Deep autoencoder enables interpretable tissue-adaptive deconvolution and cell-type-specific gene analysis

Yanshuo Chen*[1,2], Yixuan Wang*[1,3], Yuelong Chen[4], Yumeng Wei[1], Yunxiang Li[1], Ting-Fung Chan[4], and Yu Li †[1]

[1]Department of Computer Science and Engineering, CUHK, Hong Kong SAR, China
[2]School of Life Sciences, Tsinghua University, 100084 Beijing, China
[3]Department of Mathematics, HIT, 264209 Weihai, China
[4]School of Life Sciences, CUHK, Hong Kong SAR, China

Abstract

Single-cell RNA-seq has become a powerful tool for researchers to study biologically significant characteristics at explicitly high resolution, but its application on emerging data is currently limited by its intrinsic techniques. Here, we introduce TAPE, a deep learning method that connects bulk RNA-seq and single-cell RNA-seq to balance the demands of big data and precision. By taking advantage of constructing an interpretable decoder and training under a unique scheme, TAPE can predict cell-type fractions and cell-type-specific gene expression tissue-adaptively. Compared with existing methods on several benchmarking datasets, TAPE is more accurate (up to 40% performnace improvement on the real bulk data) and faster than the previous methods. It is sensitive enough to provide biologically meaningful predictions. For example, only TAPE can predict the tendency of increasing monocytes-to-lymphocytes (MLR) ratio in COVID-19 patients from mild to serious symptoms, whose estimated indices are consistent with laboratory data. More importantly, through the analysis of clinical data, TAPE shows its ability to predict cell-type-specific gene expression profiles with biological significance. Combining with single-sample gene set enrichment analysis (ssGSEA), TAPE also provides valuable clues for people to investigate the immune response in different virus-infected patients. We believe that TAPE will enable and accelerate the precise analysis of high-throughput clinical data in a wide range.

1 Introduction

RNA sequencing (RNA-seq) is a single high-throughput sequencing assay combining the identification of transcriptome and the quantification of gene expression [1]. The remarkable application of RNA-seq lies not only in surveying gene expression levels but also in discovering novel gene structures and allele-specific expression [2]. At an unprecedented resolution, single-cell RNA sequencing (scRNA-seq) is a new tool that profiles the cell-to-cell genomic variation and cell lineages [3, 4]. Cell state transitions [5] and previously obscured cellular populations, such as Foxi1+ pulmonary ionocyte [6], have been studied with the application of scRNA-seq. Although RNA-seq and scRNA-seq are powerful insights into molecular characterization, the limitations of using them independently while carrying out cell type deconvolution are obvious. RNA-seq only measures the averaged expression levels of the targeted heterogeneous mixtures, leading to the requirement of prior knowledge or pre-selected marker gene [7, 8]. scRNA-seq is an expensive technique providing quantitative cell level transcriptional profiling that may easily suffer from various noise and bias, making it difficult to study large samples [9]. Consequently, combined studies of RNA-seq and scRNA-seq hold promise for overcoming the above-mentioned challenges, which have potential applications in biomedical fields, such as in-depth studies of changes in cellular composition at different stages of cancer development and changes in cellular components before and after cancer metastasis in the same location.

*Contribute equally
†Corresponding Author. Email: liyu@cse.cuhk.edu.hk

Broad application prospects proliferate a great number of computational cell-type deconvolution algorithms that utilize scRNA-seq as references in the past few years [10–16]. The existing methods can be roughly divided into three categories: non-negative least squares (NNLS) based, support vector regression-based, and deep learning-based. Based on a weighted-NNLS framework, MuSiC has satisfactory accuracy when encountering tissues that comprise closely related cell types, owing to effectively weighing the genes that have high consistency between subjects and cells[10]. DWLS introduces a dampening constraint to prevent the infinite weight when using the classic least squares approach, thus boosting the accuracy of detecting rare cell types and making predictions[11]. Bisque is also a regression-based technique that learns gene-specific bulk expression transformations from scRNA-seq or single-nucleus RNA-seq (snRNA-seq) data to deconvolve RNA-seq data robustly, which strengthens decomposition performance when there is distinct technical variation in reference profiles and observed bulk RNA-seq data generations[12]. CIBERSORT and its updated version, CIBERSORTx, employ the support vector regression (SVR) to perform feature selection[13] and the latter introduces new functionalities for normalization of cross-platform data as well as in silico cell purification[14]. The likelihood-based RNA-Sieve[15] provides a new perspective on solving probabilistic models to deconvolve cell-type under diverse scenarios while increasing the flexibility for continued development, yet is rather slow in generating the reference dictionary and making predictions of the mixture samples.

Deep neural networks (DNNs) can not only extract optimal representations without relying on strictly linear input data but also learn to represent the potential features that are robust to bias and noise[16]. Scaden is a state-of-the-art deep learning-based method that utilizes in silico tissue datasets generated by merging cells from scRNA-seq and predicts cell type proportions of the input expression samples[16]. It is composed of a five-layer neural network with three combinations of layer sizes to predict cell-type proportions. The application of DNNs remarkably reduces dependence on the optimal design of gene expression profiles (GEPs) that is essential to the traditional statistical methods. Scaden inspires us to build a more interpretable deep learning-based method to achieve better bio-significance. On the other hand, in lack of parameter learning, traditional statistical methods are directly applied to the various dataset without necessary adaptations. This motivates us to develop a tissue-adaptive method with DNNs' structural superiority.

The encoder and decoder of an AutoEncoder (AE) are regarded as a couple of inverse functions, that is, we can use one of them to explain another in math. The encoding predictions can be validated, illustrated, and refined by the reconstructed input generated by the decoder of an AE[17]. Here, we present a more accurate, efficient, and interpretable tissue-adaptive deep learning algorithm, Tissue-AdaPtive autoEncoder (TAPE), which is based on AE. Being trained on simulated bulk RNA-seq data with ground truth like Scaden, TAPE predicts the cell composition of the input targeted samples up to 34% more accurately on simulated bulk RNA-seq data with noise and up to 40% on real bulk RNA-seq data in comparison with Scaden. Notice that TAPE runs three times faster than all of the other statistical methods. To further demonstrate the clinical rationality of TAPE, we use three datasets, namely ROSMAP dataset[18], COVID-19 PBMC dataset[19], and COVID-19 infected cultured pancreas islet dataset[20], to show that TAPE is sensitive to biological changes. For instance, TAPE is the only method that predicts the increasing tendency of MLR value which is suitable within the clinical report (0.29-0.88)[21] in COVID-19 PBMC dataset. Unlike prior approaches, TAPE automatically extracts representation information, which does not reliant on pre-defined GEP, and even predicts tissue-adaptively cell-type-specific gene expression. More specifically, TAPE only requires simulated data from healthy samples to train but it could predict cell-type-specific gene expression in pathological conditions once given corresponding bulk RNA-seq data. With the selection of "high-resolution" mode of the prediction of cell-type specific GEP, TAPE should provide a valuable reference for biologists who want to investigate differentially expression gene at a cell-type-specific level. We further prove the versatility of TAPE by implementing TAPE on virus-infected PBMC RNA-seq data to analyze the specific function differences in specific cells. This leads to the promising contribution of treatment and prevention of these infections.

In summary, we built an interpretable tissue-adaptive autoencoder that enables precise prediction of cell fractions and cell-type-specific gene expression called TAPE. Compared with the state-of-art methods, TAPE not only shows a competitive performance and the fastest processing speed on both pseudo and real bulk datasets, but is also sensitive to the biological changes in the bulk RNA-seq data, and can produce biologically significant results. Another noteworthy contribution is that TAPE predicts the cell-type-specific gene expression tissue-adaptively, allowing the dissection of bulk gene expression into different cell types and discovery of the potential differential gene expression among cell types. It compensates for the weak interpretability of the existing deep learning-based deconvolution method.

