AUTOMATED OPTIMAL PARAMETERS FOR T-DISTRIBUTED STOCHASTIC NEIGHBOR EMBEDDING IMPROVE VISUALIZATION AND ALLOW ANALYSIS OF LARGE DATASETS

ANNA C. BELKINA[1,2], CHRISTOPHER O. CICCOLELLA[4], RINA ANNO[5], RICHARD HALPERT[6], JOSEF SPIDLEN[6], JENNIFER E. SNYDER-CAPPIONE[2,3]

[1]Department of Pathology, [2]Flow Cytometry Core Facility and [3]Department of Microbiology, Boston University School of Medicine, Boston, MA; [4]Omiq, Inc, Santa Clara, CA, [5]Department of Mathematics, Kansas State University, Manhattan, KS, [6]FlowJo, LLC, Ashland, OR

CORRESPONDING AUTHOR: ANNA C. BELKINA, M.D., PH.D. BELKINA@BU.EDU

ABSTRACT. In single cell analysis, visualization of high-dimensional data is essential for information extraction and interpretation. t-Distributed Stochastic Neighbor Embedding (t-SNE) is a dimensionality reduction algorithm that facilitates visualization of complex high-dimensional cytometry data as a two-dimensional distribution or 'map'. t-SNE maps can be interrogated by either expert-driven or automated techniques to categorize single cell data into relevant biological populations and discern biologically relevant differences between individual samples. The use of t-SNE for high parameter mass and fluorescent cytometry datasets enables a comprehensive and unbiased view of results as compared to traditional biaxial gating. However, successful t-SNE visualization depends on heuristic titration of multiple parameters, as non-optimal embeddings can carry artifacts that make the map difficult or impossible to interpret. Moreover, standard t-SNE implementations fail to produce clear visualizations of datasets when millions of datapoints are projected on the map, often making this method unusable for larger biological datasets. To overcome current t-SNE limitations, we formulated opt-SNE, an array of automated tools for optimal parameter selection in t-SNE visualization. For optimal and fastest data embedding, opt-SNE utilizes Kullback-Liebler (KL) divergence evaluation in real time by tailoring the early exaggeration stage of t-SNE gradient computation in a dataset-specific manner. Here, we demonstrate that precise timing of early exaggeration and scaling the gradient descent learning rate step to the size of the dataset together dramatically improve computation time and enable high quality visualization of both large cytometry and transcriptomics datasets. Also, our results explain why existing software solutions with hard-coded t-SNE parameters produce poorly resolved and potentially misleading maps of fluorescent and mass cytometry data. In sum, our novel approach to t-SNE enables the required fine-tuning of the algorithm to ensure optimal resolution of t-SNE maps and more precise data interpretation.

Keywords: mass cytometry, flow cytometry, cytometry analysis, t-SNE, machine learning, viSNE, scRNA-seq, dimensionality reduction, data visualization

1. Introduction.

Visual exploration of high-dimensional data is imperative for the comprehensive analysis of single cell datasets. With state-of-art technologies, fluorescence, mass and sequencing-based cytometric data analysis requires tools able to reveal the combinations of proteomic or transcriptomic markers that define the cell phenotype in a multiparametric dataset. Traditional biaxial presentation and gating has been the standard analysis method for parsing cytometry data, however, with the advent of modern high-parameter era, tools that can clearly present multidimensional data became

essential. Therefore, most of the techniques that belong to the family of dimensionality reduction (DR) methods have been borrowed and tested on cytometry data with variable success. A popular DR tool is principal component analysis (PCA); however, PCA generates a low-dimensional representation of data with a linear mapping matrix and is therefore mostly unsuitable for cytometry data visualization as it cannot faithfully present the non-linear relationships in the data.

t-Distributed Stochastic Neighbor Embedding (t-SNE) is a state-of-art dimensionality reduction algorithm for non-linear data representation (van der Maaten and Hinton 2008; Wattenberg 2016) that allows visualization of complex high-dimensional cytometry data as a two-dimensional distribution or "map", where each 'island' roughly corresponds to a population of cells defined by cytometric signature that existed in high-dimensional space and was preserved and flattened in t-SNE space (Amir el et al. 2013). These maps can be interrogated by human-guided or automated techniques to categorize single cell data into relevant biological populations and visualize important differences between samples.

t-SNE was developed for a broad range of data analysis tasks that involve machine learning and has been adopted for cytometry use. Once the data structure has been revealed and visualized through t-SNE, the quantification and interpretation of the data can be performed by analyzing t-SNE coordinates with clustering algorithms such as DensVM (Becher et al. 2014; Chen et al. 2016). Cytometry clustering algorithms that directly interrogate high-dimensional data, such as FlowSOM (Van Gassen et al. 2015) and PhenoGraph (Levine et al. 2015), employ t-SNE maps to present annotated clusters to the viewer.

