Embedding to Reference t-SNE Space Adresses Batch-Effects in Single-Cell Classification

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

[1] University of Ljubljana, SI-1000 Ljubljana, Slovenia
{pavlin.policar,martin.strazar,blaz.zupan}@fri.uni-lj.si
[2] Baylor College of Medicine, Houston, TX 77030, USA

Abstract. Dimensionality reduction techniques, such as t-SNE, can construct informative visualizations of high-dimensional data. When working with multiple data sets, the straightforward application of these methods often fails, and instead of revealing underlying classes, the resulting visualizations expose domain-specific clusters. To circumvent these batch effects, we here propose an embedding procedure that takes a t-SNE visualization constructed on reference data set and uses it as a scaffold for new embeddings. New, secondary data is embedded one data-instance at the time, thus disregarding any interactions between data items in the secondary data and implicitly mitigating batch effects. We demonstrate the utility of this procedure on analysis of six recently published single-cell gene expression data sets containing up to tens of thousands of cells and thousands of genes. In these data sets, the batch effects were particularly strong as the data comes from different institutions and was obtained using different experimental protocols. The visualizations constructed by the proposed approach are absent of batch effects, and the cells from secondary data sets correctly co-cluster with cells from the primary data sharing the same cell type.

Keywords: Batch effects · Embedding · t-SNE · Visualisation · Single-Cell Transcriptomics · Data Integration · Domain Adaptation.

1 Introduction

Two-dimensional embeddings and their visualizations may assist in the analysis and interpretation of multi-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 local embeddings such as t-distributed stochastic neighbor embedding (t-SNE) [17] or uniform manifold approximation and projection [19] have recently complemented traditional data transformation and embedding approaches such as principal component analysis (PCA) and multi-dimensional scaling [24, 3]. While useful for visualizing the data from a single coherent source, these methods may encounter problems if the data comes from different sources. Here, if executing the dimensionality reduction procedures on a merged data, the resulting visualizations would typically depict

source-specific clusters instead of reveal clusters that are semantically enriched, say, with same-type of data instances across different sources. This source-specific confounding is often referred to as *domain shift* [8], *covariate shift* [4] or *dataset shift* [20]. In the bioinformatics literature, the domain-specific differences are more commonly referred to as *batch effects* [5, 9, 23].

Massive, multi-variate biological data sets typical suffer from source-specific biases. Consider an example from single-cell genomics, a domain we will focus on in this manuscript and that was — besides current scientific challenges — selected also due to availability and the abundance of recently published data. Single-cell data sets may include thousands of cells whose profiles consist of tens of thousands of cell-specific gene expressions. Single cell studies typically start with the analysis of cell types, where we expect that cells of the same type would cluster in two-dimensional data visualisation [23]. For instance, Fig. 1.a shows t-SNE embedded data from mouse brain cells originating from visual cortex [10] and hypothalamus [6]. 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. 1.a — 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 pancreas cells in Fig. 1.b, which renders the data from human cell atlas [2] and similarly-typed cells from diabetic patients [26]. Just like with data from brain cells, pancreas cells cluster first according to the data source, again resulting in an uninformative visualization that primarily exposes the batch effect.

Current solutions to embedding the data from various data sources address the batch effect problems up-front. The data is typically pre-processed and transformed, possibly to the latent space, so that the batch effects are removed. The recently proposed procedures for batch-effect removal include canonical correlation analysis [5] and mutual nearest-neighbors [9, 23]. In these works, a proof for the success of batch effect removal is a good mixing of cells from different sources in a t-SNE visualization. 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 data analysis pipeline has to be rerun, usually resulting in different layout and clusters that have little resemblance to original visualization and thus require reinterpretation.

We propose a direct solution of rendering t-SNE visualizations that addresses batch effects. Our approach considers one of the data sets as a *reference* and aims to visually classify the cells in the other, *secondary data set*. We construct the t-SNE embedding using the reference data set, and then use it as a scaffold for the embedding of data points from the secondary data. Embedding is performed one data point at a time. Independence of each new embedding of data instances from secondary data set causes clustering landscape to only depend on a 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 the "gravitational field" to independently embedded each