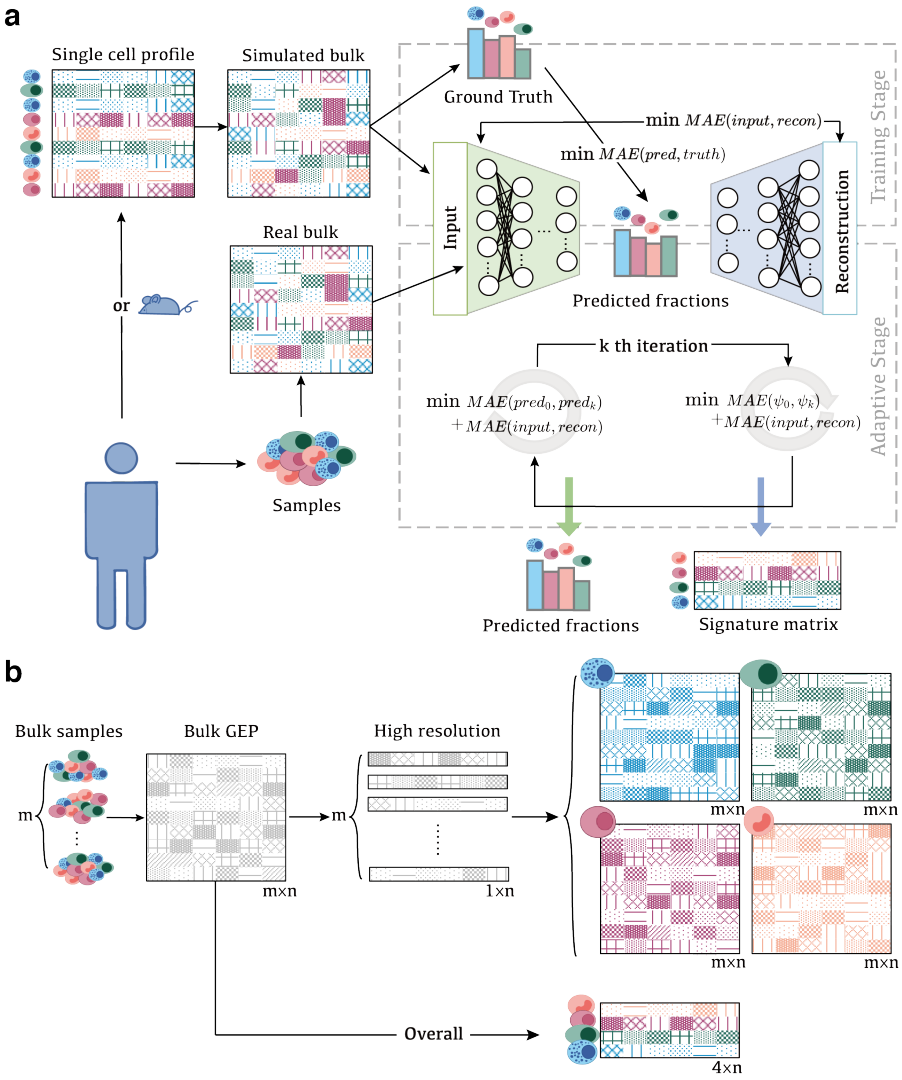

Figure 1: **TAPE workflow and clarification of adaptive stage. a** TAPE takes scRNA-seq data from human or mouse samples and RNA-seq data from human samples as input, then performs deconvolution as well as prediction of cell-type specific GEPs via a training stage and an adaptive stage. **b** Generation of cell-type-specific GEPs has two separate modes. The first is the high-resolution mode: TAPE takes RNA-seq data from one sample at a time as input and outputs an adapted cell-type-specific signature matrix for each sample. The second the overall mode: TAPE takes all the RNA-seq data at one time as input and outputs one signature matrix adapted to all samples.

2 Results

2.1 Method overview

Unlike Scaden[16] only predicts cell proportions, we aimed at building a model that could not only performs deconvolution but also is capable of illustrating why it makes this prediction tissue-adaptively. These demands could be satisfied with AutoEncoder (AE) indirectly. AE's encoder and decoder are a pair of inverse functions. If we have the explicit matrix-form of the decoder, it would greatly enhance the interpretability of the encoder and indirectly tell us why the encoder makes such a prediction. Consequently, we built our TAPE method based on the architecture.

As shown in Figure 1, the basic architecture of our method is a DNN-based AutoEncoder, taking bulk gene expression profile as input and outputting cell-type proportions and cell-type-specific gene expression profiles (GEPs). There are three stages of using TAPE. The first stage is to create training

data through simulation. Simulated bulk data is a linear combination of single-cell gene expression profiles with predefined cell fractions and total cell numbers. The single-cell profile and the real bulk profile should come from the same tissue. The next is the training stage. We want to train the model to output the proper cell fractions after encoder and use the cell fractions to reconstruct the bulk profile. More than only using the reconstruction loss in the classic AE model, we try to minimize mean absolute error (MAE) between the ground truth and the predicted cell fractions to make it supervised. When the model is required to predict cell fractions and cell-type-specific GEPs on the real bulk data, it enters the adaptive stage. In this process, inspired by the classic AE's training process, we only use real bulk data to train the model in an unsupervised manner. More specifically, the model is iteratively greedily optimized on the decoder and the encoder. That is, it would not optimize the parameters of the encoder until it achieves the temporally best parameters on the decoder. As for the decoder, we require it to reconstruct the real bulk data and maintain the concordance with itself, while the encoder is required to predict proper cell fractions and similar to the primary prediction after the training stage. Since we require the decoder to output bulk gene expression based on the cell fractions, the parameters of the decoder are the cell-type-specific GEPs as a matter of course, so we could directly output these parameters as the GEPs after the adaptive stage.

2.2 Performance evaluation on pseudo-bulk data

We used TAPE to analyze pseudo-bulk data generated in silico from single-cell gene expression profiles. To simulate the real deconvolution scenario, we added artificial noise, like Gaussian noise and drop-out events when constructing the pseudo-bulk data. On simulated data generated from single-cell data of five mouse organs[22], we compared TAPE's performance to that of four representative cell deconvolution methods, namely Scaden, RNAsieve, CSx, and DWLS, with 5-fold validation. Performance was evaluated by the mean absolute error (MAE) and Lin's concordance correlation coefficient (CCC) [23] between prediction and ground truth. More details of the simulation process, dataset, and metrics for evaluation are in the Method part.

As shown in Figure 2, TAPE achieves the best performance on all the datasets. TAPE reduces 13%-34% of the MAE and raises 2%-34% of the CCC, compared to Scaden. This lies in the decoder we bring in, which regularizes the output fractions of the encoder to approach the specific biological significance via minimizing the reconstruction loss. Regarding handling the artificial noise, deep learning-based methods are more robust than the statistical linear models, which is highly related to the structural superiority of neural networks. The likelihood-based inference method, RNA-Sieve, is 5%-40% worse than TAPE while performing better than CSx and DWLS. CSx shows excellent performance on pseudo-bulk data that simply sum the single-cell RNA-seq read counts up while showing less accuracy when encountering artificial noise. The input requirements for DWLS are very strict, and our data needed to be logged to avoid overflow.

2.3 Accurate and stable deconvolution on real bulk data

We further evaluated TAPE and the other four representative cell deconvolution methods on real tissue expression datasets with corresponding ground truth. First, we assessed deconvolution performance on two human PBMC bulk RNA-seq datasets, SDY67 [24] and the S13 cohort from Monaco [25]. Another PBMC microarray dataset was obtained from *Newman et al* [13]. Second, we deconvolved the ROSMAP human brain RNA-seq dataset [26] with both human brain single-cell RNA-seq and mouse brain single-cell RNA-seq as reference. Through immunohistochemistry analysis, the cell-type fractions of 41 samples of the ROSMAP dataset were recently given [18]. Detailed deconvolution software comparison and settings are in the Method part.

Among all the real datasets considered, TAPE achieves the best MAE on SDY67 (0.061), Monaco (0.044), and ROSMAP (human scRNA-seq as reference 0.036; mouse scRNA-seq as reference 0.044), while Scaden performs (0.070) slightly better than TAPE (0.073) only in Newman. Newman is a microarray dataset. Unlike the scRNA-seq or RNA-seq that we use for training and prediction, the differentiation among the expression of genes is very small. Also, Scaden is an ensemble of 3 different models, while TAPE uses only one model. RNAsieve obtains the worst MAE (0.26 and 0.25) in the PBMC datasets, SDY67 and Monaco, for the poor learning ability for unknown cell type ratios but achieves better performance (human scRNA-seq as reference 0.09; mouse scRNA-seq as reference 0.06) in ROSMAP than other statistical deconvolution methods. CSx is a powerful computational tool that only has slightly

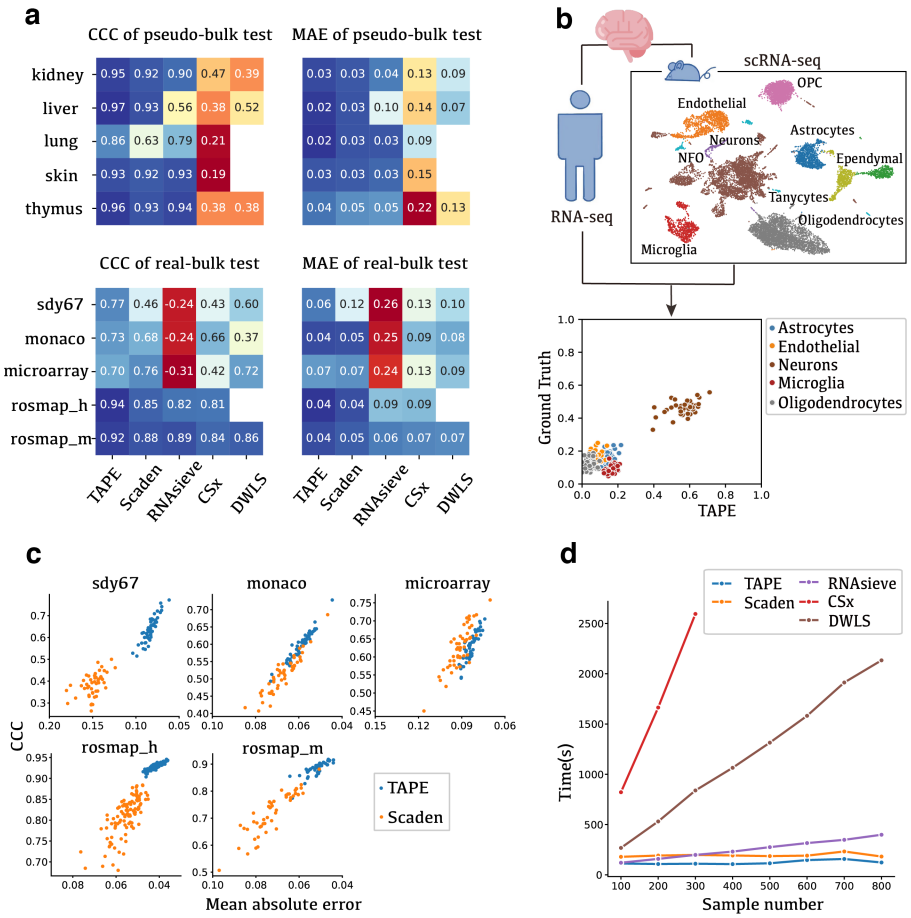

Figure 2: **Comparison of deconvolution algorithms on benchmark datasets.** **a** Deconvolution results on simulated data and real data. CCC represents the Lin's concordance correlation coefficient, measuring the concordance between predicted fraction and ground truth. MAE represents mean absolute error, measuring the accuracy of prediction. Blue represents a good performance, and red represents a bad performance. **b** Deconvolution procedure diagram. Bulk RNA-seq data and single-cell data should come from a homologous tissue. **c** Detailed comparison between Scaden and TAPE on 5 real bulk datasets using 50 different random seeds. The points on the upper right represent better performance than points on the bottom left. **d** Time complexity analysis of different methods. These tests are conducted on simulated datasets. The time limit is set to 2500s. Any longer time test was not conducted. TAPE performs as a constant time algorithm.

worse results with Scaden when deconvolving bulk RNA-seq. For instance, CSx gets an MAE of 0.14 in SDY67, 0.089 in Monaco, human scRNA-seq as reference 0.088; mouse scRNA-seq as reference 0.065 in ROSMAP. The benchmarking results valued by MAE and CCC are detailed in Figure 2a.

