Machine Learning
https://doi.org/10.1007/s10994-021-06043-1

Embedding to reference t-SNE space addresses batch effects in single-cell classification

Pavlin G. Poličar[1] · **Martin Stražar**[1] · **Blaž Zupan**[1,2]

Received: 14 February 2020 / Revised: 26 June 2021 / Accepted: 26 July 2021

Abstract
Dimensionality reduction techniques, such as t-SNE, can construct informative visualizations of high-dimensional data. When jointly visualising multiple data sets, a straightforward application of these methods often fails; instead of revealing underlying classes, the resulting visualizations expose dataset-specific clusters. To circumvent these batch effects, we propose an embedding procedure that uses a t-SNE visualization constructed on a reference data set as a scaffold for embedding new data points. Each data instance from a new, unseen, secondary data is embedded independently and does not change the reference embedding. This prevents any interactions between instances in the secondary data and implicitly mitigates batch effects. We demonstrate the utility of this approach by analyzing six recently published single-cell gene expression data sets with up to tens of thousands of cells and thousands of genes. The batch effects in our studies are particularly strong as the data comes from different institutions using different experimental protocols. The visualizations constructed by our proposed approach are clear of batch effects, and the cells from secondary data sets correctly co-cluster with cells of the same type from the primary data. We also show the predictive power of our simple, visual classification approach in t-SNE space matches the accuracy of specialized machine learning techniques that consider the entire compendium of features that profile single cells.

Keywords Batch effects · Embedding · t-SNE · Visualization · Single-cell transcriptomics · Data integration · Domain adaptation.

Editors: Petra Kralj Novak, Tomislav Šmuc.

✉ Pavlin G. Poličar
pavlin.policar@fri.uni-lj.si

Martin Stražar
martin.strazar@fri.uni-lj.si

Blaž Zupan
blaz.zupan@fri.uni-lj.si

[1] Faculty of Computer and Information Science, University of Ljubljana, Večna pot 113, Ljubljana, Slovenia

[2] Baylor College of Medicine, Houston, TX 77030, USA

Published online: 24 August 2021

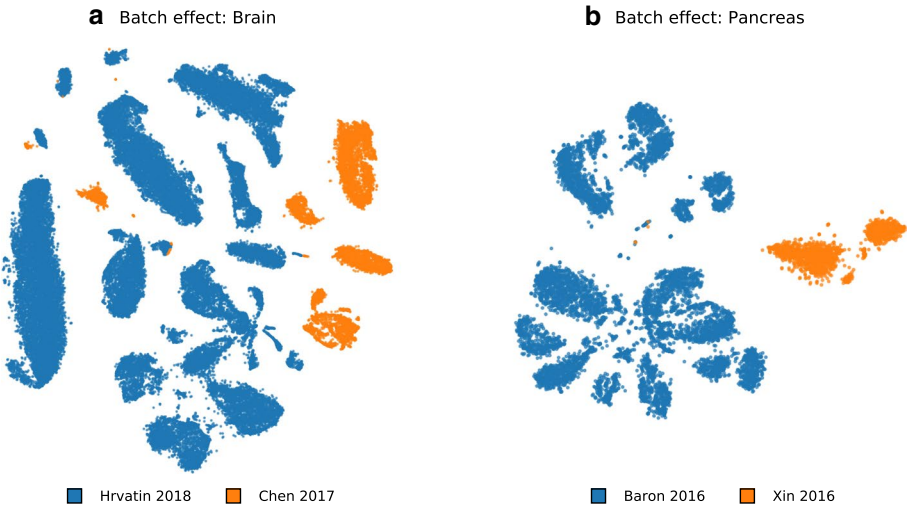

Fig. 1 Batch effects are a driving factor of variation between the data sets. We depict a t-SNE visualization of two pairs of data sets. In each pair, the data sets share cell types, so we would expect cells from the reference data (blue) to mix with the cells in a secondary data sets (orange). Instead, t-SNE clusters data according to the data source

1 Introduction

Two-dimensional embeddings and their visualizations may assist in the analysis and interpretation of high-dimensional data. Intuitively, two data instances should be co-located in the resulting visualization if their multi-dimensional profiles are similar. For this task, non-linear embedding techniques such as t-distributed stochastic neighbor embedding (t-SNE) (van der Maaten & Hinton, 2008) or uniform manifold approximation and projection (McInnes & Healy, 2018) have recently complemented traditional data transformation and embedding approaches such as principal component analysis (PCA) (Wold et al., 1987) and multi-dimensional scaling (Cox & Cox, 2008). While useful for visualizing data from a single coherent source, these methods may encounter problems with multiple data sources. Here, when performing dimensionality reduction on a merged data set, the resulting visualizations would typically reveal source-specific clusters instead of grouping data instances of the same class, regardless of data sources. This source-specific confounding is often referred to as *domain shift* (Gopalan et al., 2011), *covariate shift* (Bickel et al., 2009) or *data set shift* (Quionero-Candela et al., 2009). In bioinformatics, the domain-specific differences are more commonly referred to as *batch effects* (Butler et al., 2018; Haghverdi et al., 2018; Stuart et al., 2019).

Massive, multi-variate biological data sets often suffer from these source-specific biases. The focus of this work is single-cell genomics, a domain that was selected due to high biomedical relevance and abundance of recently published data. Single-cell RNA sequencing (scRNA-seq) data sets are the result of isolating RNA molecules from individual cells, which serve as an estimate of the expression of cell's genes. The studies can exceed thousands of cells and tens of thousands of genes, and typically start with cell type analysis. Here, it is expected that cells of the same type would cluster together in two-dimensional data visualization (Wolf et al., 2018). For instance, Fig. 1a shows t-SNE embedded data

from mouse brain cells originating from the visual cortex (Hrvatin et al., 2018) and the hypothalamus (Chen et al., 2017). The figure reveals distinct clusters but also separates the data from the two brain regions. These two regions share the same cell types and—contrary to the depiction in Fig. 1a—we would expect the data points from the two studies to overlap. Batch effects similarly prohibit the utility of t-SNE in the exploration of pancreatic cells in Fig. 1b, which renders the data from a pancreatic cell atlas (Baron et al., 2016) and similarly-typed cells from diabetic patients (Xin et al., 2016). Just like with data from brain cells, pancreatic cells cluster primarily by data source, again resulting in a visualization driven by batch effects.

Current solutions to embedding the data from various data sources address the batch effect problems up-front. The data is typically preprocessed and transformed such that the batch effects are explicitly removed. Recently proposed procedures for batch effect removal include canonical correlation analysis (Butler et al., 2018) and mutual nearest-neighbors (Haghverdi et al., 2018; Stuart et al., 2019). In these works, batch effects are deemed removed when cells from different sources exhibit good mixing in a t-SNE visualization. The elimination of batch effects may require aggressive data preprocessing which may blur the boundaries between cell types. Another problem is also the inclusion of any new data, for which the entire analysis pipeline must be rerun, usually resulting in a different embedding layout and clusters that have little resemblance to original visualization and thus require reinterpretation.

