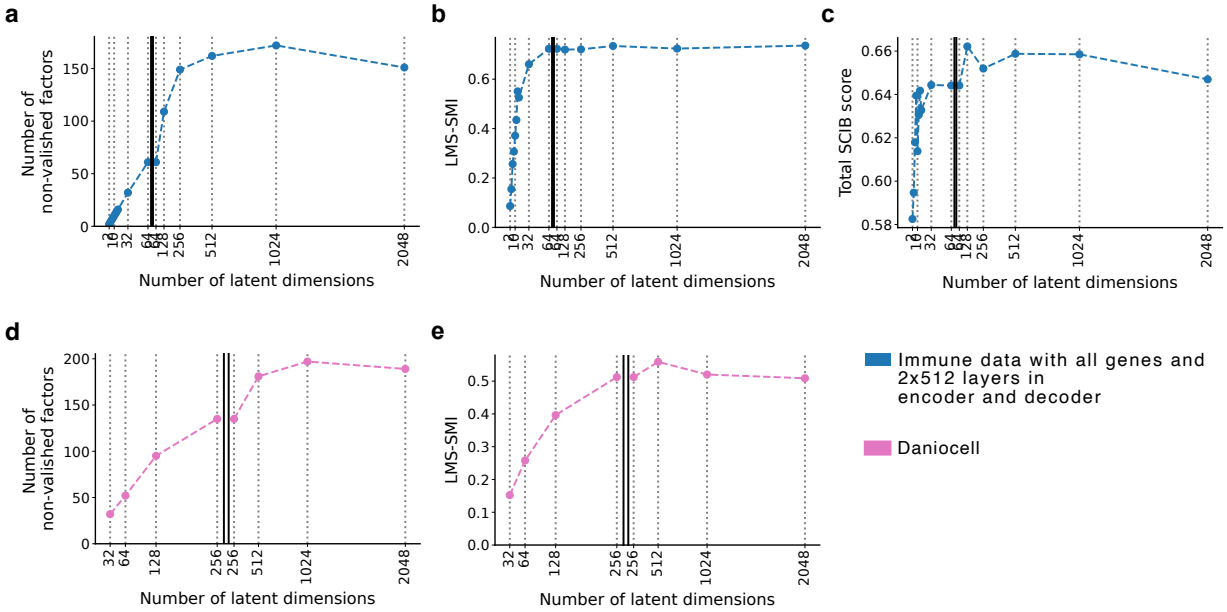
Supplemental Figure 27: **DRVI's latent space effectively models interacting double perturbations. a,** When the **CBL+CNN1** gene combination is perturbed, there's a significant increase in the expression of **hemoglobin** (DR 24+). **b,** Perturbing **CBL+CNN1** leads to the overexpression of the iron importer **SLC25A37** (DR 5+), which supports the role of CBL+CNN1 in regulating hemoglobin expression. **c, DUSP9** acts as an inhibitor of **ETS2**, as its effect (DR 15-) is neutralized when DUSP9 is also perturbed. **d, DUSP9** and **MAPK1** exhibit a mutual inhibitory relationship, which is evident from the decreased response signatures of both genes (DR 15- and DR 8-) when they are simultaneously perturbed.