Furthermore, since TAPE and Scaden are both DNN-based methods, we made a head-to-head comparison between them using different random seeds to evaluate TAPE's stability. In Figure 2c, the two colors stand for different methods, and the two coordinates of each dot represent MAE and CCC, respectively. As shown in the figure, the dots of TAPE occupy the upper-right of the figures with low variance, showing TAPE's better performance and stability than Scaden.

2.4 Efficient deconvolution on large cohort RNA-seq data

In practice, except for accuracy and stability, the methods' scalability is also important. Therefore, we evaluated the time consumption of the four representative methods mentioned above on the same pseudo-bulk samples. We ran TAPE, Scaden, RNASieve, and DWLS on the same workstation with Intel(R) Xeon(R) Gold 6226 CPU @ 2.70GHz, CentOS Linux release 7.9.2009 (Core), Nvidia 3090 GPU.

CSx was tested through the web-based application. Detailed implementations are in the Method part.

Notice that TAPE runs the fastest among all the methods. According to the time recoded, TAPE deconvolves 800 samples in about 120 seconds, which is one-third faster than Scaden. Besides optimization of the sampling process, TAPE is lighter than Scaden, which combines 3 models. Once the training process is finished, TAPE takes little time compared to the statistical methods. Written in Python, RNASieve runs faster than DWLS and CSx. RNASieve can deconvolve 800 samples in less than 400 seconds, which is quite good. However, the time consumption of RNASieve increases linearly, Thus, it is incapable of dealing with massive samples. DWLS and CSx run at a relatively slow speed. Since the web station of CSx did not provide a timer, we did manual timing. And it needs more than 40 mins to deconvolve 300 samples. Figure 2d is truncated at 2,000 seconds.

2.5 Biologically significant deconvolution on clinical RNA-seq data

We further evaluate whether TAPE could predict cell-type proportions consistent with clinical prior knowledge. Here, we selected three datasets with clinical information or related prior knowledge: 1. ROSMAP dataset [27] is obtained from patients with Alzheimer's Disease (AD); 2. COVID-19 PBMC dataset[19] (GEO accession code: GSE157859) contains clinical information about different severity of COVID-19 (mild, moderate, and serious) and different stages of patients, which include treatment stage, convalescence stage, and rehabilitation stage; 3. COVID-19 infected cultured pancreas islet dataset[20] (GEO accession code: GSE159717), which have RNA-seq data of islet from three different conditions: normal, infected, and infected tissue with Remdesivir treatment. Detailed information of these datasets is in the Method part.

For the ROSMAP dataset, we used the human brain single-cell profile from *Darmanis et al.*[28] as reference. As we know, neuron cell loss is a significant symptom in patients with AD. In the ROSMAP dataset, the braak stage is given as a measurement of the severity of AD[29]. So we expected neuron fraction would decrease with the development of AD. On the other hand, we investigated each sample's braak stage to the estimated fraction of microglia whose proportion will increase with AD severity, while the previous study also showed that the microglia activation will decrease at braak stage 6[30, 31]. The results (Figure 3a, 3b) show that among 532 samples with clinical information, TAPE can predict the tendency of neuron loss and have a good prediction of microglia activation. More than this, according to the immunohistochemistry analysis of AD from a previous study, the proportion of neurons or microglia cells ranging from 0.32-0.55 or 0.06-0.12 respectively[18]. Impressively, only TAPE could predict proportions in this range which shows the great accuracy of TAPE's prediction.

Next, we used PBMC data8k[32] dataset as the single-cell reference to deconvolve the COVID-19 PBMC dataset. According to recent studies, indices like neutrophil-to-lymphocyte ratio (NLR), monocyte-to-lymphocyte ratio (MLR) are directly associated with COVID-19, and these could be used to diagnose the severity of COVID-19 patients[33, 34]. In practice, patients with the higher NLR or MLR show a more serious symptom in COVID-19. Since the data we used is obtained from peripheral blood mononuclear cells (PBMC), which do not contain neutrophils, we only tested the correlation of MLR and the severity of COVID-19 patients. More specifically, we only considered the treatment stage data because patients in the convalescence or rehabilitation stage do not represent the same pathology characteristics as the real infected circumstances. We used the estimated MLR value predicted from different models to compare the tendency between different severity (Figure 3c). Only TAPE predicted the increasing tendency of MLR value, and the value range is suitable with the clinical report (0.29-0.88)[21].

To deconvolve the COVID-19 islet dataset, we used the endocrine cells (alpha, beta, gamma, delta, and epsilon cells) of the single-cell profile from *Baron et al.*[35] to generate training data. Since the infection of SARS-CoV-2 usually causes metabolic dysregulation and Mellitus, researchers used cultured islet tissue to investigate the detailed mechanisms[20]. Here, we used the sequencing data of these *in vitro* cultured islets and expected TAPE to predict the decrease of beta-cell proportion after infection. Furthermore, the proportion of beta-cell should restore after treatment with *Remdisivir*, a very famous antiviral medication used to treat COVID-19 (Figure 3d). We found that only TAPE and Scaden can predict both beta-cell loss and restoration in this experiment. The accurate deconvolution results of these controlled experiments demonstrate that TAPE is sensitive to the biological changes in the bulk RNA-seq data and can produce biologically significant results.

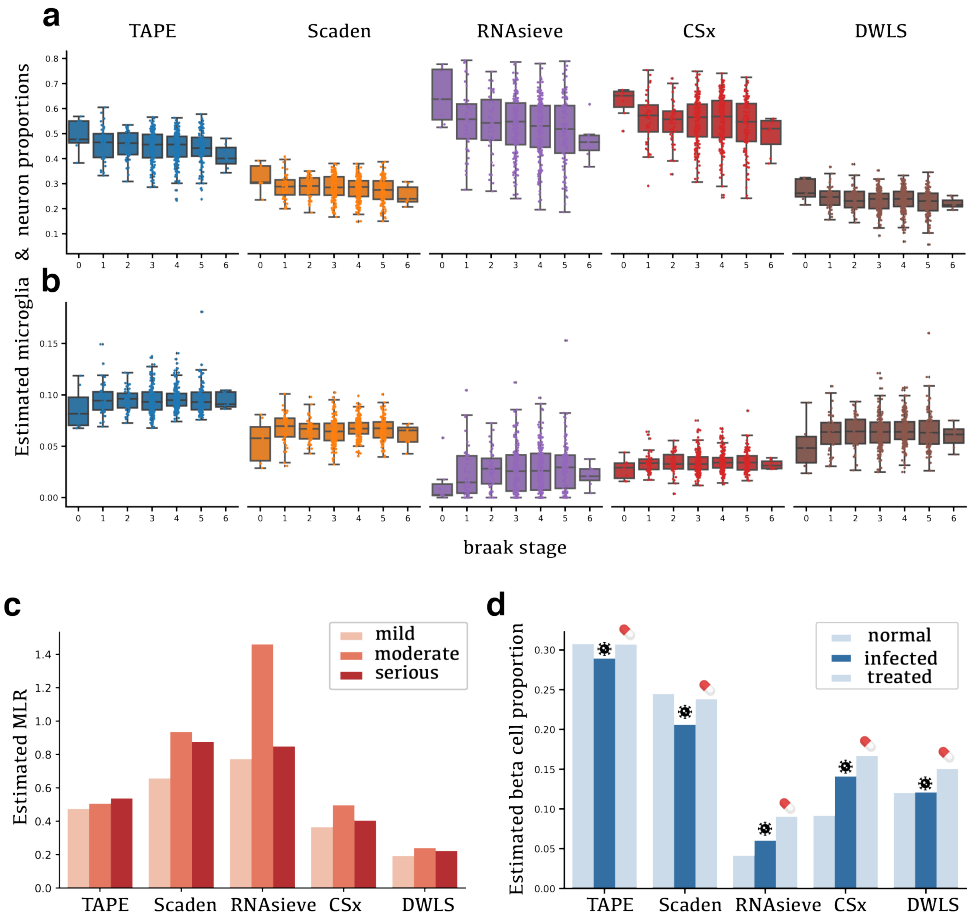

Figure 3: **Deconvolution benchmark on datasets with clinical information.** **a** Comparison of estimated neuron cell proportion on different braak stages between different models on the ROSMAP dataset. Neuron content is expected to decrease along with the development of AD. **b** Microglia content estimated by different methods on braak stage. The fraction is expected to increase from braak stage 0 to 5 while there will be a decrease from braak stage 5 to 6. **c** Estimated MLR values calculated from the estimated monocytes fraction divided by the sum of estimated proportions of $CD4^+$ T cell, $CD8^+$ T cell, and B cell. Only TAPE's prediction has a proper tendency and is accordant with laboratory data. **d** Estimated beta-cell fractions of cultured islet in different conditions. The middle one represents samples infected with SARS-CoV-2, and the right one means samples treated with *Remdesivir* after infection. The model should predict the restoration of beta-cell content after being treated with medication.