We propose a direct solution of rendering t-SNE visualizations to address batch effects. Our approach treats one of the data sets as a *reference* and embeds the cells from another, *secondary data set* to a reference-defined low-dimensional space. We construct a t-SNE embedding using the reference data set, which is then used as a scaffold to embed the secondary data. The key idea underpinning our approach is that secondary data points are embedded independently of one another.

Independent embedding of each secondary datum causes the clustering landscape to depend only on the reference scaffold, thus removing data source-driven variation. In other words, when including new data, the scaffold inferred from the reference data set is kept unchanged and defines a "gravitational field", independently driving the embedding of each new instance. For example, in Fig. 2, the cells from the visual cortex define the scaffold (Fig. 2a) into which we embed the cells from the hypothalamus (Fig. 2b). Unlike in their joint t-SNE visualization (Fig. 1a), the hypothalamic cells are dispersed across the entire embedding space and their cell type correctly matches the prevailing type in reference clusters.

The proposed solution implements a mapping of new data into an existing t-SNE visualization. While the utility of such an algorithm was already hinted at in recent publication (Kobak & Berens, 2019), we here provide its practical and theoretically-grounded implementation. Considering the abundance of recent publications on batch effect removal, we present surprising evidence that a computationally more direct and principled embedding procedure solves the batch effects problem when constructing interpretable visualizations from different data sources.

Our contributions are twofold:

1. We introduce a theoretically-grounded extension of the t-SNE visualization algorithm that supports embedding new data points into existing reference visualizations. Our extension is readily incorporated into existing approximation schemes, enabling its applications to large data sets. We show that optimization using the default t-SNE

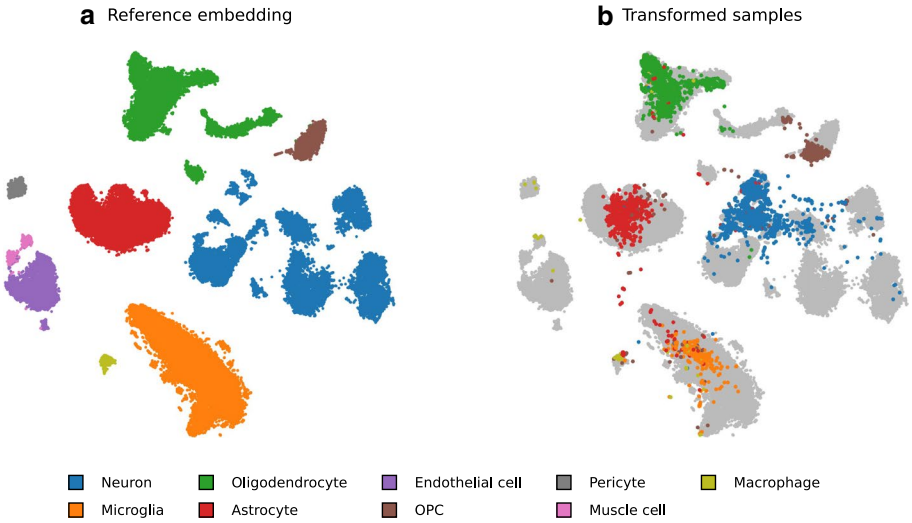

Fig. 2 A two-dimensional embedding of a reference containing brain cells (**a**) and the corresponding mapping of secondary data containing hypothalamic cells (**b**). The majority of hypothalamic cells were mapped to their corresponding reference cluster. For instance, astrocyte cells marked with red on the right were mapped to an oval cluster of same-typed cells denoted with the same color in the visualization on the left

parameters is highly unstable and proposes parameter values leading to stable convergence.
2. We show that the proposed t-SNE extensions can mitigate batch effects in the data sets and demonstrate this feature in treating single-cell gene expression data.

2 Related work

Batch effects are systematic biases between biological data sets caused by technical factors in the data collection and preparation process. It has been well documented that even small differences in the experimental setup of cell-dissociation, handling protocols, library-preparation technologies, or sequencing platforms can significantly affect the resulting gene-expression measurements (Tung et al., 2017; Hicks et al., 2018). When performing downstream comparative analyses, batch effects may confound real biological variability and introduce spurious correlations, leading to misleading conclusions.

Due to their severity, numerous computational approaches have been proposed to directly remove batch effects when performing joint analysis on two or more data sets. Batch effect removal is typically performed as a preprocessing step. Existing approaches involve either modifying the original data matrix or finding a joint lower-dimensional space, where batch effects are removed. Current methods broadly fall into two categories:

1. Mutual nearest neighbor-based approaches aim to identify matching populations of cells across the data sets, using them to either find and correct the data sets (Haghverdi et al., 2018) or directly construct a batch-corrected k-nearest neighbor graph used in downstream analyses (Park et al., 2018).

2. Embedding multiple data sets into a joint lower-dimensional space, where batch effects are removed. Some of these approaches opt for linear dimensionality-reduction methods such as PCA (Korsunsky et al., 2019) or MultiCCA (Butler et al., 2018), while others employ non-linear techniques from deep learning (Li et al., 2020; Lopez et al., 2018). Still, other approaches use a combination of the two (Stuart et al., 2019; Hie et al., 2019). Note that these approaches bear similarity with transfer learning (Weiss et al., 2016), which has also been used in domain adaptation (Liu et al., 2019).

Besides computational techniques, approaches for the removal of batch effects can also use domain knowledge. For example, in the analysis of single-cell gene expression data, these approaches act on a subset of representative marker genes for a specific cell type. Instead of considering the entire gene-expression profile, which may be noisy and affected by batch effects, the idea is to profile the cells with a handful of genes that can collectively determine the cell type. One such procedure is scMap-Cluster, a consensus-based k-nearest neighbor method tailored explicitly to scRNA-seq gene-expression data (Kiselev et al., 2018). scMap-Cluster uses three correlation-based distance measures and uses a voting scheme to perform classification. To identify novel cell types, scMap-Cluster heuristically determines a distance threshold.

Our approach to batch effect removal falls into the second category, as we lose the batch effects through dimensionality reduction. Alongside scMap-Cluster, we also benefit from a standard single-cell data preprocessing pipeline that profiles the cells with representative genes. Unlike other batch effect removal procedures, the primary purpose of our approach is not classification but the visualization of the various cell-types. If required, we can apply a k-nearest neighbor classifier to the resulting visualizations to obtain accuracy estimates and compare our approach to other classification methods. However, the classification aspect of our approach is secondary: the primary purpose of t-SNE is to aid in scientists in exploratory data analysis and help them better understand the underlying data landscape.

3 Methods

We describe an end-to-end pipeline that uses fixed t-SNE coordinates as a scaffold for embedding new (secondary) data, enabling joint visualization of multiple data sources while mitigating batch effects. Our proposed approach starts by using t-SNE to embed a reference data set, with the aim of constructing a two-dimensional visualization to facilitate interpretation and cluster classification. Then, the placement of each new sample is optimized independently via the t-SNE loss function. Independent treatment of each data instance from a secondary data set disregards any interactions present in that data set, and prevents the formation of clusters that would be specific to the secondary data. Below, we start with a summary of t-SNE and its extensions (Sect. 3.1), introducing the relevant notation, upon which we base our secondary data embedding approach (Sect. 3.2).

