Supervised Deep Learning with Gene Annotation for Cell Classification

Zhexiao Lin
Department of Statistics, University of California, Berkeley, CA

Wei Sun
Biostatistics Program, Fred Hutch Cancer Center, Seattle, WA
Department of Biostatistics, University of Washington, Seattle, WA
Department of Biostatistics, University of North Carolina, Chapel Hill, NC

July 15, 2024

Abstract

Gene-by-gene differential expression analysis is a popular supervised learning method for analyzing single-cell RNA sequencing (scRNA-seq) data. However, the large number of cells in scRNA-seq studies often results in numerous differentially expressed genes with extremely small p-values but minimal effect sizes, complicating interpretation. To address this challenge, we developed a method called Supervised Deep Learning with gene ANnotation (SDAN). SDAN integrates gene annotation and gene expression data using a graph neural network to identify gene sets to classify cells, and then the corresponding individuals. We demonstrated the usage of SDAN by identifying gene sets associated with severe COVID-19, dementia, and cancer patients' responses to immunotherapy.

Introduction

For many biomedical applications, an important step in exploring single-cell RNA-seq (scRNAseq) data is gene-by-gene differential expression (DE) analysis, which is typically followed by an additional step to identify biological processes over-represented by DE genes. Due to the large sample size (i.e., a large number of cells) in scRNA-seq data, many DE genes may have small p-values but minimal effect sizes, and thus additional *ad hoc* filtering is needed. To address this challenge, we aim to identify gene sets that can accurately classify cells. The results in terms of gene sets are easier to interpret than a larger number of DE genes. Additionally, classification accuracy is more intuitive and meaningful for applications than p-values, where a very small p-value may correspond to a negligible effect size. To achieve this goal, we have developed a graph neural network (GNN) method named Supervised Deep learning with gene ANnotation (SDAN). SDAN encodes gene annotations using a gene-gene interaction graph and incorporates gene expression as node attributes. It then learns gene sets such that the genes in a set share similar expression and are located close to each other in the graph. These gene sets are used for supervised learning (i.e., classification of cells), with gene set learning and supervised learning steps combined within a deep learning framework. Unlike typical neural network models that function as a black box, SDAN's results are simply gene sets and are therefore readily explainable.

A key benefit of deep learning is representation learning, i.e., learning new features that can be complex (often black-box) functions of original features. In scRNA-seq data, the original features are genes, which are meaningful and easy to interpret. Therefore, there is limited benefit in learning new features. This may explain why most deep learning methods for scRNA-seq data focus on unsupervised tasks such as dimension reduction and denoising [1, 2, 3], clustering [4, 5], batch correction, and harmonization [6, 7]. Our method can be considered as a combination of un-supervised clustering of genes and supervised classification of cells. We use a deep learning framework to combine these two tasks so that the clusters of genes are informative for the cell classification task.

Results

Overview of SDAN

SDAN can be considered as a two-step method (Figure 1). In the first step, we combine scRNA-seq data and gene annotations to identify latent components in scRNA-seq data using a GNN. Each component is a linear combination of all the genes. The latent components are estimated through a graph pooling operation of the GNN, where nodes (genes) with similar expression and similar annotations (i.e., connected in the graph) are pooled into one latent component. To facilitate interpretation, a penalty term is included in the objective function to encourage sparse assignments, making the assignments of genes to latent components almost binary. Therefore, these latent components can be considered as gene sets. In the second step, we project scRNA-seq data to these gene sets and then use the gene-set-level data to classify cells by a multilayer perceptron (MLP). These two neural networks (GNN for latent component identification and MLP for outcome prediction) are trained together.

Gene expression of CD8+/CD4+ T cells can classify severe versus mild COVID-19 patients

Several studies have shown that T cell response induced by vaccination or natural infection is crucial in protecting against severe COVID-19 [8, 9, 10]. An ineffective or delayed T cell response may lead to the over-activation of the innate immune system, causing tissue damage. In contrast, a timely and effective T cell response can clear the virus and thus prevent severe COVID-19. We conjecture that the T cells that react to SARS-Cov-2 have gene expression patterns that can distinguish them from other T cells. We used scRNA-seq data of $\sim$ 42,000 CD4+ T cells and $\sim$ 24,000 CD8+ T cells from 49 mild and 32 severe COVID-19 patients [11], and split the data so that half of the patients were used for training and the other half for testing. Furthermore, 10% of cells in the training data were used as validation data to decide when to stop deep learning training. See Supplementary Materials for more details.

