University of California
Santa Barbara

Single Cell Multi-modal Analysis Using scDMVAE with an Emphasis on SCoPE2 Technology

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Statistics and Applied Probability

by

Yi Zheng

Committee in charge:

Professor Alexander Franks, Chair
Professor Sang-Yun Oh
Professor Yuedong Wang

March 2023

The Dissertation of Yi Zheng is approved.

Professor Sang-Yun Oh

Professor Yuedong Wang

Professor Alexander Franks, Committee Chair

January 2023

Single Cell Multi-modal Analysis Using scDMVAE with an Emphasis on SCoPE2 Technology

by

Yi Zheng

I dedicate my dissertation work to my beloved parents, Lianping Zheng and Shulan Fan, who have been guiding me with their words, action and kind hearts; my girlfriend, Xiaofan Zhang, who accompany me through the entire Ph.D journey with love and happiness

Acknowledgements

I would like to thank the following people without whom I would not have been able to made it through my Ph.D career and complete the research: My advisor, Professor Alexander Franks, who have patiently guided me from the start with his wisdom, passion, diligence and expertise; My committee members Professor Yuedong Wang and Professor Sang-Yun Oh, who were more than generous with their expertise and precious time; Megan Elcheikhali, Professor Nikolai Slavov, Saad Khan and other members of Professor Slavov's lab, who helped me understand the biology behind the data and gave me constructive advice on statistical modelling.

Curriculum Vitæ

Yi Zheng

Education

2023	Ph.D. in Applied Statistics (Expected), University of California, Santa Barbara.
2017	M.S. in Statistics, Arizona State University.
2013	B.S in Applied Physics, University of Science and Technology of China

Publications

Abstract

Single Cell Multi-modal Analysis Using scDMVAE with an Emphasis on SCoPE2 Technology

by

Yi Zheng

Effective multi-modal integration of single cell datasets is critical for uncovering the biological properties of cells from different molecular perspectives. However, this poses significant challenges, including how to preserve shared information and account for differences between differently distributed datasets, how to integrate datasets linked by different anchors (cells or features) and how to improve the quality of datasets for integration. In this dissertation, we introduce two novel models that address these challenges. First, we present scDMVAE, a neural network model that can capture both shared and data-specific aspects of datasets in a latent space. scDMVAE can handle both cell-linked and feature-linked datasets through its embedding learning and attention-based matching components, respectively. We demonstrate the effectiveness of scDMVAE on a cell-linked CITE-seq dataset to reveal different cell type relations between mRNA and protein, and on feature-linked SCoPE2 proteomics and scRNA-Seq mRNA human testis datasets to transfer labels from mRNA to protein. Additionally, we present PCRID, a principal curve based model that aligns the retention time of peptides to improve confidence estimates of peptide-spectrum-matches (PSMs) in SCoPE2 technology. PCRID outperforms existing models like DART-ID by handling non-linearities in retention time more effectively, increasing the identification rate of peptides by 154.53 % at a PEP threshold of 0.01 while controlling false discoveries. Together, these models represent significant advances in single cell data analysis and have broad applications across related fields.

Contents

Chapter 1

Introduction

Since the publication of the first scRNA-seq study in 2009 (Tang et al., 2009), single-cell technologies have become vital tools for scientists to understand biological mechanism at cellular level. These high throughput technologies allow researchers to acquire information on different molecules within hundreds of thousands individual cells, including genome, transcriptome, DNA methylation landscape, chromatin accessibility and proteomes. When combined, these different datasets can reveal the nature of cells from different aspects and provide a comprehensive picture of basic biological process. However, integrating these datasets presents several challenges. One of the challenges is technical effect, which varies among different single-cell technologies and make it hard to combine datasets from the same modality but different technologies (e.g. batch effect). It is even more challenging to integrate datasets from different types of molecules (modalities) due to differences in their distribution and various biological connections among datasets. Additionally, a significant challenge is how to handle different linkages between different datasets (Argelaguet, Cuomo, Stegle, & Marioni, 2021). Some datasets contain information from different modalities in the same cell and thus are referred to as "anchored by cells" (vertical integration). Some datasets are profiled from independent groups of cells

and linked by the same gene set. They are referred to as "anchored by genomic features" (horizontal integration). Other datasets that have no anchor are referred to as diagonal integration.

In 2021, a new generation single cell proteomics technology, SCoPE2 (Petelski et al., 2021), was introduced. This mass-spectrometry based technology uses an isobaric carrier to enhance peptide sequence identification and can quantify over 1000 proteins per cell with high efficiency. Compared to the classical approaches that employ antibodies and can only quantify 50-100 proteins per cell (Levy & Slavov, 2018), SCoPE2 drastically increases the number of proteins identified per cell and brings more possibilities to exciting new biological discoveries.

New opportunities in single-cell proteomics, such as SCoPE2, also bring new challenges. One challenge is integrating SCoPE2 datasets with other modalities for downstream analysis. For example, analyzing protein and mRNA datasets to study post transcriptional regulation requires horizontal integration by transforming the protein features to the corresponding gene features and anchor the datasets by genes. Another challenge is improving the quality of SCoPE2 datasets. SCoPE2 uses peptides as the medium to identify proteins but low abundance peptides generate only a few fragment ions, making confident identification difficult and reducing the number of identified proteins. To increase the number of confident peptide identifications, researchers seek to use other highly informative features such as retention time (RT) to align peptides from different experiments, boosting the confidence of low abundance peptides that are consistent with general RT patterns.

In this thesis, we propose two models to address the aforementioned challenges. The first model is the single cell disentangled multi-modal variational autoencoder (scDMVAE), a neural network model that can handle both horizontal and vertical integration and uncover shared and modality specific information. In scDMVAE, we utilize the struc-

ture of disentangled multi-modal variational autoencoder (DMVAE) (Lee & Pavlovic, 2021) to create a parametric VAE variant that has both shared and modality-specific latent space. Common information is forced into shared latent space using a technique called Product of Experts (PoE) (Hinton, 2002). Vertical integration can be dealt with using scDMVAE directly. For horizontal integration, we learn from Seurat V3 (Stuart et al., 2019) and LIGER (Welch et al., 2019) and treat the overlapping genes as observations. The resulting latent space becomes gene embeddings and the decoder weights mapping gene embeddings to the reconstructed data can be viewed as cell embeddings. We then use a novel attention-based mechanism together with the Shared Nearest Neighbor (SNN) based cell mapping technique in Seurat V3 to align the cell embeddings from different modalities. We will show that scDMVAE can be applied not only to SCoPE2 and sc-RNA-seq datasets but also to other situations such as CITE-seq datasets.

The second model we propose is the principal-curve-based retention time alignment and posterior error probability (PEP) re-estimation model, PCRID. PCRID is a non-linear improvement over DART-ID (Chen, Franks, & Slavov, 2019) and uses principal curves (Tibshirani, 1992) to fit peptide retention times and update the PEP based on the posterior distribution. PCRID increases the identification rate of peptides, reduces the number of missing values and improve the quality of SCoPE2 datasets.

Chapter 2

Multi-Modal integration using scDMVAE

2.1 Background and Related Work

Data integration is a crucial step in multi-omics analysis and various methods have been developed to address this challenge. These methods aim to project datasets from different batches or modalities into a shared latent space of cells. The resulting embeddings can be used in downstream analysis such as cell type clustering, pseudo-time analysis of cell trajectories and so on. Seurat V3 (Stuart et al., 2019) and LIGER ((Welch et al., 2019)) are two popular methods that use non-parametric models to integrate datasets and obviate the distributional differences. Seurat V3 applies Canonical Correlation Analysis (CCA) (Hardoon, Szedmak, & Shawe-Taylor, 2004) for initial dimension reduction and then uses mutual nearest neighbor (MNN) (Haghverdi, Lun, Morgan, & Marioni, 2018) and anchor cells to harmonize the embeddings from different datasets. LIGER resorts to integrative Non-negative Matrix Factorization (iNMF) (Yang & Michailidis, 2016) to take both shared and dataset-specific information into consideration. It then

uses a novel shared nearest neighbor (SNN) method to align embeddings to the same space. Models like SCVI (Lopez, Regier, Cole, Jordan, & Yosef, 2018) and scMVAE (Zuo & Chen, 2021) adopt the deep learning concept Variational Auto Encoder (VAE) (Kingma & Welling, 2013). A VAE consists of two main components: encoder and decoder. An encoder projects dataset to the latent space and decoder reconstructs dataset from latent space. By utilizing encoders and decoders, those models can incorporate VAE to integrate multi-modal/multi-batches datasets and easily specify the best parametric distribution for each dataset. SCVI focuses on horizontal integration with batch correction by introducing learnable batch parameters l_m's for each dataset. scMAVE, on the other hand, emphasizes on vertical integration and adopt the MVAE (Wu & Goodman, 2018) concept. It comprises one set of encoder and decoder for each modality and learns a shared latent space. However, neither of these models can solve both horizontal and vertical integration nor do they utilize dataset-specific information. To address these limitations, We propose scDMVAE, which is a neural network model that can solve both horizontal and vertical integration as well as uncover shared and modality-specific information.

2.2 scDMVAE Framework

scDMVAE consists of two main components, the embedding learning component (Figure 2.1) and the embedding matching component (Figure 2.2). The embedding learning component is responsible for vertical integration, while horizontal integration is achieved by combing the two components and treat features as observations. In this section, we will first define the problem at hand and then then provide a detailed overview of each component.

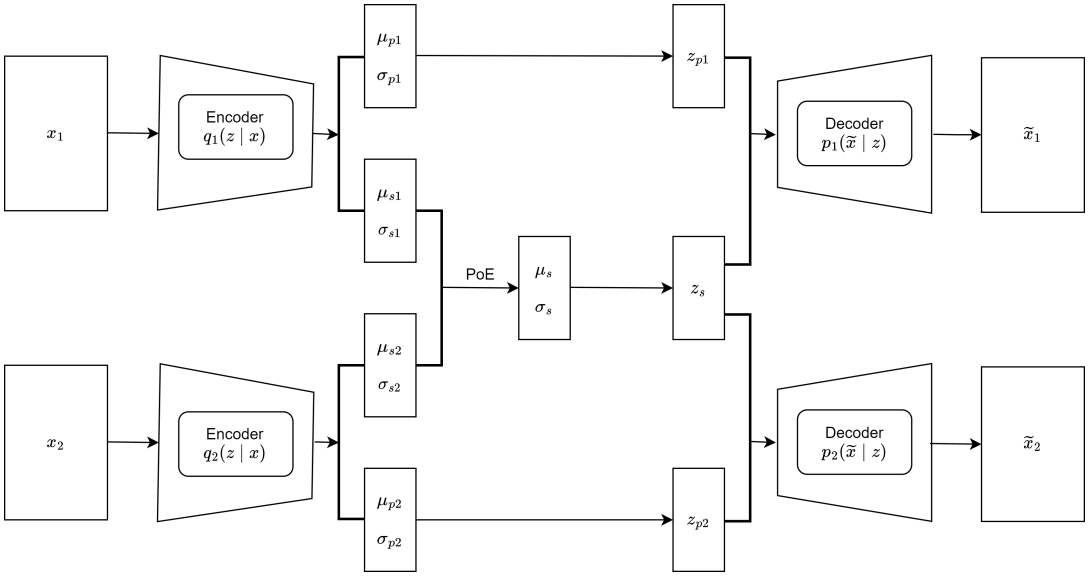


Figure 2.1: **scDMVAE Embedding Learning Framework.** Datasets are fed into encoders and projects to shared and modality-specific latent spaces. Shared information is obtained by using Product-of-Expert (PoE). The Decoders reconstruct datasets based on the latent space.

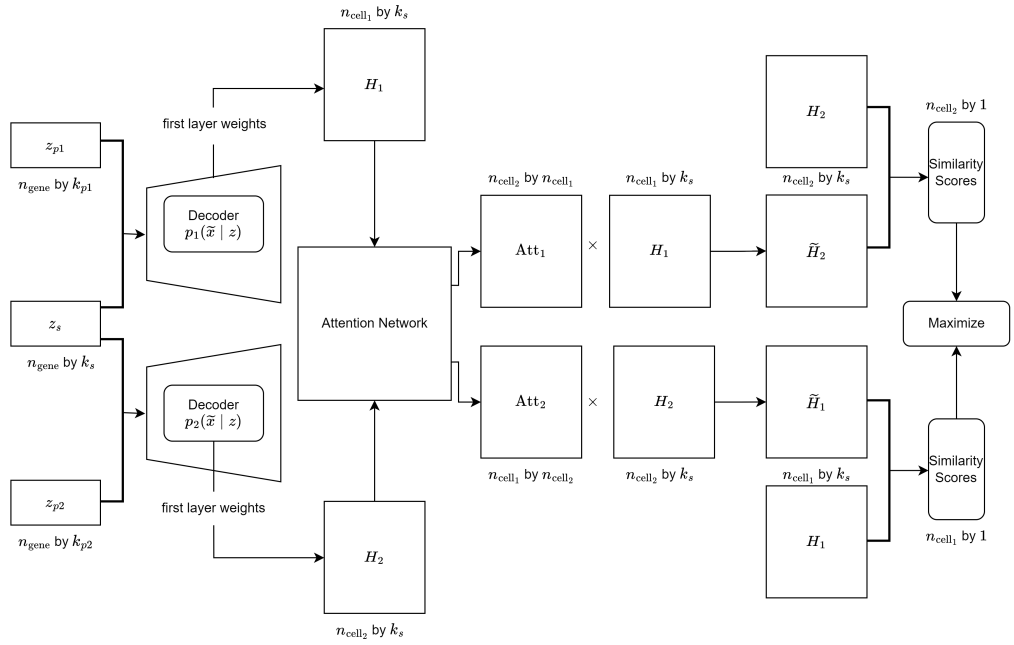


Figure 2.2: **scDMVAE Embedding Matching Framework.** In horizontal integration, the weights of the first layers of decoders H's are treated as cell-level embeddings. They are then fed into the attention network to compute the attention scores. The model tries to find the best attention scores that maximize the given similarity scores between the weighted H matrix and the H matrix from the other modality.