3.1 Data embedding by t-SNE and its extensions

Local, non-linear dimensionality reduction by t-SNE is performed as follows. Given a multi-dimensional data set $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\} \in \mathbb{R}^D$ where N is the number of data points in the reference data set, t-SNE aims to find a low dimensional embedding $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N\} \in \mathbb{R}^d$ where $d \ll D$, such that if points $\mathbf{x}_i$ and $\mathbf{x}_j$ are close in the

multi-dimensional space, their corresponding embeddings $\mathbf{y}_i$ and $\mathbf{y}_j$ are also close. Since t-SNE is primarily used as a visualization tool, d is typically set to two. The similarity between two data points in t-SNE is defined as:

$$p_{j|i} = \frac{\exp\left(-\frac{1}{2}\mathcal{D}(\mathbf{x}_i, \mathbf{x}_j)/\sigma_i^2\right)}{\sum_{k\neq i}\exp\left(-\frac{1}{2}\mathcal{D}(\mathbf{x}_i, \mathbf{x}_k)/\sigma_i^2\right)}, \quad p_{i|i} = 0 \tag{1}$$

where $\mathcal{D}$ is a distance measure. This is then symmetrized to

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}. \tag{2}$$

The bandwidth of each Gaussian kernel σ_i is selected such that the perplexity of the distribution matches a user-specified parameter value

$$\text{Perplexity} = 2^{H(P_i)} \tag{3}$$

where $H(P_i)$ is the Shannon entropy of P_i,

$$H(P_i) = -\sum_i p_{j|i}\log_2(p_{j|i}). \tag{4}$$

Different bandwidths σ_i enable t-SNE to adapt to the varying density of the data in the multi-dimensional space.

The similarity between points $\mathbf{y}_i$ and $\mathbf{y}_j$ in the embedding space is defined using the t-distribution with one degree of freedom

$$q_{ij} = \frac{\left(1 + ||\mathbf{y}_i - \mathbf{y}_j||^2\right)^{-1}}{\sum_{k\neq l}\left(1 + ||\mathbf{y}_k - \mathbf{y}_l||^2\right)^{-1}}, \quad q_{ii} = 0. \tag{5}$$

The t-SNE method finds an embedding $\mathbf{Y}$ that minimizes the Kullback-Leibler (KL) divergence between $\mathbf{P}$ and $\mathbf{Q}$,

$$C = \text{KL}(\mathbf{P}\,||\,\mathbf{Q}) = \sum_{ij} p_{ij}\log\frac{p_{ij}}{q_{ij}}. \tag{6}$$

The time complexity needed to evaluate the similarities in Eq. 5 is $\mathcal{O}(N^2)$, making its application impractical for large data sets. We adopt a recent approach for low-rank approximation of gradients based on polynomial interpolation which reduces its time complexity to $\mathcal{O}(N)$. This approximation enables the visualization of massive data sets, possibly containing millions of data points (Linderman et al., 2019).

The resulting embeddings substantially depend on the value of the perplexity parameter. Perplexity can be interpreted as the number of neighbors for which the distances in the embedding space are preserved. Small values of perplexity result in tightly-packed clusters of points and effectively ignore the long-range interactions between clusters. Larger values may result in a more globally consistent visualizations—preserving distances on a large scale and organizing clusters in a more meaningful way—but can lead to merging small clusters and thus obscuring local aspects of the data (Kobak & Berens, 2019).

The trade-off between the local organization and global consistency may be achieved by replacing the Gaussian kernels in Eq. 1 with a mixture of Gaussians of varying bandwidths (Lee et al., 2015). Multi-scale kernels are defined as

$$p_{j|i} \propto \frac{1}{L} \sum_{l=1}^{L} \exp\left(-\frac{1}{2}\mathcal{D}(\mathbf{x}_i, \mathbf{x}_j)/\sigma_{i,l}^2\right), \quad p_{i|i} = 0 \tag{7}$$

where L is the number of mixture components as specified by the user. The bandwidths $\sigma_{i,l}$ are selected in the same manner as in Eq. 1, but with a different value of perplexity for each l. In our experiments, we used a mixture of two Gaussian kernels with perplexity values of 50 and 500. A similar formulation of multi-scale kernels was proposed in Kobak and Berens (2019), and we found the resulting embeddings are visually very similar to those obtained with the approach described above (not shown for brevity).

When using t-SNE on larger data sets, the standard learning rate $\eta = 200$ has been shown to lead to slower convergence and requires more iterations to achieve consistent embeddings (Belkina et al., 2019). We follow the recommendation of Belkina et al. and use a higher learning rate $\eta = N/12$ when visualizing larger data sets.

3.2 Adding new data points to reference embedding

Our algorithm, which embeds new data points to a reference embedding, consists of estimating similarities between each new point and the reference data and optimizing the position of each new data point in the embedding space. Unlike parametric models such as principal component analysis or autoencoders, t-SNE does not define an explicit mapping to the embedding space, and embeddings need to be found through loss function optimization.

The position of a new data point in embedding space is initialized to the median reference embedding position of its k nearest neighbors. While we found the algorithm to be robust to choices of k, we use $k = 10$ in our experiments.

We adapt the standard t-SNE formulation from Eqs. 1 and 5 with

$$p_{j|i} = \frac{\exp\left(-\frac{1}{2}\mathcal{D}(\mathbf{x}_i, \mathbf{v}_j)/\sigma_i^2\right)}{\sum_i \exp\left(-\frac{1}{2}\mathcal{D}(\mathbf{x}_i, \mathbf{v}_j)/\sigma_i^2\right)}, \tag{8}$$

$$q_{j|i} = \frac{\left(1 + ||\mathbf{y}_i - \mathbf{w}_j||^2\right)^{-1}}{\sum_i \left(1 + ||\mathbf{y}_i - \mathbf{w}_j||^2\right)^{-1}}, \tag{9}$$

where $\mathbf{V} = \left\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_M\right\} \in \mathbb{R}^D$ where M is the number of samples in the secondary data set and $\mathbf{W} = \left\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M\right\} \in \mathbb{R}^d$. Additionally, we omit the symmetrization step in Eq. 2. This enables new points to be inserted into the embedding independently of one another. The gradients of $\mathbf{w}_j$ with respect to the loss (Eq. 6) are:

$$\frac{\partial C}{\partial \mathbf{w}_j} = 2 \sum_i \left(p_{j|i} - q_{j|i}\right)\left(\mathbf{y}_i - \mathbf{w}_j\right)\left(1 + ||\mathbf{y}_i - \mathbf{w}_j||^2\right)^{-1} \tag{10}$$

In the optimization step, we refine point positions using batch gradient descent. We use an adaptive learning rate scheme with momentum to speed up the convergence, as proposed

by Jacobs (1988) and van der Maaten (2014). We run gradient descent with momentum α of 0.8 for 250 iterations, where the optimization converged in all our experiments. The time complexity needed to evaluate the gradients in Eq. 10 is $\mathcal{O}(N \cdot M)$, however, by adapting the same polynomial interpolation based approximation, this is reduced to $\mathcal{O}(\max\{N, M\})$. The time complexity can further be reduced to $\mathcal{O}(M)$ by exploiting the fact that the reference embedding remains fixed.