2180

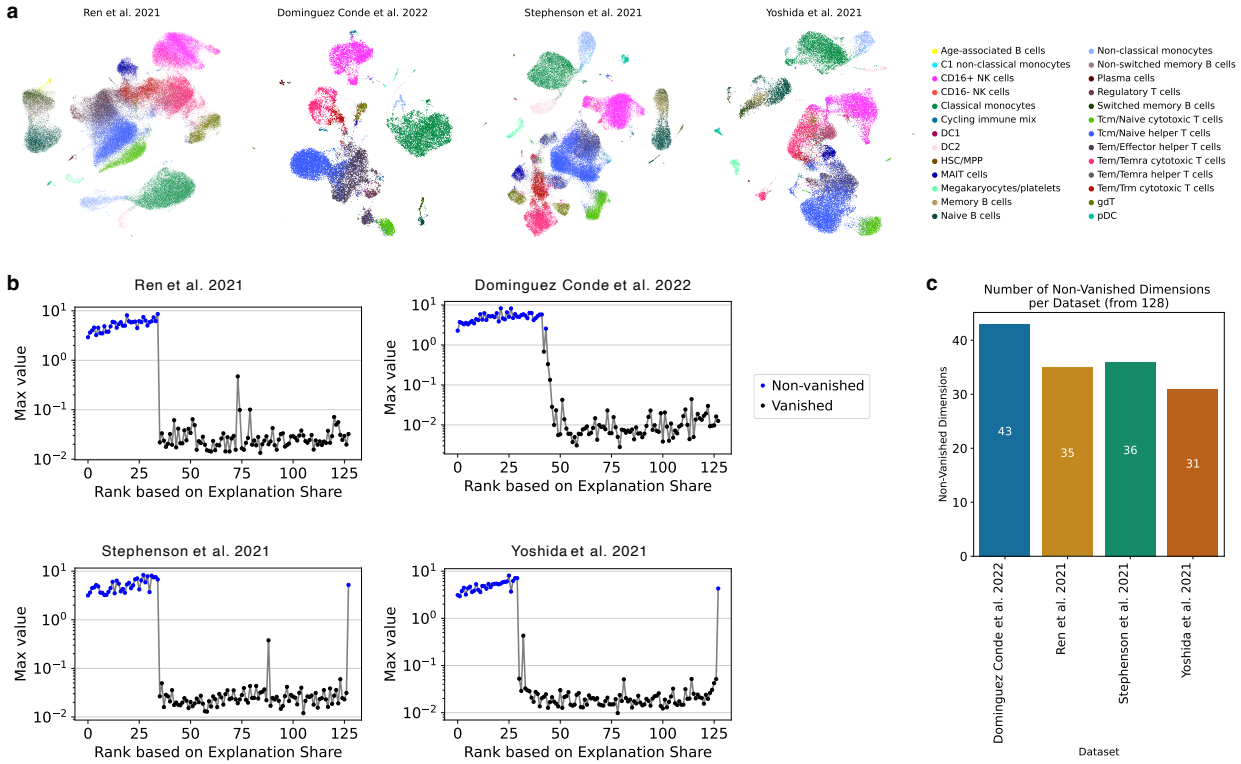


Supplemental Figure 28: **The saturation of the number of employed latent dimensions by DRVI. a,** The vanishing dimensions phenomenon. DRVI is trained with 32, 64, and 128 latent dimensions on the immune dataset. Each point is a latent dimension. The top row indicates max absolute values, and the bottom row indicates log-scale standard deviations of latent dimensions. Latent dimensions are ranked based on max absolute value. Each column indicates a run with a different number of latent dimensions. **b-d,** Saturation of the number of non-vanished dimensions (b), disentanglement score (c), and integration score (d) when more latent dimensions are provided to DRVI trained on the immune dataset. Red lines annotate integration scores for benchmarked methods with default parameters. The x-axes on the right-hand side of the plots are shrunk by a factor of 5. **e-g,** Saturation of the number of non-vanished dimensions (e), disentanglement score (f), and integration score (f) when more latent dimensions are provided to DRVI trained on the benchmarking datasets. LMS, latent matching score; SMI, scaled mutual information.

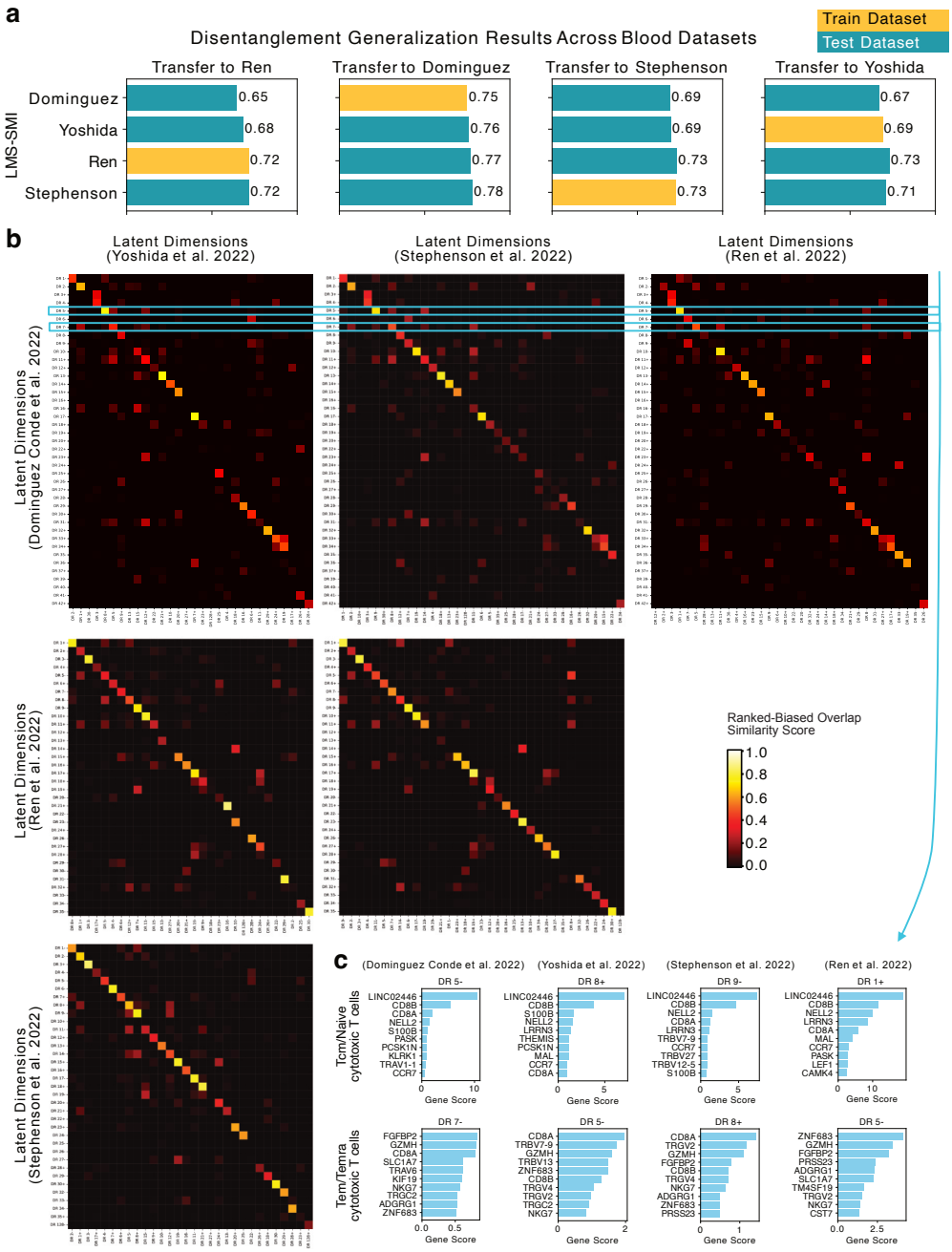
2181



Supplemental Figure 29: **Further analysis on the saturation of employed latent dimension. a-c,** DVRI's behavior trained on all genes of the immune dataset with different latent dimensions. The number of non-vanished dimensions (a), Disentanglement quality (b), and integration performance (c) are plotted when we change the number of latent dimensions. DRVI is trained on the full gene set of the immune dataset. Encoder and decoders contain 2 layers of 512 hidden neurons. As we observe, DRVI saturates at some specific point. At this point, disentanglement performance also saturates. Integration performance saturates and drops a bit. **d,e,** DVRI's behavior trained on Daniocell data beyond 512 latent dimensions. The number of non-vanished dimensions (d) and disentanglement quality (e) are plotted when we change the number of latent dimensions. As we observe, even in Daniocell data where there are 156 cell types, DRVI saturates at some point. At this point, disentanglement performance also saturates. LMS, latent matching score; SMI, scaled mutual information.