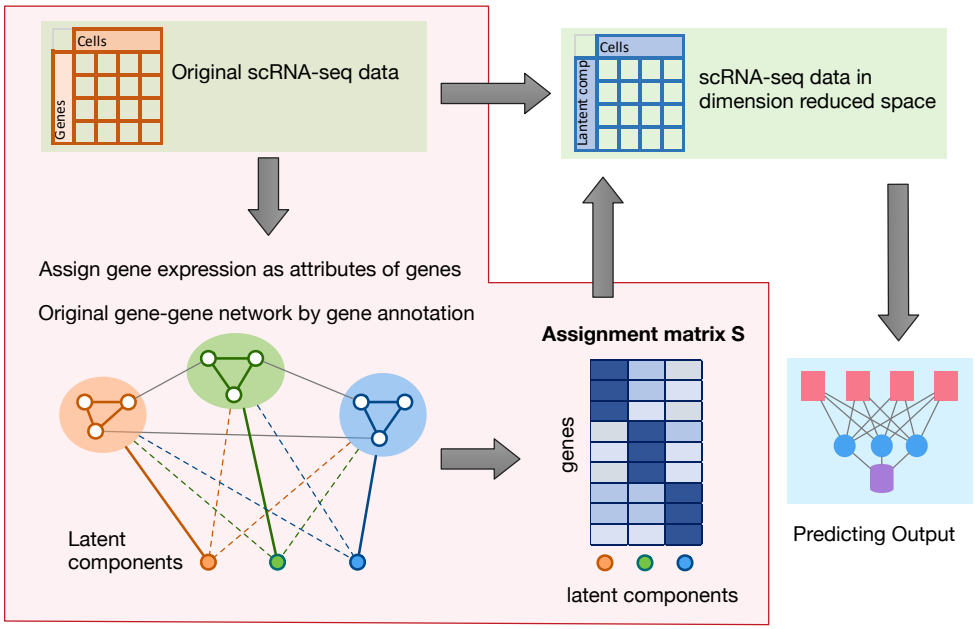

Figure 1: An overview of Supervised Deep Learning with Gene Annotation (SDAN) can be divided into two steps. In the first step (shaded area), SDAN combines gene expression data and gene annotations (gene-gene interaction graph) to learn a gene assignment matrix **S**, which specifies the weights of each gene for all latent components. In the second step, SDAN uses the gene assignment matrix **S** to reduce the gene expression data of each cell to a low-dimensional space (from the number of genes to the number of latent components) and then makes predictions in the low-dimensional space using a feed-forward neural network.

The loss function of SDAN includes two unsupervised components and a supervised component. The unsupervised loss function is the loss for the graph neural network, which encourages genes connected in the graph to be included in the same gene set. The supervised loss function is the cross-entropy loss for classification accuracy. See the Online Methods Section for details. We adjusted the contributions of the two losses by adding a weight for the unsupervised loss and explored a series of values for this weight from 0 to 10. Note that when this weight is 0, our method becomes a baseline approach of unsupervised clustering using gene annotations.

Next, we illustrated the consequence of different weights using the example of predicting severe COVID-19 by CD8+ T cell gene expression. The conclusion was similar when we explored other applications. SDAN estimates 40 latent components (gene sets) by default. When the weight for the unsupervised loss is small, many components are empty, i.e., without contributing genes. As the weight for the unsupervised loss increases, the number of non-empty latent components increases and stabilizes at 40 (Figure 2(A)). We assessed whether the genes within a latent component were more connected than expected by chance using a quantile measurement. Denote the connectivity of an observed gene set G by C_G, which is the average degree in the gene set, i.e., $2 \times$ number of edges/number of genes. We randomly selected many gene sets of the same size and calculated the distribution of their connectivities, denoted by $f(G)$. Then we calculated what quantile of $f(G)$ is C_G. As expected, as the weight for the unsupervised loss increases, the quantile become larger (Figure 2(B)). Finally, we aggregated the cell-level prediction scores to individual-level prediction scores by taking the average, and used the individual level prediction scores to predict COVID-19 severeness. The accuracy of the prediction decreases slightly as the weights for the unsupervised loss increase (Figure 2(C)). The CD4+ T and CD8+ T cell-level AUCs vary from 0.89 to 0.85 and from 0.91 to 0.83 respectively, though individual level AUC are more stable and are between 0.94 and 0.97 for both cell types (see Supplementary Materials Section 1.2 for more details).

While balancing the prediction accuracy and the utilization of gene annotation, we chose to use a weight 2 for the unsupervised loss. Clustering analysis of CD8+ T cells cannot reveal disease status (Figure 2 (D-E)). In contrast, the SDAN prediction scores are very