Special care must be taken to reduce the learning rate η as the default value in most implementations ($\eta = 200$) may cause points to "shoot off" from the reference embedding. This phenomenon is caused due to the embedding to a previously defined t-SNE space, where the distances between data points and corresponding gradients of the optimization function may be quite large. When running standard t-SNE, points are initialized and scaled to have variance 0.0001. The resulting gradients tend to be very small during the initial phase, resulting in stable convergence. When embedding new samples, the span of the embedding is much larger, resulting in substantially larger gradients, and the default learning rate causes points to move very far from the reference embedding. In our experiments, we found that decreasing the learning rate to $\eta \sim 0.1$ produces stable solutions. Alternatively, we can employ gradient clipping to achieve similar behaviour. This is especially important when using the interpolation-based approximation, which places a grid of interpolation points over the embedding space, where the number of grid points is determined by the span of the embedding. Clearly, if even one point "shoots off" far from the embedding, the number of required grid points may grow dramatically, increasing the runtime substantially. The reduced learning rate suppresses this issue, and does not slow the convergence because of the adaptive learning rate scheme, provided the optimization is run for a sufficient number of steps.

4 Experiments and discussion

We apply the proposed approach to t-SNE visualizations of single-cell data. Data in this realm include a variety of cells from specific tissues and are characterized through gene expression. In our experiments, we considered several recently published data sets where cells were annotated with the cell type. Our aim was to construct t-SNE visualizations where similarly-typed cells would cluster together, despite systematic differences between data sources. To that end, we focus on comparing different ways of using t-SNE rather than differences to embeddings like PCA or MDS, which have been substantially covered before (van der Maaten & Hinton, 2008; Becht et al., 2019). Below, we list the data sets used in our experiments, and display the resulting data visualizations. Due to the unique nature of single-cell data, we apply a specialized single-cell pipeline for all our experiments, as described in Appendix A. Finally, we discuss the success of the proposed approach in alleviating the batch effects.

4.1 Data

We use three pairs of reference and secondary single-cell data sets originating from different organisms and tissues. The data in each pair were chosen so that the majority of cell types from the secondary data set were included in the reference set (Table 1). The cells in the data sets originate from the following three tissues:

Table 1 Data sets used in our experiments

Study	Organism/tissue	Protocol	Cells	Cell types	Sparsity (%)
Hrvatin et al.	Mouse brain	inDrop	48,266	9	94
Chen et al.		Drop-seq	14,437	6	93
Baron et al.	Human pancreas	inDrop	8569	9	91
Xin et al.		SMARTer	1492	4	86
Macosko et al.	Mouse retina	Drop-seq	44,808	12	97
Shekhar et al.		Drop-seq	27,499	5	96

The first data set in each pair (Hrvatin et al., Baron et al., and Macosko et al.) was used as a reference. We relied on the quality control and annotations from the original publication and report the number of cell types after preprocessing. The cell annotations were made consistent to annotations from the Cell Ontology (Bard et al., 2005). Notice that different RNA sequencing protocols were used to estimate gene expressions

Mouse brain. The data set from Hrvatin et al. (2018) contains cells from the visual cortex exploring transcriptional changes after exposure to light. This was used as a reference for the data from Chen et al. (2017), containing cells from the mouse hypothalamus and their reaction to food deprivation. From the secondary data, we removed cells with no corresponding types in the reference: tanycytes, ependymal, epithelial, and unlabelled cells.

Human pancreas. Baron et al. (2016) created an atlas of pancreatic cell types. We used this set as a reference for data from Xin et al. (2016), who examined transcriptional differences between healthy and type 2 diabetic patients.

Mouse retina. Macosko et al. (2015) created an atlas of mouse retinal cell types. We used this as a reference for the data from Shekhar et al. (2016), who built an atlas for retinal bipolar cells.

4.2 t-SNE transform successfully alleviates batch effects

Figures 2, 3, and 4 show the embeddings of the reference data sets and their corresponding embeddings of the secondary data sets. In all the figures, the cells from the secondary data sets were positioned in the cluster of same-typed reference cells, providing strong evidence of the success of our approach. There are some deviations to these observations; for instance, in Fig. 2 several oligodendrocyte precursor cells (OPCs) were mapped to oligodendrocytes. This may be due to differences in annotation criteria by different authors, or due to inherent similarities of these types of cells. Examples of such erroneous placements can be found in other figures as well, but are uncommon and constitute less then 5% of the cells (less than 5% in brain, 1% in pancreas and 2% in retina secondary data).

Notice that we could simulate the split between reference and secondary data sets using one data set only and perform cross-validation, however this type of experiment would not incorporate batch effects. We want to remind the reader that handling batch effects were central to our endeavor and that the disregard of this effect could lead to overly-optimistic results

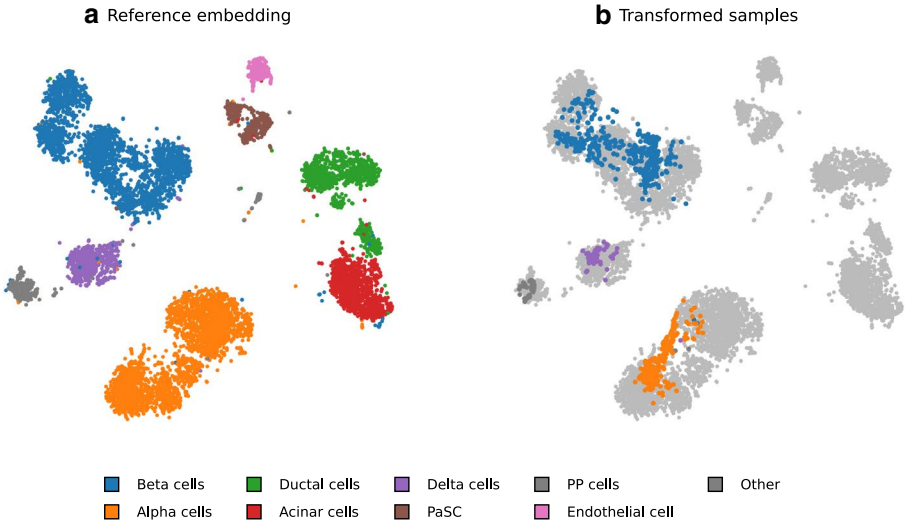

Fig. 3 Embedding of pancreatic cells from Baron et al. (2016) and cells from the same tissue from Xin et al. (2016). Just like in Fig. 2, the vast majority of the cells from the secondary data set were correctly mapped to the same-typed cluster of reference cells