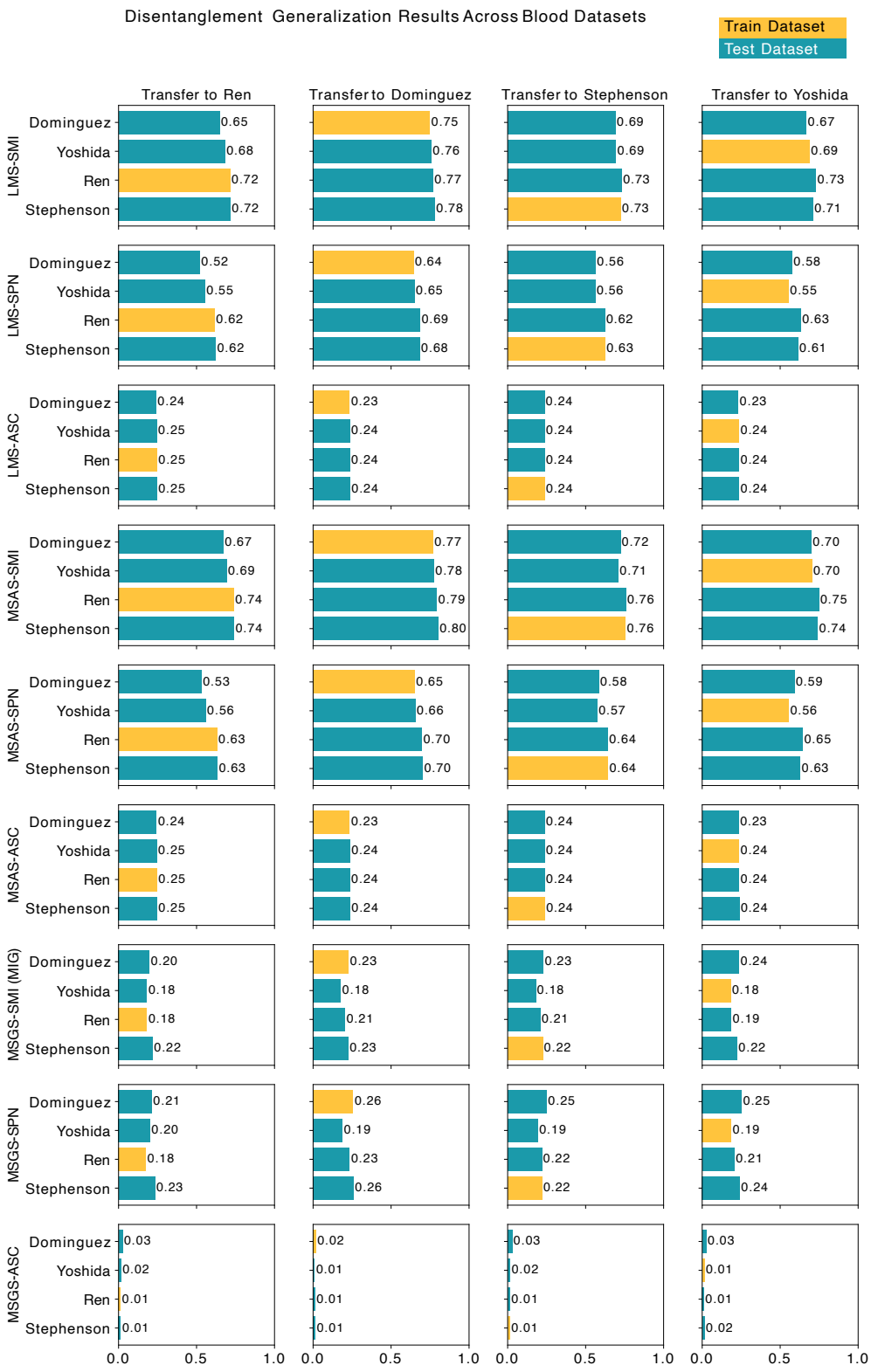


Supplemental Figure 30: **Training DRVI on four datasets of the Blood data. a,** UMAP visualizations of individual Blood datasets, integrated by separate DRVI models. Cells are colored according to their curated annotations. **b,** Visualization of the vanished dimensions for each of the four DRVI models trained on the Blood datasets. **c,** The total number of non-vanished dimensions for each model trained on an individual Blood dataset.