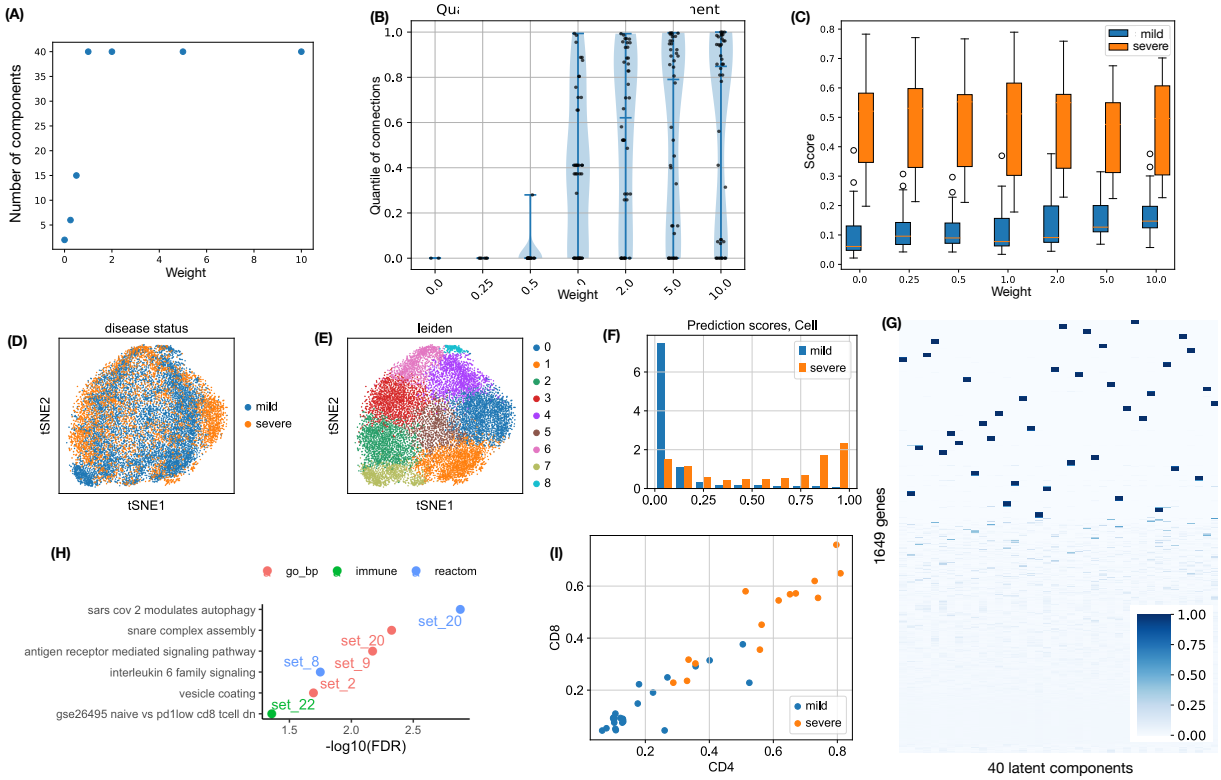

Figure 2: Summary of results for CD8+ T cells. **(A)** The number of latent components for different weights for the unsupervised loss. **(B)** Violin plots of connection quantiles. Each point represents one component. The connection quantile is computed based on the number of connections from randomly selected gene sets. **(C)** Boxplots of individual-level scores (average of cell level scores) for severe disease, across different weights for the unsupervised loss. **(D-E)** tSNE projection of CD8+ T cells in testing data, colored by disease status or clusters generated by the Leiden algorithm. **(F)** Distribution of cell-level prediction scores (for CD8+ T cells using weight 2.0) for severe disease in testing data, stratified by the status of severe/mild diseases of the corresponding individual. **(G)** Assignment matrix of 1,649 genes to 40 latent components. Genes are ordered by hierarchical clustering. **(H)** Enriched functional categories of identified gene sets. **(I)** Individual level scores COVID-19 patients calculated based on gene expression of CD4+ or CD8+ T cells.

different between the cells from severe patients and the cells from mild patients (Figure 2 (F)). These scores were calculated by a neural network using 40 latent components of each cell as the input. SDAN automatically selected genes to be included in each component (Figure 2 (G)). Further analysis shows several components are enriched with functional categories from gene ontology (biological processes), reactome pathways, or immunologic gene sets (Figure 2 (H)). For example, gene set 20 is enriched with genes from the reactome pathway of SARS-Cov-2 modulates autophagy, and GO term SNARE complex assembly. Autophagy is activated during SARS-Cov-2 infection to destroy the infected cells. However, some proteins encoded by SARS-Cov-2 can prevent this process by blocking the assembly of the SNARE complex, which is required for the autophagy process [12]. The enriched functional categories are similar for larger weights for the unsupervised loss (Supplementary Figure 2).

The results from CD4+ T cells also support the use of weight 2 for the unsupervised loss (Supplementary Figure 3). Individual-level prediction scores made by CD8+ T cells and CD4+ T cells are highly consistent (Figure 2 (I)).

We have used the average of cell-level prediction scores to calculate individual-level prediction scores. While this is a reasonable approach, it is important to explore the variation of cell level prediction scores. When using either CD8+ T cells (Figure 2(F)) or CD4+ T cells (Supplementary Figure 3(D)) to predict severe COVID-19, the T cells from mild patients tend to have smaller prediction scores concentrating around 0, and the prediction scores from severe patients have a peak around 1, but also spread in the range from 0 to 1. This implies that a subset of T cells from severe patients might be responsible for disease progression.

Gene expression of astrocytes or microglia can classify dementia status

Alzheimer's disease, which is associated with the accumulation of amyloid beta plaques and hyperphosphorylated tau, is the most common cause of dementia in older individuals [13]. Many recent studies have pointed out that the human immune response plays an

important role in the initiation, progression, and pathology of Alzheimer's. Two types of non-neuronal cells, microglia and astrocyte, are involved in this process [14, 15]. We used SDAN to analyze the snRNA-seq data from the Seattle Alzheimer's Disease Cell Atlas (SEA-AD) consortium [16], focusing on $\sim$ 70,000 astrocyte (Astro) cells and $\sim$ 40,000 microglia/perivascular macrophage (micro-PVM) cells from 84 donors (42 with dementia and 42 without dementia). We sought to classify dementia status using astrocyte or micro-PVM expression. Similar to the COVID-19 study, we randomly selected half of the individuals as training data and used the remaining ones as testing data. We also evaluated the results across different weights for the unsupervised loss and concluded that a weight of 2.0 for the unsupervised loss is an appropriate choice (Supplementary Figures 3-4). We further explored in detail the results with a weight of 2.0.

In contrast to predicting severe COVID-19 using T cells, predicting dementia using astrocyte or microglia-PVM is more challenging. This can be illustrated by comparing cell level and gene level prediction scores in Figure 2(C) and (F) (CD8+ T cells to predict COVID-19) and Figure 3(A-D). Many astrocyte and microglia-PVM cells have similar prediction scores between dementia and non-dementia donors (Figure 2(A-B)), though a subset of them have higher scores in dementia donors. The individual-level prediction scores cannot separate dementia and non-dementia donors very well (with AUC 0.744 and 0.735 for astrocyte and microglia-PVM, respectively; see Supplementary Materials Section 2 for more details). However, when jointly considering the prediction scores by astrocyte and microglia-PVM, a subset of donors stands out. They have high scores using either astrocyte or microglia-PVM (Figure 3(E)). We refer to this subset of dementia donors as dementia-i where i stands for immune system, and refer to the remaining dementia donors as dementia-s, where s stands for immune silence.