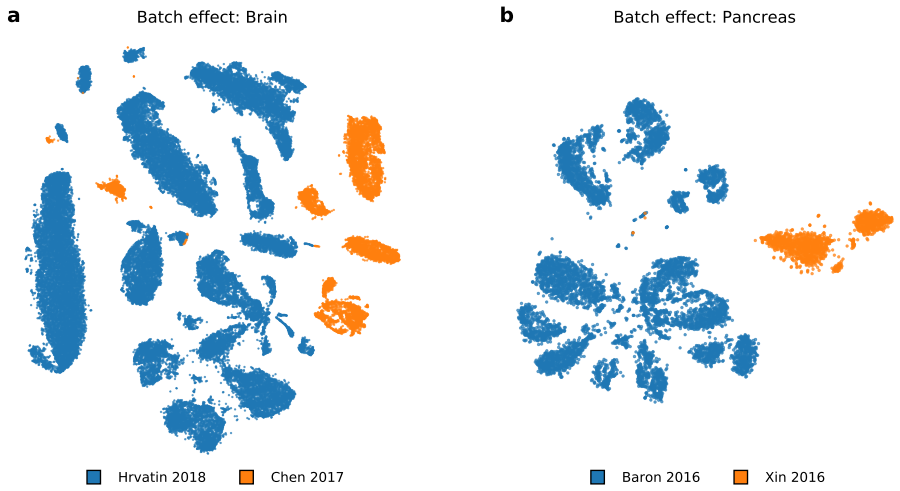

Fig. 1. Batch effects are a driving factor of variation between the data sets. Depicted is a t-SNE visualisation of two pairs of data sets. In each pair, the data sets share cell types, so it would be expected that the cells from the reference data (blue) would mix with the cells in a secondary data sets (orange). Instead, t-SNE visualisation clusters data according to the data source.

new data item. For example, in Fig. 2, the cells from visual cortex define the scaffold (Fig. 2.a) into which we embed the cells from hypothalamus (Fig. 2.b). Unlike in their joint t-SNE visualization (Fig. 1.a), the hypothalamus cells are dispersed across entire embedding space and their cell type correctly matches the prevailing type in reference clusters.

At the core of the proposed solution is a mapping that implements embedding of new data to existing t-SNE visualization. In the following, we introduce t-SNE, then describe its recently proposed multi-scale extension, and outline our algorithm that embeds new data into a previously trained scaffold. While a utility of such an algorithm was already hinted to in recent publication [14], we here provide its practical and theoretically-grounded implementation. Considering the abundance of recent publications in batch effect removal, we present surprising evidence that a computationally more direct and elegant embedding procedure solves the batch effects problem when constructing interpretable visualizations from different data sources.

2 Methods

We describe an end-to-end pipeline that uses t-SNE embeddings as a scaffold for new data samples, and enables visualisation of data from different sources while mitigating batch effects. Our proposed approach starts with t-SNE to embedd a reference data set, with the aim of constructing a two-dimensional

visualisation to facilitate interpretation and cluster classification. We regard the resulting two-dimensional embedding as a scaffold to which we add samples from the new, secondary data source. Each new sample is then placed into the reference embedding and optimized independently using the t-SNE loss function. Independent embedding of each data item from a secondary data set disregards any interactions present in that data, and prevents forming clusters that would be data-souce specific. Below, we start with a summary of t-SNE and its extensions (Sec. 2.1, emphasizing the notation and relevant parts upon which we base our secondary data embedding approach (Sec. 2.2).

2.1 Data embedding by t-SNE and its extensions

The embedding by t-SNE is local and non-linear, tailored to visualisation of high dimensional data sets. 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 samples 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:

$$p_{j|i} = \frac{\exp\left(-\frac{1}{2}\mathcal{D}(\mathbf{x}_i, \mathbf{x}_j)/\sigma_i^2\right)}{\sum_{k \neq_1} \exp\left(-\frac{1}{2}\mathcal{D}(\mathbf{x}_i, \mathbf{x}_k)/\sigma_i^2\right)}, \quad p_{i|i} = 0 \tag{1}$$

where $\mathcal{D}$ is some 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 are 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 data transformation by t-SNE finds an embedding $\mathbf{Y}$ that minimizes the Kullback-Leibler (KL) divergence between $\mathbf{P}$ and $\mathbf{Q}$,

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

The time complexity needed to evaluate the similarities in Eq. 5 is $\mathcal{O}(N^2)$, making the applications of the algorithm impractical for any reasonably-sized data. To address larger data sets, we instead adapt a recent approach for low-rank approximation of similarity matrix based on polynomial interpolation that reduces the time complexity of t-SNE to $\mathcal{O}(N)$. This approximation enables the visualization of massive data sets possibly containing millions of data points [16].