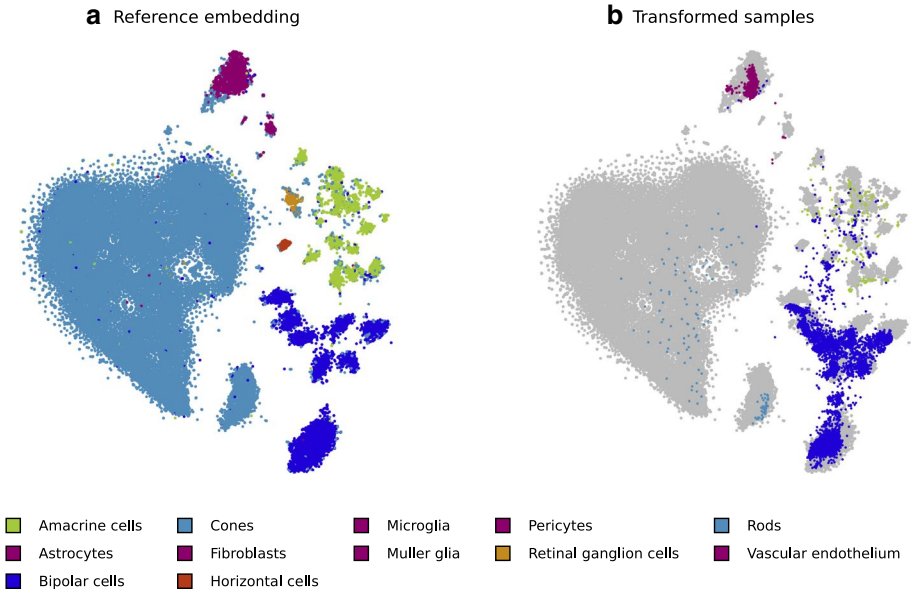

Fig. 4 An embedding of a large reference of retinal cells from Macosko et al. (2015) (**a**) and mapping of cells from a smaller study that focuses on bipolar cells from Shekhar et al. (2016) (**b**). We use colors consistent with the study by

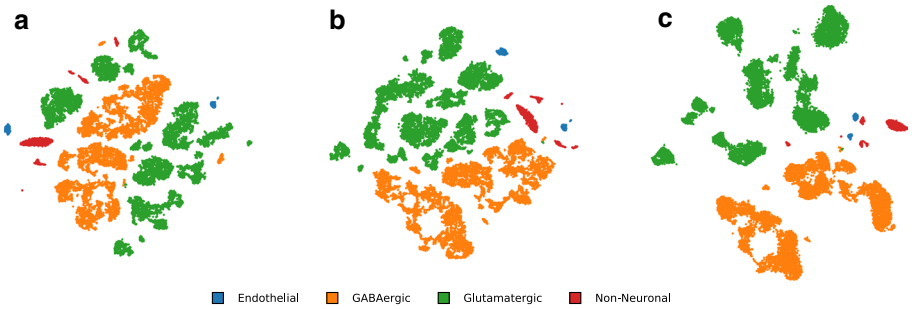

Fig. 5 A comparison of standard and multi-scale t-SNE on data from the mouse neocortex (Tasic et al., 2018). **a** Standard t-SNE using random initialization places clusters arbitrarily. The resulting clustering structure is not globally consistent, as clusters of the same type of cells are dispersed throughout the landscape. Non-Neuronal clusters, for instance, are mixed with clusters of GABAergic and Glutamaergic neurons. **b** By utilizing a globally consistent initialization for t-SNE, the clusters are organized in a more meaningful layout, where clusters of cells of the same type appear closer together. **c** Augmenting t-SNE with multi-scale similarities and using proper initialization provides a more meaningful layout of the clusters. Non-Neuronal and Endothelial cell types are now placed in the same region of the embedding. There are two clear sub-groups of GABAergic neurons corresponding to their developmental origins, which was not as apparent when using clever initialization alone

and data visualizations strikingly different from ours. For example, compare the visualizations from Figs. 1a and 2b, or Figs. 1b and 3b.

4.3 Construction of a reference embedding

We use a number of additional, recently proposed modifications to enhance the t-SNE visualization of the reference data set. Kobak and Linderman have shown that the global consistency of embeddings produced by popular visualization algorithms are largely dependent on their initialization (Kobak & Linderman, 2021). By utilizing PCA-based initialization, t-SNE is able to achieve more meaningful layouts of the resulting clusters (Fig. 5b) as opposed to using randomly initialized embeddings (Fig. 5a). Another important extension is the use of multi-scale similarities, which, in addition to considering short range interactions, also models wider point neighborhoods. Coupled with PCA-based initialization, this produces even more meaningful visualizations where clusters form interpretable structures. For instance, consider Fig. 5c, which reveals two meaningful subgroups of GABAergic neurons, corresponding to their developmental origin, as discussed in Tasic et al. (2018), while this division is less apparent when using PCA-based initialization alone in Fig. 5b.

We also observed the important role of gene selection in crafting the reference embedding spaces. We found that when selecting an insufficient number of genes, the resulting visualizations display overly-fragmented clusters. When the selection is too broad and includes lowly expressed genes, the subclusters tend to overlap. These effects can all be attributed to sparseness of the data sets and may be intrinsic to single-cell data. In our studies, we found that selection of 3000 genes yields most informative visualizations (Fig. 6).

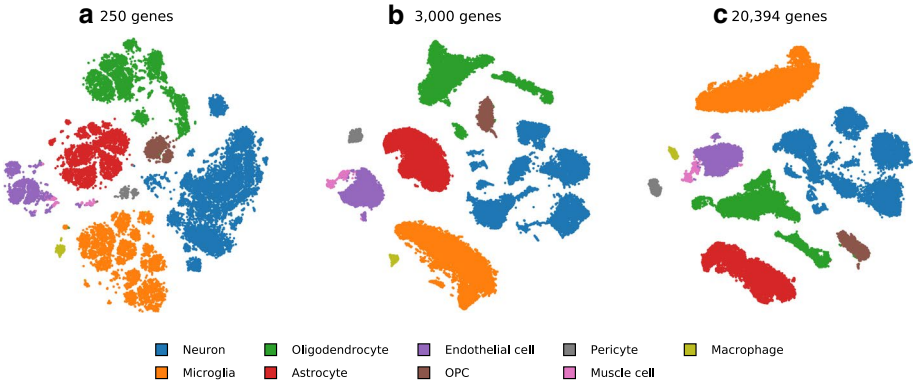

Fig. 6 Gene selection plays an important role when constructing the reference embedding. **a** Using too few genes results in fragmented clusters. **b** Using an intermediate number of genes reveals clustering mostly consistent with cell annotations. **c** Including all the genes may lead to under-clustering of the more specialized cell types. In our example, the neuronal subclusters are more clearly defined in (**b**)

4.4 Optimization is crucial to producing meaningful point embeddings

In principle, our theoretically-grounded embedding of secondary data into the scaffold defined by the reference embedding could be simplified with the application of the nearest neighbors-based procedure. For example, while describing a set of tricks for t-SNE (Kobak & Berens, 2019) proposed positioning new points into a known embedding by placing them in the median position of their 10 nearest neighbors, where the neighborhood was estimated in the original data space. Notice that we use this approach as well, but only for the initialization of positions of new data instances that are subject to further optimization. Despite both nearest-neighbors search and t-SNE optimization can be computed in linear time, the former dominates the runtime (mouse retina example; 44,808 reference, 26,830 secondary cells, 9min NN-search, 13 s optimization).