Unfortunately, t-SNE is that it does not specifically address the challenge of large cytometry datasets, and the results of running readily available t-SNE implementations on large datasets are often unsatisfactory. Having been developed and benchmarked on relatively small (<50,000 datapoints) datasets, t-SNE does not satisfy the needs of cytometry data, when hundreds of thousands or millions of cytometric events are collected for analysis. Firstly, t-SNE learns the embedding non-parametrically, and hence new pieces of data cannot be added to an existing analysis. This necessitates the whole dataset to be analyzed within one embedding computation. When the full dataset is comprised of multiple samples, each representing a subject in a large cohort or an independent experimental condition, retaining statistically significant representation of small subpopulations in each sample requires inflating the size of the dataset (Donnenberg and Donnenberg 2007). Second, even when the experimental setup and reagent selection permits detection of rare populations, downsampling the data risks preventing these scarce subsets from being identified.

These limitations are not satisfactorily addressed with existing practices for applying t-SNE. Not only are large datasets computationally expensive to analyze, but also the resulting t-SNE maps provide poor visualization and incomprehensive representation of high-dimensional data.

As a result, when being not able to produce quality t-SNE visualizations from the large datasets, the researchers often resort to either downsampling their data up to the very limit of detection of rare populations (DiGiuseppe et al. 2015) or to exporting specific populations from their dataset, thus compromising the 'unbiased' data analysis approach (Lin et al. 2015).

Despite the fact that t-SNE has already been widely adopted by the scientific community, to our knowledge no rigorous theoretical or empirical testing of the t-SNE adaptation for cytometry applications has been performed to date. In 2013, Amir et al reported the use of t-SNE (or viSNE, as it was renamed (Amir el et al. 2013)) on mass cytometry data; since then, t-SNE has been

implemented in the majority of commercial and open-source platforms for cytometry analysis (FlowJo, Cytobank, FCSExpress, cytofkit, etc). In most implementations, few or no adjustments were made to the Barnes-Hut t-SNE algorithm. This includes retaining the default and hard-coded parameter settings that were originally tested and optimized with non-cytometry datasets like CIFAR (image dataset) or MNIST (handwritten digits). This lack of rigorous analysis of the full potential and applicability of t-SNE to cytometry data is the primary motivation for this work.

In this paper, we propose a method to automatically find optimal t-SNE parameters via fine-tuning of the early exaggeration stage of t-SNE embedding in real time. We call our approach opt-SNE, for *opt*imal t-SNE algorithm. We find that our adjustments can tremendously shorten the number of iterations required to obtain a visualization of superior quality. Our approach also eliminates the need for trial-and-error runs intended to empirically find the optimal selection of t-SNE parameters, which can potentially save hundreds of hours of computation time.

2. Materials and methods

2.1. Datasets

All datasets used in the study are summarized in Table 1.

Dataset	Data type	Number of parameters	References
Mass41parameter	Mass cytometry	41 total, 14 used for t-SNE	(Bendall et al. 2011)
Flow18parameter	Flow cytometry	18 total, 11 used for t-SNE	primary data available upon request
Flow20M	Flow cytometry	18 total, 15 used for t-SNE	primary data available upon request
10X	scRNAseq	20 PCA vectors	https://support.10xgenomics.com/single-cell-gene-expression/datasets
OMIP-37	Flow cytometry	18 total, 10 used for t-SNE	(Belkina and Snyder-Cappione 2017)

2.2. Primary samples

Flow18parameter data were collected as described (Belkina and Snyder-Cappione 2017) with minor modifications of the flow cytometry reagent list.