2.6 Tissue-adaptive cell-type-specific gene expression prediction

More than only predicting cell fractions of bulk RNA-seq data like the existing deep-learning method, TAPE could also predict the cell-type-specific gene expression tissue-adaptively. That is, TAPE only needs simulated data from healthy samples to train, but it can predict cell-type-specific gene expression in pathological conditions if the corresponding bulk RNA-seq data is given. This feature enables TAPE to dissect bulk gene expression into different cell types and discover some potential differentially expressed genes in different cell types.

Specifically, in the adaptive stage, TAPE will take the bulk RNA-seq data as input and deconvolve it into gene expression at a cell-type level. This adaptive process has two modes, one is the overall mode, and the other is the high-resolution mode (Figure 1b). In the overall mode, the model will take all the RNA-seq data as input and output a signature matrix adapted to all the samples, while in the high-resolution mode, the model will take each RNA-seq data as input and output adapted signature matrix for each sample.

We began with testing the correctness of the predicted cell-type-specific GEP. To test this, we mea-

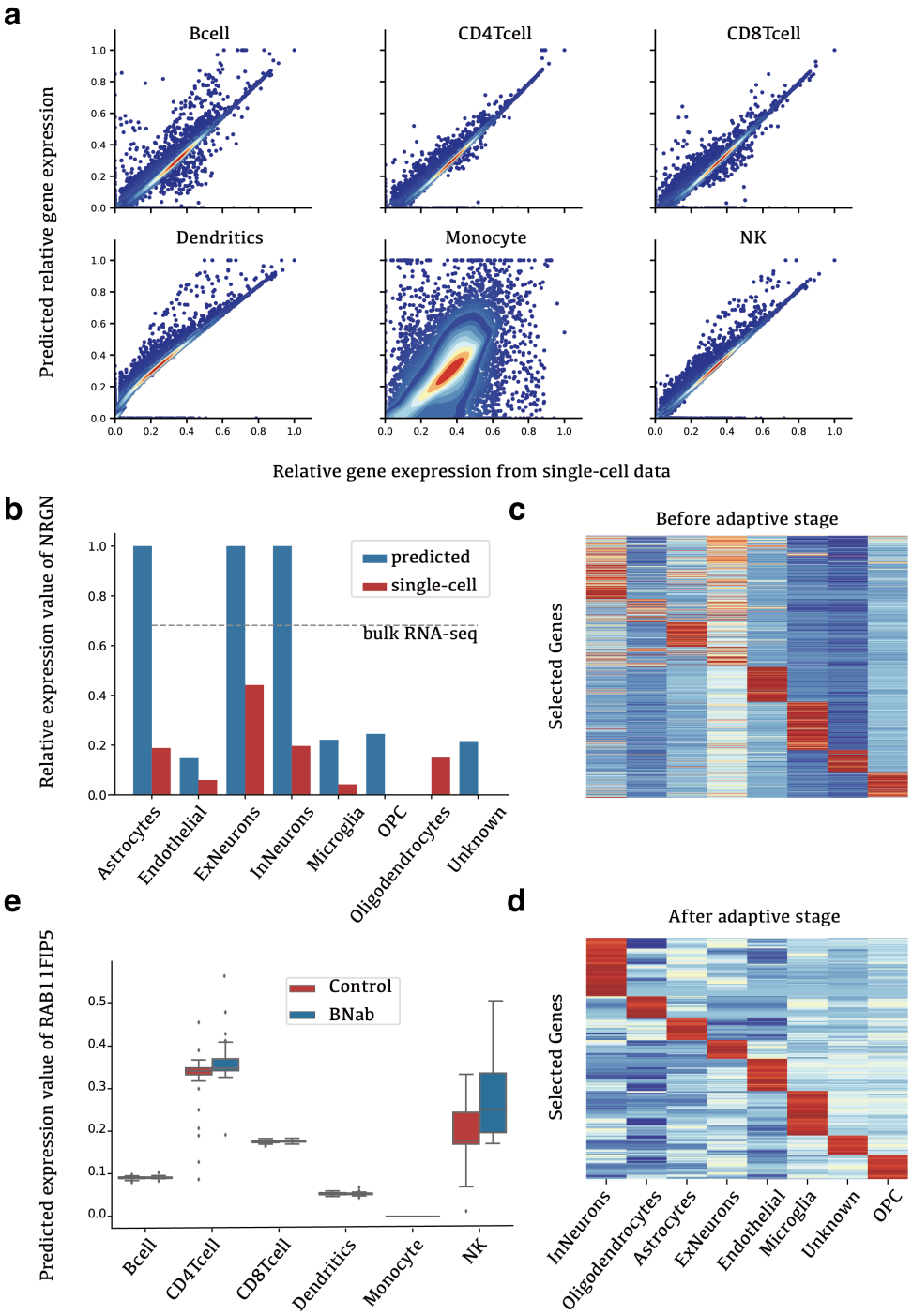

Figure 4: **Cell-type-specific gene expression analysis.** **a** Concordance between predicted relative gene expression value and relative gene expression value from single-cell data. The relative gene expression value is the original expression value after Log_2 and $MinMaxScaler()$ transformation. **b** Relative gene expression values of NRGN from different sources. The dashed line represents the total relative NRGN expression value in the AD patients' brain tissue. **c, d** Estimated signature matrix after the adaptive stage in the overall mode. The gene expression is normalized with Z-Score. The genes are selected by the differential expression in different cell types after the adaptive stage. The differences between the before and after adaptive stage only represent TAPE could make the signature matrix adapted to new data and also maintain concordance with the original one. **e** Boxplot of estimated RAB11FIP5 gene expression values in different cell types. This gene has been proved to have differential expression in NK cells through experiments. TAPE predicts this phenomenon by digital analysis without physical experiments.

sured the concordance between the predicted gene expression value of each cell type and the original gene expression value obtained from single-cell RNA-seq (Fig 4a). Here, the PBMC bulk data is from *Monoco et al.*[25], while the single-cell data is the data8k dataset from the 10X website[32]. Since in the training stage, we transformed the input RNA-seq data into 0-1 value using Log_2 and $MinMaxScaler()$ (more in Method part), the sums of gene expression value grouped by cell types are also transformed in this way to compare to the predicted relative gene expression value. Note that only gene expression in monocyte does not have a good concordance. This may be caused by the batch effect from different individuals. The concordance shown in the figure proves that TAPE predicts the signature matrix correctly and establishes the base for further gene expression analysis.

More than the basic concordance, we also expect that TAPE can assign certain gene expression values in bulk data to different values at the cell-type level. To test this and the model's robustness, we used the ROSMAP RNA-seq dataset[27] and human brain single-cell profile[28] to perform adaptive training in the overall mode. The deconvolution result (Fig 4d) of cell-type-specific GEPs shows that TAPE indeed predicted the differentially expressed genes in different cell types. However, since TAPE takes single-cell gene expression as input, these differences may be inherent from single-cell data. So, we compared the original signature matrix from single-cell data to the adapted signature matrix using the heatmap (Fig 4c). We further investigate whether TAPE just inherits the data distribution from bulk RNA-seq data and the different distributions of different cell types are randomly assigned. We selected the NRGN gene to study it. Since the NRGN gene has been shown to be closely associated with AD[36], we expect it to have a high gene expression level in neurons or other nerve cells. Interestingly, comparing the relative NRGN expression value in bulk GEP, single-cell GEPs, and predicted GEPs (Fig 4b), we found that TAPE can successfully predict the highly expressed NRGN in neurons and the lower expression in endothelial cells. More specifically, this shows that the prediction of cell-type-specific GEPs is a product of two-side information from both bulk and single-cell profile, not randomly assigned or guessed.

2.7 Cell-type-specific differentially gene expression profiling at high-resolution

For HIV-infected patients, they can be classified into two different classes based on the existence of broadly neutralizing antibodies (BNab). Recently, a study about the development mechanism of BNab in HIV patients used bulk RNA-seq and population sorted RNA-seq to investigate the most differentially expressed gene (DEG)[37]. In this study, researchers first analyzed the differentially expressed genes in HIV patients with or without BNab. They found that RAB11FIP5 is the most differentially expressed gene. Then, they used qPCR to investigate which cell type overexpressed RAB11FIP5 in PBMC. The results showed that RAB11FIP5 is the most differentially expressed in natural killer (NK) cells. After that, they designed a series of experiments to prove the overexpressed RAB11FIP5 in NK cells is indeed related to BNab development. The steps they used to find the relation between RAB11FIP5 and NK could be replaced with the cell-type-specific gene expression analysis at high resolution. We used TAPE to tissue-adaptively deconvolve the HIV PBMC data[37] (GEO accession code: GSE115449). To avoid batch effects and harmful effects caused by the low-quality single-cell data, we combined data6k, data8k, and data10k PBMC single-cell data[32, 38, 39] as reference. In the high-resolution mode of TAPE, TAPE could predict cell-type-specific GEPs for each sample. After obtaining the predicted GEPs at high resolution, we calculated the adjusted p-value and fold change for each cell type and the original bulk RNA-seq data (Fig 4e). The results show that TAPE successfully predicts RAB11FIP5 differentially expressed in NK cells, which means that TAPE could tell NK as a source of producing the most differentially expressed gene in PBMC correctly and precisely. Additionally, the differentially expressed RAB11FIP5 in each cell type also proves that TAPE does not just copy the gene expression pattern in bulk RNA-seq data. Instead, it combines the single-cell profile with the real bulk data to predict the tissue-adaptive GEPs. Although the GEPs prediction is not as reliable as real experiments, TAPE could be a valuable reference for biologists investigating differentially expressed genes at a cell-type-specific level.