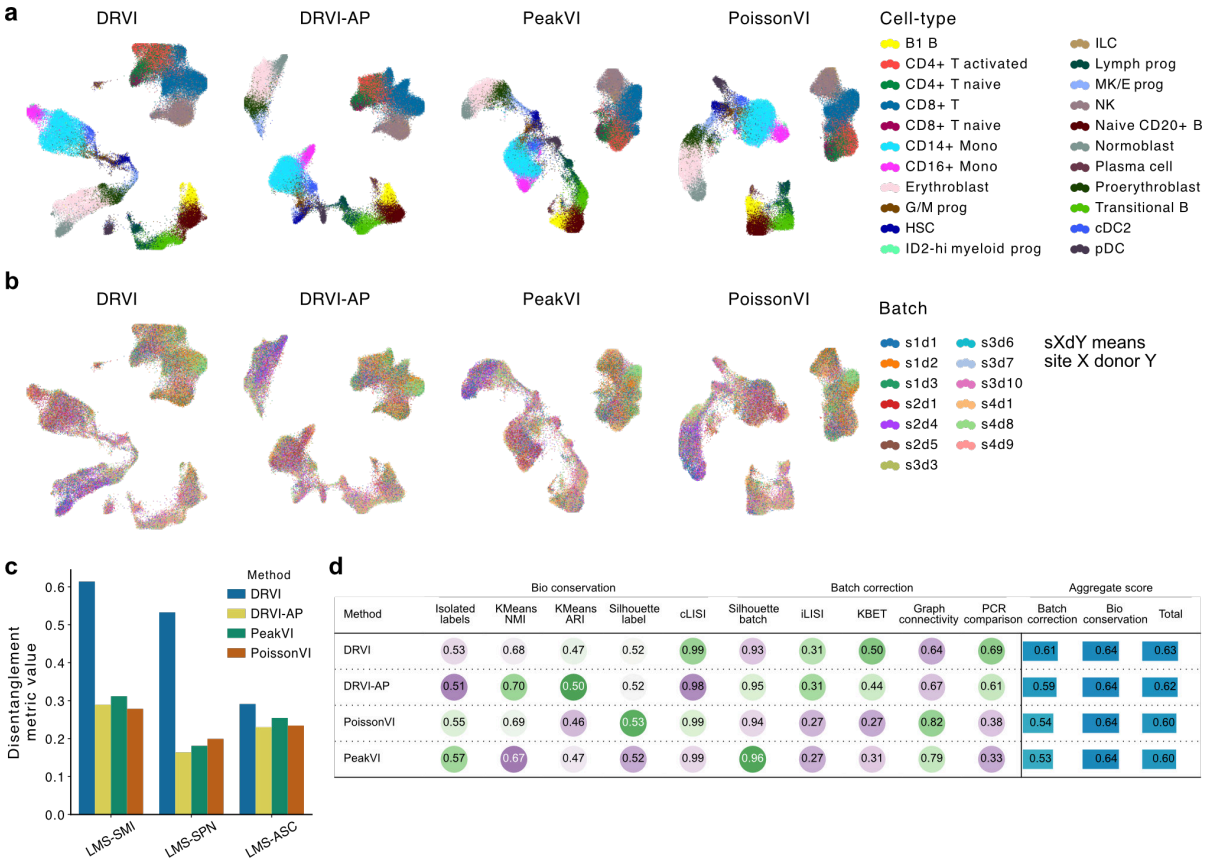


Supplemental Figure 31: **Assessing the generalization and disentanglement of latent programs in unseen Blood data a,** A separate DRVI model is trained on each of the Blood datasets. This trained model is then used to generate latent representations for the remaining, unseen datasets. The quality of disentanglement is measured for each of these transfer scenarios. The results demonstrate that the disentangled concepts are robustly transferred to the unseen test datasets. **b,** For each trained DRVI model, we identify the relevant genes associated with each latent dimension. We then compare these gene sets across datasets using the Rank-Biased Overlap (RBO) similarity score [90]. The heatmap indicates a favorable correspondence between identified concepts across datasets. **c,** This panel displays the top relevant genes for dimensions representing Tcm/Naive cytotoxic T cells and Tem/Temra cytotoxic T cells across the four models. The positive RBO scores observed between the dimensions representing these two cell types (as shown in panel **b**) are a result of the intrinsic gene overlap between these closely related subtypes. LMS, latent matching score; SMI, scaled mutual information.