2.3. Data pre-processing.

For the mass41parameter dataset, we used singlet events from five files recorded from replicate conditions of mouse bone marrow samples. For Flow18parameter dataset and OMIP37 dataset, flow cytometry samples from a single donor per file were recorded. Data were digitally concatenated when applicable and a randomly downsampled file of 1,000,000 events was created and used for analyses. Flow cytometry data were compensated with acquisition-defined

compensation matrices. Prior to t-SNE analysis, all cytometry data were transformed using asinh or biexponential transformation.
For Flow20M dataset, 18-parameter flow cytometry PBMC data from 27 subjects were compensated with acquisition-defined compensation matrix and concatenated. Light scatter parameters were log-transformed and fluorescence parameters were asinh-transformed.

2.4. Data analysis.

A desktop C++ Barnes-Hut implementation of t-SNE for Mac OS was used for most t-SNE analyses (Van Der Maaten 2014). All datasets were embedded in 2D space. Original code was edited to allow user input for early exaggeration stop iteration, perplexity, total number of iterations, early exaggeration factor value, and learning rate value. KLD value and t-SNE coordinates were reported at each generation or as frequently as requested. To allow generation of visually comparable t-SNE maps, random seed was not used unless specified. For cross-validation and to benchmark against standard platforms, we utilized cloud-based Cytobank, FlowJo V10.3-10.5 and FlowJo V9.9.6. Cytobank and FlowJo platforms were used to generate FCS files from tabular data. Logs of t-SNE runs were batch-processed in VBA and analyzed with GraphPad Prism 7. Expert-guided (manual) analysis of cell populations was performed in FlowJo 10.3-10.5 as described previously for specific datasets (Belkina and Snyder-Cappione 2017; Bendall et al. 2011) or as explained below and used for map annotations.

For scRNAseq analysis, we used SeqGeq 1.3 package. We used PCA projections from 10X Genomics dataset to calculate the t-SNE embedding and annotated it using marker genes for major cell types.

The quality of the embeddings was assessed by a nearest neighbor classifier strategy similar to that found in previous reports (Jian et al. 2016; Van Der Maaten 2014; van der Maaten and Hinton 2008). Briefly, for each observation, the k nearest neighbors (by Euclidean distance) were calculated using the 2D coordinates of the t-SNE map and the class assigned by expert gating was compared to the most common class of its k neighborhood. The rate of correct matches was tallied and represented as the overall nearest neighbor accuracy. The accuracy was also calculated on a per-class basis; different values for k (1,10,20,30,40,50) were reported.

3. Results and discussion.

3.1. An overview of t-SNE setup for cytometry.

As described in detail in van der Maaten 2014, t-SNE computes low-dimensional coordinates of high-dimensional data so that similar data points that are close to one another in the raw data space are mapped close in the reduced space, and dissimilar points are mapped at longer distances. First, t-SNE models the probabilities as a Gaussian distribution around each data point in the high-dimensional space, and models the target distribution of pairwise similarities in the lower-dimensional space using Cauchy distribution (Student t-distribution with 1 degree of freedom). Then the Kullback-Liebler Divergence (KLD) between the distributions is iteratively minimized via gradient descent. The gradient computation is essentially an N-body simulation problem with attractive forces (approximated to nearest neighbors using vantage-point trees) pulling similar

points together and repulsive forces (approximated at each iteration using the Barnes-Hut algorithm) pushing dissimilar points apart.

An important part of gradient descent is "early exaggeration" (EE) that was proposed by van der Maaten and Hinton (2008) to battle the "overcrowding" artifact of embedding. With early exaggeration, all probabilities modeling distances in high-dimensional space are multiplied by a factor (early exaggeration factor, EEF) for the duration of the first (typically 250 or 25% of the total number of) iterations. EE coerces data to form tight and widely separated clusters in the map and is considered to enable the map to find a better global structure.

3.2. The standard t-SNE configuration fails to visualize large datasets

Multiple software platforms allow t-SNE analysis of cytometry data, including commonly used cytometry analysis desktop packages (FlowJo, FCS Express), and a cloud-based analysis platform Cytobank. Implementations of t-SNE are available as open-source packages in popular programming languages such as R (rtsne) and Python (sci-kit learn). Most of implementations wrap or re-write original C++ code of t-SNE (van der Maaten, 2014) and produce identical analysis results. We opted to use the C++ code and customize it to implement parameter adjustments discussed below, and we also intergerated equivalent adjustments in the FlowJo and SeqGeq implementations of t-SNE.