2.8 Functional investigation across various types of virus infection

To further prove the versatility of TAPE, we applied TAPE on the PBMC RNA-seq data of three kinds of virus-infected samples, including SARS-CoV-2 infected, which is the severe acute respiratory syndrome coronavirus 2 that has been sweeping the world, hepatitis C virus (HCV) infected, which caused 290,000 death in 2019, and human immunodeficiency virus (HIV) infected, which will lead acquired

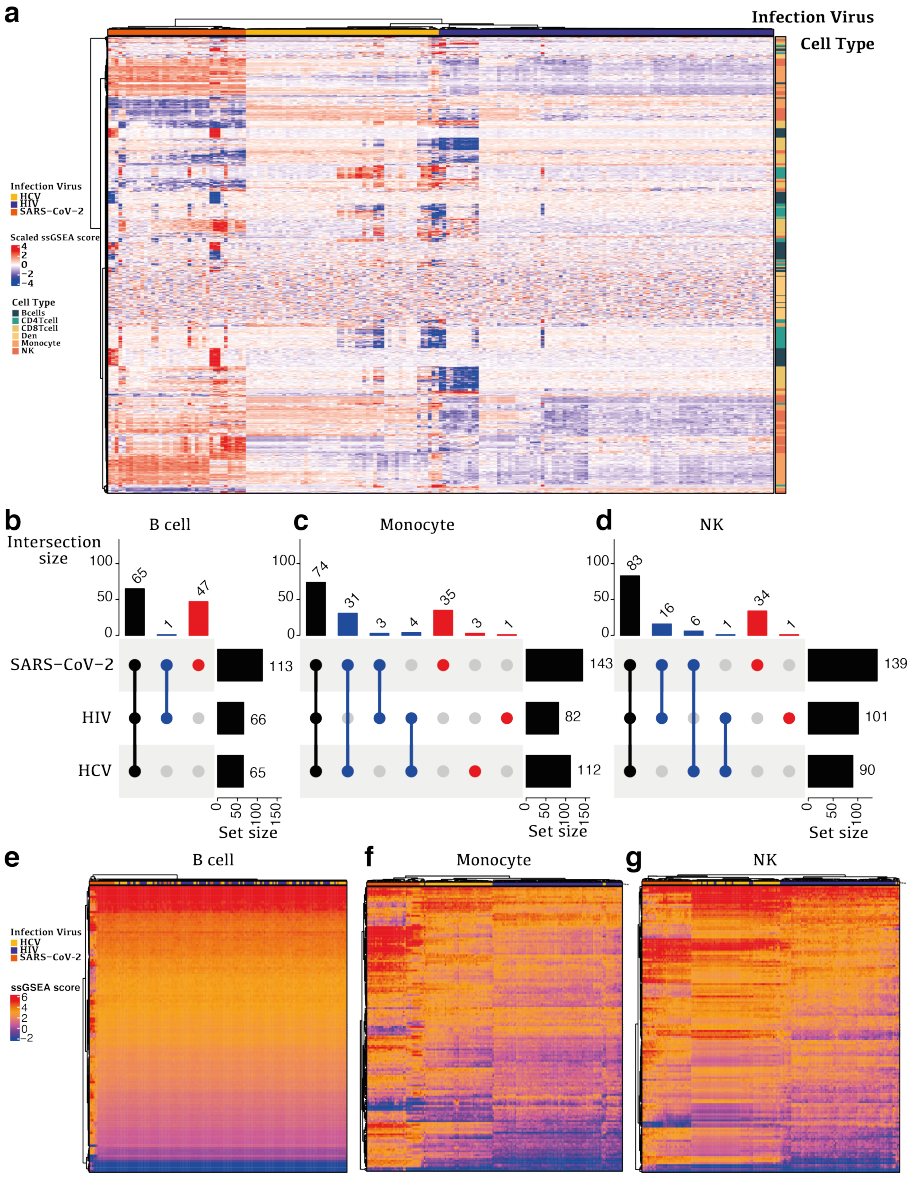

Figure 5: **Function enrichment of cell-specific GEP. a** Heatmap of enrichment scores for various cell types, including B cells, CD4 T cells, CD8 T cells, Dendritic cells, Monocytes, and NK cells within different virus-infection samples. The enrichment scores have been scaled by z-normalization. The top row annotation represents the virus types of the infection. The left column annotation represents the corresponding cell type of the enriched pathway. **b-d** Significantly enriched pathway upset plots for b) B cell, c) Monocyte, and d) NK cell in three kinds of virus-infection. **e-g** Heatmap of enrichment scores for **e** B cell, **f** Monocyte, and **g** NK cell.

immunodeficiency syndrome (AIDS). These three virus infections damage the host immune system but lead to different syndromes. Knowing the specific function differences in specific cells could help us in both the treatment and prevention of these infections.

Besides the differential expressed genes, we also investigated the functions of each cell type by incorporating cell-specific GEPs and ssGSEA. As the ssGSEA algorithm only needs the gene rank, which could be provided by our cell-specific GEPs, we could predict the activities of each function pathway for each sample. Considering function pathways that significantly ($p_adj < 0.05$) (in)activated in at least one sample, we found the samples that were infected by different viruses clustered (Pearson for distance, ward.D2 for cluster) together (Figure 5a). Comparing SARS-CoV-2 infected samples with the other

two virus-infected samples, we found that functional pathways were potential to activate then to inactivate. Also, the HIV-infected samples are similar to HBV-infected samples, showing the difference from the SARS-CoV-2 infection. Besides, subsets of samples within each virus-infected sample could also be identified, which might present the heterogeneous samples within the same virus infection.

Even the activities of the significant function pathways show differences among the three virus infections, the proportions of common significant enriched pathways were large in different cell types (Figure 5b, 5c, 5d). More significant enriched function pathways were observed in SARS-CoV-2 infected samples, compared to the other two virus-infection samples. In the B cells, HIV-infected samples share 99% of significantly enriched pathways with HBV-infected samples, while SARS-CoV-2 occupied more than 40% of the significantly enriched pathways privately (Figure 5b). These SARS-CoV-2 private enriched pathways contributed to the identification of the subset samples (Figure 5e).

Of note, monocytes and NK cells contributed to distinguishing these three kinds of virus-infected samples (Figure 5f, 5g). We noticed that the number of common enriched pathways in these two cell types are much larger than the numbers of mono-enriched or di-enriched pathways, indicating the activation differences, rather than functional differences, make the various of three virus-infection samples.

3 Discussion

We develop TAPE as a novel deep-leaning algorithm for digital tissue dissection. Key features distinguishing it from previous methods include: 1. highly accurate and sensitive deconvolution to capture the biologically significant changes in clinical data and 2. tissue-adaptive cell-type-specific gene expression profile prediction to identify potential gene expression differences at cell-type level. TAPE benefits from the architecture of autoencoder and the unique training method in the adaptive stage. The encoder-decoder architecture enables us to design an interpretable decoder to answer why encoder makes such predictions. More interestingly, the decoder is a natural cell-type-specific signature matrix which can be learned after the training stage and then adapted to the bulk data after the adaptive stage. Notice that the special training process of TAPE makes it fundamentally different from other methods, which only predict cell fractions or need large cohort bulk RNA-seq data to impute cell-type-specific GEPs. Another advantage of TAPE is the constant running time when deconvolving a large number of samples. Running on the popular graphic card, TAPE is much faster than traditional statistical methods and 3 times faster than the previous deep-learning method.

As highlighted before, TAPE can predict cell-type-specific GEPs tissue-adaptively. But admittedly, it can be improved further. Firstly, a normalized gene expression value has lost the original information about total reads count, which is hard for researchers to analyze DEGs perfectly. Secondly, although we have proved that TAPE could capture the biological significance of the highly expressed genes and highly differetially expressed genes, the fidelity of the insignificant genes is still unknown.

In summary, TAPE represents a widely applicable framework for deciphering heterogeneity of tissues at a cell-type level and also provides a practical training scheme for supervised autoencoder to perform domain adaptation. Considering it can be integrated with other tools seamlessly, we believe that TAPE will be helpful to investigate the connection between the single-cell data and the abundant bulk data.

4 Methods

4.1 Datasets and preprocessing

In this work, we used several public single-cell RNA-seq datasets, bulk RNA-seq datasets, and microarray datasets to perform our experiments. In the pseudo-bulk test, a single-cell dataset of mouse atlas from Tabular Muris[22] was used. This dataset consists of 20 organs and tissues with cell type labels provided by the authors. Only five tissues' (kidney, liver, lung, skin, thymus) data produced by the fluorescence activated cell sorting (FACS) method was used to perform the pseudo-bulk test.