2.2.1 Problem Definition

scDMVAE is designed to perform both the vertical and horizontal integration in single-cell context. In vertical integration, multiple types of molecules are measured in the same cell, with cells serving as the anchors. Our aim with scDMVAE is to learn a cell latent space that contains both shared and modality-specific information separately. This allows researchers to identify cell types that are common to both modalities using the shared space, and to identify sub-cell types that belong to the same cell types but differ slightly in one of the modalities.

In horizontal integration, we have multiple datasets in different modalities but with the same set of features (such as genes for mRNA and proteomics readings), making the features the anchors. The goal is to find a latent space for cells in each modality and match cells in those latent spaces across different modalities. The matched results can then be used to study reading changes across modalities within the same clusters or cell types. Assume we have multi-modal datasets $\mathcal{X} = \{\mathbf{x_1}, \ldots, \mathbf{x_M}\}$ and the formal definitions of the two problems are as follows.

In vertical integration $\mathbf{x_m}$ is the dataset in modality m with n cells and d_m features. The input matrix is $\mathbf{X}_{n\times D}$, where $x_m s$ are concatenated by cells and $D = d_1+d_2+\cdots+d_M$. The goal is to find latent space $\mathbf{Z}_m = (\mathbf{Z}_s, \mathbf{Z}_{p_m})$ in which $\mathbf{Z}_s$ is the shared latent space and $\mathbf{Z}_{p_m}$ is the private latent space of modality m, for m in $\{1, \ldots, M\}$. The dimensions for shared and private spaces are k_s and k_p respectively. The cells are mapped to the same space and alignment is not needed.

In horizontal integration $\mathbf{x_m}$ is a n_m by d matrix for m in $\{1, \ldots, M\}$. The input matrix $\mathbf{X}_{N\times d}$, where $x'_m s$ are concatenated by features and $N = n_1 + n_2 + \cdots + n_M$. The target cell latent spaces are $\mathbf{H}_m = (\mathbf{H}_{s_m}, \mathbf{H}_{p_m})$ for m in $\{1, \ldots, M\}$ with k_s and k_p as the corresponding dimensions. Alignment is required for the resulting latent spaces, which

will be discussed in details in section 2.2.3.

2.2.2 Embedding Learning

The embedding learning component of scDMVAE is built on DMVAE (Lee & Pavlovic, 2021), which is a multimodal variation of variational autoencoder (VAE). In the following section, we will provide a brief overview of VAE and then proceed to discuss the structure of the embedding learning component of scDMVAE. Specifically, we will describe the details of the model inference and reconstruction, as well as the objective function.

Variational AutoEncoder (VAE)

A variational autoencoder is a generative model first introduced by Kingma and Welling. It aims to learn a probabilistic latent space z that maximizes the probability distribution of the input data x, $p(x)$, using the encoder and decoder structure (Figure 2.3). The encoder network, $p(z|x)$, projects data onto latent space while the decoder, $p(\tilde{x}|z)$, reconstructs data from latent space z. The objective of VAE is to maximize $\log p(x)$ but this is intractable. However, the likelihood has a lower bound called evidence lower bound (ELBO): $\log p(x) \geq ELBO = \mathbb{E}_{q(z|x)} \log p(x|z) - KL[q(z|x)||p(z)]$. Here, $q(z|x)$ belongs to an approximation to the posterior $p(z|x)$ and $p(z)$ is the prior. Assuming both $q(z|x)$ and $p(z)$ have Gaussian distributions, the ELBO becomes tractable and we maximize the ELBO term instead of the intractable likelihood of observed data x. The first term in ELBO can be viewed as reconstruction $p(x|z)$ based on the encoder distribution $q(z|x)$. The second term, KL divergence, forces $q(z|x)$ to resemble the Gaussian prior $p(z)$ so that the latent space will not be over fitted with variance zero. Therefore, the KL divergence term can be seen as a regularization term.

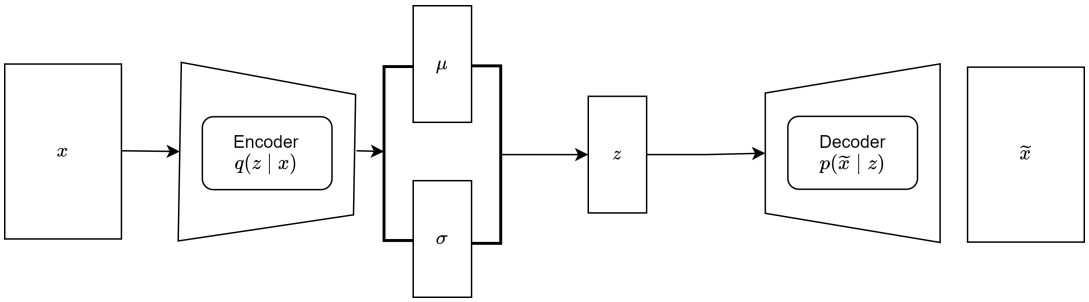


Figure 2.3: **Structure of Variational AutoEncoder (VAE).**

Structure

The embedding learning component of scDMVAE is constructed using the DMVAE framework, which posits that in a multi-modal scenario, the latent space is comprised of a shared space that is common to all modalities, as well as a private space that is specific to each modality. Specifically, the shared latent space should only contain information that is shared across modalities and is constrained using the Product-of-Expert technique, which was introduced by MVAE (Wu & Goodman, 2018). In contrast, the private latent spaces supplement the reconstruction for each modality and contain information that is unique to that modality. Fig 2.1 is an illustration of scDMVAE when there are two modalities. The encoders project observations $x = (x_1, x_2)$ to latent spaces z_1 and z_2 respectively. We assume $z_1 \sim q_{\phi_1}(z|x_1)$ and $z_2 \sim q_{\phi_2}(z|x_2)$ and can be factored into $z_1 = (z_{s_1}, z_{p_1})$, $z_2 = (z_{s_2}, z_{p_2})$. z_{s_1} and z_{s_2} are then aligned by PoE into z_s. The decoders takes (z_s, z_{p_1}) and (z_s, z_{p_2}) as input and reconstruct each modality to $\tilde{x}_1$ and $\tilde{x}_2$

Latent Space Inference

The inference of latent space in scDMVAE consists of two parts. The first part involves using $q(z_s|x_1, \ldots, x_M)$ to approximate the true shared latent space $p(z_s|x_1, \ldots, x_M)$, while the second part involves using $q(z_{p_m}|x_m)$ to approximate the true private latent space $p(z_{p_m}|x_m)$ of modality m for m in $\{1, \ldots, M\}$. One of the challenges in this process is

to establish connections between the jointed posterior $p(z_s|x_1,\ldots,x_M)$ and the shared single-modality inference networks $q(z_{s_m}|x_m)$ for for m in $\{1,\ldots,M\}$. In order to overcome this challenge, MVAE uses Product-of-Experts (PoE) with conditionally independence assumption $p(x_1,\ldots,x_M,z_s)=p(z_s)p(x_1|z_s)\cdots p(x_M|z_s)$:

$$\begin{aligned} p(z_s|x_1,\ldots,x_M) &= \frac{p(x_1,\ldots,x_M|z)p(z_s)}{p(x_1,\ldots,x_M)} = \frac{p(z_s)}{p(x_1,\ldots,x_M)}\prod_{m=1}^{M} p(x_m|z_s) \\ &= \frac{p(z_s)}{p(x_1,\ldots,x_M)}\prod_{m=1}^{M}\frac{p(z_s|x_m)p(x_m)}{p(z_s)} \propto \frac{\prod_{m=1}^{M} p(z_s|x_m)}{\prod_{m=1}^{M-1} p(z_s)} \end{aligned} \tag{2.1}$$

If we further assume that $p(z_s|x_m)$ can be correctly approximated by $q(z_s|x_m)\equiv\tilde{q}(z_s|x_m)p(z_s)$:

$$p(z_s|x_1,\ldots,x_M) \propto \frac{\prod_{m=1}^{M} p(z_s|x_m)}{\prod_{m=1}^{M-1} p(z_s)} \approx \frac{\prod_{m=1}^{M} \tilde{q}(z_s|x_m)p(z_s)}{\prod_{m=1}^{M-1} p(z_s)} = p(z_s)\prod_{m=1}^{M}\tilde{q}(z_s|x_m) \tag{2.2}$$

Equation 2.2 suggests that we can use a product of experts, "prior expert" together with "modality-specific experts", to approximate the joint posterior. While the PoE is generally intractable, it has a closed form solution when each term is assumed Gaussian, which is the distributional assumption of latent space in VAE. In this case, the product of Gaussian experts is also Gaussian with mean $\mu=(\sum_i \mu_i T_i)(\sum_i T_i)^{-1}$ and covariance $V=(\sum_i T_i)^{-1}$, where μ_i, V_i are the parameters of the ith Gaussian expert and $T_i=V_i^{-1}$.

In this context, vertical integration refers to integrating information from different modalities features at the cell level, while horizontal integration refers to integrating information of different cells from different modalities at the feature level. In both case, we aim to construct cell level latent space. However, the treatment of the data will differ depending on the type of integration being performed. In vertical integration, cells are treated as observations and $z_m=(z_s,z_{p_m})$ is the desired latent space for each modality m

in $\{1, \ldots, M\}$. On the other hand, in horizontal integration, data matrices are transposed and features are treated as observations since they are anchored by features. The resulting latent space z will be at the feature level and to obtain the corresponding cell-level latent spaces, the first layer of decoders is shaped to have $k_s + k_p$ by n_m and the transpose of the corresponding weight matrices of the first layers $H_m = W_m^T$ are taken as the n_m by $k_s + k_p$ cell latent spaces, for m in $\{1, \ldots, M\}$. These H matrices are then aligned later to obtain correspondences among cells.

Reconstruction

One advantage of scDMVAE is that it specifies a dedicated parametric distribution for each modality (e.g. Zero-Inflated Negatvie Binomial for mRNA and Log Normal for proteomics) and seeks to find MLEs for those parameters. In this way, each modality can be treated with the most suitable distribution instead of using $L2$ norm as reconstruction loss. To be more specific, the goal is to learn the best parameter estimations through decoders that maximize the following equation:

$$
\begin{aligned}
\prod_{m=1}^{M} p(\tilde{x}_m | x_1, \ldots, x_M) &= \prod_{m=1}^{M} \int p(\tilde{x}_m, z_s, z_{p_m} | x_1, \ldots, x_M) dz \\
&= \prod_{m=1}^{M} \int p(\tilde{x}_m | z_s, z_{p_m}) p(z_s, z_{p_m} | x_1, \ldots, x_M) dz \\
&= \prod_{m=1}^{M} \int p(\tilde{x}_m | z_s, z_{p_m}) p(z_s | x_1, \ldots, x_M) p(z_{p_m} | x_m) dz \\
&= \prod_{m=1}^{M} \mathbb{E}_{p(z_s | x_1, \ldots, x_M) p(z_{p_m} | x_m)} [p(\tilde{x}_m | z_s, z_{p_m})] \\
&\approx \prod_{m=1}^{M} \mathbb{E}_{q(z_s | x_1, \ldots, x_M) q(z_{p_m} | x_m)} [p(\tilde{x}_m | z_s, z_{p_m})] \\
&= \prod_{m=1}^{M} \mathbb{E}_{q(z_s | \mathbf{x}) q(z_{p_m} | x_m)} [p(\tilde{x}_m | z_s, z_{p_m})]
\end{aligned}
\tag{2.3}
$$

Objective Function

Combining all parts of the model, the objective function can be written as:

$$\begin{aligned} ELBO(x_1, \ldots, x_M) = & \sum_{m=1}^{M} \lambda_m \, \mathbb{E}_{q_\phi(z_s|\mathbf{x}) q_\phi(z_{p_m}|x_m)} [\log p_\theta(\tilde{x}_m | z_s, z_{p_m})] \\ & - \lambda_p KL(q_\phi(z_{p_m}|x_m) || p(z_{p_m})) - \lambda_s KL(q_\phi(z_s|\mathbf{x}) || p(z_s)) \end{aligned} \tag{2.4}$$

In equation 2.4, the λ's are hyper-parameters to balance the reconstructions as well as KL divergences; θ and ϕ are the weight parameters in encoders and decoders respectively; priors $p(z)$ are chosen to be standard Gaussian distributions and KL divergences can be viewed as regularization terms.

2.2.3 Embedding Matching

In horizontal integration, the embeddings H_m of different modalities are of different scales. To establish correspondence across modalities, we build a attention-based matching model to evaluate the similarities among cells of different modalities. This matching model uses shared part of the embeddings and is designed for two modal situation. In this section, we will give an overview on the embedding matching component with two modalities. For more than two modalities, we can select one modality as reference and match other modalities to it.

Attention-based Matching Structure

In many horizontal integration models, the similarities of cells across different datasets are typically measured by the $L2$ norm in the latent space. However, this approach is not suitable for multi-modal integration, as datasets from different modalities are often

distributed differently, and the connection among datasets anchored by the same gene sets can be complicated.

To address this issue, we developed a novel attention-based method inspired by (Vaswani et al., 2017), which uses attention score as a measurement of similarity. As shown in Figure 2.2, the embedding H matrices are fed into an attention network that learns two cross-attention score matrices (one n_1-by-n_2 and one n_2-by-n_1) where the row sums equal to one. The attention scores in each row represent how much attention we should pay to the cells in the other modality when looking at the corresponding cell in current modality. It can also be viewed as weighted average of cells in the other modality.

We want this weighted average to be as similar to the target cell as possible. The default similarity metric we use is cosine similarity, which does not depend on the magnitudes of embeddings but only on the angles. However, one can use a different similarity metric based on the nature of target datasets. Based on the attention matrices, we can identify the correspondence of cells across modalities. If the attention scores are mutually high for two cells, then they are matched and considered to correspond to each other.

In the case that all the high attention scores of a cell type are not mutual, this cell type can be identified as belonging to a unique cluster that only exists in its modality. In the following section, we will discuss in detail how to formally find the corresponding cells (anchors) and use those anchors to perform integration and label transfer.

2.2.4 Integration Using Anchors

Matching cells across modality using only the attention matrices can be problematic, as not all correspondences are equally important. Some correspondences are strong, some are weak, and some may even be incorrect due to random noise. To address this issue, we adopt the anchor scoring and weighing strategy from Seurat V3 (Stuart et al.,

2019), which uses scores and weights to guide the integration. This approach allows us to prioritize the most reliable anchors and reduce the influence of noise and weak correspondences. We will discuss the details of this strategy in the following section.