The t-SNE embeddings substantially depend on the value of the perplexity parameter. The perplexity parameter 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 lead to ignoring the long-range interactions between clusters. Larger values may result in a more globally consistent visualisations, preserving distances on a large scale and organizing clusters in a more meaningful way. Larger values of perplexity may often result in merging multiple small clusters, thus obscuring local aspects of the data [14].

A 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 [15]. 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. 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. We note that a similar formulation of multi-scale kernels was proposed in [14], and we found the resulting embeddings are visually very similar to those obtained with the approach described above (for brevity, data not shown).

2.2 Adding new data points to reference embedding

An algorithm to embedd new data point to a reference embedding consists of estimating similarities between the point and the reference data and optimizing the position of the 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 embedding space, and embeddings need to be found through loss function optimization.

For each new data point, we first estimate its distance to each of the data instances in the reference data set. We initialize the position of the data point in embedding space to the median embedding position of its k nearest neighbors from the reference data set. 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

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

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

where $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_M\} \in \mathbb{R}^D$ where M is the number of samples in the new data set and $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M\} \in \mathbb{R}^d$. 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 proposed by Jacobs to speed up the convergence [12]. We run gradient descent with momentum α set to 0.8 for 250 iterations, where the optimization converged for all cases in 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 $\mathcal{O}(N)$.

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 already defined t-SNE space, where the distances between data points and corresponding gradients of the optimization function may be quite large. When running s 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 range of the embedding is much larger, resulting in much larger gradients, and the default learning rate causes points to move very far from the reference embedding. In our experiments, we found that lowering the learning rate to $\eta \sim 0.1$ produces stable solutions. This is especially important when using the interpolation-based approximation. Our approach places a grid of interpolation points over the embedding space, where the number of grid points is determined by the embedding range. Clearly, if even one point "shoots off" far from the embedding, the number of required grid points escalates, substantially increasing the runtime. The reduced learning rate suppresses this issue, and does not slow the convergence because of an adaptive learning rate scheme and running the optimization for a sufficient number of steps.

3 Experiments and Discussion

We apply the proposed transfer learning approach to t-SNE visualizations of single-cell data. In single-cell data sets, the data includes the variety of cells from specific tissue and characterizes the cells through the expression of its genes. In 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 the differences between data sources. Below, we enlist the data sets, describe single-cell specific data preprocessing procedures, and display the resulting data visualizations. Finally, we discuss the success of the proposed approach in alleviating the batch effects.

3.1 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).

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	8,569	9	91
Xin *et al.*		SMARTer	1,492	4	86
Macosko *et al.*	mouse retina	Drop-seq	44,808	12	97
Shekhar *et al.*		Drop-seq	27,499	5	96

Table 1. Data sets used in our experiments. In each pair, the first data set (Hrvatin *et al.*, Baron *et al.*, and Macoscko *et al.*) was used as a reference. In all cases, we relied on quality control and annotations from the original studies. To facilitate comparisons, the cell annotations were harmonized using the cell type annotations from cell ontology [1]. Notice that different RNA sequencing protocols were used to estimate gene expressions. We here report the number of cell types from each data set retained after preprocessing. Single-cell data is sparse, typically containing less than 10% expressed genes per cell.

The cells in the data sets were captured from the following three tissues:

Mouse brain. The data set from Hrvatin *et al.* [10] 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.* [6], containing various 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, namely ependymal cells, epithelial cells, tanycytes, and unlabelled cells.

Human pancreas. The data set from Baron *et al.* [2] was created as an atlas of pancreatic cell types. We used this set as a reference for data from Xin *et al.* [26], who examined transcriptional differences between healthy and type 2 diabetic patients.

Mouse retina. The data set from Macosko *et al.* [18] was created as an atlas of mouse retinal cell types. We used this as a reference for the data from Shekhar *et al.* [22], who built an atlas for different types of retinal bipolar cells.