Fig. 7 demonstrates a case where nearest neighbor-based positioning alone is insufficient. We construct a reference embedding using only neurons from Hrvatin et al. (2018) (Fig. 7a) and use that to position neuronal cells from the data set from Campbell et al. (2017). We utilize the weighted mean and median positions to initialize point positions from the secondary data set, as shown in Fig. 7b, c . After initialization, we optimize point positions using the procedure described above for 500 iterations. The resulting visualizations from both initializations are visually very similar, indicating stable convergence. We show one of the resulting visualizations in Fig. 7b.

Notice that both neighbor-based initialization schemes generally position data points such that their classification is unclear. Median-based initialization produces a sort of grid-like structure, while median based initialization positions the points almost continuously across the embedding space. Optimization reveals strong correspondence of several points to reference-defined clusters, while other points from the secondary data set are pushed away from their initial clusters, possibly indicating dissimilarity.

Fig. 7 Comparison of different initialization schemes for positioning new data points onto reference embeddings. **a** We construct a reference embedding using only neuronal subtypes from Hrvatin et al. (2018). **b** We position neuronal cells from Campbell et al. (2017) using the median initialization scheme from Kobak and Berens (2019) and run optimization for 500 iterations. Compare the optimized embedding with the initial median initialization (**c**) or by using a simple weighted mean initialization (**d**)

4.5 On requirement of a complete reference set

Our approach assumes that all cell types from the secondary data set are present in the reference. Intuitively, using t-SNE in such a way is conceptually similar to classification via *k*-nearest neighbor classifiers and is similarly limited. The method may fail to reveal unseen cell types in the secondary data set, likely positioning them arbitrarily close to unrelated clusters. In some instances, unknown cell types may be sufficiently different from the reference data that t-SNE will repel them from existing clusters. However, we caution that this approach is unreliable and depends heavily on the chosen preprocessing pipeline.

We illustrate this with Fig. 8, where we first fit create a reference embedding containing only neuronal cells from Hrvatin et al. (2018). We then select only non-neuronal cells from Campbell et al. (2017) and add them to the reference embedding in Fig. 8b. The

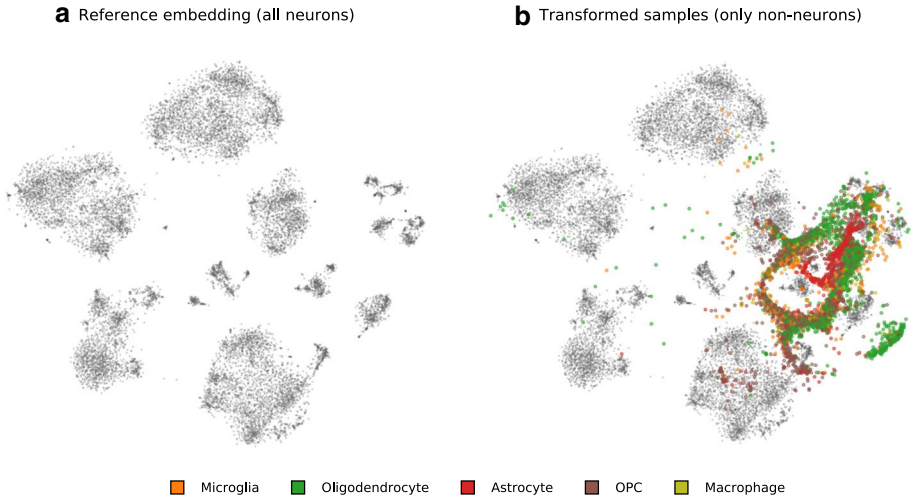

Fig. 8 A reference embedding must contain all the cell types in the secondary embedding to produce reliable results. **a** We construct a reference embedding containing only neuronal cells from Hrvatin et al. (2018). **b** We select only non-neuronal cells from Campbell et al. (2017) so that no overlap exists between the cell types between the data sets. Conceptually, t-SNE behaves similarly to a *k*-NN classifier and places the non-neuronal cells to their most similar points in the reference. In some instances, the non-neuronal cells are sufficiently different from the neuronal cells so that they are repelled from the reference clusters. Such behavior results in the "ring" seen on the right-hand side of the embedding

non-neuronal cells from Hrvatin et al. are scattered somewhat arbitrarily around several clusters in the reference embedding. Interestingly, the secondary data points form a "ring" around one of the clusters, indicating that these data points are very different from the cells in this cluster. Notice also that the points from the secondary data set exhibit little to no clustering and the different cell types seem to be mixed among each other. We hypothesize that this effect is due primarily to the single-cell preprocessing pipeline and not the limitations of our procedure itself, as the informative genes selected to create the reference neuronal embedding likely do not differentiate supportive glial cells from the secondary data set. This effect is similar to procedures such as scMap-Cluster, a consensus *k*-nearest neighbor method, which heuristically determines a distance threshold to identify unknown cell types (Kiselev et al., 2018).

Our procedure is, therefore, asymmetrical in the choice of reference and secondary data set. In practice, however, newly produced *secondary* data would be embedded into previously-prepared reference landscapes. Large collections of data *e.g.* the Human Cell Atlas initiative (Rozenblatt-Rosen et al., 2017) make it possible to scale up our approach to wider sets of cell types. Identifying potential failure cases where rare cell-types may still be missing from constructed reference embeddings is a problem that plagues the bioinformatics community and is an active area of research.

4.6 Comparison to other similar batch-effect methods

To quantitatively evaluate the predictive accuracy of the described procedure, we fit *k*-nearest neighbors classifiers on each reference t-SNE embedding from Figs. 2a, 3a and 4a and

Table 2 We compare our approach (t-SNE) to three other methods, evaluating performance using classification accuracy and the adjusted rand index (ARI)

Tissue	Method	Accuracy	ARI
Mouse brain	t-SNE	0.96	0.93
	KNN	0.96	0.93
	Random forest	**0.98**	**0.96**
	scMap-cluster	0.66	0.70
Human pancreas	t-SNE	**0.99**	**0.99**
	KNN	**0.99**	0.98
	Random forest	0.96	0.89
	scMap-cluster	0.95	0.93
Mouse retina	t-SNE	**0.99**	0.94
	KNN	**0.99**	0.96
	Random forest	**0.99**	**0.99**
	scMap-cluster	0.88	0.59

Notice that while the proposed approach classifies cells only based on their position on the two-dimensional plane, it performs comparably to other methods that use full compendium of features (gene expressions) that characterize the cells

Bold indicates the methods with the highest scores, which is pretty standard

use them to predict the cell types for the secondary data set embeddings from Figs. 2b, 3b and 4b. The accuracy measures are reported in Table 2. Our procedure of embedding new data points into two-dimensional t-SNE plane results in similar accuracy to approaches like random forests that use full compendium of cell-characterizing features. The results indicate that positioning of new cells onto a cell visualization plane is not only indicative but also an accurate instrument for cell type characterization.