In the experiments of real bulk data with ground truth, we used several real bulk datasets with the corresponding cell fractions. The first PBMC dataset SDY67 was created by *Zimmermann et al.*, but it was indirectly obtained from Scaden's training data with unknown fractions. The second PBMC dataset created by *Monaco et al.* could be downloaded from the GEO database with accession code GSE107011. The corresponding cell fractions data was provided as supplementary information of the original paper. More specifically, the unknown fraction was calculated by one minus sum of known proportions and cell

types of the same kind were added together to fit cell types in training data. For example, monocytes C, monocytes I, and monocytes NC are different kinds of monocytes, so their fractions will be added together as the total fraction of monocytes. The third PBMC dataset is created by *Newman et al.* Its expression data were downloaded from GEO with code GSE65133 and its cell fractions were provided on the webpage of CIBERSORT[13]. Next, the dataset we used to deconvolve human tissue with Alzheimer's Disease (AD) is from a project called Religious Orders Study and Memory and Aging Project (ROSMAP)[27]. This dataset consists of about 600 samples of RNA-seq data from AD patients, while 41 of them have cell-type proportion information measured by immunohistochemistry in another study[18]. The gene expression data was obtained from supplementary data of Scaden rather than the original program of ROSMAP to maintain consistency during the test. As for the single-cell datasets, 8k PBMC dataset from healthy donors was downloaded from 10X Genomics[32], mouse and human brain datasets were obtained from the GEO database with accession code GSE87544 and GSE67835 respectively[28, 40]. All of these datasets were preprocessed to generate cell-type labels using the same procedure in Scaden. Notably, if the training data is available in Scaden, like PBMC and mouse brain datasets, we just used the training data provided by the author of Scaden to assess performance

In the advanced analysis of real bulk data with clinical information, three different datasets were involved. Since the ROSMAP dataset has been introduced above, here we only describe the other two datasets. The first is COVID-19 PBMC dataset[19] from a longitude study of patients with COVID-19, this dataset has 39 RNA-seq data of PBMC constitutes of different stages (treatment stage, convalescence stage, and rehabilitation stage) and different types (mild, moderate, and serious) from 16 patients. The second is the COVID-19 islet dataset which is from a study of the SARS-CoV-2 infected islet. This dataset only has six samples which are divided into three groups: normal cultured group, infected group, and Remdesivir treated group. The single-cell dataset used as reference is from *Baron et al.*[35] (GEO accession code: GSE84133) which has 14 labeled cell types in pancreas tissue. Instead of using all the cells in the dataset, we selected only endocrine cells: alpha cell, beta-cell, delta cell, gamma cell, and epsilon cell to constitute the reference dataset.

In the final analysis of tissue-adaptive GEP, we introduced an HIV PBMC dataset from the GEO database with accession number GSE115449. This dataset has PBMC data collected from 91 HIV patients, half of them have developed BNab and the others do not have BNab. Furthermore, when we used ssGSEA to analyze cellular function changes in PBMC cross different viruses infection, we used an HCV infected bulk RNA-seq data of PBMC (GEO database, accession number: GSE119117). This dataset is also from a longitude study of patients. RNA-seq data is collected from individuals before, during, and after acute HCV infection. Acute HCV infection resulted in spontaneous viral resolution (n=6) or chronic infection (n=8). Four time points were examined per subject: i) Pre-infection baseline (Variable); ii) Early acute (2-9 weeks, mean 6 weeks); iii) Late acute (15 – 20 weeks, mean 17 weeks); and iv) Follow up (25-71 weeks, mean 52 weeks). Another virus-infected PBMC dataset is the COVID-19 PBMC dataset which has been mentioned before.

Note that, all the datasets involved in this study might use different ways to represent genes. To maintain the concordance, we processed all the different representations into gene names through BioMart[41].

4.2 The TAPE framework

4.2.1 Simulation of pseudo-bulk data from a single-cell dataset

Usually, deep learning models need a large amount of training data to optimize. So, it is crucial to generate pseudo-bulk data from a single-cell dataset to train the model.

Single-cell expression data with cell-type fractions are used to generate pseudo-bulk data. By definition, pseudo-bulk expression data is the sum of single-cell expression data from a subset of cells. So, to generate pseudo-bulk data, cells should be sampled with a given cell type proportions (ground truth) and total cell number like the stratified sampling.

The random cell-type fractions were first generated using the *randn()* function from *numpy.random* package[42] and then divided by the sum of random numbers to ensure the sum of each cell type proportion is equal to 1. Next, we multiply the total cell number with the generated cell fractions for each sample to acquire the exact sampling number for each cell type. After that, we use a stratified sampling method to sample cells of each cell type with the given number. Finally, the pseudo-bulk expression profile is created by summing the expression values of the randomly selected single-cell expression profiles for each sample.

Additionally, if users want to predict tissue-adaptive GEPs and investigate the relative gene expression value (output GEP value is between 0 and 1), users need to consider the data shift between different sequencing methods. For example, counts data from the 10X sequencing platform represents the real expression value while counts data from smart-seq need to be further normalized using a method like TPM or FPKM to show the real expression value. Here we provide a simple function *counts2FPKM* (or TPM) to transform raw counts to FPKM (or TPM). Due to the original information loss of the processed single-cell expression profile, we just normalized raw counts of a certain gene with its maximum transcripts length obtained from BioMart[41]. So, we recommend users prepare a suitable single-cell profile in advance to avoid information loss.

According to the previous study from Scaden[16], different sampling distributions and single-cell datasets with a heavy bias of different cell types do not affect deconvolution performance notably. Furthermore, we investigate the impact of varying total cell numbers; the results demonstrate that there is no significant difference when total cells number ranges from 100 to 1,000.

4.2.2 Problem Definition

To illustrate our model more clearly, it is necessary to define the problem in advance. All of the symbols defined in this section are consistent throughout the article. Intuitively, we expect gene expression profile from bulk RNA-seq would be a linear combination of each cells' GEP from single-cell RNA-seq. Furthermore, if cells belonging to one kind of cell type have the same gene expression pattern, we could use the signature gene expression pattern and the number of cells for each type to reconstruct the GEP of a bulk RNA-seq data. So, given the number of k cell types, m genes, and n samples in bulk RNA-seq data, an ideal mathematic model could be defined as:

$$\mathbf{X} \cdot \mathbf{S} = \mathbf{B}, \tag{1}$$

where $\mathbf{B}$ is an $n \times m$ matrix representing GEPs of bulk RNA-seq; $\mathbf{S}$ is a $k \times m$ signature matrix; $\mathbf{X}$ is an $n \times k$ matrix representing cell-type fractions in each sample.

4.2.3 Model set-up

Given the well-defined problem, we just need to modify the equation to accommodate deep learning:

$$\begin{aligned} f_\phi(\mathbf{B}) &= \tilde{\mathbf{X}}, \\ f_\psi(\tilde{\mathbf{X}}) &= \tilde{\mathbf{X}} \cdot \mathbf{S}, \\ f_\psi(f_\phi(\mathbf{B})) &= \tilde{\mathbf{B}}. \end{aligned} \tag{2}$$

Here, f_ϕ and f_ψ represent two coordinated deep neural network, symbols with tilde like $\tilde{\mathbf{B}}$ refers to the output of the model, and $\mathbf{S}$ refers to the explicit matrix form of f_ψ. Usually, f_ϕ and f_ψ are called encoder and decoder respectively in the classical architecture of AE. f_ϕ is a regression model which is responsible for mapping the high dimensional bulk gene expression data to a low dimensional representation of cell compositions. In contrast, f_ψ is the inverse function of f_ϕ which is expected to reconstruct bulk data based on the cell fractions. Obviously, f_ψ functions like the signature matrix discussed above. Therefore, we want to make it have an explicit matrix-form to enforce the interpretability of f_ϕ. Because an ideal f_ϕ could also be represented in matrix-form due to the inverse relation in math. To achieve the progress in interpretability of deep model, f_ψ was designed without activation layers or bias, it is only the regularized value of dot product of five weight matrices, thus the signature matrix is visible in the deep model:

$$f_\psi = \mathbf{S} = \mathrm{ReLU}(\mathbf{W_1} \cdot \mathbf{W_2} \cdot \mathbf{W_3} \cdot \mathbf{W_4} \cdot \mathbf{W_5}), \tag{3}$$

where $\mathrm{ReLU}(x) = (x)^+ = \max(0, x)$. The reason to design such an equation to represent $\mathbf{S}$ rather than a single matrix is that more parameters could enable the model to learn a good signature matrix more quickly and easily and the $\mathrm{ReLU}(\cdot)$ function is used to ensure the biological meaning of the signature matrix.

We need to stress that, it seems that our model assumes that cell proportions could be inferred from the bulk data directly through the function f_ϕ without the signature matrix. However, if we consider f_ψ, we will find that parameters of f_ϕ is affected by f_ψ during optimization. Just like other statistical methods computing the pseudo-inverse of the signature matrix in the fitting process, we also use the

inverse relationship between f_ϕ and f_ψ in the training stage. Therefore, compared to the previous machine learning methods using a single function to predict fractions without regularization from the signature matrix, this architecture makes sense better

4.2.4 Input Data preprocessing

Although the input datasets varied between platforms and protocols, we utilize the same processing approach to prepare them for deep-learning models and avoid the dimensionality curse. As for the bulk data (real or simulated), it was first transformed to the Log_2 space with a pre-added one to avoid null value. Next, to maintain the meaningful signature matrix, we decide to use $MinMaxScaler()$ function provided by scikit-learn[43] to scale data into a range between 0 and 1. This function is described below:

$$\mathbf{B}_{i,j} = \frac{\mathbf{B}_{i,j} - \min(\mathbf{B}_i)}{\max(\mathbf{B}_i) - \min(\mathbf{B}_i)}, \quad j = 1, 2, 3, \ldots, m. \tag{4}$$

4.2.5 Training method

As previously stated, there are two stages of training in TAPE. The first is the training stage, in this stage, we use mean absolute error (MAE) between prediction and ground truth to optimize the parameters of encoder and MAE between reconstructed input and input to optimize both the decoder and the encoder. The loss functions are defined as:

$$\begin{aligned} \mathrm{MAE}(\mathbf{X}, \tilde{\mathbf{X}}) &= \frac{\sum_{i,j} |\mathbf{X}_{i,j} - \tilde{\mathbf{X}}_{i,j}|}{n \times k}, \\ \mathrm{MAE}(\mathbf{B}, \tilde{\mathbf{B}}) &= \frac{\sum_{i,j} |\mathbf{B}_{i,j} - \tilde{\mathbf{B}}_{i,j}|}{n \times k}. \end{aligned} \tag{5}$$

Usually, we found that $\mathrm{MAE}(\mathbf{X}, \tilde{\mathbf{X}})$ is stable after 5,000 iterations with batch size 128, and we just stopped training to avoid overfitting.