3.2 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 selection of 3,000 representative genes (see Sec. 3.3), library size normalization, log-transformation, standardization, and PCA-based representation that retains 50 principal components [23, 25]. To obtain the reference embedding, we apply multi-scale t-SNE using PCA initialization [14]. Due to high-dimensionality of the preprocessed input data we use cosine distance to estimate similarities between reference data points [7]. When adding new data points from secondary data set to the reference embedding, we select 1,000 genes present in both data sets and use these to estimate the similarities between the secondary data item and reference data points. The similarities are estimated using cosine similarity. We note that similarity is computed using the raw count matrices. The preprocessing stages are detailed in accompanying Python notebooks (Sec. 3.5).

3.3 Gene selection

Single-cell data sets suffer from a high level of technical noise and low capture efficiency, resulting in sparse expression matrices [11]. 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 [14]. Here, genes with higher than expected dropout rate are regarded as potential markers for cell subpopulations and are retained in the data.

Given an expression matrix $\mathbf{X} \in \mathbb{R}^{N \times G}$ where N is the number of samples and G is the number of genes in the data set, we compute the fraction of cells where a gene g was not expressed

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

The mean $\log_2$ expression of the genes is computed from all the cells where gene was expressed:

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

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

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

In our experiments we use $t = 0$ and $a = 1$.

3.4 Results and Discussion

Figs. 2, 3, and 4 show the embedding 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 the proposed approach. There are some deviations to these observations; for instance, in Fig. 2 several oligodendrocytes precursor cells (OPC) 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 they are not common and constitute less then 5% of the cells (less than %1 for pancreas, %2 for retina and 5% for brain secondary data sets).

Notice that we could simulate the split between reference and secondary data sets by cross-validation using one data set only, as this type of experiments would not incorporate batch effects. We want to remind the reader that handling batch effects were central to our endeavor and that disregard of this effect could lead to data visualizations strikingly different from ours. For example, compare the visualisations from Fig 1.a and Fig. 2.b, or Figs. 1.b and 3.b.

There are a few tricks that we use in our procedure for t-SNE embedding of the secondary data set that were proposed recently and enhance the original t-SNE. An important one is a multi-scale extension that besides local ordering of the data points takes care about global optimization of the cluster placement. For illustration, consider visualizations with standard and multi-scale t-SNE in Fig. 5. Notice, for instance, that in multi-scale t-SNE (Fig. 5.b) the clusters with neuron cells are clumped together, while their placement in standard t-SNE is arbitrary (Fig. 5.a).

We have also observed an important role of gene selection in crafting the reference embedding spaces. We have found that when selecting an insufficient number of genes, the resulting visualizations display 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 for single-cell data. In our studies, we found that selection of 3,000 genes yields most informative visualizations (Fig. 6).

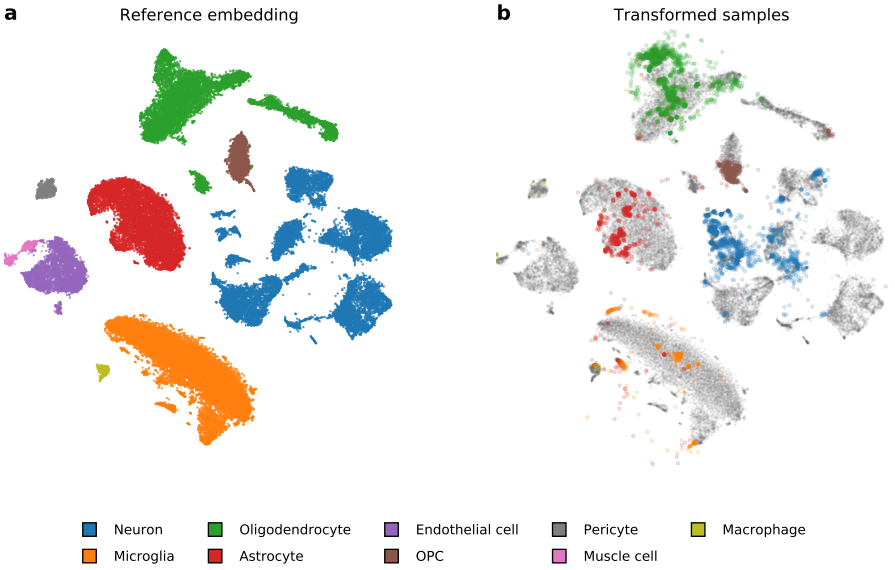

Fig. 2. Two-dimensional embedding of a reference brain cells (a) and corresponding mapping of secondary data from hypothalamus cells (b). Notice that the majority of hypothalamus cells were mapped to their corresponding reference cluster. For instance, the astrocyte cells marked with red on the right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