Identification of Anchor Correspondences

The first step in our integration approach is to identify anchors using the attention matrices. We treat attention scores as distance measurements and use them to find the K-nearest neighbors (KNNs) for each cell within the other dataset. If two cells from different modalities are within the KNNs of each other, they are referred to as anchors. The parameter k_{id} controls the number of nearest neighbors used in this step.

Anchor Scoring

The second step of our method involves evaluating the quality of anchors using anchor scores. For each anchor pairs, we calculate the within and across k_{score} nearest neighbors for each of the two cells and use the total number of overlap cells as a score for this anchor pairs. To find across nearest neighbors, we use attention scores as our similarity metric, while for the within nearest neighbors, we use the $L2$ norm in latent space. The score metric is inspired by Shared Nearest Neighbor (SNN) graph clustering algorithm (Jarvis & Patrick, 1973; Houle, Kriegel, Kröger, Schubert, & Zimek, 2010), which measures the similarities between nodes based on the number of common neighbors. This approach is more robust in high dimension than traditional distance-based similarity metrics, making it well-suited for our multi-modal integration problem where distances within and across datasets are not comparable. Seurat V3 has shown that in practice the correct anchors have significantly high scores than incorrect anchors. Therefore, we use those scores to down-weight the influence of incorrect anchors. To avoid extreme outliers, the scores are re-scaled to be between 0 and 1 using 0.01 and 0.90 quantiles.

Anchor Weighting

The third step involves establishing connections between anchor cells and other cells using the scores obtained in the second step. The strength of a connection is determined by two factors: the distances from a cell to the anchor cell within its modality and the corresponding anchor score. To quantify this connection, we use a weight matrix W of size n_{cell} by n_{pairs}. For each cell c, we identify its k_{weight} nearest anchor cells within its own modality using latent space embeddings. We calculate the weights based on the scores S_{a_i} and the distances between the cell c and its nearest anchor cells as follows:

$$\begin{aligned} D_{c,i} &= (1 - \frac{dist(c, a_i)}{dist(c, a_{k_{weight}})})S_{a_i} \\ \tilde{D}_{c,i} &= 1 - e^{-D_{c,i}/2} \\ W_{c,i} &= \frac{\tilde{D}_{c,i}}{\sum_{j=1}^{k_{weight}} \tilde{D}_{c,j}} \end{aligned} \tag{2.5}$$

Label Transferring

To transfer labels from one modality to the other, we first create a binary n_{label} by n_{pairs} classification matrix L. Each column of L correspond to an anchor pair, while each row corresponds to a label. For a given anchor pair, the column of L has a value of one in the row that corresponds to the label of the reference cell, and zeros for all other entries. To account for label imbalance, we average each row of L by the total number of cells that belong to that label. The label transfer is characterized by equation 2.6, where P contains the predicted label scores for all cells.

$$P = LW^T \tag{2.6}$$

As a conclusion, the predicted label scores for a given cell are obtained by taking

a weighted average of the reference cells' labels, where the weights are determined by the distances from the cell to the anchor cells in its modality, the corresponding anchor scores, and the attention scores across modalities.

2.3 Results

2.3.1 Data Distribution

mRNA datasets contains UMI counts of gene expressions with excessive missing values due to technical reasons. Therefore, the dataset has a huge amount of zeros which can be due to either technical issues or biological variations. To better distinguish the biological variations from technical issues, the Zero-Inflated Negative Binomial (ZINB) distribution is used to model mRNA datasets. For protein datasets, based on different technologies, the datasets have different distributions. For CITE-seq datasets containing counts of antibodies, we use Negative Binomial distribution to describe them. For SCoPE2 datasets, we model them with log-normal distribution.

The Negative Binomial Distribution can be characterized with its mean $\mu \in \mathbb{R}^+$ and dispersion $\phi \in \mathbb{R}^+$ as is shown in equation 2.7:

$$\begin{aligned} NB(y|\mu,\phi) &= \binom{y+\phi-1}{y}\left(\frac{\mu}{\mu+\phi}\right)^y\left(\frac{\phi}{\mu+\phi}\right)^\phi \\ \mathbb{E}[Y] = \mu, \mathrm{Var}[Y] &= \mu + \frac{\mu^2}{\phi} \end{aligned} \tag{2.7}$$

The ZINB is a Negative Binomial distribution with an additional spike at zero. The probability of being zero due to technical issue is denoted using π. The ZINB can be viewed as a combination of two steps. The first step is a Bernoulli distribution with success probability $1-\pi$ and π is the probability of being zero due to technical issue (missing value). With probability $1-\pi$, the variable value can be observed and is

distributed according to a Negative Binomial distribution. The probability mass function of ZINB is:

$$\begin{aligned} P(y = 0|\pi, \mu, \phi) &= \pi + (1 - \pi)NB(y = 0|\mu, \phi) \\ P(y = i|\pi, \mu, \phi) &= (1 - \pi)NB(y = i|\mu, \phi), i \in \mathbb{Z}^+ \end{aligned} \tag{2.8}$$

2.3.2 Vertical integration

scDMVAE is a powerful tool for integrating multi-modal datasets from the same cell. One of the main challenges in this field is how to balance the preservation of heterogeneity, such as unique cell types, across modalities while also uncovering homogeneous information. While vertical integration models often overlook heterogeneous information, it is crucial in downstream analysis. To address this issue, scDMVAE introduces modal-specific latent space in single cell multi-modal analysis, which provides access to information that is not present in the shared latent space. The embedding generated by scDMVAE contains a representation that reveals both shared and private information, resulting in more accurate cell characterization compared to analyzing each modality separately. In this section, we demonstrate the usefulness of scDMVAE by analyzing simulation datasets and the cord blood mononuclear cells (CBMC) dataset.

Simulation

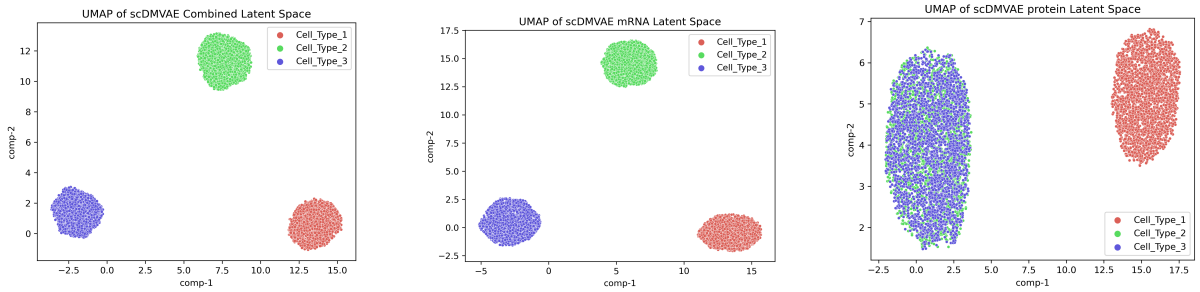
In this simulation, we evaluated the ability of scDMVAE to disentangle homogeneous and heterogeneous information by utilizing the shared and modality-specific latent space.The dataset contained both protein and mRNA information for three cell types, where cell type 2 and 3 were subtypes of a larger cell type, and could only be differentiated by mRNA data. Only two cell types could be distinguished by protein information. The dataset consisted of 7500 cells with 2500 cells in each cell type and 200 genes. The

mRNA distribution followed a Zero-Inflated Negative Binomial (ZINB) distribution with a missing probability of 0.5, while the protein data followed a log-normal distribution.

Figure 2.4 depicts the combined embedding (z_s, z_{p_1}, z_{p_2}), along with the embeddings of the shared and modality-specific space (z_s, z_{p_i}) for each modality. The combined embedding exhibited the highest resolution and identified three distinct cell types. The mRNA and protein embeddings also accurately identified their corresponding cell types as per the simulation ground truth. To further explore what information is contained in z_s, z_{p_1}, z_{p_2} respectively, we show the UMAP of shared and modality specific embeddings separately in Figure 2.5. the shared space contained only two clusters, one representing cell type 1, and the other containing both cell type 2 and 3. This was as expected since the shared space only captured shared information, i.e., two cell types. The modality-specific space, on the other hand, complemented the shared space and contained information specific to each modality. In the UMAP of the mRNA modality-specific space, two clusters were observed: a smaller one containing cell type 2 and a larger one containing cell type 1 and 3. This complemented the shared space since the combined shared and mRNA modality-specific space contained enough information to distinguish all three cell types. The protein modality-specific space yielded similar results since the shared space contained enough information to differentiate between the two cell types in the protein data. Hence, the protein modality-specific space appeared to be random in the UMAP. In conclusion, the simulation results demonstrated that scDMVAE can effectively disentangle shared and modality-specific information in vertical integration tasks, thanks to its dedicated modules for each aspect.

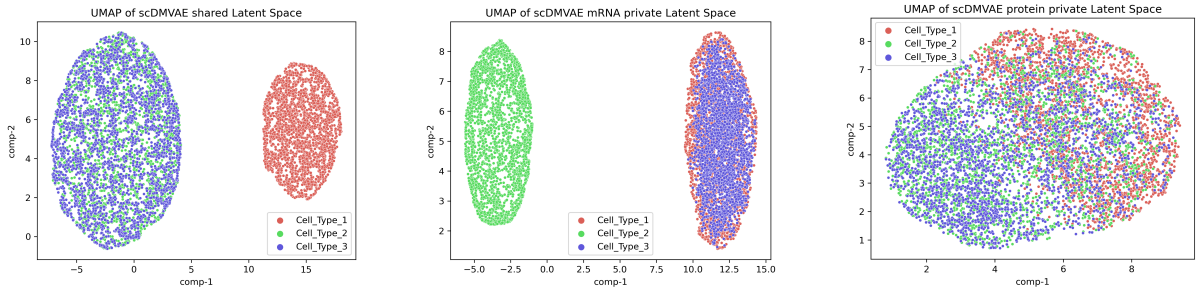
Real data: Cord blood mononuclear cells (CBMC) dataset

The CBMC dataset is generated using CITE-seq technology in (Stoeckius et al., 2017), which allows simultaneous RNA sequencing and surface protein quantification



(a) Combined Embedding (b) mRNA Embedding (c) Protein Embedding

Figure 2.4: **Simulation Results: UMAP of combined embeding, mRNA embedding and protein embedding.** (a) shows the combined embedding z_s, z_{p_1}, z_{p_2}) and all three cell types are well separated; (b) and (c) shows the mRNA and protein embedding (z_s, z_{p_i}) respectively and the clusters are in accordance with the ground truth.



(a) Shared Embedding (b) mRNA Modality-Specific Embedding (c) Protein Modality-Specific Embedding

Figure 2.5: **Simulation Results: UMAP of shared embeding, mRNA and protein modality-specific embedding.** (a) shows the shared embedding z_s and it contains the two main cell types; (b) and (c) shows the mRNA and protein modality-specific embedding z_{p_i} respectively; (b) contains enough information for distinguishing cell type 2 and 3 and information in (c) is quite noisy indicating that most of the information in protein is captured by the shared embedding.

using available antibodies at the single-cell level. The dataset consists of 20,501 mRNA genes and 10 protein antibodies from 8,617 cells.To reduce the dimensionality of the mRNA dataset, we selected the 2,000 most variable genes using Seurat's tools (Hao et al., 2021). Due to technical reasons, the mRNA dataset contains a large number of excessive missing values, resulting in a significant number of zeros. To differentiate biological variation from technical variation, we modeled the mRNA dataset using a Zero-Inflated Negative Binomial (ZINB) distribution, while the surface protein data, which contains antibody counts, was modeled using a Negative Binomial (NB) distribution.

The UMAP anlaysis of raw data (Figure 2.6a and 2.6b) shows that most cell types are separated consistently between the two modalities. However, some are mixed with other cell types in either modality. For example, CD8 T and Memory CD4 T cells overlap in mRNA but can be well-distinguished in protein. Conversely, T/Mono doublet cells are entirely mixed with CD14+ Mono cells, but can be separated in protein. In the protein UMAP embedding, Eryth and Mouse cells, as well as Memory and Naive CD4 T cells are mixed together, but they can be identified in mRNA. It is desirable in multi-modal analysis to preserve those diversified information in the learned embedding while learning the common information. Figure 2.6c, 2.6d and 2.6e show the integrated UMAP of scDMVAE embedding using 3 different dimension combinations. The previously entangled cell types in either modality are well-distinguished in the integrated analysis, enabling clear identification.

Seurat V4 (Hao et al., 2021) are designed for the same purpose using graph-based method. It builds a weighted nearest neighbor (WNN) graph, where the weights are calculated using independent KNN graph of each modality. One potential drawback of Seurat V4 is that the result is sensible to the choice of dimension reduction technique. This is because the KNN of each modality is built using different dimension reduction methods. The other potential issue is that the shared and modal-specific information

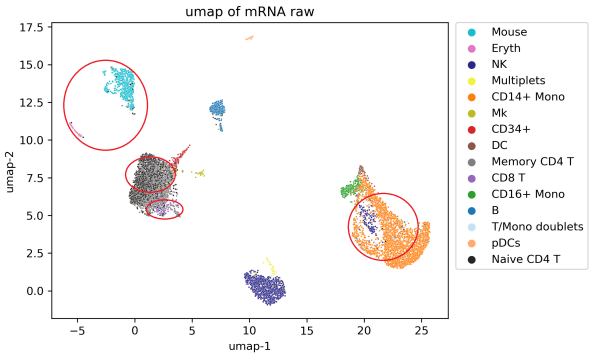
are not distinguishable in the WNN graph. Compared to Seurat v4, scDMVAE generate embeddings using an end-to-end automated approach, without weighting. The embeddings are generated using the same model and shared and modal-specific information is accessible and comparable.