We further characterize the dementia-i donors by the neuropathology measurements. Hyperphosphorylated tau aggregates into insoluble twisted fibers known as neurofibrillary tangles. Braak staging is a popular system used to classify the extent of neurofibrillary tangle pathology in Alzheimer's disease. The staging system is divided into six stages with stage IV being the most severe. All four donors at Braak stage II-IV are dementia-s; most (7 out of 8) donors at Braak stage VI are dementia-i, while donors at Braak stage V are a

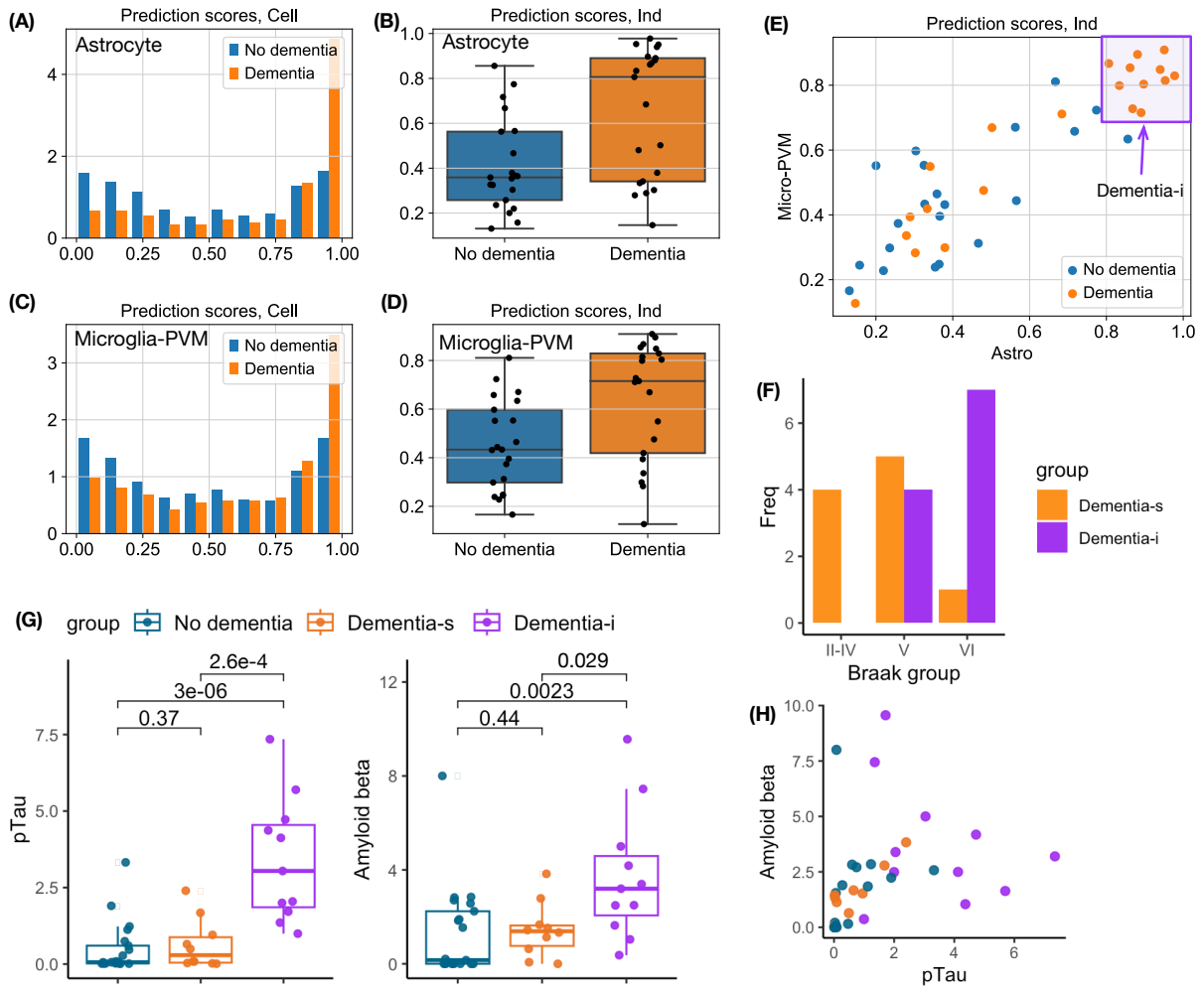

Figure 3: **(A-B)** The distributions of cell level and individual level prediction scores for dementia using astrocyte in testing data. **(C-D)** The distributions of cell level and individual level prediction scores for dementia using microglia-PVM in testing data. **(E)** Scatter plot of individual level scores by astrocyte or microglia-PVM, in testing data. We identified a Dementia subset. **(F)** The Braak staging groups for donors who belong to the Dementia subset or the remaining Dementia donors. **(G)** Percentage of pTau (Amyloid beta) positive area for three groups of donors: No dementia, Dementia-s, and Dementia-i. **(H)** Scatter plot of the percentage of pTau positive area vs. the percentage of Amyloid beta positive area.

mixture of dementia-s and dementia-i (Figure 3(F)). These results suggest that dementia-i donors have more advanced diseases.

This set of dementia-i donors also have much higher pTau and amyloid beta measurements than dementia-s or non-dementia donors (Figure 3(G)), while pTau and amyloid beta were measured as the percentage of areas that are positive for pTau (phosphorylated tau, measured by antibody AT8) or Amyloid beta (by antibody 6e10) by Gabitto et al. [16]. Combining pTau and amyloid beta measurement may give a better characterization of the dementia-i donors (Figure 3(H)), though a larger sample size is needed to confirm this conclusion.

Gene expression of CD8+ T cells can predict cancer patients' response to immunotherapy

Cancer immunotherapy has revolutionized cancer treatment, yet predicting which patients will respond to it remains a significant challenge [17]. The goal of immunotherapy is to induce or strengthen the T cell response to tumors. Previous studies have demonstrated that CD8+ T cell gene expression is informative in predicting patient responses to immunotherapy [18, 19]. To address this critical and challenging problem, we employed SDAN to identify gene sets associated with immunotherapy response. Due to the limited sample sizes in most existing studies, we used scRNA-seq data from one study, consisting of 6,350 CD8+ T cells from 17 responding tumors and 31 non-responding tumors [18], as our training data. For testing, we used scRNA-seq data from Yost et al. [19], which included 27,924 CD8+ T cells from 8 responders and 7 non-responders.