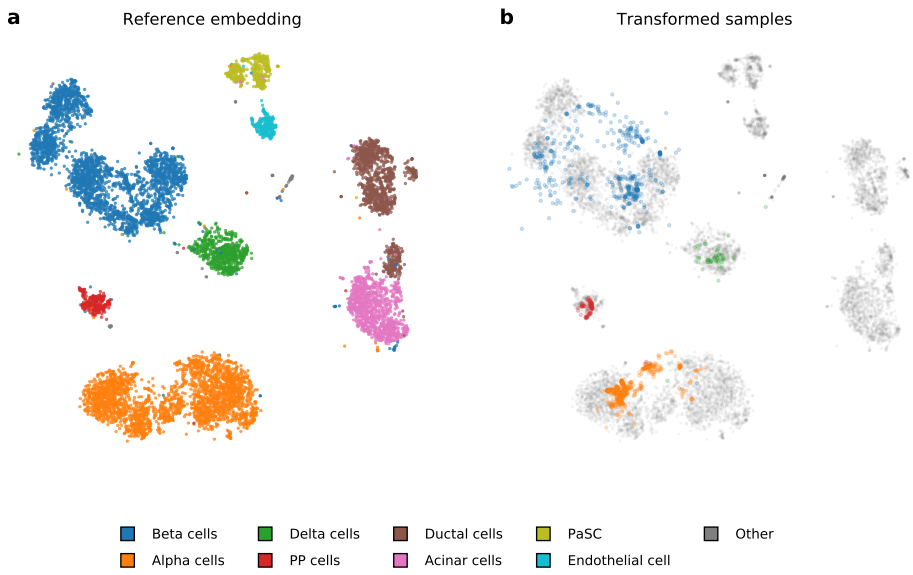

Fig. 3. Embedding of pancreas cells from Baron *et al.* [2] and cells from the same tissue from Xin *et al.*. [26]. Just like in Fig 2 vast majority of the cells from the secondary data set were correctly mapped to the same-typed cluster of reference cells.

In principle, our theoretically-grounded embedding of secondary data into the scaffold defined by 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 and Berens [14] proposed to position 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 trick as well, but only for initialization of positions of new data instances that are subject to further optimization. In Fig. 7 we demonstrate that nearest neighboring-based positioning is insufficient and may yield clumped visualizations where the optimal positioning using the t-SNE loss function is much more dispersed and rightfully shows a more considerable variation in the secondary data. Some data points may also fell into the neighboring space between differently typed clusters, while after optimization they typically converge closer to same-typed groups.

The proposed method assumes that all cell types from the secondary data set are present in the reference. The proposed method would fail to reveal novel cell types in the secondary data set, possibly positioning them arbitrarily close to unrelated clusters. Procedures such as scmap [13] were recently proposed to cope with such cases and identify the cells whose type is new and not included in the reference. Our procedure does not address such cases, and for scaling-up to