2.3.3 Horizontal integration

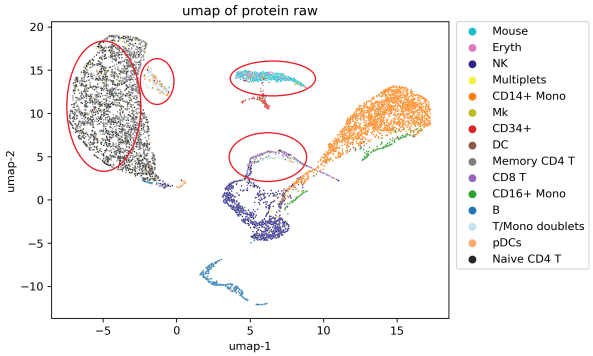
Simulation

In this simulation, we assessed the ability of scDMVAE to align mRNA cells and protein cells. We simulated 7500 mRNA cells and 7500 protein cells with 200 genes in common. The dataset comprises three cell types that are common to both mRNA cells and protein cells, with 2500 cells in each cell type and modality combination. We assumed that the means of cells from the same cell type followed the relation $\mu_{mRNA}/\mu_{protein} = 10$ for each gene to emulate the central dogma of biology. The means of the three cell types exhibited distinct patterns among the 200 common genes (Figures 2.7a, 2.7b, and 2.7c). To increase the difficulty of integration, we set the missing probability of gene 51 to 150 to be 0.9, such that the observed patterns for cell types 1 and 2 would be very similar and challenging to distinguish without modeling the missing probability (Figures 2.7d, 2.7e, and 2.7f). In this scenario, we pretended that the labels of protein cells were unknown and aimed to transfer the labels from mRNA to protein.

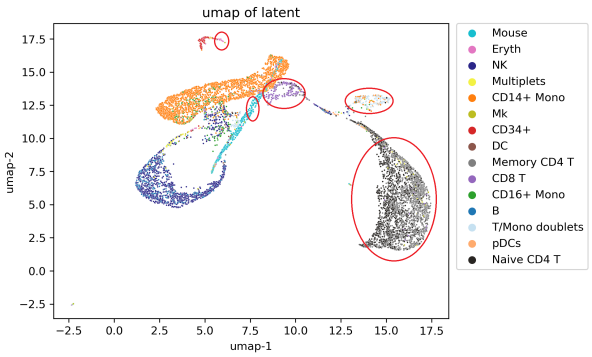
We compared scDMVAE with LIGER using the simulation dataset. In scDMVAE, the cells are aligned using the attention network together with anchor strategy and Labels are transferred from mRNA cells to protein cells. On the other hand, the LIGER-learned embeddings of mRNA and protein will be mapped to the same space by choosing one embedding space as reference and mapping the other to it. The transferred labels in LIGER are obtained by majority voting of the k-nearest mRNA cells around the target



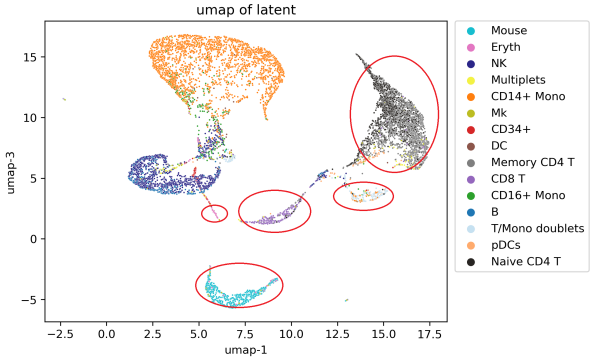
(a) Raw mRNA



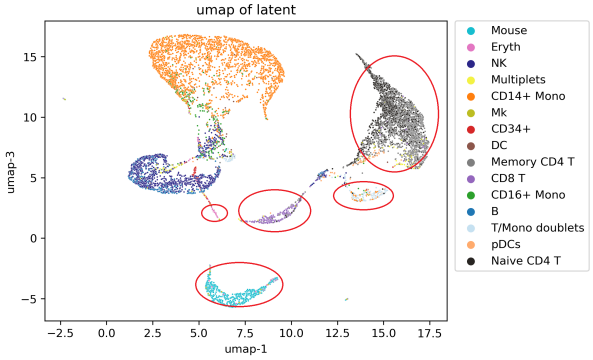
(b) Raw Protein



(c) scDMVAE: UMAP dimension 1 and 2

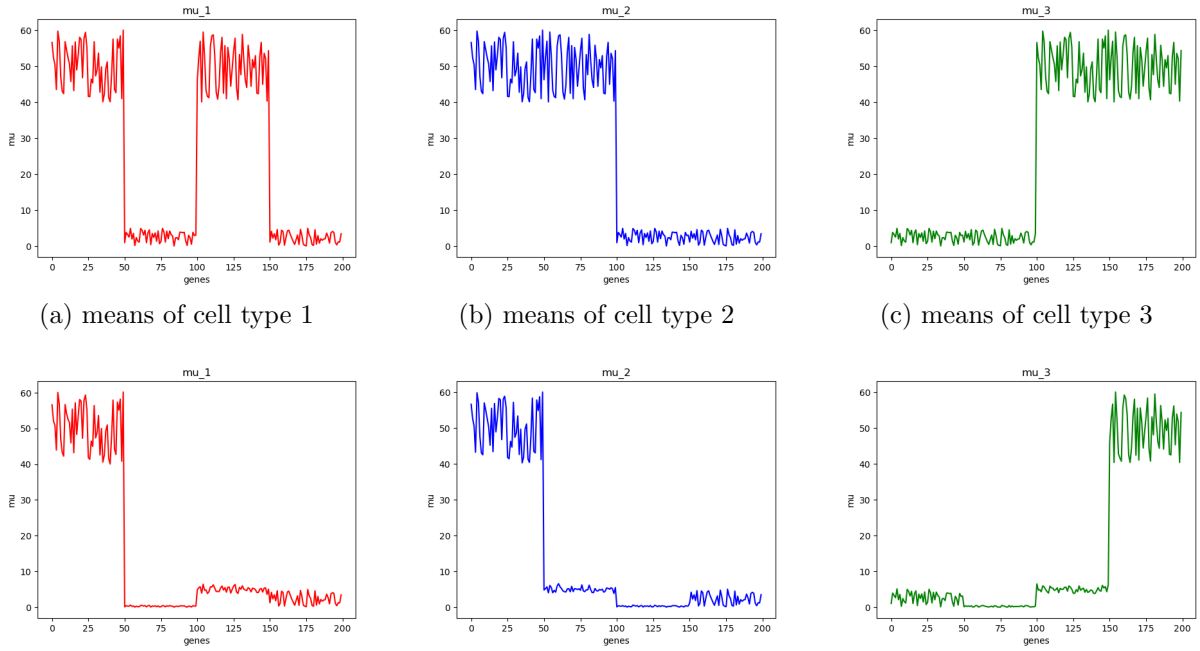


(d) scDMVAE: UMAP dimension 1 and 3



(e) scDMVAE: UMAP dimension 2 and 3

Figure 2.6: **UMAP of Raw CBMC Data.** (a) and (b) are the UMAP of mRNA and Protein raw data respectively and show the heterogeneity between the two modalities. Eryth Cells and Mouse Cells, Naive CD4 T and Memory CD4 T cells can be distinguished in (a) but not in (b); CD 14+ Mono and T/Mono doublets cells, CD8 T and Memory CD 4 T cells are well-separated in (b) but not in (a); (c)-(e) shows the UMAP of scDMVAE embedding with UMAP dimension 3. All the cell types mentioned above are disentangled and clearly separated.



(a) means of cell type 1 (b) means of cell type 2 (c) means of cell type 3

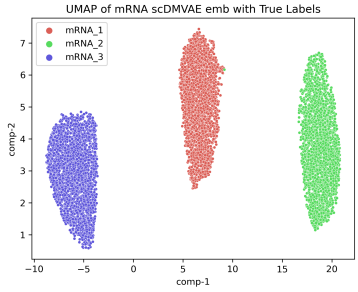
(d) means of cell type 1 with missing values (e) means of cell type 2 with missing values (f) means of cell type 3 with missing values

Figure 2.7: **Means of 200 Genes for All Three Cell Types.** The cell types are defined by the patterns of means of the 200 genes and the patterns are shown in (a) (b) and (c); For the same gene, $\mu_{mRNA}/\mu_{protein} = 10$; The observed means with missing values are shown in (d) (e) and (f) with 90% missing rate for gene 51 to 150; The patterns of cell type 1 and 2 are similar to each other with missing values.

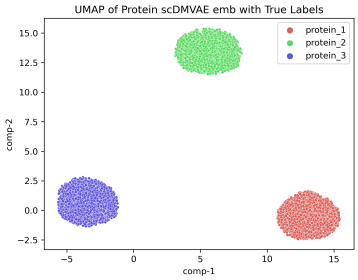
protein cells. We compared the transferred labels with the ground truth and found that scDMVAE correctly assigned all transferred labels, while LIGER had a 7.88 % error rate when using protein as the reference embedding. The alignment process in LIGER can align a proportion of cells from different cell types, leading to label assignment errors. The choice of reference embedding also affects the results, with 25 % and 7.88 % wrongly-assigned labels using mRNA and protein as reference, respectively. Figure 2.8 presents UMAP visualizations of embeddings labeled with true and transferred labels, and Figure 2.9 shows that wrongly-assigned labels in LIGER are mainly due to the alignment algorithm.

Real data: SCoPE2 and sc-RNAseq testes datasets

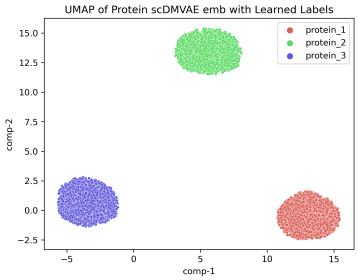
Horizontal integration is a challenging task in multi-modal analysis. Because in multi-modal single cell analysis, datasets from different modalities are distributed differently. The connection among datasets anchored by the same gene sets can be complicated. Most existing models perform pseudo multi-modal analysis which focuses on batch correction of datasets from the same modality, where the relation among datasets can be modeled with learnable scalar (Lopez et al., 2018) or linear vector correction (Haghverdi et al., 2018; Stuart et al., 2019). LIGER (Welch et al., 2019) utilizes iNMF (Yang & Michailidis, 2016) to integrate different modalities. By the parts-based nature of NMF, cells with the same factor as their highest factor loadings are clustered together. To increase the stability of clustering, LIGER introduced shared factor neighborhood (SFN) graph to assign cluster based on the neighbor averaging information together with the highest loadings. Quantile normalization is performed to normalize the loadings in joint clusters to integrate cells to the same latent space. There are three main drawbacks of LIGER. Firstly, it matches cells using only the information of its the highest loading factor and information of other dimensions is only used to build SFN graph. Secondly, it lacks of the options to make



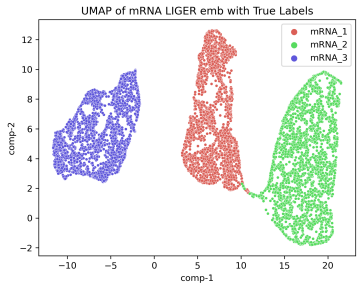
(a) scDMVAE: UMAP of mRNA Embedding with True Labels



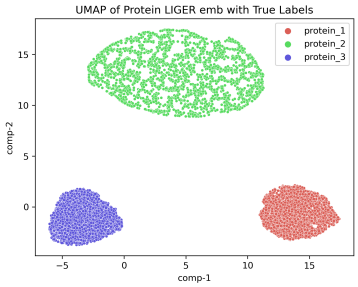
(b) scDMVAE: UMAP of Protein Embedding with True Labels



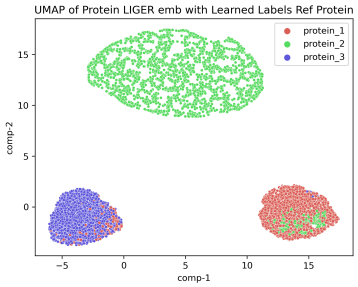
(c) scDMVAE: UMAP of Protein Embedding with Transferred Labels



(d) LIGER: UMAP of mRNA Embedding with True Labels

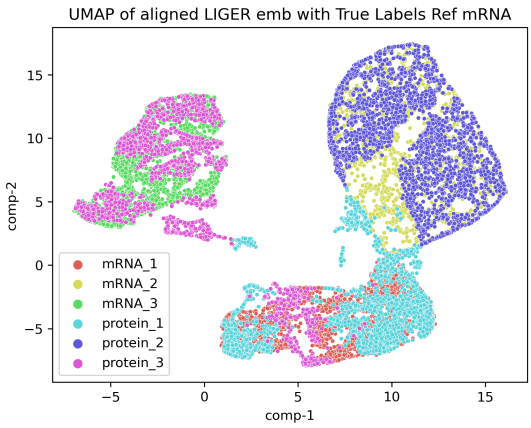


(e) LIGER: UMAP of Protein Embedding with True Labels

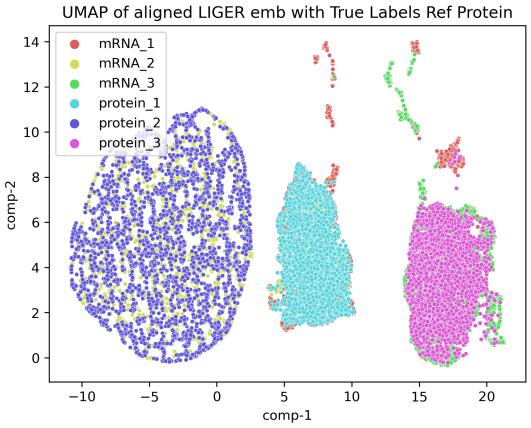


(f) LIGER: UMAP of Protein Embedding with Transferred Labels

Figure 2.8: **UMAP of Simulation Embeddings with True and Transferred Labels of scDMVAE and LIGER.** scDMVAE results are shown in (a) (b) and (c): the cell types are well separated and all the transferred labels are correct; LIGER results are shown in (d) (e) and (f): we use protein as reference because it gives better results; the mRNA embedding are slightly overlapped between cell type 1 and 2; 7.88 % of the labels are wrongly assigned