The t-SNE algorithm can be guided by a set of parameters that finely tune certain aspects of the t-SNE run (Wattenberg 2016). However, cytometry software solutions often make those parameters either inaccessible or severely restrained to provide 'one-size-fits-all' solution for t-SNE setup. Although each platform has a unique combination of possible adjustments, most of them allow changes to the number of iterations and to the perplexity (a soft measure for the number of nearest neighbors considered for each data point).

For this study, we used two large datasets, one fluorescent (18-parameter) and one mass cytometry (41-parameter), each containing 1 million events in total. We opted not to employ popular datasets that are used for algorithm testing in cytometry because most of them were too small for our needs and rarely exceeded 500,000 events (Weber and Robinson 2016).

Datasets larger than approximately 1-2 $x10^5$ events are generally observed to suffer from dramatic reduction in t-SNE map quality. As such, they are usually downsampled prior to analysis. However, we hypothesized that the resolution of t-SNE maps comprised of higher event counts could be dramatically improved with fine-tuning of t-SNE parameters.

Empirically, cytometrists have observed that increasing the number of iterations results in better quality maps. We tested that by running both datasets at default 1000 iterations per run and comparing them with "extended" 3000 iteration computation (Fig. 1A, B). To aid visualization, here and further on we overlay the maps with color-coded populations derived from expert-driven ('manual') gating of the same dataset which serves here as a ground-truth basis for data classification. As expected, 1000-iteration runs produced maps with poor visualization (overall 1-NN accuracy of embedding was 65% and as low as 18% for certain populations). Specifically, we observed massive overlaps and random fragmentation of populations. In contrary, 3000-iterations runs resulted in maps with better defined "islands", isolated populations and no random fragmentations (overall 1-NN accuracy of embedding 96%; detailed results of accuracy evaluations for all runs are presented in Suppl. Table 1).

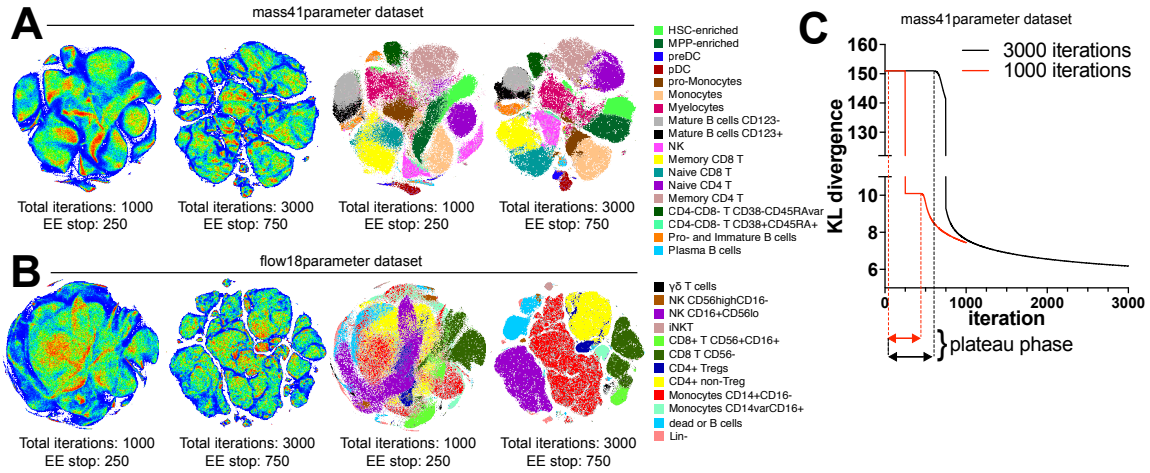

Figure 1. Performance of standard t-SNE implementation for cytometry data visualization. Comparison of standard (1000 iterations) and extended (3000 iterations) embeddings of mass cytometry (A) or flow cytometry (B) datasets. Data are shown as density plots (left) or color-coded classification overlays. C.KLD change over iteration time.
A representative example of multiple runs with varying seed values is presented.