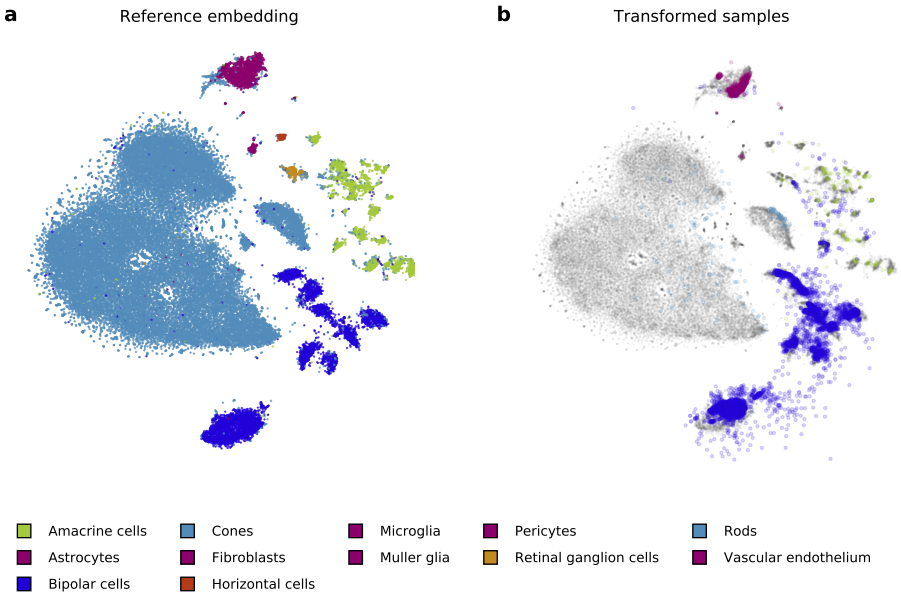

Fig. 4. Embedding of a large reference retina cells from Macosco *et al.* [18] (a) and mapping of cells from a smaller study that focuses on bipolar cells from Shekhar *et al.* [22] (b). We use colors consistent with the study by Macosko *et al.*. Notice that Shekhar includes cells of only four different types.

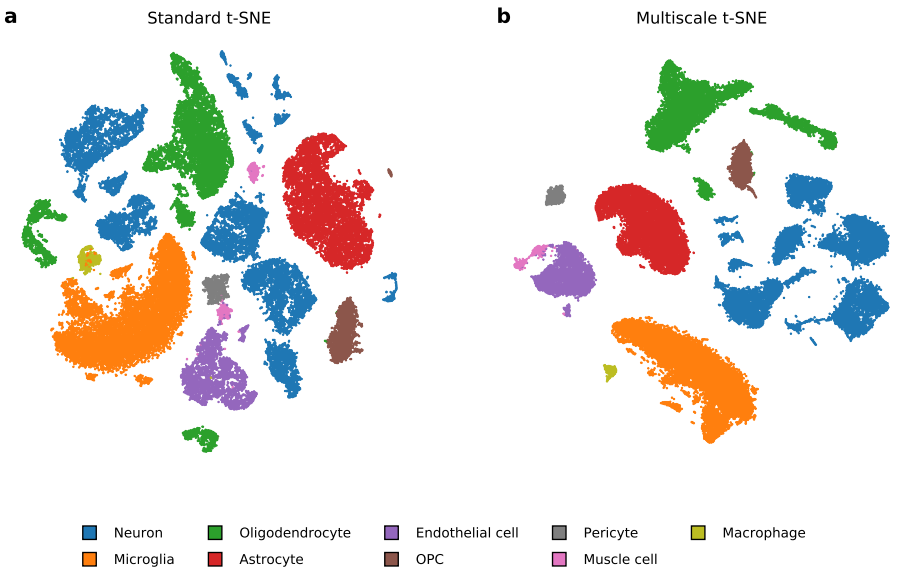

Fig. 5. Comparison of embedding by stanard and multiscale t-SNE on a data set from the mouse visual cortex [10]. **(a)** Standard t-SNE places clusters arbitrarily. **(b)** Augmenting t-SNE with multi-scale similarities provides a more meaningful layout of the clusters. Neuronal types occupy one region of the space. Oligodendrocyte precursor cells (OPCs) are mainly progenitors to oligodendrocytes, but may also differentiate into neurons or astrocytes.

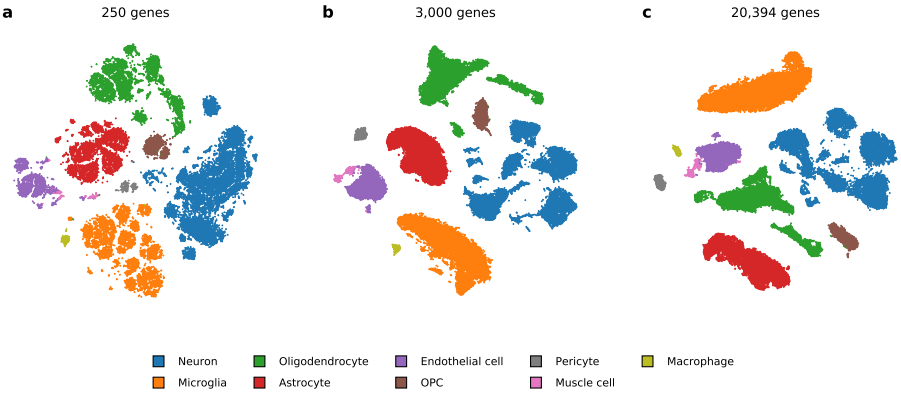

Fig. 6. Gene selection plays an important role when constructing the reference embedding. **(a)** Using too few genes results in over-clustering. **(b)** Using an intermediate number of genes reveals clustering mostly consistent with cell annotations. **(c)** Including all the genes may lead to under-clustering of the more specialized cell types.