Direct clustering of CD8+ T cells cannot separate cancer patients who respond to cancer immunotherapy from those who do not (Figure 4 (A)). In the testing data by Yost et al. [19], the classification accuracy at the cell level is low, with AUC in the range of 0.52 to 0.59 for unsupervised loss weights from 0 to 10. After aggregating information across cells, the individual-level AUC varies from 0.82 to 0.48 (Supplementary Materials Section 3). A weight of 2.0 for the unsupervised loss is still a good choice (Supplementary Figure 5). At this weight, the cell-level and individual-level AUCs are 0.53 and 0.66, respectively (Figure

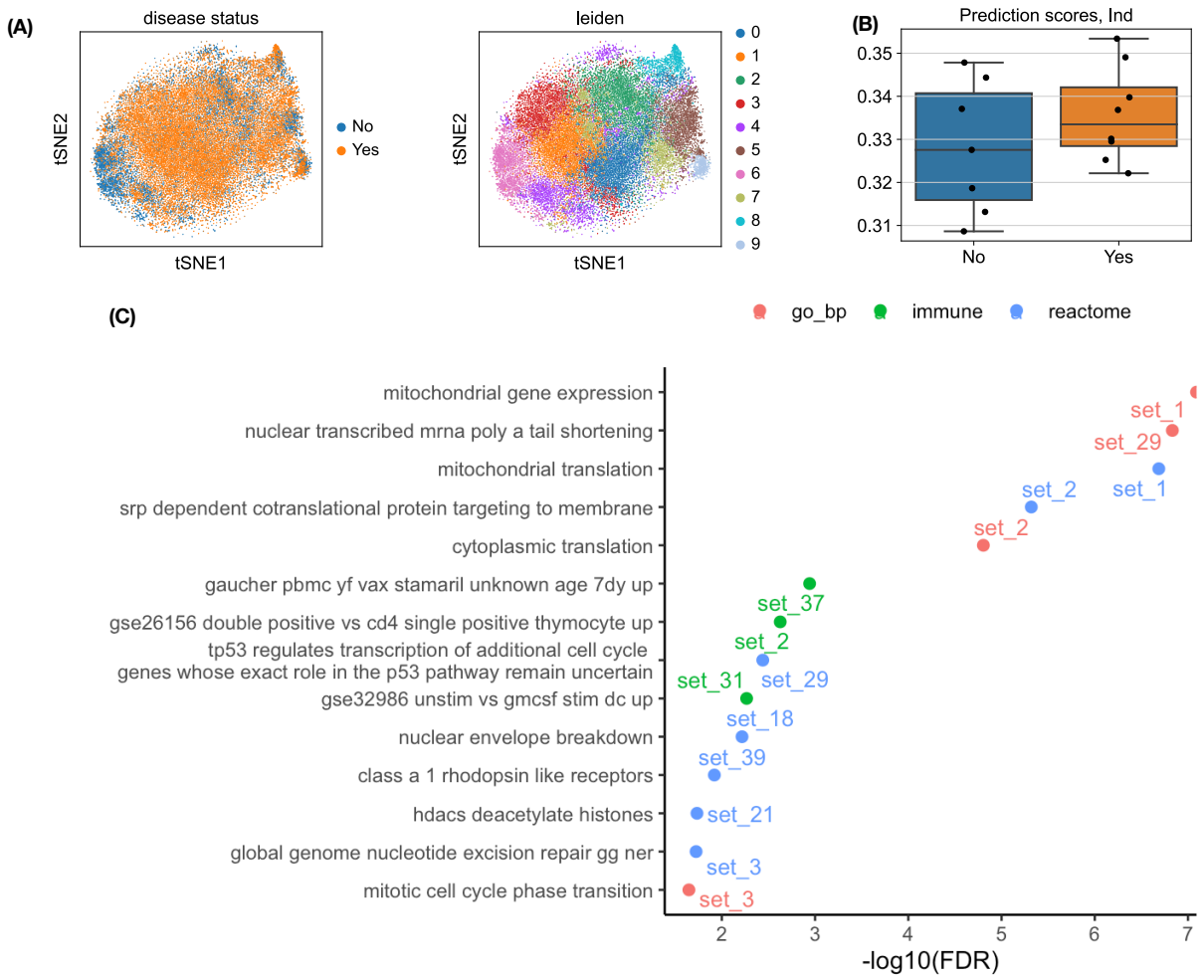

Figure 4: **(A-B)** tSNE projection of CD8+ T cells in testing data, colored by the response to immunotherapy (Yes or No) or clusters generated by the Leiden algorithm. **(B)** The distributions of individual level prediction scores for patients who respond to immunotherapy (Yes) or not (No), using weight 2.0 for unsupervised loss. **(C)** The functional categories that are over-represented by the genes in the 40 gene sets learned by SDAN, using a weight of 2.0 for unsupervised loss.

3B).

Next, we queried the functional categories that were enriched in the gene sets identified by SDAN. The top enrichment is from gene set 1, which is enriched with genes involved mitochondrial function. This enrichment is robust for unsupervised weights from 2.0 to 10 (Supplementary Figures 6-7). Gene set 1 is composed of 23 genes, including six genes that are mitochondrial ribosomal proteins or involved in mitochondrial ribosome function, three genes involved in transcription/translation in mitochondrial, a mitochondrial methionyl-tRNA formyltransferase (MTFMT), two mitochondrial tRNA synthetases ß (SARS2 and EARS2), and four genes involved in NADH:ubiquinone oxidoreductase (Supplementary Table 1). NADH plays an essential role in generating energy within cells. The collection of mitochondrial and NADH genes suggests that this gene set captures the metabolism characteristics of T cells. It is well recognized that the metabolism state of T cells is closely related to T cell immune response [20]. Recent studies have also shown that mitochondrial dysfunction could affect T cell response to neoantigens (e.g., leading to terminally exhausted T cells) [21, 22], and thus may affect cancer patients' response to immunotherapy.