We compare our approach to two machine learning techniques, namely a *k*-nearest neighbor classifier (KNN) and a random forest ensemble, and scMap-Cluster (Kiselev et al., 2018). For scMap-Cluster, we disable the distance threshold heuristic for identifying novel cell types, as our secondary data sets were chosen such that there is complete overlap between cell-types. For the two machine learning approaches, we apply the typical single-cell preprocessing pipeline described in Appendix A, i.e., library-size normalization, log-transformation, and select 1000 most informative genes. Similarly to scMap-Cluster, we use the cosine distance to find the 5 nearest neighbors in the KNN model. We used 100 trees in the random forest ensemble. The models were fit on the reference data set, and no hyper-parameter tuning was performed.

Surprisingly, both the random forest and *k*-nearest neighbor models outperform scMap-Cluster, which is specifically tailored to scRNA-seq data. However, these results may be skewed, as, in our examples, all the cell-types from the secondary data set were present in the reference data set. One of the core features of scMap-Cluster is the detection of novel cell types, which none of the other methods support. In other words, the other three methods would always assign a cell-type to a given cell, regardless of cell origin. Additionally, scMap-Cluster was primarily designed and tested on data sets produced by full-length sequencing protocols, which tend to detect a much higher number of molecules than other, sequencing protocols based on unique molecular identifiers (UMI). These two classes of sequencing protocols produce data sets with different sparsity and variance characteristics. This is consistent with the results in Table 2, as only the data sets from the human

pancreas, were produced using a full-length sequencing protocol, where scMap-Cluster achieves reasonably high accuracy.

The aim of t-SNE is to construct embeddings, in which neighborhoods are preserved, therefore it is unsurprising that the accuracy of our t-SNE based approach is largely consistent with the *k*-nearest neighbors model. While our approach is comparable to the other models in terms of accuracy, we emphasize that the goal of t-SNE embeddings is to serve as visual aids in exploratory data analysis. Therefore, it is surprising that our simple procedure performs competitively to specialized classification methods. Therefore, our procedure, in addition to providing the end-user with a cell-type prediction, allows the user to examine the low-dimensional embedding space, which may provide richer insight and interpretation of the resulting predictions.

5 Conclusion

Almost all recent publications of single-cell studies begin with a two-dimensional visualization of the data that reveals cellular diversity. While many dimensionality reduction techniques are available, different variants of t-SNE are most often used to produce such visualizations. Single-cell studies enable the exploration of biological mechanisms at a cellular level, and their publications in the past couple of years are abundant. One of the central tasks in single-cell studies is the classification of new cells based on findings from previous studies. Such transfer of knowledge is often difficult due to batch effects present in data from different sources. Addressing batch effects by adapting and extending t-SNE, the prevailing method used to present single-cell data in two-dimensional visualization, motivated the research presented in this paper.

The proposed approach uses a t-SNE embedding as a scaffold for the positioning of new cells within the visualization, and possibly for aiding in their classification. The three case studies incorporating pairs of data sets from different domains but with similar classifications demonstrate that our proposed procedure can effectively deal with batch effects to construct visualizations that correctly map secondary data sets onto an embedding of the data from an independent study that possibly uses different experimental protocol. We quantitatively evaluate the predictive accuracy of our approach by fitting a *k*-nearest neighbors model on the resulting two-dimensional embeddings and compare its predictive accuracy to other machine learning methods that use the entire compendium of gene expressions that characterize the cells. Experiments show that our approach is successful in predicting cell types and performs comparably to other methods. This encouraging result indicates that by using our procedure, scientists can quickly and accurately determine the composition of new data by merely visualizing and inspecting resulting visualizations. While we focused here on reference visualizations constructed using t-SNE, this approach can be applied using any existing two-dimensional visualization.

6 Availability and implementation

The procedures described in this paper are provided as Python notebooks that are, together with the data, available in an open repository.[1] The described methods were implemented and incorporated into openTSNE, our open-source, extensible t-SNE library for Python (Poličar et al., 2019).

Appendix A: Single-cell data preprocessing pipeline

Due to the specific nature of single-cell data, additional steps must be taken to properly apply t-SNE. We use a standard single-cell preprocessing pipeline, consisting of the selection of 3,000 representative genes (see Appendix B), library size normalization, log-transformation, standardization, and PCA-based representation that retains 50 principal components (Stuart et al., 2019; Wolf et al., 2018). To obtain the reference embedding, we apply multi-scale t-SNE using PCA initialization (Kobak & Berens, 2019). Due to high-dimensionality of the preprocessed input data we use cosine distance to estimate similarities between reference data points (Domingos 2012). When adding new data points from the secondary data set to the reference embedding, we select 1000 genes present in both data sets and use the cosine similarity to estimate the similarities between the secondary data item and reference data points. We note that similarities are computed using the raw count matrices. The preprocessing stages are detailed in accompanying Python notebooks (Sect. 5).

Appendix B: Gene selection

Single-cell data sets suffer from high levels of technical noise and low capture efficiency, resulting in sparse and noisy expression matrices (Islam et al., 2014). A common occurrence in these data sets is "dropout", where an expressed gene is not measured, and its corresponding matrix entry is set to zero. To address this problem, we use a specialized feature-selection method, which exploits the mean-dropout relationship of expression counts as recently proposed by Kobak and Berens (2019). In this context, we will refer to all genes with matrix entries set to zero as dropouts. Intuitively, if a gene has high mean expression over all cells, but is detected in only a handful of them (i.e. has high mean dropout rate), then this gene is likely specific to a specific cell-type and will serve as a good feature for any subsequent analysis where we wish to discriminate between cell types.

More formally, given an expression matrix $\mathbf{X} \in \mathbb{R}^{N \times G}$ where N is the number of cells and G is the number of genes in the data set, we compute the fraction of cells where a gene g was not expressed i.e. its dropout rate

$$d_g = \frac{1}{N} \sum_i I\big(X_{ig} = 0\big), \tag{11}$$

where I is the indicator function. The mean $\log_2$ expression of gene g considers only cells i in which gene g was expressed

[1] https://github.com/biolab/tsne-embedding.

$$m_g = \left\langle \log_2 X_{ig} \mid X_{ig} > 0 \right\rangle. \tag{12}$$

All genes expressed in less than ten cells are discarded. In order to select a desired number of $\hat{G}$ genes, we use binary search to find a parameter value of b such that

$$\sum_g I\left(d_g > \exp\left[-(m_g - b)\right] + 0.02\right) = \hat{G}. \tag{13}$$

In this way, we are able to select a desired number of genes $\hat{G}$, which appear discriminative between cell types for the given gene-expression data set.

Acknowledgements This work was supported by the Slovenian Research Agency Program Grant P2-0209, and by the BioPharm.SI project supported from European Regional Development Fund and the Slovenian Ministry of Education, Science and Sport. We would also like to thank Dmitry Kobak for many helpful discussions on t-SNE.

References

Bard, J., Rhee, S. Y., & Ashburner, M. (2005). An ontology for cell types. *Genome Biology, 6,* 2.

Baron, M., Veres, A., Wolock, S. L., Faust, A. L., Gaujoux, R., Vetere, A., et al. (2016). A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Systems, 3*(4), 346–360.