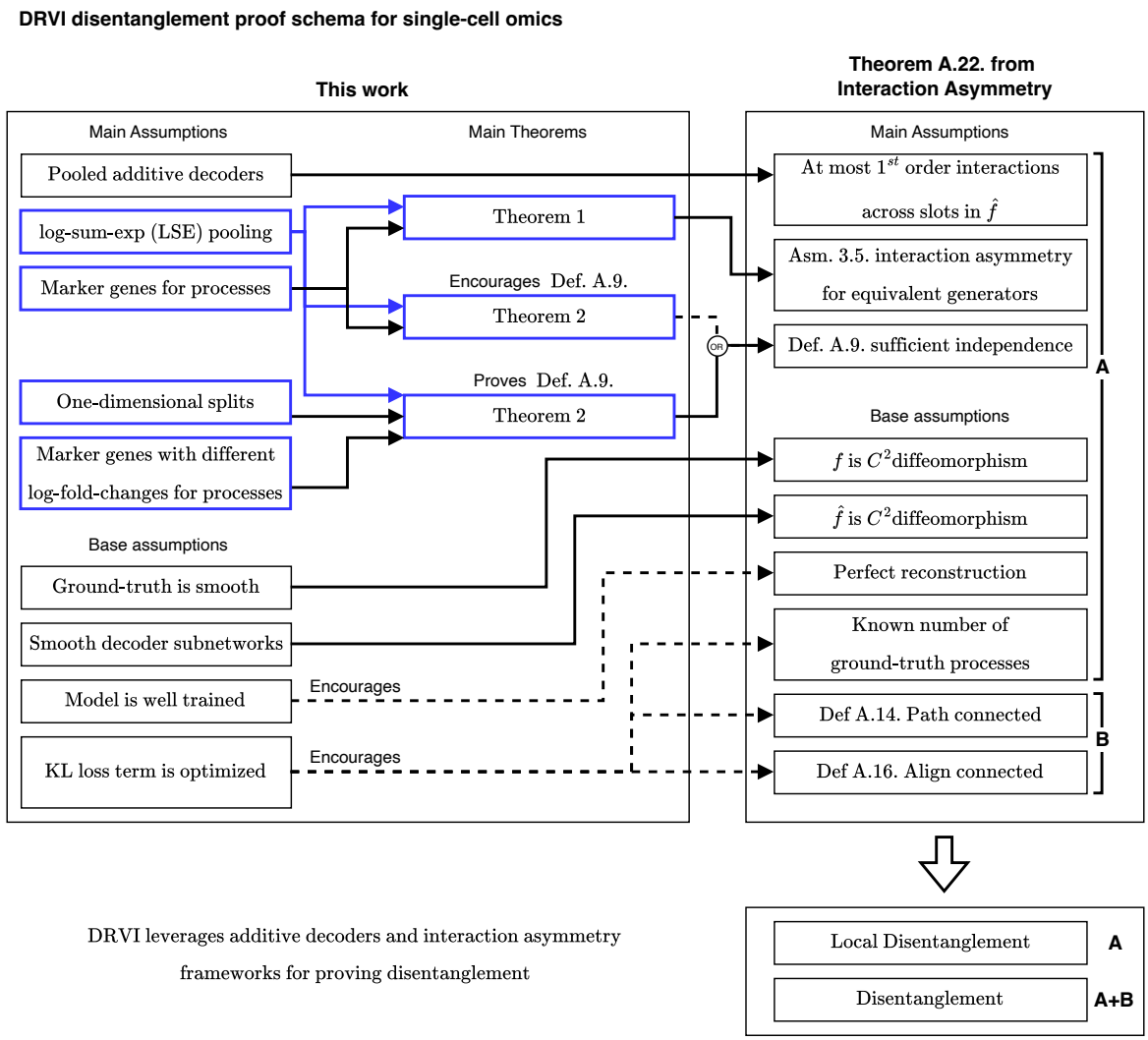
2184



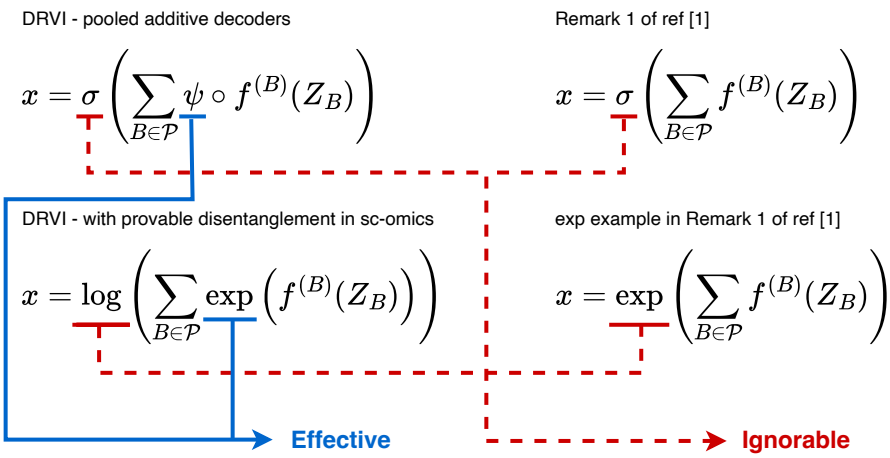
Supplemental Figure 32: **Full disentanglement metrics on generalization to unseen datasets in the Blood data.** A separate DRVI model is trained on each of the Blood datasets. This trained model is then used to generate latent representations for the remaining, unseen datasets. The quality of disentanglement is measured for each of these transfer scenarios. The results demonstrate that the disentangled concepts are robustly transferred to the unseen test datasets. ASC, absolute Spearman correlations; LMS, latent matching score; MSAS, Most Similar Averaging Score; MSGS, Most Similar Gap Score; SMI, scaled mutual information; SPN, same-process neighbors.



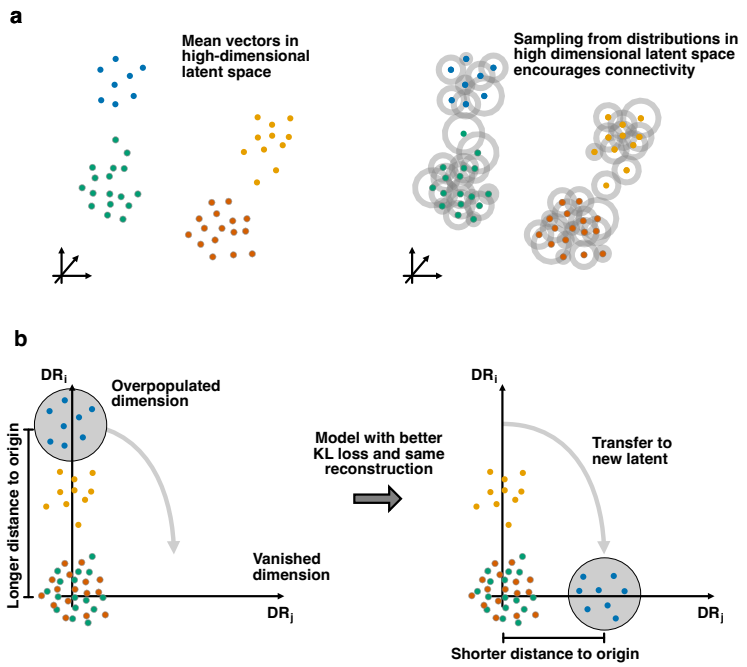
Supplemental Figure 33: **DRVI is applicable to scATAC-seq data. a,b,** UMAPs of the ATAC modality of the NeurIPS 2021 data integrated by DRVI, DRVI-AP, PeakVI, and PoissonVI colored by cell-type annotations (a) and batch identity (b). **c,** Disentanglement performance of the benchmarked methods for scATAC-seq data. DRVI performs significantly better in this task. **d,** Integration performance of the benchmarked methods based on the scIB metrics. DRVI performs better compared to the three other methods. ASC, absolute Spearman correlations; LMS, latent matching score; SMI, scaled mutual information; SPN, same-process neighbors.