Discussion

Biological knowledge, such as gene-gene interactions, is often noisy and may vary across different tissues or conditions. Consequently, when using such knowledge in machine learning methods, it is crucial to have the flexibility to learn the subset of biological knowledge that is relevant to the particular task. The SDAN method achieves this goal by combining supervised classification loss and unsupervised graph loss, which prefer gene sets that align with biological knowledge and their expression can classify cells. In contrast, many earlier works incorporate biological knowledge into neural networks by designing the neural network structure using gene-regulation networks [23, 24, 25, 26] and lack the flexibility to select the relevant subset of biological knowledge.

In this work, we label cells according to their corresponding individuals. For example, when comparing cells from severe and mild COVID-19 patients, we label the cells from individuals with severe or mild cases as severe or mild cells, respectively. However, some

of these cell labels, inherited from their corresponding individuals, may be incorrect due to cellular heterogeneity. Despite these noisy labels, we have demonstrated that SDAN can still learn and extract useful signals. An interesting future direction is to explicitly handle such noisy labels [27, 28, 29].

Online Methods

Neural network architecture and loss functions

Let n be the number of cells in the training data, p be the number of genes, and d be the number of components after dimension reduction. Each gene is regarded as a node in the gene-gene interaction graph, with the node features being the expression data for each gene. We utilized a graph pooling operator to reduce the gene expression from p genes to d components. Denote the transposed gene expression data matrix by $X \in \mathbb{R}^{p \times n}$. Gene annotations are summarized by an adjacency matrix $A \in \mathbb{R}^{p \times p}$ (including self-connections). Some annotations, such as protein-protein interactions, are already in this adjacent matrix form. For other annotations that group genes into categories, such as gene ontology, gene-gene similarities may also be defined as a continuous measurement [30], and thus create a weighted gene-gene interaction graph. For our method, A does not need to be binary; it can also represent a continuous measurement of similarity. Let $\tilde{A}$ be the normalized adjacency matrix $\tilde{A} = D^{-1/2} A D^{-1/2}$, where D is the degree matrix of A. The goal is to learn an assignment matrix $S \in \mathbb{R}^{p \times d}$ to assign some of the p genes to the d components.

The GNN has two steps: graph convolution followed by graph pooling. We apply the graph convolution networks [31] to reduce the high-dimensional data of each gene (i.e., its expression in n cells) to a small number of hidden states h:

$$H^{(1)} = \mathrm{ReLU}(\tilde{A}XW + XW_{\text{skip}}),$$

where $H^{(1)} \in \mathbb{R}^{p \times h}$ is the output of the first layer, $W \in \mathbb{R}^{n \times h}$ and $W_{\text{skip}} \in \mathbb{R}^{n \times h}$ are two weight matrices to be estimated, and ReLU denotes the non-linear activation function Rectified Linear Unit. $\tilde{A}X$ represents the graph-convoluted gene expression data. The same weight matrix W is applied to all the genes, resembling the operation in a convolutional

neural network where the same convolutional operation is applied across all segments of an image. The additional XW_{skip} allows each gene to give more weight to its own expression. This graph convolution layer can be applied multiple times. For the ℓ-th layer, the formulation is $H^{(\ell)} = \text{ReLU}(\tilde{A}H^{(\ell-1)}W^{(\ell-1)} + H^{(\ell-1)}W_{\text{skip}}^{(\ell-1)})$. The output of the final layer is denoted by H. In our implementation, we used two hidden layers for the graph convolution networks, with the number of hidden states being 64 in each layer.

The second step is graph pooling. The simplest case is one-step graph pooling. The assignment matrix S is obtained by $S = \text{Softmax}(HW_H) \in \mathbb{R}^{p\times d}$, where the Softmax operation is applied on each row (i.e., each gene) to assign the weights of each gene to d latent components, and $W_H \in \mathbb{R}^{h\times d}$ is a weight matrix to be estimated. Graph pooling can also involve multiple layers, which becomes hierarchical graph pooling. Let $X_r = X^\top S$ be the projection of scRNA-seq data in the latent space. We used X_r to predict class labels via a MLP with three hidden layers, each containing 64 hidden nodes and a ReLU activation function.

Now we define the loss function as a combination of two components: an unsupervised graph loss and a supervised classification loss. For the graph loss, we adopt the "minCUT" loss, which is formulated to resemble spectral clustering [32]. The graph loss is the sum of two terms: $\mathcal{L}_{\text{c}}$ and $\mathcal{L}_{\text{o}}$, defined as follows:

$$\mathcal{L}_{\text{c}} := -\frac{\text{Tr}(S^\top \tilde{A} S)}{\text{Tr}(S^\top \tilde{D} S)} \quad \text{and } \mathcal{L}_{\text{o}} := \left\| \frac{S^\top S}{\|S^\top S\|_F} - \frac{I_d}{\sqrt{d}} \right\|_F,$$

where $\|\cdot\|_F$ denotes the Frobenius norm, $\tilde{A}$ is the normalized adjacency matrix, $\tilde{D}$ is the degree matrix of $\tilde{A}$, and I_d is the d-dimensional identity matrix. The term $\mathcal{L}_{\text{c}}$ encourages strongly connected nodes to be grouped together. The term $\mathcal{L}_{\text{o}}$ encourages gene assignment to be orthogonal (i.e., each gene is assigned to only one latent component) and ensures that all latent components have similar sizes. The classification loss is defined by the cross-entropy loss, denoted as $\mathcal{L}_{\text{clf}}$. Therefore, the final loss function is $\mathcal{L}_{\text{clf}} + w(\mathcal{L}_{\text{mc}} + \mathcal{L}_{\text{o}})$, where w adjusts the weight for the unsupervised graph loss. We set $w = 2$ by default.

Evaluation and generalization

Let $X_{\text{test}} \in \mathbb{R}^{n' \times p'}$ be the test data, where p' may not be equal to p; that is, the gene lists of the training data and the test data may be different. When the gene list of the test data is the same as that of the training data, the dimension reduction of the test data is performed as $X_{\text{test,r}} = X_{\text{test}} S$ using the trained assignment matrix S. We then apply the trained MLP to $X_{\text{test,r}}$ to obtain the scores for each cell in the test data. If the gene list of the test data differs from that of the training data, we identify the common genes between the test data and the training data. We then use the corresponding sub-matrices of X_{test} and S to perform the dimension reduction on test data.