(a) UMAP of Aligned LIGER embedding with True Labels with mRNA as Reference

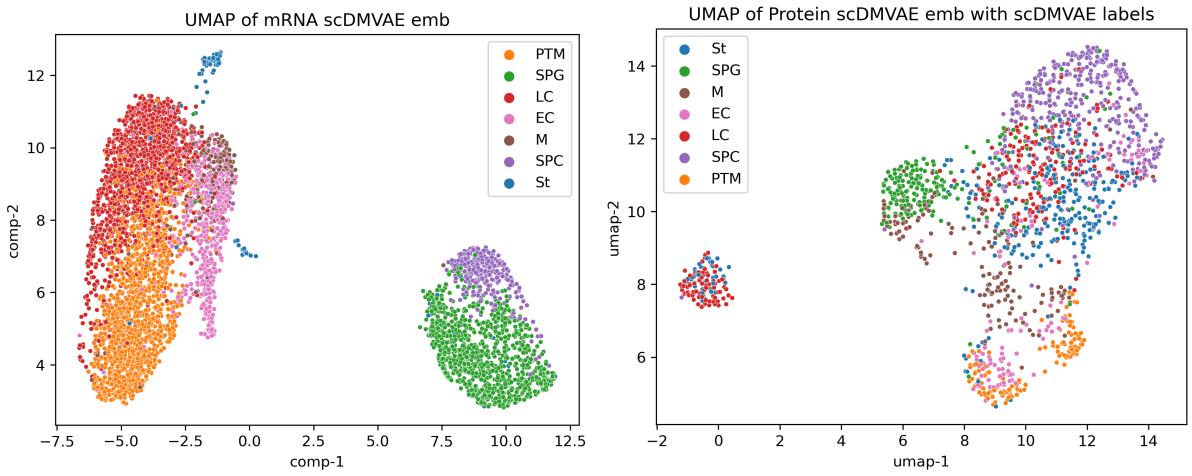


(b) UMAP of Aligned LIGER embedding with True Labels with Protein as Reference

Figure 2.9: **UMAP of LIGER Aligned Embeddings with True Labels.** The result is sensitive to the choice of reference: (a) and (b) are using mRNA and protein as reference with 25 % and 7.88 % wrongly-assigned labels respectively; In both cases the alignment process of LIGER aligns a proportion of cells from different cell types.

dedicated changes for a specific problem (e.g. distributions and similarity metrics for matching cells across modalities). Thirdly, it assumes linearity in the decomposition of modalities. However the relation can be complicated and nonlinear in reality. scDMVAE is more flexible than LIGER in the sense that it can utilize all dimensions in similarity metrics and can change distributions and similarity metrics accordingly. In addition, scDMVAE can incorporate nonlinear transformations in the encoder to learn a non-linear transformed representation. We will compare scDMVAE and LIGER on their ability to integrate the testes dataset sequenced by SCoPE2 and an independently generated sc-RNA dataset.

The testes dataset of SCoPE2 contains 1547 cells with 2428 proteins, while the mRNA dataset have 4955 cells with 32738 genes.Since the datasets have different dimensions, we perform diagonal integration by selecting the highly variable proteins, converting them to their corresponding genes, and finding the intersection of genes sets of both modalities. The final common gene set contains 174 genes. The mRNA dataset contains

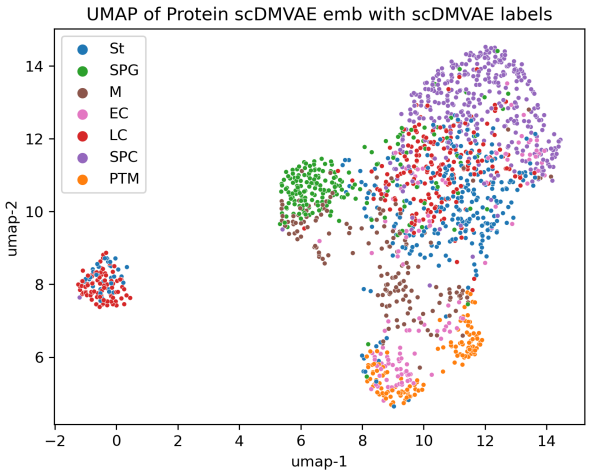


(a) scDMVAE mRNA (b) scDMVAE Protein

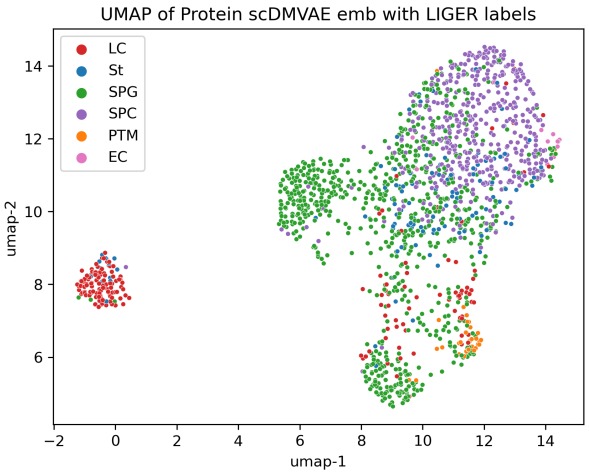
Figure 2.10: **UMAP of scDMVAE embeddings with protein labels learned using scDMVAE.** (a) and (b) are the UMAP of scDMVAE mRNA and Protein embeddings respectively. General cell type relations in mRNA are preserved in transferred protein labels with some noises in the St and LC cells.

expert annotated labels, whereas protein dataset doesn't. Therefore, in this task, we focus on transferring label information of mRNA dataset to SCoPE2 dataset to facilitate downstream analysis. In scDMVAE, we assume ZINB distribution (equation 2.8) for mRNA and Lognormal distribution for SCoPE2 dataset. In the matching step, cosine similarity is used because cells use mRNA to synthesis proteins and we expect the gene measurements of protein and mRNA cells from the same cell types to be approximately proportional. By the nature of reconstruction in scDMVAE, the similar property applies to the shared part H embedding matrices. Therefore we used the shared part of H matrices for integration, which is intuitive because we can only transfer the shared information from one modality to the other.

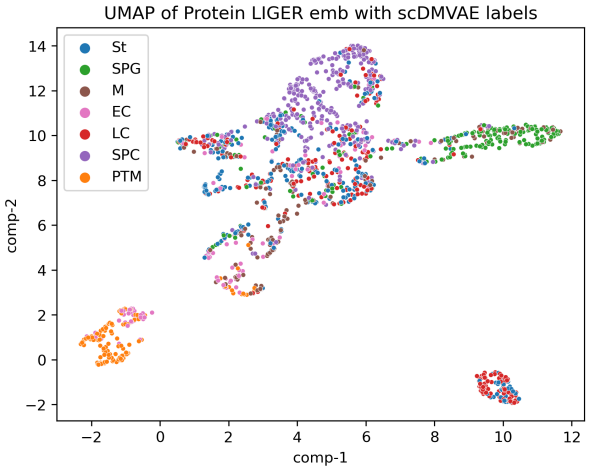
The horizontally integrated scDMVAE representations for mRNA and Protein are shown in Figure 2.10, which demonstrates that general cell type relations are preserved with some noises. However, it should be noted that 2D plots may not be sufficient to reveal some high dimensional biological variations. A comparison of the protein em-



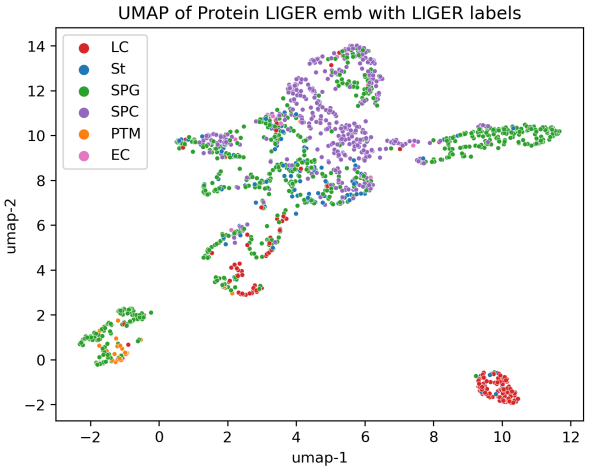
(a) scDMVAE Protein embedding with scDMVAE labels



(b) scDMVAE Protein embedding with LIGER labels



(c) LIGER Protein embedding with scDMVAE labels



(d) LIGER Protein embedding with LIGER labels

Figure 2.11: **UMAP of scDMVAE and LIGER embeddings labeled using both strategies.** The scDMVAE labels are better separated than LIGER labels and are more consistent with cell type relations in mRNA. The label SPG takes a big proportion of LIGER labels and scatters at many places. The SPG labels mingling with PTM cells should be labeled as PTM or EC; In both scDMVAE and LIGER labels, LC and St are scattered in the middle part and in the isolated small cluster

beddings of both scDMVAE and LIGER annotated by labels learned by LIGER and scDMVAE is presented in Figure 2.11. The scDMVAE labels are better separated than LIGER labels and are more consistent with cell type relations in mRNA. The SPG labels takes a significant proportion of LIGER labels and are dispersed across many places. Some of them mingling with PTM cells should be labeled as PTM or EC cells, which is evidenced in the heat maps presented in Figure 2.12. It is worth noting that both scDMVAE and LIGER labels show that LC and St are scattered in the middle part and in the isolated small cluster. The reason behind this could either be biological or systematic noise.

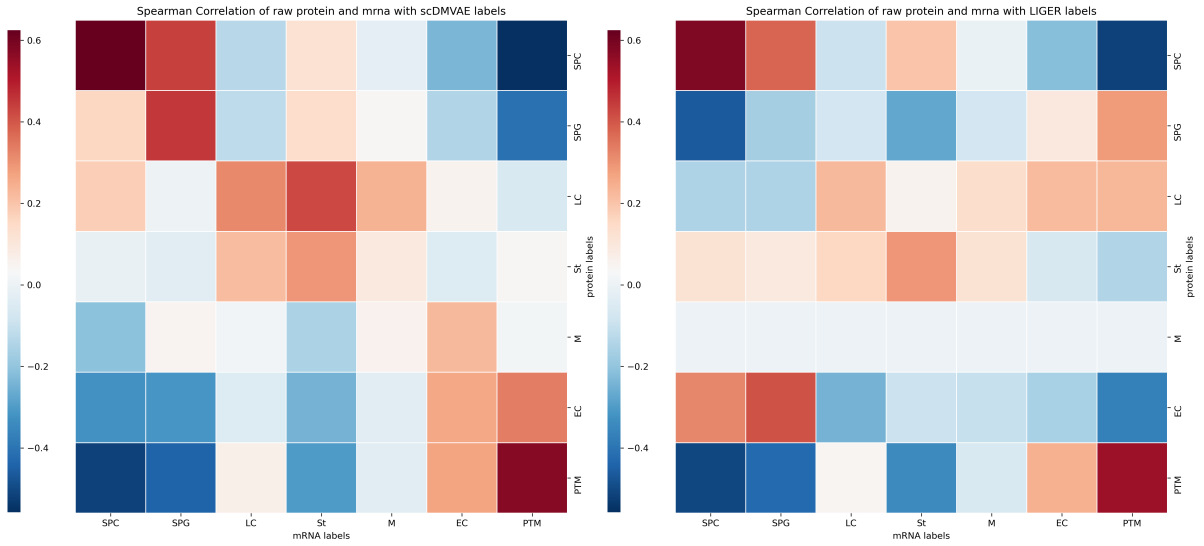
To more rigorously compare the quality of label transformations, we calculated the mean of each cell type in both mRNA and protein raw datasets and computed the pairwise correlation among cell types. Since the datasets have different distributions, we used Spearman ranked correlation to account for this difference. We expect cells from the same cell type but different modalities to have high correlations by their biological nature. The heat maps in Figure 2.12 show that the labels provided by scDMVAE are more consistent with the raw structure of the datasets, as indicated by the highlighted diagonal. The M cell type in scDMVAE is more correlated with EC compared to M in mRNA because M and EC are not distinguishable in scDMVAE mRNA embedding (Figure 2.10a). The off-diagonal correlations are also consistent with biological cell type clusters, with three major clusters formed by SPC and SPG, EC and St, M and EC and PTM. These three main clusters are also shown in Figure 2.10a. However, in LIGER labels, the SPGs are more similar to PTMs than SPGs, indicating that the widely assigned SPG labels are inappropriate. EC labels in LIGER are also not consistent with mRNA. Furthermore, no M label is assigned in LIGER, and the corresponding Spearman correlation values are marked as zeros. LC and St labels scatter in two parts in Figure 2.10b: one in the isolated cluster, the other in the middle part. This may be caused by the existence of subtypes

of cells that were not distinguished by the experts when annotating mRNA cells. Figure 2.10a supports this speculation because there are two clusters of St cells. Figure 2.12a also supports this because LC and St cells only correlate with each other. We conducted similar correlation analysis using cosine similarity in latent spaces of scDMVAE and LIGER. The corresponding heat maps are shown in Figure 2.13. The high diagonal values in scDMVAE heat map (Figure 2.13a) indicate high correspondences between the same cell types of different modalities in the shared latent space, which are not presented in the LIGER embeddings (Figure 2.13b).

We employed two metrics to assess the performance of scDMVAE and LIGER. The first metric evaluated the quality of label assignment by measuring split correlations of proteins using peptide information. Peptides are the building blocks of proteins and provide information that characterizes them. We randomly split each protein's peptides into two groups, calculated the means of each group in each cell type, and computed the correlation between the group means. The median of all protein split correlations served as the evaluation metric. A high correlation indicates a good label assignment, while a random label assignment would result in a correlation of approximately 0.