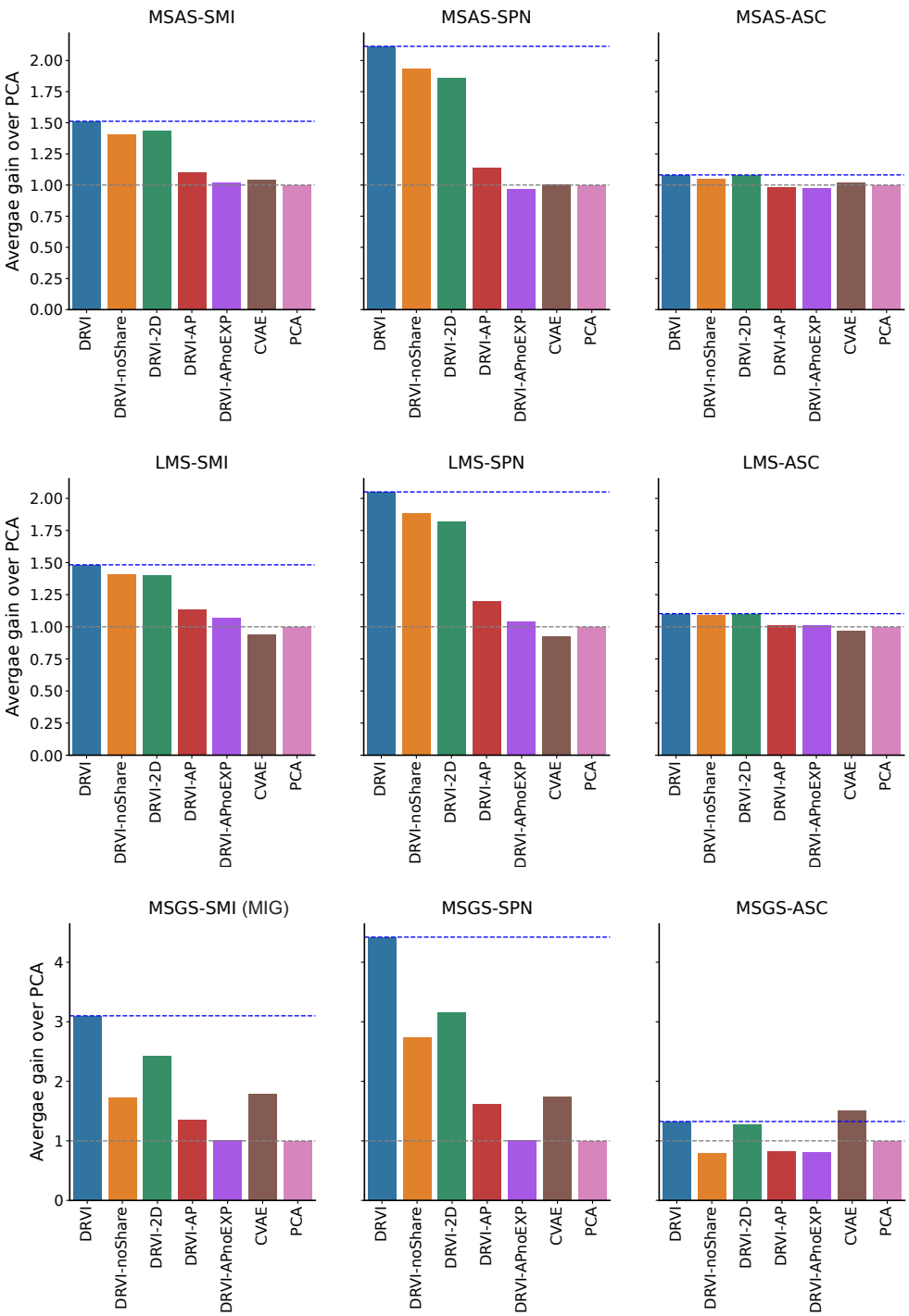
Supplemental Figure 34: **DRVI leverages additive decoders and interaction asymmetry to prove disentanglement.** These frameworks prove disentanglement based on abstract assumptions about the generative process and decoder subnetworks. The main and base assumptions for the interaction asymmetry [48], along with the types of derived disentanglement, are illustrated on the right side of the sketch. By introducing log-sum-exp (LSE) pooling as a novel architectural constraint in DRVI, combined with domain assumptions like the presence of marker genes in ground-truth processes, we prove (Theorems 1, 2, and 3) that the abstract criteria for interaction asymmetry are met, or in some cases encouraged, as illustrated on the left side of the sketch. Altogether, we leverage additive decoders and interaction asymmetry to prove disentanglement (when splits are one-dimensional) or show that it is encouraged (in the general case) for DRVI in single-cell omics.