To find the cause of this difference, we examined the behavior of KL divergence (KLD) over the duration of t-SNE embedding. We made several observations. First, as shown on Fig 1C, KLD value is inflated during the EE in the beginning of the "default" t-SNE run (red line). In the "extended" run with 3000 iterations (black line), the EE lasts over 750 iterations since most cytometry platforms have EE scaled to 25% of total iteration number (although some use original hard-coded value of 250 iterations). Second, KLD does not immediately minimize at the start of the EE; instead, the graph of KLD over time is a plateau that is followed by a curve that captures the incremental decrease of KLD indicating the gradient descent. In the "default" run, the plateau was interrupted when the EE was stopped and KLD dropped, and then continued with non-exaggerated value of KLD.

3.3. KLD plateau phase resolves global cluster structure in t-SNE visualization

According to the authors, EE was introduced as a "trick" to improve resolution of the global structure of the data visualization that would not converge to separated clusters otherwise (van der Maaten and Hinton 2008). Since the poor-quality cytometry t-SNE maps as shown in Fig. 1A, B suffer from same problem, we hypothesized that with conventional platforms, by increasing total iterations the analyst may inadvertently increase the number of EE iterations and that may beneficially influence visualization quality. To test this hypothesis, we compared multiple 3000-iteration runs that differed in timing of the EE stop (Fig 2A). To assess the effect of early exaggeration on map optimization, we sampled each run at the iteration when the EE stops, and assessed the effects of our perturbations on both mass cytometry (Fig 2A) and flow cytometry (Fig 2B) data visualization.

We found impressive differences in map quality between shorter and longer EE runs. Although the map after EE200/total3000 iterations appears visually more pleasing than EE250/total1000 (Fig 1A, B) and could be considered a successful visualization, ground-truth labeling indicates that it suffered from cluster fragmentation. When cluster fragments were plotted on a biaxial plot against