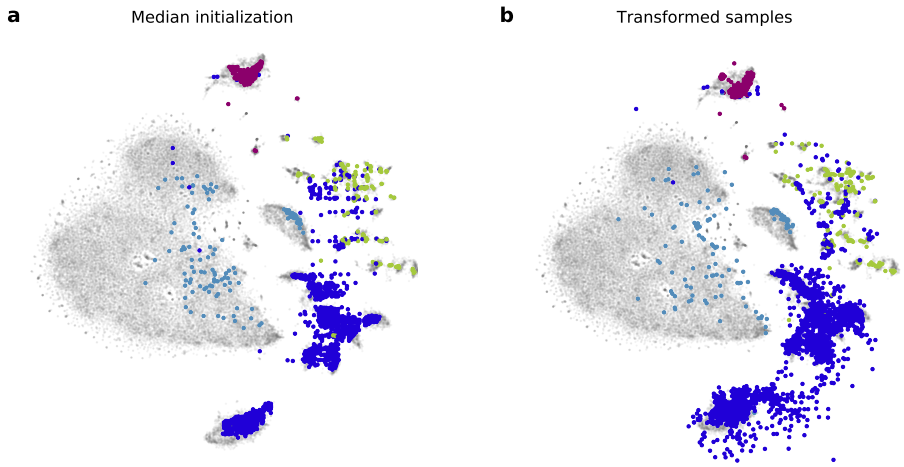

Fig. 7. Comparison of data placement by nearest neighbors approach from Kobak and Berens [14] and optimized placement by our algorithm. **(a)** Data points are placed to the median position of their 10 nearest neighbors in the reference set. **(b)** Point positions are optimized, revealing a different, more dispersed placement that better reflects the variety of cells in the secondary data set.

a wider collection of cell types relies on emerging availability of large collections of the reference data such as those managed by Human Cell Atlas initiative [21].

3.5 Implementation

The procedures described in this paper are provided as Python notebooks that are, together with the data, available in an open repository [3]. All experiments were run using openTSNE [4], our open and extensible t-SNE library for Python.

4 Conclusion

Almost all recent publications of single-cell studies start with a two-dimensional visualization of the data that exposes the diversity as well as different types of cells from the study. While any dimensionality reduction technique can be used to render such a visualization, different variants of t-SNE are most often used. Due to the ability to explore biological mechanisms at the cellular level, single-cell studies are increasingly widespread, 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 finding from previous studies. Such transfer of knowledge is often difficult due to batch effects present in data from different

[3] https://github.com/biolab/tsne-embedding
[4] https://github.com/pavlin-policar/openTSNE

sources. Solving the problem of the batch-effects together with prevailing means of presenting single-cell data in two-dimensional visualizations motivated the research presented in this paper.

Our proposed approach uses t-SNE embedding as a scaffold for the positioning of new cells within the visualization, and possibly for aiding 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 a secondary data set onto a reference data set from an independent study that possibly uses different data collection protocol. While we here focused on t-SNE constructed from reference data sets, the approach could be applied to any existing two-dimensional visualization.

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 helpful discussions on t-SNE extensions.