In the adaptive stage, we aimed to train the parameters to adapt new data rather than predicting cell fractions with the same parameters in all situations. To achieve this goal, we designed a greedily iterative optimizing method in a new manner: *step 1.* optimize the decoder with loss function $\mathrm{MAE}(\mathbf{B}, \tilde{\mathbf{B}}) + \mathrm{MAE}(\tilde{\mathbf{S}}, \tilde{\mathbf{S}}_\mathbf{0})$ until $\mathrm{MAE}(\mathbf{B}, \tilde{\mathbf{B}})$ did not decrease; *step 2.* optimize the encoder with loss function $\mathrm{MAE}(\mathbf{B}, \tilde{\mathbf{B}}) + \mathrm{MAE}(\tilde{\mathbf{X}}, \tilde{\mathbf{X}}_\mathbf{0})$ until $\mathrm{MAE}(\mathbf{B}, \tilde{\mathbf{B}})$ did not decrease. Usually, repeating step 1 and step 2 several times would make the parameters of TAPE adapt to the new data. In our practice, after the adaptive stage, the prediction of cell fractions will improve a little and it always outputs an adaptive signature matrix.

4.2.6 Predict tissue-adaptive cell-type-specific GEP in different modes

Generally, there are two different ways to analyze cell-type-specific GEP: 1. The "overall" mode: using all the samples at once to capture an overall cell-type-specific GEP in a certain condition. 2. The "high-resolution" mode: predicting all the samples one by one to maintain the differences between each sample. Certainly, the latter will consume more time than the first one. The choice of different modes mainly depends on users' demands. For example, if users want to discover the differentially expressed genes at a cell-type level, they should choose to predict GEPs in the "high-resolution" mode, thus they could calculate the p-value. On the other hand, if users only need to investigate the highly expressed genes in different cell types or have an overall look at the cell-type level, they should choose to predict GEPs in the "overall" mode.

It is worth noting that, the value of GEP predicted by TAPE is between 0 and 1 which represents the relative expression value among a single sample. This limitation is caused by the *Min-Max* scaling method when we preprocess the input data. Using this GEP may encounter some problems when users need to analyze the fold change of a certain gene due to the information loss induced by the nonlinear scaling function.

4.2.7 Architecture and hyperparameters

TAPE's encoder and decoder are both made up of five fully connected layers with the same weight size in the corresponding position. For example, the first layer of the encoder and the last layer of the decoder each have 512 nodes. More specifically, the number of nodes in each encoder layer is 512, 256, 128, 64, and the number of cell types in sequential order. Before the first four fully connected layers, each has a dropout function with a probability of 0.5; after each layer, each has a nonlinear activation function CELU$(\cdot)$, defined as $\mathrm{CELU}(x) = \max(0, x) + \min(0, e^{x-1})$. Decoder, on the other hand, does not contain any bias in the fully connected layers or nonlinear functions except the ReLU$(\cdot)$ function, as we mentioned before. During the training stage, we use Adam with a learning rate $1 times 10^{-4}$ to optimize parameters. Other parameters of Adam are set as default in PyTorch. We train the network for 5,000 iterations with batch size 128. These training hyperparameters are succeeded from Scaden. While in the adaptive stage, we use Adam with the same learning rate 1×10^{-4} to fine-tune the parameters on the new data. We train both the encoder and the decoder for 300 steps within each iteration. The max iteration number is flexible for users and we recommend users set it at least 2 to make it output a well-adapted signature matrix.

4.3 Performance evaluation

Within the main text above, we combined mean absolute error (MAE) with Lin's concordance correlation coefficient (CCC)[23] to evaluate different algorithms' performance because only one metric is hard to assess performance reasonably in all situations. For instance, if there are only two kinds of cell type in tissue and one type's fraction range from 80%-90% in the ground truth, if the model predicts this cell type fraction is 100%, then the CCC value will perform well but the MAE is not so good. So, to avoid the situation of discarding fractions of minor cell types, it is necessary to combine MAE with CCC. Generally, a higher CCC value and a lower MAE suggest a better deconvolution performance. These metrics are defined as follows:

$$\begin{aligned} \mathrm{MAE}(\mathbf{X}, \tilde{\mathbf{X}}) &= \frac{\sum_{i,j} |\mathbf{X}_{i,j} - \tilde{\mathbf{X}}_{i,j}|}{n \times k}, \\ \mathrm{CCC}(x, \tilde{x}) &= \frac{2 \times cov(x, \tilde{x})}{\sigma_x^2 + \sigma_{\tilde{x}}^2 + (\mu_x - \mu_{\tilde{x}})}, \end{aligned} \tag{6}$$

where $cov(x, \tilde{x})$ stands for the covariance between these two vectors. Notably, these two metrics are applied to all data points of the predicted matrix $\tilde{X}$ and the ground truth matrix X. More specifically, as for the CCC value, we reshape the matrix into a vector and then calculate the total CCC between two vectors. This calculation pattern usually results in a higher CCC value than computing the average CCC value for each cell type.

4.4 Software comparison and settings

To evaluate the performance of TAPE compared with other methods, we selected several representative methods for comparison. Except for Scaden, other methods were tested following the instruction and tutorials provided by each package.

For deconvolution performance on the pseudo-bulk and real bulk data with ground truth, we benchmark Scaden, RNAsieve, CIBERSORTx, and DWLS[11, 14–16]. We will describe the details of the benchmarking procedure below.

For Scaden, since the package provided by the author did not have a clear API document, we implemented a PyTorch-based Scaden. The training hyperparameters were set following the instruction of the original article and source code. Though we tried our best to make it the same as the original Scaden, it still had some different behaviors, for example, loss plot and deconvolution performance on the SDY67 dataset were different from reported data. These differences were probably caused by the different deep learning backends (Keras or PyTorch). In general, since the differences were not huge, the implementation of Scaden is acceptable.

For RNA-Sieve, we used the python package provided by the author and used the default settings to deconvolve data. We first validated its performance on pseudo-bulk data and the original data provided by the authors. The results showed that it could perform well on the simulated data but it could not

reproduce the same results as they reported on Newman's dataset and Monaco's dataset. There may exist some problems in the python-based RNA-Sieve.

For CIBERSORTx (CSx), we used the web-based application to test. As for the pseudo-bulk test, we did not perform the 5-fold evaluation on CSx because it consumed too much time to deconvolve. We only used the first batch (800 samples) of simulated data to deconvolve. We first used a single-cell profile to generate a signature matrix and then we used the corresponding bulk data to deconvolve with S mode batch-correction. Other settings were default.

For DWLS, we used the core functions and packages written in R programming language with the default settings to generate signature matrices and therefore deconvolving the targeted pseudo-bulk data and the real ones. To guarantee the rationality of our implementation, we carefully followed the example of the intestine stem cell provided by the manual of DWLS. Since the deconvolution function provided by DWLS only deconvolves one sample at a time, a for-loop is brought in because we need to deal with some large samples. What's more, we have run into several overflows when carrying out DWLS, thus the logarithmic operation on bulk data ensuring that the maximum gene expression is in the teens.

It should be pointed out that, for all statistical methods (RNA-sieve, CSx, and DWLS), all PBMC datasets were deconvolved using a signature matrix generated from PBMC data8k dataset, reference of mouse brain dataset is generated from *Chen et al.*[40], and signature matrix of human brain dataset is generated from *Darmanis et al.*[28]

5 Data availability

All the datasets we used are listed in the Method part. Only the ROSMAP human brain dataset is not public, researchers need to download it from Synapse (ID: syn3219045) with a request. For convenience, we listed these datasets on the webpage: https://sctape.readthedocs.io/datasets/.

6 Code availability

The open source implementation of TAPE is available at https://github.com/poseidonchan/TAPE, and the experiments conducted to produce the main results of this article are also stored in this repository. The documentation of TAPE is published at https://sctape.readthedocs.io/.

7 Acknowledgements

References

[1] Conesa, A. *et al.* A survey of best practices for rna-seq data analysis. *Genome Biol* **17**, 13 (2016). URL https://www.ncbi.nlm.nih.gov/pubmed/26813401.

[2] Kukurba, K. R. & Montgomery, S. B. Rna sequencing and analysis. *Cold Spring Harb Protoc* **2015**, 951–69 (2015). URL https://www.ncbi.nlm.nih.gov/pubmed/25870306.

[3] Saliba, A. E., Westermann, A. J., Gorski, S. A. & Vogel, J. Single-cell rna-seq: advances and future challenges. *Nucleic Acids Res* **42**, 8845–60 (2014). URL https://www.ncbi.nlm.nih.gov/pubmed/25053837.

[4] Han, W. *et al.* Self-supervised contrastive learning for integrative single cell rna-seq data analysis. *bioRxiv* (2021). URL https://www.biorxiv.org/content/early/2021/07/27/2021.07.26.453730. https://www.biorxiv.org/content/early/2021/07/27/2021.07.26.453730.full.pdf.

[5] Nguyen, Q. H. *et al.* Single-cell rna-seq of human induced pluripotent stem cells reveals cellular heterogeneity and cell state transitions between subpopulations. *Genome Res* **28**, 1053–1066 (2018). URL https://www.ncbi.nlm.nih.gov/pubmed/29752298.