The second metric, called the agreement score, was defined in (Welch et al., 2019) and evaluated embedding quality. It measures the similarity of cell neighbors before and after integration. We first reduced the dimension of the raw dataset using appropriate dimension reduction techniques. KNN graphs were then constructed on both integrated and independent latent space. The agreement score represents the overlap of cells' nearest neighbors in each KNN graph. To ensure a fair comparison, we used NMF for LIGER and PCA for scDMVAE as dimension reduction tools. A high agreement score indicates less distortion in the integrated embedding space compared to the independently reduced space of raw data.

scDMVAE outperforms LIGER in both metrics. Specifically, we computed split cor-



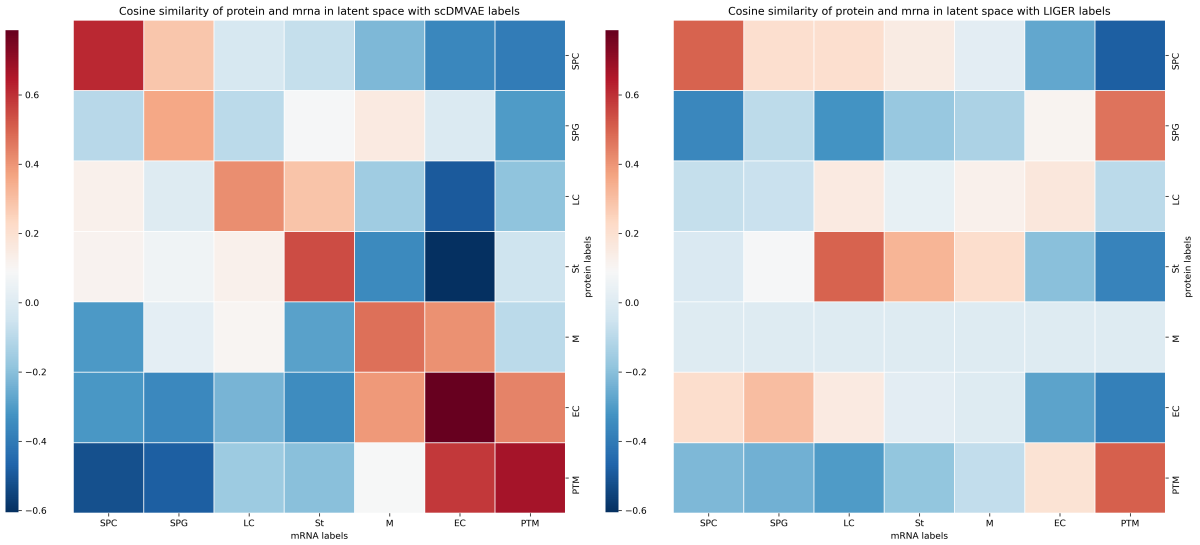
(a) scDMVAE Protein labels (b) LIGER Protein labels

Figure 2.12: **Heat maps of Spearman Correlations of cell types between Protein and mRNA raw datasets.** In scDMVAE, the cell labels are consistent in raw data with high correlations in the diagonal. Similar clusters are formed compared to the UMAP. On the other hand, LIGER-transferred protein labels are not consistent with the mRNA labels.

relation scores for the 174 common proteins and found that scDMVAE achieved a score of 0.801, while LIGER obtained a score of 0.763, indicating that scDMVAE has better label assignment. Additionally, we evaluated the agreement score of the protein embedding for both methods and found that scDMVAE had an agreement score of 0.143, compared to LIGER's score of 0.110. This indicates that scDMVAE's embedding better preserves the structures present in the raw datasets compared to LIGER's embedding.

2.4 Hyper Parameter Tuning

In scDMVAE, hyperparameters are used to adjust the model performance on different datasets in all three components. In this section, we will provide a general guide on the selection of hyperparameter values and how they affect the model's performance.



(a) scDMVAE Protein labels

(b) LIGER Protein labels

Figure 2.13: **Heat maps of cosine similarities of cell types between Protein and mRNA embeddings.** The scDMVAE model yields cell labels that exhibit high correlation with the learned embeddings, as evidenced by the strong diagonal pattern. The resulting clusters are similar to those observed in the UMAP. However, the protein labels transferred using LIGER are not consistent with the mRNA labels in the latent space.

In embedding learning and matching components, hyper parameters include the latent space dimension k_s for shared space, k_p for modality-specific space, attention hidden space dimension k_{att}, the penalties λ_m and λ_s on the modality-specific and shared parts of the objective function, respectively. The choice of k_s and k_p varies depending on the dataset, but they should be much smaller than the number of input features to serve the purpose of dimension reduction. According to (Vaswani et al., 2017), k_{att} should be less than or equal to k_s. In vertical integration, the relative values of λ's control the flow of information among the shared and modality-specific latent spaces. A larger λ_s penalizes more on the shared latent space, resulting in less information flowing through it. In an extreme case where $\lambda_s/\lambda_m \approx +\infty$, the embedding learning model is equivalent to two independent VAEs, and no shared information will be learned. On the other hand, if λ_s is relatively small compared to the λ_m's, the shared latent space will contain more information, including modality-specific information. When $\lambda_m/\lambda_s \approx +\infty$, useful information only flows through the shared latent space, and the modality-specific latent spaces are deprecated. The embedding learning model is equivalent to an MVAE model. Therefore, balancing the penalties is crucial in practice for disentangling the information.

In the component using anchors to integrate datasets, the choice of k's for anchor identification, anchor scoring and anchor weighting. k_{id} is used for anchor identification using the mutual nearest neighbors. If k_{id} is small, fewer anchor pairs will be identified, harming the alignment and finding fewer cell types the target dataset. However, if k_{id} is large, more noisy anchor pairs will be found, increasing computation time without additional contribution. Noisy anchor pairs can mislead the label transfer when their anchor scores are moderate. k_{score} is used for anchor scoring in shared nearest neighbors. Small k_{score} will result in a conservative scoring strategy where only a small proportion of anchor pairs will have high scores, leading to the domination of several cell types in transferred labels. If k_{score} is large, the scoring strategy is too aggressive and many more

noisy pairs will have high scores, misleading the label transfer. k_{weight} is used for setting a boundary when weighting anchors for a target cell. When k_{weight} is small, only a few anchor cells are used and the alignment will be unstable. However, k_{weight} should not be too large as the weighting process with large k_{weight} values will be vulnerable to outlier anchors that are far from the target cell. In this case, the distance effects of most anchors in equation 2.5 will be 1, and all anchors except for the outliers will be weighted on their score only, leading to extremely noisy transferred labels.

Chapter 3

Retention Time Alignment Using PCRID

3.1 Background and Related Work

In SCoPE2 technology, cell analysis is conducted through multiple experiments/runs using mass spectrum. Each experiment identifies a proportion of all peptides, and the identified peptides (peptide spectrum matches or PSMs) in each experiment come with confidence scores called posterior error probability (PEP/Spectrum PEP). This score represents the probability that the observed ion was assigned to the wrong peptide. However, lowly abundant peptides tend to have high PEPs due to the small number of fragment ions they generate, leading to a reduction in the quantity and quality of protein identifications.

To overcome this limitation, researchers utilize additional information about peptides to enhance low-confidence identifications. Retention time (RT) is a highly accurate measurement of the time taken for a peptide to pass through a chromatography column. It characterizes peptides in a way that RT patterns of peptides are similar across different

experimental conditions. If we plot the RTs of peptides in a high-dimensional space with experiments as axes, they will form a data cloud that resembles a high-dimensional curve. We can increase the confidence of peptide identification if the peptide follows the RT pattern, particularly when the same peptide has high confidence in other experiments. Conversely, the confidence of peptide identification should be decreased if it does not follow the RT pattern.

In their work, Chen et al. developed a Bayesian framework called DART-ID that utilizes retention time (RT) information to update peptide confidence. Specifically, DART-ID introduces a latent variable, referred to as the reference retention time μ_i, for each peptide i. They assume that the corresponding observed retention time in experiment j is $\rho_{ij} = g_j(\mu_i) + \epsilon_{ij}$, where ϵ_{ij} is an independent mean-zero random error and the function g_j is approximated by a two-segment linear regression model. The Bayesian framework is then used to update the confidence.

While DART-ID performs relatively well compared to other methods, the two-segment linear regression approximation can be inaccurate when patterns of RTs become more non-linear (Figure 3.1). To account for this non-linearity, we propose PCRID, a model that learns from mixture-model-based principal curves (Tibshirani, 1992). The PCRID framework works well with RT alignment and is compatible with the nature of peptide identification, where the underlying truth of the same PSMs across experiments can be different. PCRID also accounts for the existence of missing values, where peptides are only observed in a proportion of all experiments and treated as missing values in other experiments.

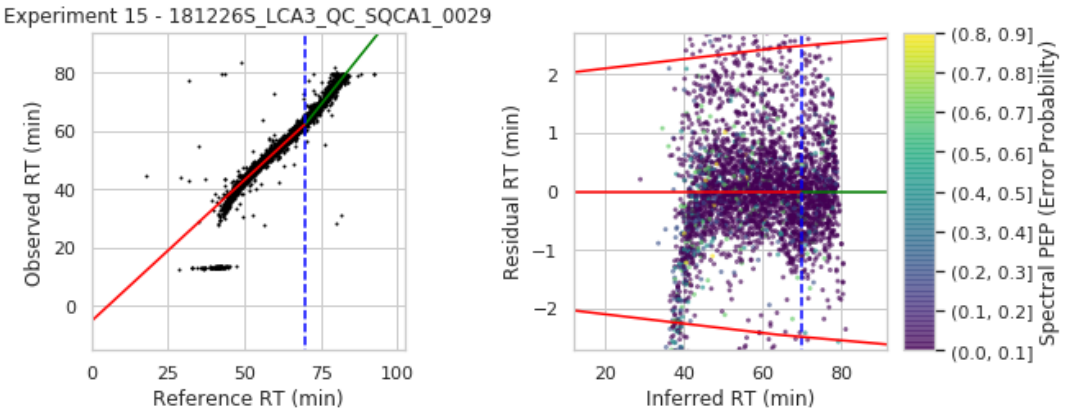


Figure 3.1: **DART-ID fitting result for non-linear data.** The left panel shows the two-segment regression line and we can see that the non-linear feature around reference time 40 is not captured; The right panel shows the residual plots and large residuals around 40 indicates inaccuracy of the fitting.

3.2 PCRID Framework

3.2.1 One-Dimensional Curves

A one-dimensional curve in p-dimensional space, as defined in (Hastie & Stuetzle, 1989), is a vector $\boldsymbol{f}(s) = (f_1(s), \ldots, f_p(s))$, where the latent variable s provides an ordering along the curve. The p functions $f_j(s)$, with $j = 1, \ldots, p$, are called coordinate functions. If the coordinate functions are smooth, then $\boldsymbol{f}$ forms a one-dimensional smooth curve in high-dimensional space. An important property of one-dimensional curves is that we can apply any monotone transformation to s and modify the coordinate functions accordingly, the curve will remain unchanged. Specifically, if $s' = g(s)$ is the new latent variable where g is a monotone function, we can modify the coordinate function to $f(g^{-1})$ so that the curve remains the same $f(g^{-1}(s')) = f(s)$. To avoid over-fitting in this case, we need to control the acceleration. The acceleration of a curve at s is defined as the vector of its second derivatives, denoted as $\boldsymbol{f}''(s)$. The acceleration controls the curvature of a one-dimensional curve, where a straight line has an acceleration of 0. The higher

the value of acceleration, the more wiggly the curve is at s.

3.2.2 Definition of Principal Curves

(Tibshirani, 1992) proposes a definition of principal curves based on mixture model. Let $\boldsymbol{Y} = (Y_1, Y_2, \ldots, Y_p)$ be a random vector with density $g_{\boldsymbol{Y}}(y)$. We assume that $\boldsymbol{Y}$ was generated in two stages: 1. A latent variable s was generated according to a continuous distribution $g_S(s)$ and 2. $\boldsymbol{Y} = (Y_1, Y_2, \ldots, Y_p)$ was generated from a conditional distribution $g_{\boldsymbol{Y}|S}(y|s)$ with mean $\boldsymbol{f}(s)$, a point on a curve in $\mathbb{R}^p$ with $Y_1, Y_2, \ldots, Y_p$ conditionally independent given s.

Definition 3.2.1 (Principal Curves)**.** The principal curves is defined to be a triplet $\{g_S, g_{Y|S}, \boldsymbol{f}\}$ satisfying the following conditions:

1. $Y_1, Y_2, \ldots, Y_p$ are conditionally independent given s
2. $\boldsymbol{f}(s)$ is a curve in $\mathbb{R}^p$ parameterized over $s \in \Gamma$, a closed interval in $\mathbb{R}$, satisfying $\boldsymbol{f}(s) = \mathbb{E}(\boldsymbol{Y}|S = s)$.

This definition decomposes the density $g_{\boldsymbol{Y}}$ into g_S and $g_{Y|S}$, which suits the context of retention time alignment. We can treat the latent variable s as a representation of unknown ontology of peptides, called reference retention time. It characterizes the different types of peptides with different values of s. The observed retention time of peptide i in experiment j is a random variable Y_{ij}, $Y_{ij} = f_j(s_i) + \epsilon_{ij}$. f_j represents the systematic effect on the measurement of retention time in experiment j, which may include effects such as temperature, humidity and so on. ϵ_{ij} represents the random effect on the measurement. A principal curve is fitted using EM algorithm and the resulting weight matrix $W = \{w_{ik}\}$, where $w_{ik} = P(s_i = a_k|\boldsymbol{y})$, can be considered as the probability that the observation assigned to peptide i is generated by peptide k given the additional information on observed retention time $\boldsymbol{y}$.

However, this principal curve definition is not applicable to practical retention time alignment for the following reasons:

- **Incorrect Identifications** In practice, there are false identifications for one kind of peptide across one or more experiments. Therefore the underlying true reference retention time for s_i can be different across experiments.
- **Missing Values** In the above principal curve definition, there is no accommodation for missing values which are commonly seen in retention time alignment.
- **Using PEP as Prior Information** The classical principal curve model doesn't use external information such as PEP to boost its performance.

3.2.3 PCRID

In PCRID, the data generating process is different from principal curve model in that it treats different observation separately. For observation i, instead of using only one latent variable, there will be p latent variables $s_{ij}, j \in \{1, \ldots, p\}$ for the p experiments. Each s_{ij} represents the underlying truth of corresponding experiment. In this way, peptides with the same index can have different underlying true reference peptides in different experiments.

Let $\boldsymbol{Y}_i = (Y_{i1}, \ldots, Y_{ip})$ be a random vector with density $g_{\boldsymbol{Y}_i}(y_i)$. We assume that each $\boldsymbol{Y}_i$ value was generated in two stages:

1. p latent variables $s_{i1}, s_{i2}, \ldots, s_{ip}$ were generated independently according to some distributions $g_{S_{ij}}(s)$, $j \in \{1, \ldots, p\}$
2. $\boldsymbol{Y}_i = (Y_{i1}, \ldots, Y_{ip})$ was generated from a conditional distribution $g_{Y_{ij}|S_{ij}}$ having mean $f_j(s_{ij})$ with $Y_{i1}, \ldots, Y_{ip}$ conditionally independent given $s_{ij}, j \in \{1, \ldots, p\}$.