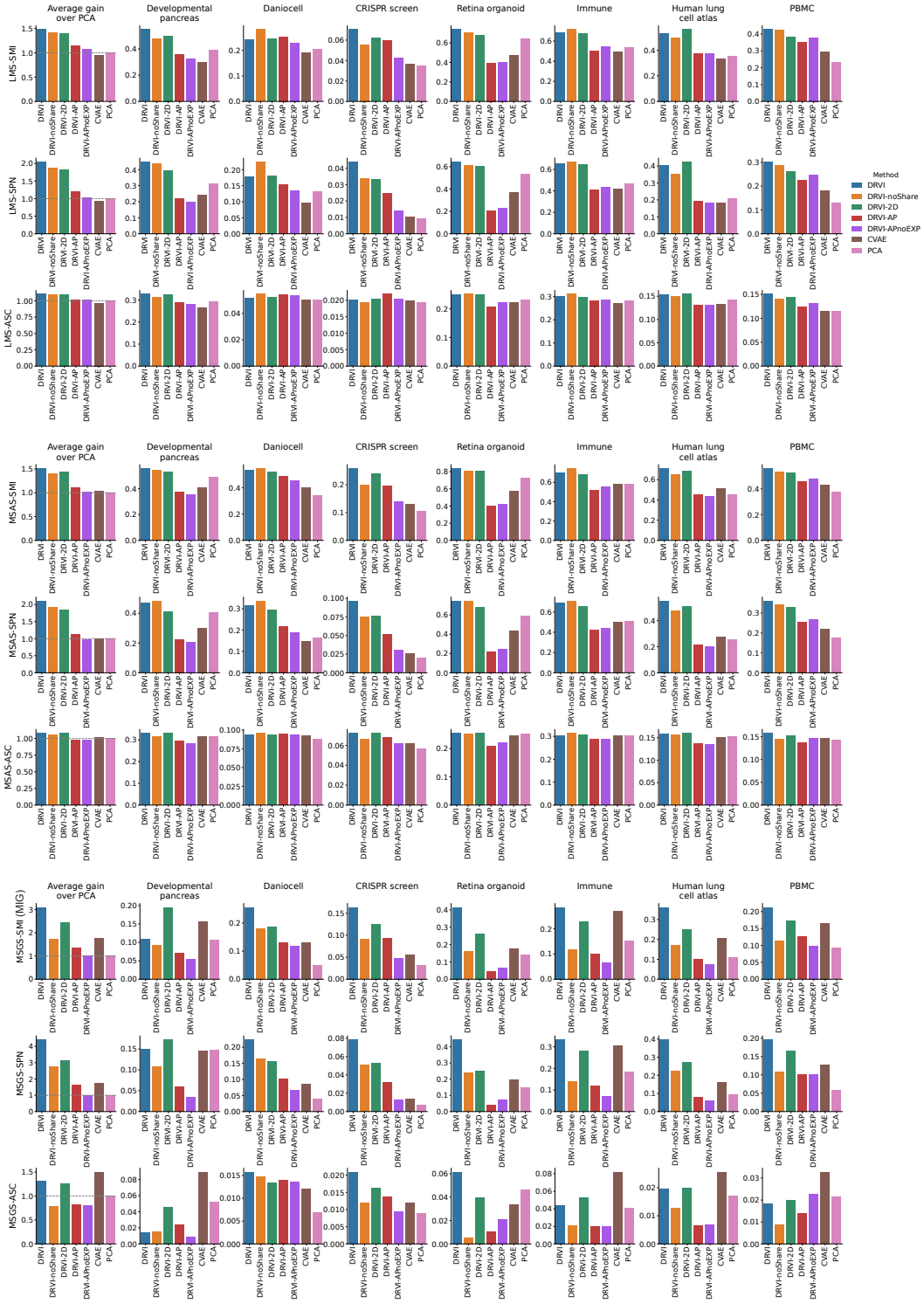
Supplemental Figure 35: **The effective and ignorable parts of the pooling function.** Remark 1 from [37] suggests that activation functions applied after summing additive decoders can be disregarded. In contrast, here, we make use of a shared activation function (ψ) before the sum operator, which is not ignorable. This novel constraint, and as we demonstrate in Theorems 1 and 3, plays a crucial role in enforcing disentanglement.



Supplemental Figure 36: **How latent space sampling and KL loss promote global disentanglement. a,** In DRVI, like other VAE-based methods, the latent representation of each cell isn't a vector, but rather a distribution, typically represented by a multivariate Gaussian with mean μ and a diagonal covariance matrix derived from a σ vector. The KL loss term encourages μ values to be near zero and σ values to be near one. As the model converges, these non-negligible σ values cause the distributions of neighboring cells to overlap. This prevents the model from indexing individual cells and promotes a continuous manifold of nearby cells. Thus, the KL loss term encourages path connectivity. **b,** Assuming the model has achieved local disentanglement and a sufficient number of latent dimensions are available, the KL loss term indirectly encourages irrelevant processes to be represented in individual latent dimensions. If two irrelevant processes are present within a single latent dimension and an additional dimension is available, further optimization of the model is possible. By relocating the process further from the center to an available dimension, one can achieve the same reconstruction quality while simultaneously improving the KL loss.