Implementation

The gene-gene interaction graph was constructed based on protein-protein interaction annotation from the BioGRID database: `BIOGRID-ORGANISM-Homo_sapiens-4.4.204.tab3.txt.gz`. We matched genes by their official symbols in the BioGRID database. There is an edge between two genes if their interaction exists in BioGRID.

Given the labels for all the cells (e.g., severe or mild COVID-19, dementia or not, or responding to immunotherapy or not), we first conducted differential expression analysis to identify differentially expressed genes. This step preselects the genes to facilitate model training since the original gene list is large. Specifically, we selected marker genes for each label/class separately. For example, in the case of severe and mild COVID-19, we performed a one-sided Mann-Whitney U rank test on each gene using the training data. To select the marker genes for severe (mild), the alternative hypothesis is that the expression in the severe (mild) group is stochastically greater than that in the other group. This test can also be generalized to a multi-class setting, with one group versus all other groups. We accounted for multiple testing by controlling the false discovery rate (FDR) using the Benjamini-Hochberg procedure and then selected the genes with FDR ≤ 0.05. As the number of marker genes may be large, we kept a maximum number of highly variable marker genes for each label, ranking the selected marker genes by normalized variance (computed using normalized data across all class labels). We selected at most 1,000 marker genes for each

class. We then combined all the marker genes for different labels to create the gene list used later.

For the dimension reduction, we specified the number of components to be 40. For the unsupervised clustering using the Leiden algorithm to evaluate the latent components, we set the number of neighbors to 20 to compute a neighborhood graph and used a resolution value of 1.

The method was implemented using PyTorch. We used the Adam algorithm for optimization, with a learning rate of 0.0001 and weight decay of 0.0001. To avoid overfitting, we added dropout and early stopping. Dropout modules were added after each layer of the MLP, except the final layer, with a dropout probability of 0.5. We set the maximum number of epochs to 50,000 and the minimum number of epochs to 10,000. Training stops after no improvement in validation loss for 3,000 epochs.

When making predictions for each individual, we took the mean value of the scores across all their cells as the individual-level score.

Data Availability

Su et al. COVID-19 dataset

Gene expression data were downloaded from ArrayExpress: `https://www.ebi.ac.uk/arrayexpress/experiments/E-MTAB-9357/` on 2/22/2022. Patient information were extracted from Table S1 of Su et al. [11]: `https://data.mendeley.com/datasets/tzydswhhb5/5` on 2/22/2022.

SEA-AD dataset

The snRNA-seq data were downloaded from cellxgene website`https://cellxgene.cziscience.com/collections/1ca90a2d-2943-483d-b678-b809bf464c30` on 2021/11/21. Donor meta data and clinical data were downloaded from SEA-AD website `https://portal.brain-map.org/explore/seattle-alzheimers-disease/seattle-alzheimers-disease-brain-cell-atlas-down` on 2022/1/7.

Cancer immunotherapy dataset

The gene expression data of Sade-Feldman et al. 2018 [18] were downloaded from `https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE120575.` on 2023/11/6. Cell and sample information were extracted from Supplementary Table 1 of Sade-Feldman et al. 2018 [18]. Gene expression and meta data of Yost et al. 2019 [19] were downloaded from `https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE123813`.

References

[1] Gökcen Eraslan, Lukas M Simon, Maria Mircea, Nikola S Mueller, and Fabian J Theis. Single-cell rna-seq denoising using a deep count autoencoder. *Nature communications*, 10(1):1–14, 2019.

[2] Jingshu Wang, Divyansh Agarwal, Mo Huang, Gang Hu, Zilu Zhou, Chengzhong Ye, and Nancy R Zhang. Data denoising with transfer learning in single-cell transcriptomics. *Nature methods*, 16(9):875–878, 2019.

[3] Divyansh Agarwal, Jingshu Wang, Nancy R Zhang, et al. Data denoising and post-denoising corrections in single cell rna sequencing. *Statistical Science*, 35(1):112–128, 2020.

[4] Tian Tian, Ji Wan, Qi Song, and Zhi Wei. Clustering single-cell rna-seq data with a model-based deep learning approach. *Nature Machine Intelligence*, 1(4):191–198, 2019.

[5] Xiangjie Li, Kui Wang, Yafei Lyu, Huize Pan, Jingxiao Zhang, Dwight Stambolian, Katalin Susztak, Muredach P Reilly, Gang Hu, and Mingyao Li. Deep learning enables accurate clustering with batch effect removal in single-cell rna-seq analysis. *Nature communications*, 11(1):1–14, 2020.

[6] Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018.

[7] Chenling Xu, Romain Lopez, Edouard Mehlman, Jeffrey Regier, Michael I Jordan, and Nir Yosef. Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Molecular systems biology*, 17(1):e9620, 2021.

[8] Valerie Oberhardt, Hendrik Luxenburger, Janine Kemming, Isabel Schulien, Kevin Ciminski, Sebastian Giese, Benedikt Csernalabics, Julia Lang-Meli, Iga Janowska, Julian Staniek, et al. Rapid and stable mobilization of CD8+ T cells by SARS-CoV-2 mrna vaccine. *Nature*, 597(7875):268–273, 2021.

[9] Alison Tarke, Camila H Coelho, Zeli Zhang, Jennifer M Dan, Esther Dawen Yu, Nils Methot, Nathaniel I Bloom, Benjamin Goodwin, Elizabeth Phillips, Simon Mallal, et al. SARS-CoV-2 vaccination induces immunological T cell memory able to cross-recognize variants from alpha to omicron. *Cell*, 185(5):847–859, 2022.

[10] Paul Moss. The T cell immune response against SARS-CoV-2. *Nature immunology*, 23(2):186–193, 2022.