Definition 3.2.2 (PCRID)**.** PCRID is defined to be a triplet $\{g_{S_{ij}}, g_{Y_{ij}|S_{ij}}, f_j\}$ satisfying the following conditions:

1. $Y_{i1}, Y_{i2}, \ldots, Y_{ip}$ conditionally independent given $s_{ij}, j \in \{1, \ldots, p\}$.

2. $\boldsymbol{f}(s)$ is a curve in $\mathbb{R}^p$ parametrized over $\boldsymbol{s} \in \Gamma$, a closed hyper-rectangle in $\mathbb{R}^p$, satisfying $f_j(s_{ij}) = \mathbb{E}(Y_{ij}|S_{ij} = s_{ij})$.

We can incorporate the PEP information in $g_{S_{ij}}(s)$ with $P(s_{ij} = a_i) = 1 - PEP$ and $P(s_{ij} = a_k) = PEP/(n-1)$ for $k \neq i$

3.2.4 Algorithm

Assume we have n observations and p experiments Let $a_k, k \in \{1, \ldots, n\}$ be the reference retention times of all possible underlying peptides. Denote the parameters $\theta = \theta(s) = (\boldsymbol{f}(s), \Sigma(s))$. We assume $\Sigma(s)$ is a diagonal matrix with entries $\sigma_j(s) = \sigma_j, j = 1, 2, \ldots, p$. The complete data log-likelihood is

$$l(\theta) = \sum_i \sum_j \log(g_{y_{ij}|S_{ij}}(y_{ij}|\theta(a_k)) + \sum_i \sum_j \log(g_{s_{ij}}(s_{ij}))$$

.

To avoid over-fitting, as in the classical principal curves, we add a penalty term $-(c_1 - c_2)\sum_j \int_{c_1}^{c_2} [f_j''(s)]^2 ds$ to the complete data log-likelihood and the objective function becomes:

$$j(\theta) = l(\theta) - (c_1 - c_2)\sum_j \int_{c_1}^{c_2} [f_j''(s)]^2 ds \tag{3.1}$$

We can maximize Equation 3.1 via the EM algorithm. The E step starts with initial

value $f_j^0, j = 1, 2, \ldots, p$ and computes the Q function

$$Q(\theta|\theta^0) = \mathbb{E}\{j(\theta)|y, \theta^0\}$$

where y denotes the observation matrix. The M step maximize $Q(\theta|\theta^0)$ over θ.

Let $w_{ijk} = P(s_{ij} = a_k|y_{ij})$, and $v_{ijk} = P(s_{ij} = a_k)$, we may write Q as

$$\begin{aligned} Q(\theta|\theta^0) = &\sum_i \sum_j \sum_k w_{ijk} \log(g_{y_{ij}|s_{ij}}(y_{ij}|\theta(a_k)))+ \\ &\sum_i \sum_j \sum_k w_{ijk} \log v_{ijk} - (c_1 - c_2) \sum_j \int_{c_1}^{c_2} [f_j''(s)]^2 ds \end{aligned} \tag{3.2}$$

Using Bayes' Theorem

$$w_{ijk} = \frac{P(y_{ij}|s_{ij} = a_k)P(s_{ij} = a_k)}{\sum_k P(y_{ij}|s_{ij} = a_k)P(s_{ij} = a_k)}$$

. In practice we can incorporate PEP information in the distribution of s_{ij}, $P(s_{ij} = a_i) = 1 - PEP_{ij}$ and $P(s_{ij} = a_k) = PEP_{ij}/(n-1)$ for $k \neq i$

If we assume the conditional distributions are Gaussian, the solution for each iteration will have closed forms. Let $b_{jk} = \sum_i w_{ijk}$ and D_j be a diagonal matrix with entries $b_{j1}, b_{j2}, \ldots, b_{jn}$ and $\bar{\boldsymbol{y}}_{\boldsymbol{j}}$ be an n-vector with kth component $\sum_i w_{ijk} y_{ij} / \sum_i w_{ijk}$. Then

$$\hat{f}_j = (D_j + (c_2 - c_1)\lambda_j K_j)^{-1} D_j \{D_j^{-1} \bar{\boldsymbol{y}}_{\boldsymbol{j}}\} \tag{3.3}$$

$$\hat{\sigma}_j^2 = \sum_i \sum_k w_{ijk} y_{ij} / \sum_i \sum_k w_{ijk} \tag{3.4}$$

The matrix K_j is the quadratic penalty matrix associated with a cubic smoothing spline. Equation 3.3 means that $\hat{f}_j$ is obtained by applying a weighted cubical smoothing

spline to the quantity $D_j^{-1}\bar{\boldsymbol{y}}_{\boldsymbol{j}}$ with weights $b_{j1}, b_{j2}, \ldots, b_{jn}$. The variances are assumed to be related only to experiments.

While the aforementioned model shows promise for retention time alignment, there are still obstacles to its practical implementation. Dealing with missing values is one such challenge, which we address by initializing the curve using a variant of PCA that accommodates missing values (Josse & Husson, 2012). Another challenge is computational efficiency. The matrix $W = \{w_{ijk}\}$ can get extremely large as number of unique peptides and experiments increase. To improve its calculation efficiency without losing much accuracy, we only compute the entries associated with observed peptides in each experiment, setting other entries to zero. To find the maximizing values $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_n$, we need to use optimization method such as the Newton-Raphson procedure. However, due to the dependence between the a_k's, σ_j^2's and f_j's, iterating through Equation 3.3, 3.4 and the Newton-Raphson procedure to fully maximize the Q function is computationally unattractive. To address this, we refer to (Tibshirani, 1992) and use a generalized EM algorithm, which seeks to increase Q function at each iteration by applying Equation 3.3 and 3.4 together followed by one Newton-Raphson step for the a_k's.

The algorithm is summarized in Algorithm 1. The outputs of the algorithm are the weight array $W = \{w_{ijk}\}$ together with fitting information. For a observation assigned to peptide i in experiment j, the weight matrix gives the probability that the observation is actually generated from peptide k conditioning on the retention time of the observation.

Algorithm 1: PCRID algorithm

Input : Data
Result: weight array $W = \{w_{ijk}\}$
init : Initialize the curve with the first principal component and $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_n$ are the projected values onto the it
while $\Delta Q > tol$ **do**

(a) Compute $\hat{w}_{ijk}$ that associated with observed values

(b) Fix $\hat{f}_j$ and σ_j^2 apply a Newton-Raphson step to obtain a new set of support points $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_n$ for iterations larger than one.

(c) Compute $\hat{f}_j$ and σ_j^2 according to Equation 3.3 and 3.4

end

3.3 Results

3.3.1 SCoPE2 Data

We run PCRID on a SCoPE2 dataset that contains 301594 peptide observations from 44 experiments. The number of unique peptides is 482855 and missing value proportion is 85.8%. The dataset contain decoys for calibration of false positive identifications (Elias & Gygi, 2010). Decoys are manufactured sequences do not exist in nature. They have noises RTs and thus tend not to be identified as useful proteins after alignment. In practice, a large number of decoys will be wrongly perceived as useful proteins in both experiment and alignment. Thus, we only compare the relative change of number of falsely identified decoys. Higher identification of decoys indicating that the alignment algorithm is aggressive and tends to generate more false positive identifications. We filtered the dataset so that the peptides observed in less than 3 experiments are excluded. These peptides are legal as model input but won't provide enough additional information to the alignment. On the contrary, they will slow down computation. There are many outliers in retention time of peptides. Therefore we applied robust L_1 smoothing spline

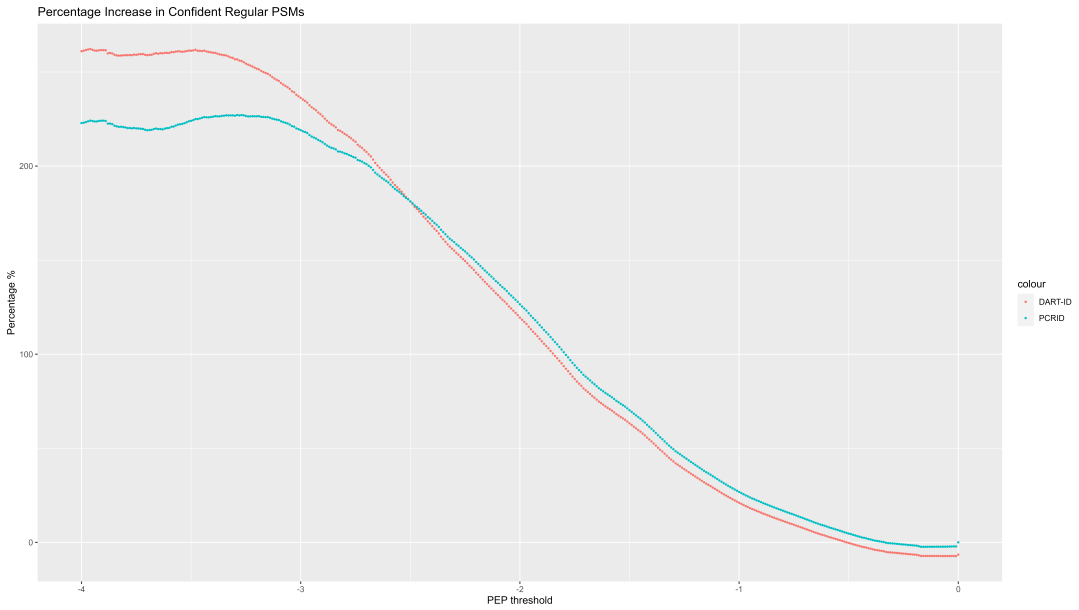


Figure 3.2: **Percentage Increase of Confident PSMs.** Both PCRID and DART-TID can significantly increase the number of PSMs. PCRID demonstrates superior performance compared to DARID when the PEP threshold is set at values greater than 0.0043. Typically, PEP thresholds in single cell proteomics experiments are set at values greater than 0.01.

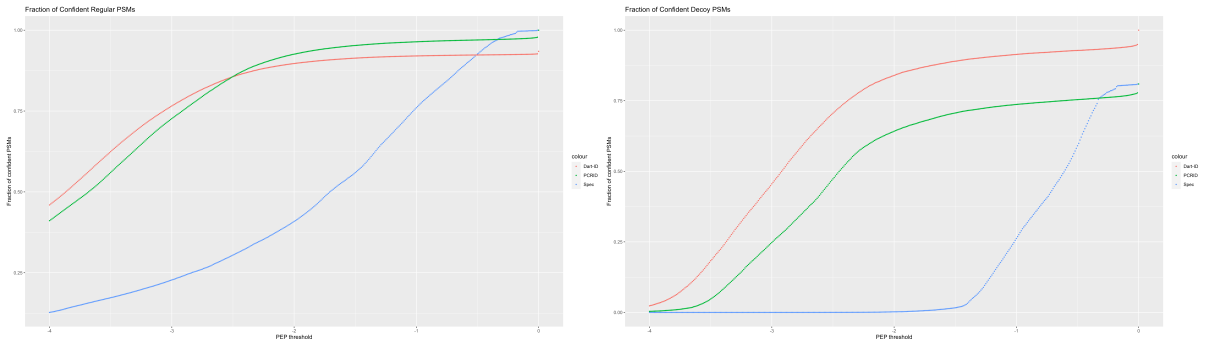
instead of cubic smoothing spline.

	Peptide Hits	Decoy Hits	Percent Change
PCRID	0.926	0.694	154.53%
DART-ID	0.897	0.839	146.85%

Table 3.1: **Alignment Summary at PEP threshold 0.01.**

PCRID drastically increase the number of identified peptides (PSMs)

In Figure 3.2, we plot the percentage increase of confident PSMs against different PEP threshold (usually ≥ 0.01). It shows that both PCRID and DART-ID can remarkably increase the number of PSMs. PCRID performs better than DART-ID for PEP threshold larger than 0.0043. Table 3.1 shows the percentage increase at PEP threshold 0.01, with PCRID 154.53% and DART-ID 146.85% respectively.

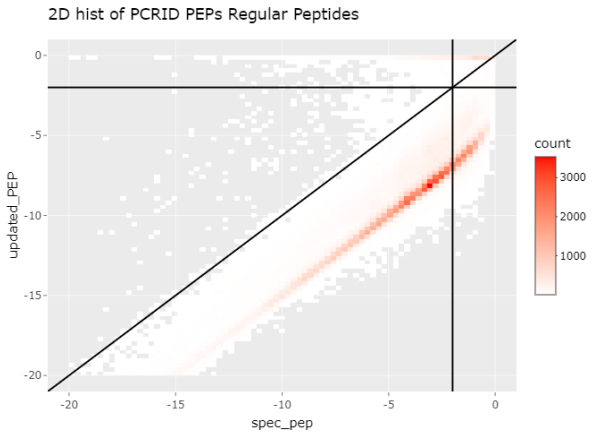


(a) Regular Peptides (b) Decoys

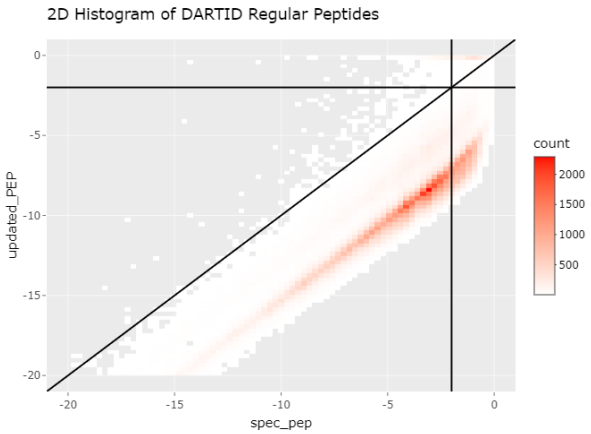
Figure 3.3: **Fraction of Confident PSMs of Regular Peptides and Decoys.** In regular peptides, PCRID has higher fraction of increased PSMs than DARTID for PEP greater than 0.01. On the other hand, in faked decoys, the fraction of increased PSMs in PCRID is consistently much lower than that in DARTID. This indicates that PCRID can outperform DARID in regular peptides and controls the false positive identifications at the same time.