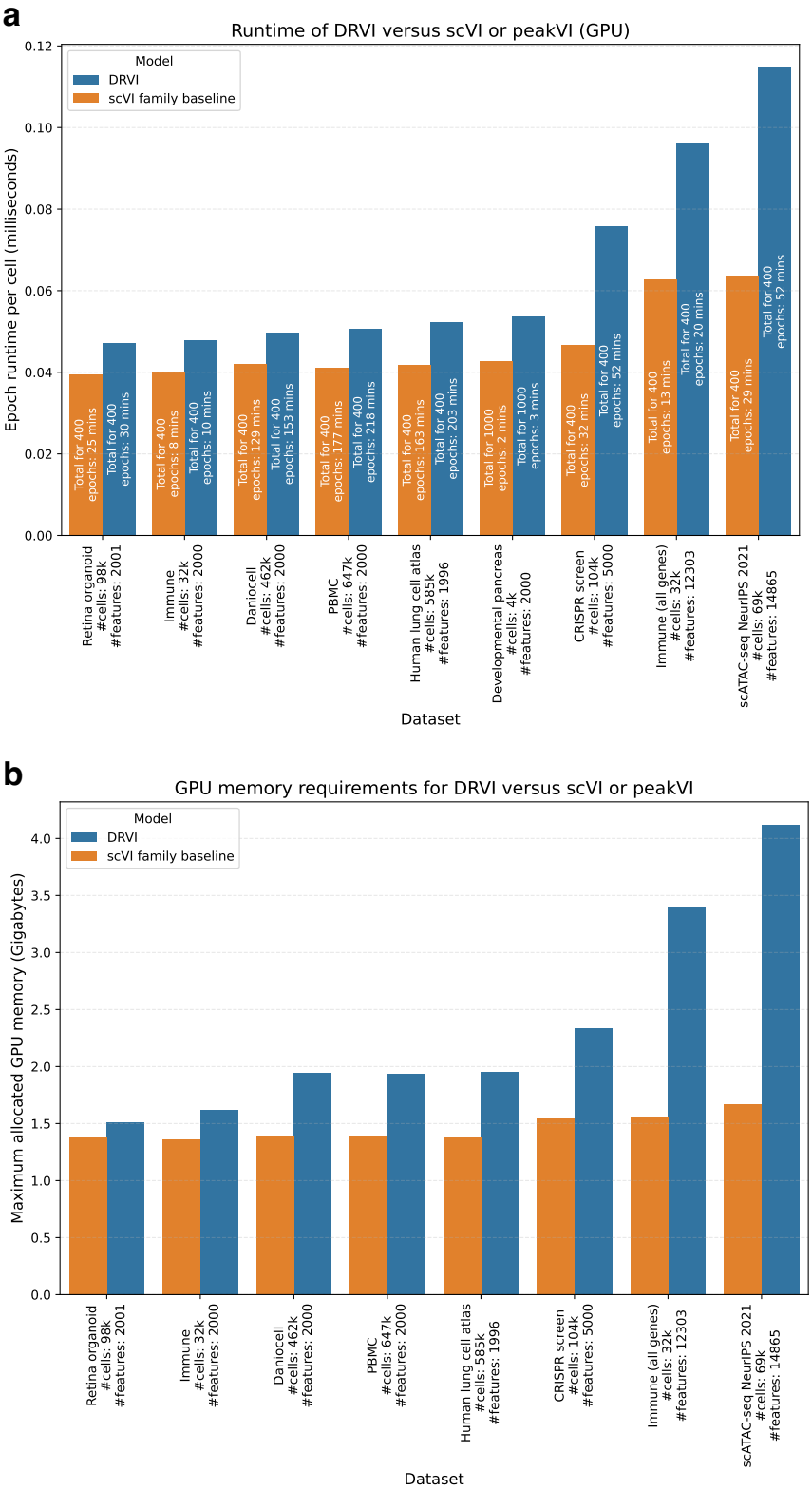


Supplemental Figure 37: **Summary of all metrics for the ablation study.** This plot displays the average gain achieved by each of the DRVI variants over PCA. The gains are shown for every combination of the SMI, SPN, and ASC similarity functions with the LMS, MSAS, and MSGS aggregation functions. Results indicate the importance of employing log-sum-exp (LSE) pooling to achieve disentanglement. ASC, absolute Spearman correlations; LMS, latent matching score; MSAS, Most Similar Averaging Score; MSGS, Most Similar Gap Score; SMI, scaled mutual information; SPN, same-process neighbors; MIG, mutual information gap.

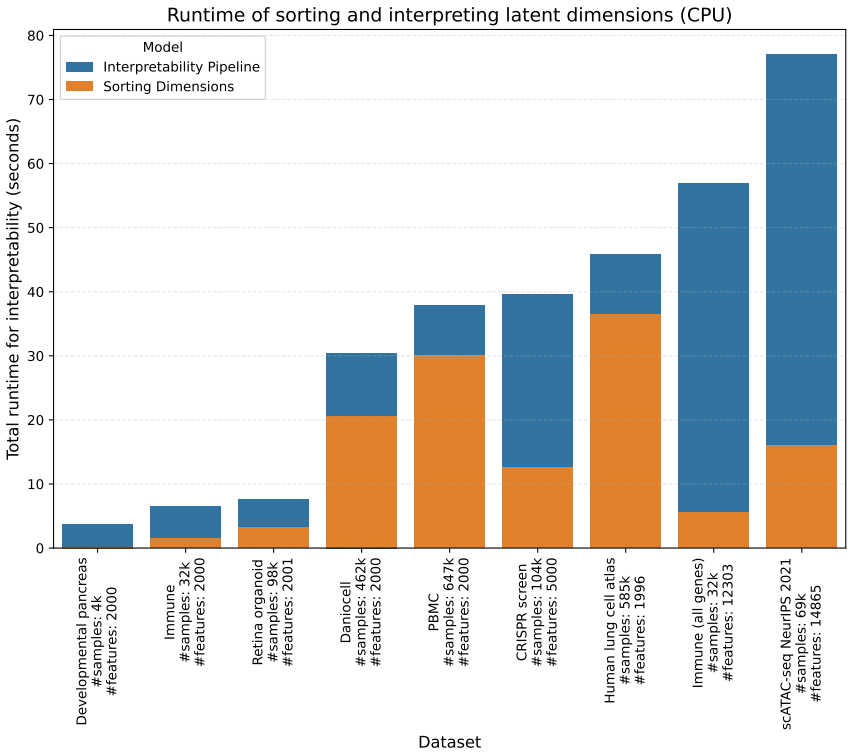


Supplemental Figure 38: **Details of disentanglement metrics for the ablation study.** For all benchmarked datasets and each of the DRVI variants, detailed results of the LMS, MSAS, and MSGS aggregation functions over the SMI, SPN, and ASC similarity functions are provided. ASC, absolute Spearman correlations; LMS, latent matching score; MSAS, Most Similar Averaging Score; MSGS, Most Similar Gap Score; SMI, scaled mutual information; SPN, same-process neighbors; MIG, mutual information gap.

2191



Supplemental Figure 39: **Runtime analysis of DRVI. a,b,** The runtime (a) and maximum allocated GPU memory (b) for DRVI, scVI, and peakVI are shown for each benchmarked dataset. To normalize for varying dataset sizes and training lengths, the y-axis in (a) represents the runtime divided by the number of cells and the number of epochs (runtime per sample per epoch). The total runtime for each experiment is also displayed on the respective bars. DRVI's training runtime is on the same order as scVI or peakVI up to a small scaling factor, with a maximum twofold increase observed in our experimental setups. In addition, training DRVI remains manageable when applied to the full gene set.



Supplemental Figure 40: **Runtime for interoperability and sorting latent dimensions.** The time required to sort the latent dimensions and interpret gene programs for all latent dimensions is calculated. This demonstrates that the interpretability runtime is negligible.