[6] Montoro, D. T. *et al.* A revised airway epithelial hierarchy includes cftr-expressing ionocytes. *Nature* **560**, 319–324 (2018). URL https://www.ncbi.nlm.nih.gov/pubmed/30069044.

[7] Shen-Orr, S. S. *et al.* Cell type-specific gene expression differences in complex tissues. *Nat Methods* **7**, 287–9 (2010). URL https://www.ncbi.nlm.nih.gov/pubmed/20208531.

[8] Zhong, Y., Wan, Y. W., Pang, K., Chow, L. M. & Liu, Z. Digital sorting of complex tissues for cell type-specific gene expression profiles. *BMC Bioinformatics* **14**, 89 (2013). URL https://www.ncbi.nlm.nih.gov/pubmed/23497278.

[9] Stegle, O., Teichmann, S. A. & Marioni, J. C. Computational and analytical challenges in single-cell transcriptomics. *Nat Rev Genet* **16**, 133–45 (2015). URL https://www.ncbi.nlm.nih.gov/pubmed/25628217.

[10] Wang, X., Park, J., Susztak, K., Zhang, N. R. & Li, M. Bulk tissue cell type deconvolution with multi-subject single-cell expression reference. *Nat Commun* **10**, 380 (2019). URL https://www.ncbi.nlm.nih.gov/pubmed/30670690https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6342984/pdf/41467_2018_Article_8023.pdf.

[11] Tsoucas, D. *et al.* Accurate estimation of cell-type composition from gene expression data. *Nat Commun* **10**, 2975 (2019). URL https://www.ncbi.nlm.nih.gov/pubmed/31278265https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6611906/pdf/41467_2019_Article_10802.pdf.

[12] Jew, B. *et al.* Accurate estimation of cell composition in bulk expression through robust integration of single-cell information. *Nat Commun* **11**, 1971 (2020). URL https://www.ncbi.nlm.nih.gov/pubmed/32332754https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7181686/pdf/41467_2020_Article_15816.pdf.

[13] Newman, A. M. *et al.* Robust enumeration of cell subsets from tissue expression profiles. *Nat Methods* **12**, 453–7 (2015). URL https://www.ncbi.nlm.nih.gov/pubmed/25822800.

[14] Newman, A. M. *et al.* Determining cell type abundance and expression from bulk tissues with digital cytometry. *Nat Biotechnol* **37**, 773–782 (2019). URL https://www.ncbi.nlm.nih.gov/pubmed/31061481.

[15] Erdmann-Pham, D. D., Fischer, J., Hong, J. & Song, Y. S. A likelihood-based deconvolution of bulk gene expression data using single-cell references. *Genome Res* (2021). URL https://www.ncbi.nlm.nih.gov/pubmed/34301624.

[16] Menden, K. *et al.* Deep learning-based cell composition analysis from tissue expression profiles. *Sci Adv* **6**, eaba2619 (2020). URL https://www.ncbi.nlm.nih.gov/pubmed/32832661.

[17] Hinton, G. E. & Salakhutdinov, R. R. Reducing the dimensionality of data with neural networks. *Science* **313**, 504–7 (2006). URL https://www.ncbi.nlm.nih.gov/pubmed/16873662https://science.sciencemag.org/content/313/5786/504.long.

[18] Patrick, E. *et al.* Deconvolving the contributions of cell-type heterogeneity on cortical gene expression. *PLoS Comput Biol* **16**, e1008120 (2020). URL https://www.ncbi.nlm.nih.gov/pubmed/32804935.

[19] Zheng, H. Y. *et al.* Longitudinal transcriptome analyses show robust t cell immunity during recovery from covid-19. *Signal Transduct Target Ther* **5**, 294 (2020). URL https://www.ncbi.nlm.nih.gov/pubmed/33361761.

[20] Muller, J. A. *et al.* Sars-cov-2 infects and replicates in cells of the human endocrine and exocrine pancreas. *Nat Metab* **3**, 149–165 (2021). URL https://www.ncbi.nlm.nih.gov/pubmed/33536639.

[21] Dávila-Collado, R., Jarquín-Durán, O., Solís-Vallejo, A., Nguyen, M. A. & Espinoza, J. L. Elevated monocyte to lymphocyte ratio and increased mortality among patients with chronic kidney disease hospitalized for covid-19. *Journal of personalized medicine* **11**, 224 (2021).

[22] Tabula Muris, C. *et al.* Single-cell transcriptomics of 20 mouse organs creates a tabula muris. *Nature* **562**, 367–372 (2018). URL https://www.ncbi.nlm.nih.gov/pubmed/30283141.

[23] Lin, L. I. K. A concordance correlation coefficient to evaluate reproducibility. *Biometrics* **45** (1989).

[24] Zimmermann, M. T. *et al.* System-wide associations between dna-methylation, gene expression, and humoral immune response to influenza vaccination. *PLoS One* **11**, e0152034 (2016). URL https://www.ncbi.nlm.nih.gov/pubmed/27031986.

[25] Monaco, G. *et al.* Rna-seq signatures normalized by mrna abundance allow absolute deconvolution of human immune cell types. *Cell Rep* **26**, 1627–1640 e7 (2019). URL https://www.ncbi.nlm.nih.gov/pubmed/30726743.

[26] Bennett, D. A. *et al.* Religious orders study and rush memory and aging project. *J Alzheimers Dis* **64**, S161–S189 (2018). URL https://www.ncbi.nlm.nih.gov/pubmed/29865057https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6380522/pdf/nihms-1009988.pdf.

[27] De Jager, P. L. *et al.* A multi-omic atlas of the human frontal cortex for aging and alzheimer's disease research. *Sci Data* **5**, 180142 (2018). URL https://www.ncbi.nlm.nih.gov/pubmed/30084846.

[28] Darmanis, S. *et al.* A survey of human brain transcriptome diversity at the single cell level. *Proc Natl Acad Sci U S A* **112**, 7285–90 (2015). URL https://www.ncbi.nlm.nih.gov/pubmed/26060301.

[29] Braak, H. & Braak, E. Neuropathological stageing of alzheimer-related changes. *Acta Neuropathol* **82**, 239–59 (1991). URL https://www.ncbi.nlm.nih.gov/pubmed/1759558.

[30] Hansen, D. V., Hanson, J. E. & Sheng, M. Microglia in alzheimer's disease. *J Cell Biol* **217**, 459–472 (2018). URL https://www.ncbi.nlm.nih.gov/pubmed/29196460.

[31] Navarro, V. *et al.* Microglia in alzheimer's disease: Activated, dysfunctional or degenerative. *Front Aging Neurosci* **10**, 140 (2018). URL https://www.ncbi.nlm.nih.gov/pubmed/29867449.

[32] 8k pbmcs from a healthy donor, single cell gene expression dataset by cell ranger 2.1.0. *10X Genomics* (2017). URL https://www.10xgenomics.com/resources/datasets/8-k-pbm-cs-from-a-healthy-donor-2-standard-2-1-0.

[33] Lissoni, P. *et al.* Evidence of abnormally low lymphocyte-to-monocyte ratio in covid-19-induced severe acute respiratory syndrome. *J Immuno Allerg* **1**, 1–6 (2020).

[34] Yang, A. P., Liu, J. P., Tao, W. Q. & Li, H. M. The diagnostic and predictive role of nlr, d-nlr and plr in covid-19 patients. *Int Immunopharmacol* **84**, 106504 (2020). URL https://www.ncbi.nlm.nih.gov/pubmed/32304994.

[35] Baron, M. *et al.* A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Syst* **3**, 346–360 e4 (2016). URL https://www.ncbi.nlm.nih.gov/pubmed/27667365.

[36] Sun, X. *et al.* Association of neurogranin gene expression with alzheimer's disease pathology in the perirhinal cortex. *Alzheimers Dement (N Y)* **7**, e12162 (2021). URL https://www.ncbi.nlm.nih.gov/pubmed/33860070.

[37] Bradley, T. *et al.* Rab11fip5 expression and altered natural killer cell function are associated with induction of hiv broadly neutralizing antibody responses. *Cell* **175**, 387–399 e17 (2018). URL https://www.ncbi.nlm.nih.gov/pubmed/30270043.

[38] 6k pbmcs from a healthy donor single cell gene expression dataset by cell ranger 1.1.0. *10X Genomics* (2016). URL https://www.10xgenomics.com/resources/datasets/6-k-pbm-cs-from-a-healthy-donor-1-standard-1-1-0.

[39] 10k pbmcs from a healthy donor (v3 chemistry), single cell gene expression dataset by cell ranger 3.0.0. *10X Genomics* (2018). URL https://www.10xgenomics.com/resources/datasets/10-k-pbm-cs-from-a-healthy-donor-v-3-chemistry-3-standard-3-0-0.

[40] Chen, R., Wu, X., Jiang, L. & Zhang, Y. Single-cell rna-seq reveals hypothalamic cell diversity. *Cell Rep* **18**, 3227–3241 (2017). URL https://www.ncbi.nlm.nih.gov/pubmed/28355573.

[41] Smedley, D. *et al.* The biomart community portal: an innovative alternative to large, centralized data repositories. *Nucleic Acids Res* **43**, W589–98 (2015). URL https://www.ncbi.nlm.nih.gov/pubmed/25897122.

[42] Harris, C. R. *et al.* Array programming with numpy. *Nature* **585**, 357–362 (2020). URL https://www.ncbi.nlm.nih.gov/pubmed/32939066.

[43] Pedregosa, F. *et al.* Scikit-learn: Machine learning in python. *the Journal of machine Learning research* **12**, 2825–2830 (2011).