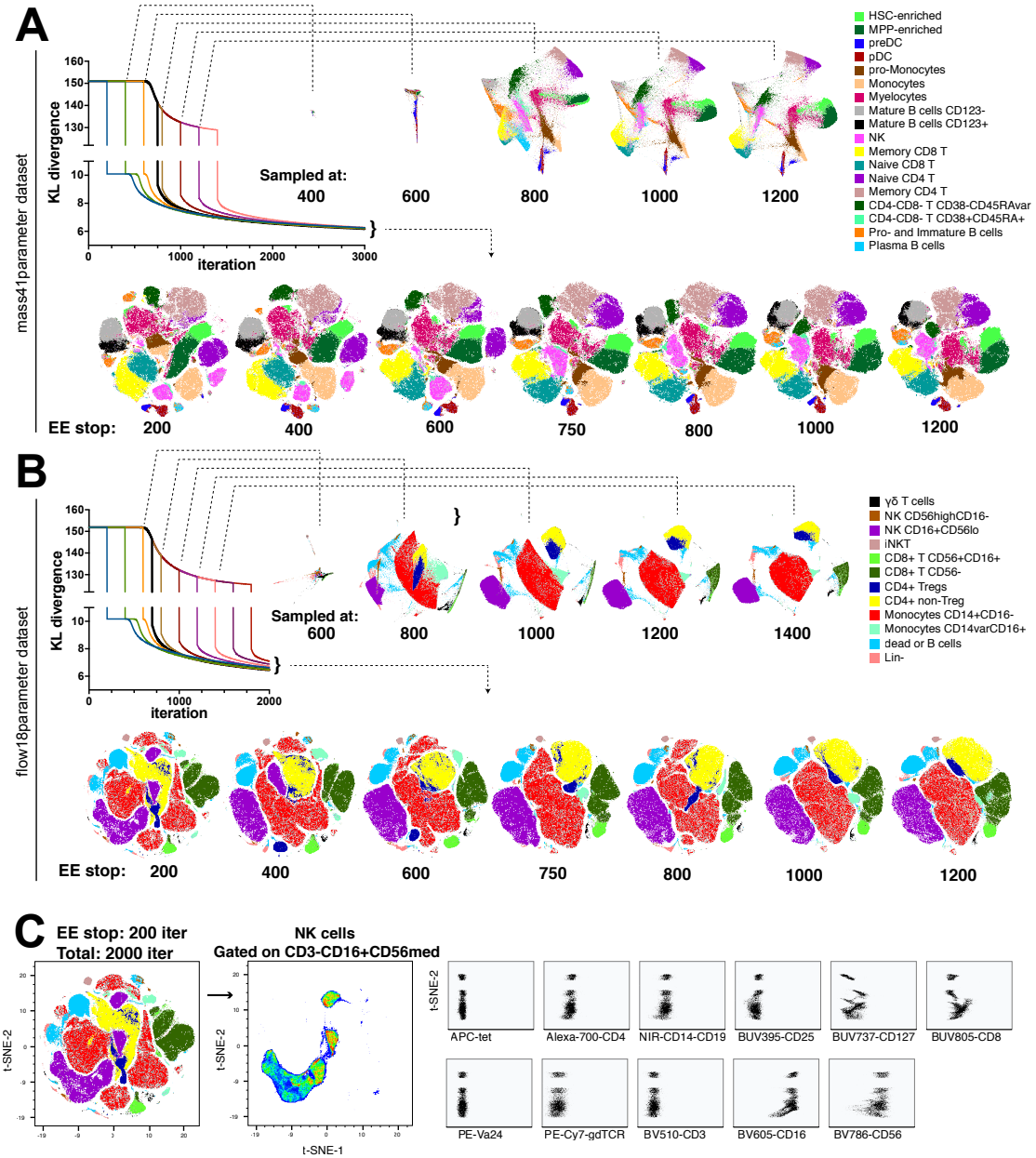

Figure 2. Effect of EE plateau duration on t-SNE visualization. EE was stopped after varying number of iterations and embedding output was sampled at several timepoints for mass cytometry (A) and fluorescent cytometry (B) data visualization. Data plots are shown as color-coded classification overlays and KLD change over iteration time. is reported for each perturbation. C. Clusters corresponding to CD3-CD16+CD56med NK cells were subsetted from the dataset and assessed as biaxial plots of different parameters plotted versus t-SNE-2 axis.

parameters that were used in t-SNE dimension reduction, we were not able to identify parameters that immediately contributed to their fragmentation (Fig. 2C).

Conversely, tight clusters that form at the end of the plateau remain mostly unchanged until the EE continues (Fig 2 A, B). The KLD minimization in that case could be explained by the gradual shrinking of the 2D space (data not shown). Once EE is removed, the attractive forces within each cluster are weakened and the local structure of the data is fully resolved within each cluster. Overall,

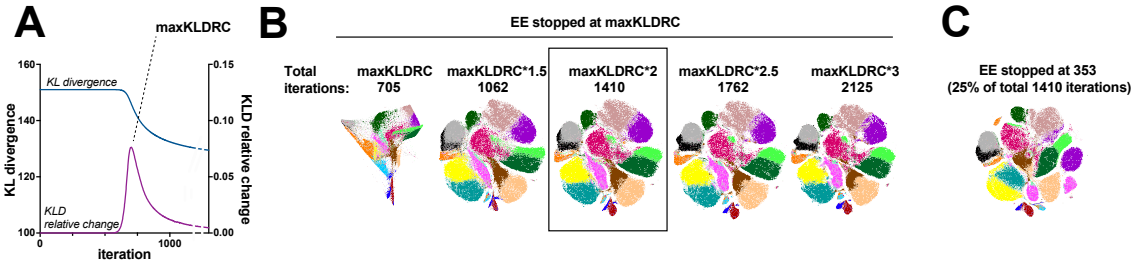

Figure 3. Early exaggeration plateau ensures optimal quality of visualization. A, KLDand KLD relative change plotted against iteration time. B. Mass cytometry data visualiztions generated with varying duration of post-EE iteration time. C. Mass cytometry data visualization generated with default (25%) duration of EE stage. A representative example of multiple runs with varying seed values is presented.

these observations suggest that the EE stage of gradient descent is essential for clusterization of datapoints while the non-exaggerated descent results in resolution of local structures.