Becht, E., McInnes, L., Healy, J., Dutertre, C. A., Kwok, I. W. H., Ng, L. G., et al. (2019). Dimensionality reduction for visualizing single-cell data using UMAP. *Nature Biotechnology, 37*(1), 38–47.

Belkina, A. C., Ciccolella, C. O., Anno, R., Halpert, R., Spidlen, J., & Snyder-Cappione, J. E. (2019). Automated optimized parameters for T-distributed stochastic neighbor embedding improve visualization and analysis of large datasets. *Nature Communications, 10*(1), 1–12.

Bickel, S., & Brückner, M. & Scheffer, T. . (2009). Discriminative learning under covariate shift. *Journal of Machine Learning Research, 10,* 2137–2155.

Butler, A., Hoffman, P., Smibert, P., Papalexi, E., & Satija, R. (2018). Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nature Biotechnology, 36*(5), 411.

Campbell, J. N., Macosko, E. Z., Fenselau, H., Pers, T. H., Lyubetskaya, A., Tenen, D., et al. (2017). A molecular census of arcuate hypothalamus and median eminence cell types. *Nature Neuroscience, 20*(3), 484.

Chen, R., Xiaoji, W., Jiang, L., & Zhang, Y. (2017). Single-cell RNA-Seq reveals hypothalamic cell diversity. *Cell Reports, 18*(13), 3227–3241.

Cox, M. A. A., & Cox, T. F. (2008). Multidimensional scaling. In C. Chen, W. Härdle, and A. Unwin (eds.) *Handbook of data visualization* (pp. 315–347). Springer

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM, 55*(10), 78–87.

Gopalan, R., Li, R., & Chellappa, R. (2011). Domain adaptation for object recognition: An unsupervised approach. In *2011 International conference on computer vision* (pp. 999–1006). IEEE

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

Hicks, S. C., Townes, F. W., Teng, M., & Irizarry, R. A. (2018). Missing data and technical variability in single-cell RNA-sequencing experiments. *Biostatistics, 19*(4), 562–578.

Hie, B., Bryson, B., & Berger, B. (2019). Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. *Nature Biotechnology, 37*(6), 685–691.

Hrvatin, S., Hochbaum, D. R., Nagy, M. A., Cicconet, M., Robertson, K., Cheadle, L., et al. (2018). Greenberg. Single-cell analysis of experience-dependent transcriptomic states in the mouse visual cortex. *Nature Neuroscience, 21*(1), 120–129.

Islam, S., Zeisel, A., Joost, S., La Manno, G., Zajac, P., Kasper, M., et al. (2014). Quantitative single-cell RNA-seq with unique molecular identifiers. *Nature Methods, 11*(2), 163.

Jacobs, R. A. (1988). Increased rates of convergence through learning rate adaptation. *Neural Networks, 1*(4), 295–307.

Kiselev, V. Y., Yiu, A., & Hemberg, M. (2018). Scmap: Projection of single-cell RNA-seq data across data sets. *Nature Methods, 15*(5), 359–362.

Kobak, D., & Berens, P. (2019). The art of using t-SNE for single-cell transcriptomics. *Nature Communications, 10*(1), 5416.

Kobak, D., & Linderman, G. C. (2021). Initialization is critical for preserving global data structure in both t-SNE and UMAP. *Nature Biotechnology, 39*(2), 156–157.

Korsunsky, I., Millard, N., Fan, J., Slowikowski, K., Zhang, F., Wei, K., et al. (2019). Fast, sensitive and accurate integration of single-cell data with Harmony. *Nature Methods, 16*(12), 1289–1296.

Lee, J. A., Peluffo-Ordóñez, D. H., & Verleysen, M. (2015). Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure. *Neurocomputing, 169,* 246–261.

Linderman, G. C., Rachh, M., Hoskins, J. G., Steinerberger, S. & Kluger, Y. (2019). Fast interpolation-based t-SNE for improved visualization of single-cell RNA-seq data. *Nature Methods*, *16*(3), 243–245

Liu, J., Huang, Y., Singh, R., Vert, J.-P. & Noble, W. S. (2019) Jointly embedding multiple single-cell omics measurements. *The Workshop on Algorithms in Bioinformatics, 143*

Li, X., Wang, K., Lyu, Y., Pan, H., Zhang, J., Stambolian, D., et al. (2020). Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis. *Nature Communications, 11*(1), 1–14.

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

Macosko, E. Z., Basu, A., Satija, R., Nemesh, J., Shekhar, K., Goldman, M., et al. (2015). Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell, 161*(5), 1202–1214.

McInnes, L., & Healy, J. (2018). James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. arXiv.

Polański, K., Young, M.D., Miao, Z., Meyer, K.B., Teichmann, S.A., & Park J.E. (2019). BBKNN: fast batch alignment of single cell transcriptomes. *Bioinformatics*, *36*(3), 964–965.

Poličar, P. G., Stražar, M., & Zupan, B. (2019). OpenTSNE: A modular Python library for t-SNE dimensionality reduction and embedding. *bioRxiv*, 1–2

Quionero-Candela, J., Sugiyama, M., Schwaighofer, A., & Lawrence, N. D. (2009). *Dataset shift in machine learning*. The MIT Press.

Rozenblatt-Rosen, O., Stubbington, M. J. T., Regev, A., & Teichmann, S. A. (2017). The Human Cell Atlas: From vision to reality. *Nature, 550*(7677), 451–453.

Shekhar, K., Lapan, S. W., Whitney, I. E., Tran, N. M., Macosko, E. Z., Kowalczyk, M., et al. (2016). Comprehensive classification of retinal bipolar neurons by single-cell transcriptomics. *Cell, 166*(5), 1308-1323.e30.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck, W. M., et al. (2019). Comprehensive integration of single-cell data. *Cell, 177*(7), 1888-1902.e21.

Tasic, B., Yao, Z., Graybuck, L. T., Smith, K. A., Nguyen, T. N., Bertagnolli, D., et al. (2018). Shared and distinct transcriptomic cell types across neocortical areas. *Nature, 563*(7729), 72–78.

Tung, P.-Y., Blischak, J. D., Hsiao, C. J., Knowles, D. A., Burnett, J. E., Pritchard, J. K., & Gilad, Y. (2017). Batch effects and the effective design of single-cell gene expression studies. *Scientific Reports, 7,* 39921.

van der Maaten, L. (2014). Accelerating t-SNE using tree-based algorithms. *Journal of Machine Learning Research, 15*(93), 3221–3245.

van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research, 9*(86), 2579–2605.

Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). A survey of transfer learning. *Journal of Big Data, 3*(1), 1–40.

Wold, S., Esbensen, K., & Geladi, P. (1987). Principal component analysis. *Chemometrics and Intelligent Laboratory Systems, 2*(1–3), 37–52.
Wolf, F. A., Angerer, P., & Fabian, J. (2018). Theis. SCANPY: Large-scale single-cell gene expression data analysis. *Genome Biology, 19*(1), 15.
Xin, Y., Kim, J., Okamoto, H., Ni, M., Wei, Y., Adler, C., et al. (2016). RNA sequencing of single human islet cells reveals type 2 diabetes genes. *Cell Metabolism, 24*(4), 608–615.