[11] Yapeng Su, Daniel Chen, Dan Yuan, Christopher Lausted, Jongchan Choi, Chengzhen L Dai, Valentin Voillet, Venkata R Duvvuri, Kelsey Scherler, Pamela Troisch, et al. Multi-omics resolves a sharp disease-state shift between mild and moderate covid-19. *Cell*, 183(6):1479–1495, 2020.

[12] Guangyan Miao, Hongyu Zhao, Yan Li, Mingming Ji, Yong Chen, Yi Shi, Yuhai Bi, Peihui Wang, and Hong Zhang. Orf3a of the covid-19 virus sars-cov-2 blocks hops complex-mediated assembly of the snare complex required for autolysosome formation. *Developmental cell*, 56(4):427–442, 2021.

[13] Colin L Masters, Randall Bateman, Kaj Blennow, Christopher C Rowe, Reisa A Sperling, and Jeffrey L Cummings. Alzheimer's disease. *Nature reviews disease primers*, 1(1):1–18, 2015.

[14] Ashley N Brandebura, Adrien Paumier, Tarik S Onur, and Nicola J Allen. Astrocyte contribution to dysfunction, risk and progression in neurodegenerative disorders. *Nature Reviews Neuroscience*, 24(1):23–39, 2023.

[15] Katherine E Prater, Kevin J Green, Sainath Mamde, Wei Sun, Alexandra Cochoit, Carole L Smith, Kenneth L Chiou, Laura Heath, Shannon E Rose, Jesse Wiley, et al. Human microglia show unique transcriptional changes in alzheimer's disease. *Nature Aging*, 3(7):894–907, 2023.

[16] Mariano I Gabitto, Kyle J Travaglini, Victoria M Rachleff, Eitan S Kaplan, Brian Long, Jeanelle Ariza, Yi Ding, Joseph T Mahoney, Nick Dee, Jeff Goldy, et al. Integrated multimodal cell atlas of alzheimer's disease. *Research Square*, 2023.

[17] Jonathan J Havel, Diego Chowell, and Timothy A Chan. The evolving landscape of biomarkers for checkpoint inhibitor immunotherapy. *Nature Reviews Cancer*, 19(3):133–150, 2019.

[18] Moshe Sade-Feldman, Keren Yizhak, Stacey L Bjorgaard, John P Ray, Carl G de Boer, Russell W Jenkins, David J Lieb, Jonathan H Chen, Dennie T Frederick, Michal Barzily-Rokni, et al. Defining t cell states associated with response to checkpoint immunotherapy in melanoma. *Cell*, 175(4):998–1013, 2018.

[19] Kathryn E Yost, Ansuman T Satpathy, Daniel K Wells, Yanyan Qi, Chunlin Wang, Robin Kageyama, Katherine L McNamara, Jeffrey M Granja, Kavita Y Sarin, Ryanne A Brown, et al. Clonal replacement of tumor-specific t cells following pd-1 blockade. *Nature medicine*, 25(8):1251–1259, 2019.

[20] Evanna L Mills, Beth Kelly, and Luke AJ O'Neill. Mitochondria are the powerhouses of immunity. *Nature immunology*, 18(5):488–498, 2017.

[21] Santosha A Vardhana, Madeline A Hwee, Mirela Berisa, Daniel K Wells, Kathryn E Yost, Bryan King, Melody Smith, Pamela S Herrera, Howard Y Chang, Ansuman T Satpathy, et al. Impaired mitochondrial oxidative phosphorylation limits the self-renewal of t cells exposed to persistent antigen. *Nature immunology*, 21(9):1022–1033, 2020.

[22] Hao Wu, Xiufeng Zhao, Sophia M Hochrein, Miriam Eckstein, Gabriela F Gubert, Konrad Knöpper, Ana Maria Mansilla, Arman Öner, Remi Doucet-Ladevèze, Werner

Schmitz, et al. Mitochondrial dysfunction promotes the transition of precursor to terminally exhausted t cells through hif-1α-mediated glycolytic reprogramming. *Nature communications*, 14(1):6858, 2023.

[23] Jianzhu Ma, Michael Ku Yu, Samson Fong, Keiichiro Ono, Eric Sage, Barry Demchak, Roded Sharan, and Trey Ideker. Using deep learning to model the hierarchical structure and function of a cell. *Nature methods*, 15(4):290–298, 2018.

[24] Jie Hao, Youngsoon Kim, Tae-Kyung Kim, and Mingon Kang. Pasnet: pathway-associated sparse deep neural network for prognosis prediction from high-throughput data. *BMC bioinformatics*, 19(1):1–13, 2018.

[25] Haitham A Elmarakeby, Justin Hwang, Rand Arafeh, Jett Crowdis, Sydney Gang, David Liu, Saud H AlDubayan, Keyan Salari, Steven Kregel, Camden Richter, et al. Biologically informed deep neural network for prostate cancer discovery. *Nature*, 598(7880):348–352, 2021.

[26] Mohammad Lotfollahi, Sergei Rybakov, Karin Hrovatin, Soroor Hediyeh-zadeh, Carlos Talavera-López, Alexander Misharin, and Fabian J Theis. Biologically informed deep learning to infer gene program activity in single cells. *bioRxiv*, 2022.

[27] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31, 2018.

[28] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? *International Conference on Machine Learning*, pages 7164–7173, 2019.

[29] Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *International Conference on Learning Representations*, 2019.

[30] Guangchuang Yu, Fei Li, Yide Qin, Xiaochen Bo, Yibo Wu, and Shengqi Wang. GOSemSim: an R package for measuring semantic similarity among go terms and gene products. *Bioinformatics*, 26(7):976–978, 2010.

[31] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[32] Filippo Maria Bianchi, Daniele Grattarola, and Cesare Alippi. Spectral clustering with graph neural networks for graph pooling. In *International Conference on Machine Learning*, pages 874–883. PMLR, 2020.