References

1. Bard, J., Rhee, S.Y., Ashburner, M.: An ontology for cell types. Genome Biology **6**(2), R21 (2005)
2. Baron, M., Veres, A., Wolock, S.L., Faust, A.L., Gaujoux, R., Vetere, A., Ryu, J.H., Wagner, B.K., Shen-Orr, S.S., Klein, A.M., et al.: A single-cell transcriptomic map of the human and mouse pancreas reveals inter-and intra-cell population structure. Cell Systems **3**(4), 346–360 (2016)
3. Becht, E., McInnes, L., Healy, J., Dutertre, C.A., Kwok, I.W., Ng, L.G., Ginhoux, F., Newell, E.W.: Dimensionality reduction for visualizing single-cell data using umap. Nature Biotechnology **37**(1), 38 (2019)
4. Bickel, S., Brückner, M., Scheffer, T.: Discriminative learning under covariate shift. Journal of Machine Learning Research **10**(Sep), 2137–2155 (2009)
5. Butler, A., Hoffman, P., Smibert, P., Papalexi, E., Satija, R.: Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nature Biotechnology **36**(5), 411 (2018)
6. Chen, R., Wu, X., Jiang, L., Zhang, Y.: Single-cell rna-seq reveals hypothalamic cell diversity. Cell Reports **18**(13), 3227–3241 (2017)
7. Domingos, P.M.: A few useful things to know about machine learning. Communications fo the ACM **55**(10), 78–87 (2012)
8. Gopalan, R., Li, R., Chellappa, R.: Domain adaptation for object recognition: An unsupervised approach. In: 2011 International Conference on Computer Vision. pp. 999–1006. IEEE (2011)
9. Haghverdi, L., Lun, A.T., Morgan, M.D., Marioni, J.C.: Batch effects in single-cell rna-sequencing data are corrected by matching mutual nearest neighbors. Nature Biotechnology **36**(5), 421 (2018)
10. Hrvatin, S., Hochbaum, D.R., Nagy, M.A., Cicconet, M., Robertson, K., Cheadle, L., Zilionis, R., Ratner, A., Borges-Monroy, R., Klein, A.M., et al.: Single-cell

analysis of experience-dependent transcriptomic states in the mouse visual cortex. Nature Neuroscience **21**(1), 120 (2018)
11. Islam, S., Zeisel, A., Joost, S., La Manno, G., Zajac, P., Kasper, M., Lönnerberg, P., Linnarsson, S.: Quantitative single-cell rna-seq with unique molecular identifiers. Nature Methods **11**(2), 163 (2014)
12. Jacobs, R.A.: Increased rates of convergence through learning rate adaptation. Neural Networks **1**(4), 295–307 (1988)
13. Kiselev, V.Y., Yiu, A., Hemberg, M.: scmap: projection of single-cell rna-seq data across data sets. Nature Methods **15**(5), 359 (2018)
14. Kobak, D., Berens, P.: The art of using t-sne for single-cell transcriptomics. bioRxiv p. 453449 (2018)
15. Lee, J.A., Peluffo-Ordóñez, D.H., Verleysen, M.: Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure. Neurocomputing **169**, 246–261 (2015)
16. Linderman, G.C., Rachh, M., Hoskins, J.G., Steinerberger, S., Kluger, Y.: Fast interpolation-based t-sne for improved visualization of single-cell rna-seq data. Nature Methods **16**(3), 243 (2019)
17. van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(Nov), 2579–2605 (2008)
18. Macosko, E.Z., Basu, A., Satija, R., Nemesh, J., Shekhar, K., Goldman, M., Tirosh, I., Bialas, A.R., Kamitaki, N., Martersteck, E.M., et al.: Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. Cell **161**(5), 1202–1214 (2015)
19. McInnes, L., Healy, J., Melville, J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. ArXiv e-prints (Feb 2018)
20. Quionero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N.D.: Dataset shift in machine learning. The MIT Press (2009)
21. Rozenblatt-Rosen, O., Stubbington, M.J., Regev, A., Teichmann, S.A.: The human cell atlas: from vision to reality. Nature News **550**(7677), 451 (2017)
22. Shekhar, K., Lapan, S.W., Whitney, I.E., Tran, N.M., Macosko, E.Z., Kowalczyk, M., Adiconis, X., Levin, J.Z., Nemesh, J., Goldman, M., et al.: Comprehensive classification of retinal bipolar neurons by single-cell transcriptomics. Cell **166**(5), 1308–1323 (2016)
23. Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck III, W.M., Hao, Y., Stoeckius, M., Smibert, P., Satija, R.: Comprehensive integration of single-cell data. Cell (2019)
24. Wattenberg, M., Viégas, F., Johnson, I.: How to use t-sne effectively. Distill **1**(10), e2 (2016)
25. Wolf, F.A., Angerer, P., Theis, F.J.: Scanpy: large-scale single-cell gene expression data analysis. Genome Biology **19**(1), 15 (2018)
26. Xin, Y., Kim, J., Okamoto, H., Ni, M., Wei, Y., Adler, C., Murphy, A.J., Yancopoulos, G.D., Lin, C., Gromada, J.: Rna sequencing of single human islet cells reveals type 2 diabetes genes. Cell Metabolism **24**(4), 608–615 (2016)