PCRID identifies more peptides with lower decoy hits compared to DARTID

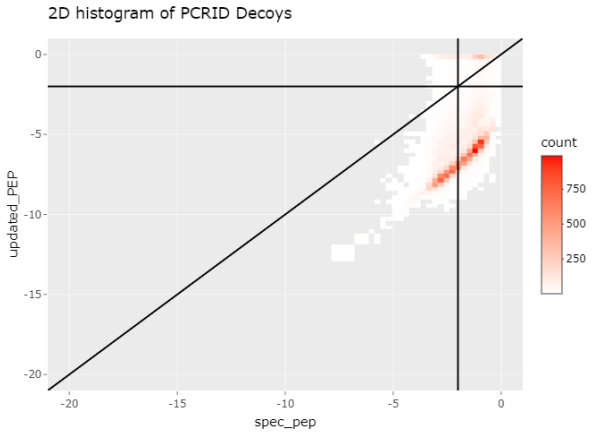
We need to further investigate the alignment of regular peptides and decoys respectively to check if the methods overly-upgrade PEPs. We analyzed the fractions of PSMs for both regular peptides and decoys before and after alignment with different PEP thresholds. When calculate the DART-ID fractions, we must take cautious and adjust the denominator of total number of peptides. This is because DAR-ID tends to filter out the observations that are hard to align using the two-segment straight lines but easy for PCRID. Figure 3.3 and 3.4 show that PCRID identifies more regular peptides and less decoys compared to DART-ID in commonly used PEP threshold range. Table 3.1 shows the scores for PCRID and DART-ID at PEP threshold 0.01. The peptide hits (0.926) and decoy hits (0.694) of PCRID are significantly different while those of DART-ID are not. This result supports that PCRID can distinguish true peptides from decoys much better than DART-ID. The increment of decoy hits is inevitable because there are still many decoys with small PEPs and consistent retention time across experiments.



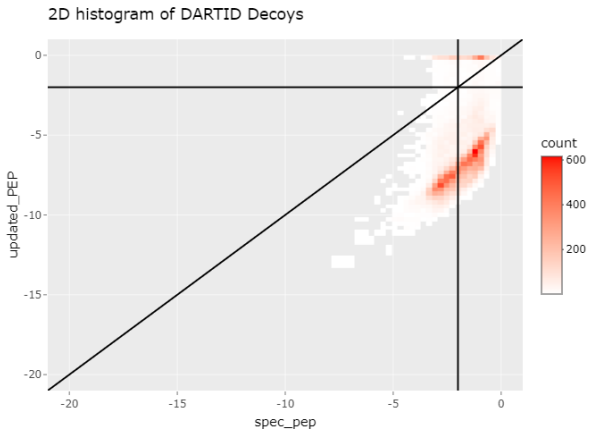
(a) Regular Peptides PCRID



(b) Regular Peptides DART-ID



(c) Decoys PCRID



(d) Decoys DART-ID

Figure 3.4: **2D Histogram of updated PEPs Compared to Spectrum PEPs.** The two rectangles at top and right represent downgrade and upgrade respectively; PCRID has more regular peptides upgraded and fewer decoy upgraded compared to DARTID.

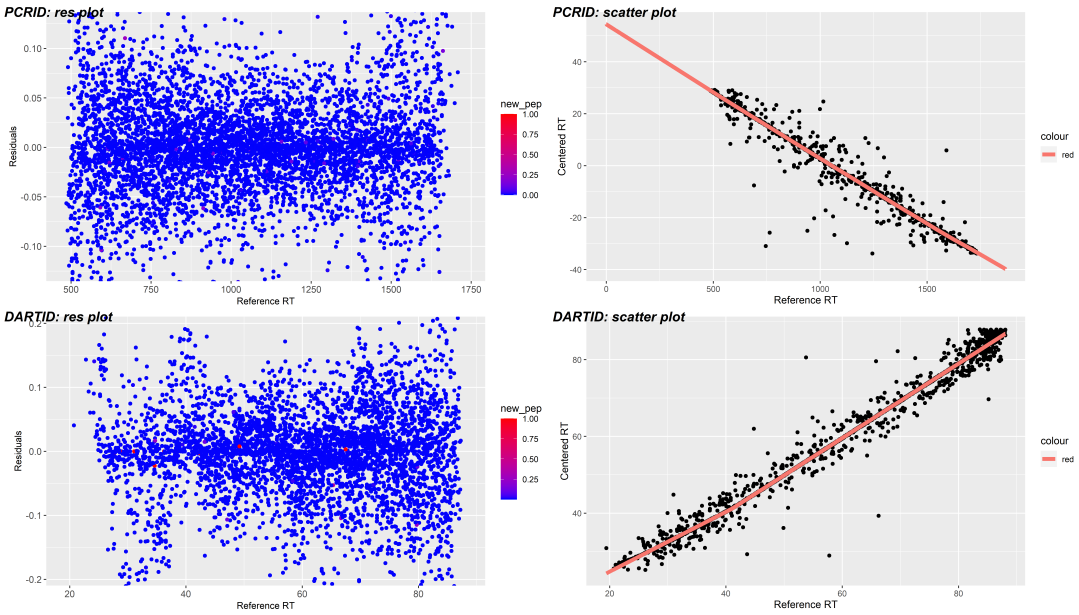


Figure 3.5: **Residual Plots of PCRID and DART-ID.** Top: PCRID results and the residuals are symmetric with consistent variance and linear pattern; Bottom: DART-ID results and the residuals are not symmetric with non-linear pattern indicating bad fit of retention time.

PCRID fit nonlinear data better than DART-ID

Figure 3.5 shows the residual plots of PCRID and DART-ID of a typical experiment. It is clear that nonlinear pattern exists in DART-ID residuals, which will weaken its ability of alignment. Figure 3.6 plots empirical CDF of residuals from PCRID and DART-ID, indicating that the residuals of PCRID are much closer to zero than those of DART-ID. As a conclusion, PCRID have a better goodness of fit than DART-ID especially in the presence of non-linearity.

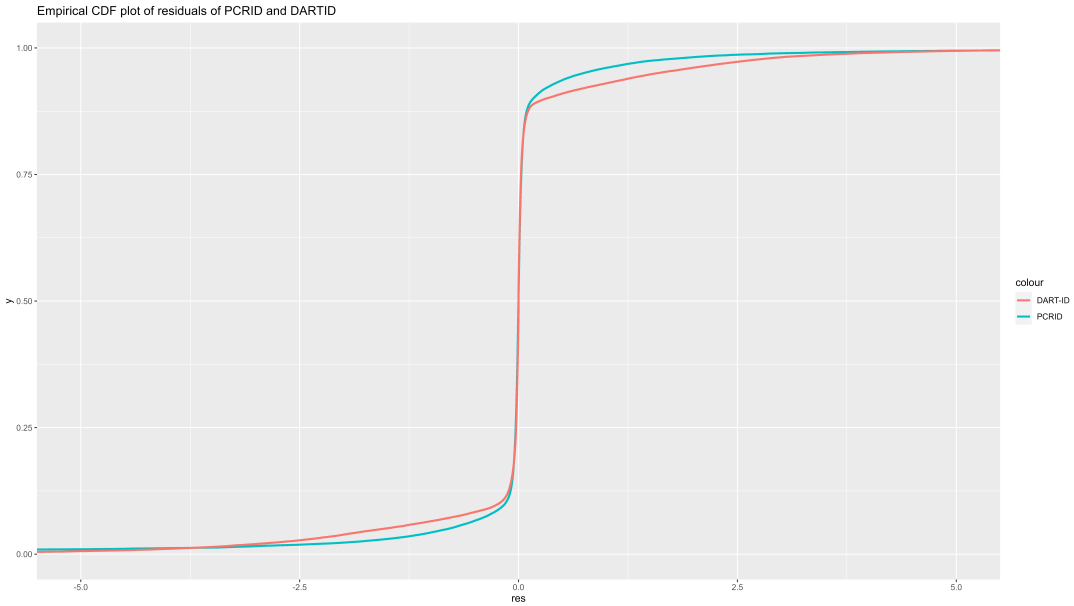


Figure 3.6: **Empirical CDF of residuals from PCRID and DART-ID fits.** The empirical CDF of residuals shows that more residuals of PCRID are scattered around zero compared to that of DARTID, indicating that PCRID fits the data better than DARTID

Chapter 4

Discussion

In this dissertation, we have developed two models to improve the analysis of single cell data. PCRID enhances the data quality of SCoPE2 technologies by significant increase the number of identified peptide using retention time. It remarkably increases the performance of RT alignment in non-linear situations and has the potential to be applied to other experiments with different characterization information other than retention time. On the other hand, scDMVAE is designed to integrate single cell dataset from different modalities. In vertical integration, it can detect heterogeneity among datasets and preserve both shared and modal specific information. In horizontal integration, scDMVAE provides a end-to-end neural network solution from embedding learning to embedding matching. It can easily incorporate differently distributed datasets and align them using similarity scores that fits the biological context. The embeddings learned and labels transferred are consistent with original structure of the data. scDMVAE can be further improved by adding projection module that harmonizes feature-anchored cells to the same space, expanding the possibilities for downstream analysis. Another potential improvement is to incorporate dataset specific space of H matrices in horizontal integration to discover sub cell types. In the current version, the dataset specific spaces in

H matrices have very high degree of freedom and rotates differently in different modality; therefore we did not include them in horizontal integration. However, with proper regularization, they can provide more information for horizontal integration. Combined together, scDMVAE and PCRID can significantly improve analysis quality of SCoPE2 data as well as data in related fields. We hope these two models can broaden the horizon in related field and help to solve biological puzzles.

References

Argelaguet, R., Cuomo, A. S., Stegle, O., & Marioni, J. C. (2021). Computational principles and challenges in single-cell data integration. *Nature biotechnology*, *39*(10), 1202–1215.

Chen, A. T., Franks, A., & Slavov, N. (2019). Dart-id increases single-cell proteome coverage. *PLoS computational biology*, *15*(7), e1007082.

Elias, J. E., & Gygi, S. P. (2010). Target-decoy search strategy for mass spectrometry-based proteomics. In *Proteome bioinformatics* (pp. 55–71). Springer.

Haghverdi, L., Lun, A. T., Morgan, M. D., & Marioni, J. C. (2018). Batch effects in single-cell rna-sequencing data are corrected by matching mutual nearest neighbors. *Nature biotechnology*, *36*(5), 421–427.

Hao, Y., Hao, S., Andersen-Nissen, E., III, W. M. M., Zheng, S., Butler, A., ... Satija, R. (2021). Integrated analysis of multimodal single-cell data. *Cell*. Retrieved from `https://doi.org/10.1016/j.cell.2021.04.048` doi: 10.1016/j.cell.2021.04.048

Hardoon, D. R., Szedmak, S., & Shawe-Taylor, J. (2004). Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, *16*(12), 2639–2664.

Hastie, T., & Stuetzle, W. (1989). Principle curves. *Journal of the American Statistical Association*, *84*.

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural computation*, *14*(8), 1771–1800.

Houle, M. E., Kriegel, H.-P., Kröger, P., Schubert, E., & Zimek, A. (2010). Can shared-neighbor distances defeat the curse of dimensionality? In *International conference on scientific and statistical database management* (pp. 482–500).

Jarvis, R. A., & Patrick, E. A. (1973). Clustering using a similarity measure based on shared near neighbors. *IEEE Transactions on computers*, *100*(11), 1025–1034.

Josse, J., & Husson, F. (2012). Handling missing values in exploratory multivariate data analysis methods. *Journal de la Société Française de Statistique*, *153*(2), 79–99.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Lee, M., & Pavlovic, V. (2021). Private-shared disentangled multimodal vae for learning of latent representations. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 1692–1700).

Levy, E., & Slavov, N. (2018). Single cell protein analysis for systems biology. *Essays in biochemistry*, *62*(4), 595–605.

Lopez, R., Regier, J., Cole, M. B., Jordan, M. I., & Yosef, N. (2018). Deep generative modeling for single-cell transcriptomics. *Nature methods*, *15*(12), 1053–1058.

Petelski, A. A., Emmott, E., Leduc, A., Huffman, R. G., Specht, H., Perlman, D. H., & Slavov, N. (2021). Multiplexed single-cell proteomics using scope2. *Nature protocols*, *16*(12), 5398–5425.

Stoeckius, M., Hafemeister, C., Stephenson, W., Houck-Loomis, B., Chattopadhyay, P. K., Swerdlow, H., ... Smibert, P. (2017). Simultaneous epitope and transcriptome measurement in single cells. *Nature methods*, *14*(9), 865–868.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., III, W. M. M., ... Satija, R. (2019). Comprehensive integration of single-cell data. *Cell*, *177*, 1888-1902. Retrieved from `https://doi.org/10.1016/j.cell.2019.05.031` doi: 10.1016/j.cell.2019.05.031

Tang, F., Barbacioru, C., Wang, Y., Nordman, E., Lee, C., Xu, N., ... others (2009). mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, *6*(5), 377–382.

Tibshirani, R. (1992). Principal curves revisited. *Statistics and computing*, *2*(4), 183–190.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

Welch, J. D., Kozareva, V., Ferreira, A., Vanderburg, C., Martin, C., & Macosko, E. Z. (2019). Single-Cell Multi-omic Integration Compares and Contrasts Features of Brain Cell Identity. *Cell*, *177*, 1873-1887.

Wu, M., & Goodman, N. (2018). Multimodal generative models for scalable weakly-supervised learning. *Advances in Neural Information Processing Systems*, *31*.

Yang, Z., & Michailidis, G. (2016). A non-negative matrix factorization method for detecting modules in heterogeneous omics multi-modal data. *Bioinformatics*, *32*(1), 1–8.

Zuo, C., & Chen, L. (2021). Deep-joint-learning analysis model of single cell transcriptome and open chromatin accessibility data. *Briefings in Bioinformatics*, *22*(4), bbaa287.