3.4. Stopping early exaggeration after the plateau phase produces maps of optimal quality

We have demonstrated that when the EE is too short, cell clusters continue being resolved simultaneously with local structure of each cluster being unfolded, which leads to fragmented, overlapped or deformed "islands" in the resulting map. This conclusion allows an equation to be construction to find optimal early exaggeration timing. We track the relative rate of KLD change ($KLDRC_N = 100\% *((KLD_{N-1} – KLD_N)/ KLD_{N-1})$ where N is the iteration number) and locate the local maximum (maxKLDRC) (Fig. 3A). Since KLD is computed at each iteration, the maxKLDRC 'sensor' can be easily added to the algorithm programmatically and would stop EE at the next iteration past maxKLDRC. For mass41parameter dataset of 1M datapoints, the maxKLDRC was detected at iteration 705. We ran t-SNE with EE stop at iteration 706 and sampled map development at 706, 1.5x706, 2x706, 2.5x706 and 3x706 iterations (Fig 3B). As expected, at maxKLDRC iteration the map contained the primordial clusters only; it was fully shaped at 2 x max KLDRC and there was no visible improvement in map quality past that step and the visualization was very similar to EE750/3000 map at Fig. 1A). As expected, when compared to the map ran with 'default' settings of EE taking 25% of the run but running the same number of iterations (1410), the EE-triggered t-SNE produced superior results within the same computation time, and it also eliminated extensive trial-and-error calibration of t-SNE parameters (Fig. 3C). We propose a conservative approach to finalize the embedding when KLDRC < KLD/10,000. Alternatively, t-SNE projection output can be evaluated in real time to justify the termination of embedding.

3.5. Moderate adjustments of EE factor and perplexity do not impact visualization

Once we found EE to be crucial for map optimization, we decided to examine if the value of EE factor α can also be tuned to improve the results of t-SNE. We have altered the C++ t-SNE code since in the original Barnes-Hut C++ t-SNE implementation the EE factor is hardcoded, and all our previous results were obtained with default value of $\alpha = 12$. We chose the parameters defined above ($\alpha = 12$, EE = 706 iterations, 1410 iterations total) as our baseline for comparison since they provided optimal balance of map quality/computation time. First, we tested how the optimization would proceed without EE ($\alpha = 0$). We expected the run to fail or produce extremely crowded results as explained in the original t-SNE report (van der Maaten and Hinton 2008), however, we

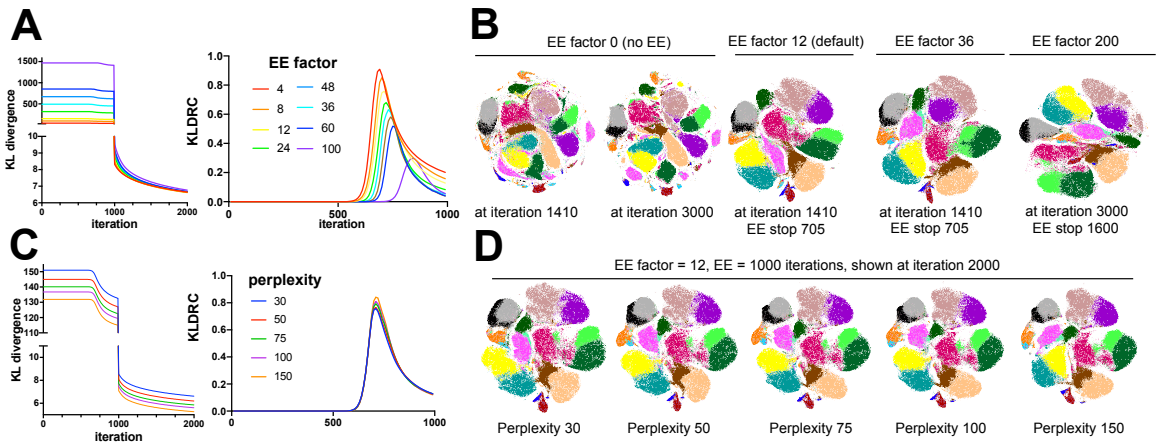

Figure 4. Effects of perplexity and EE factor adjustments on t-SNE visualization of cytometry data. A, B. KLD, KLDRC and t-SNE biaxial plots generated with varying EE factor values. C, D. KLD, KLDRC and t-SNE biaxial plots generated with varying perplexity. A representative example of multiple runs with varying seed values is presented.

did not see much overlap in cluster positioning, probably due to the fact that we ran a substantial number of map iterations (Fig 4B). Nevertheless, the resulting map showed a lot of fragmentation proving to be an extreme case of interrupted plateau phase. Even when run for as many as 3000 iterations, the fragmentation could not be remedied, once again demonstrating the necessity of EE.

As expected, higher values of α lead to much higher KLD during EE, however, the KLD of resulting maps at 2000 iterations when α is varied between 4 and 60 was comparable (Fig. 4A). Larger α prolongs the plateau phase and become detrimental for KLD values when over 100. Visually α = 200 results in a distorted map with smaller populations lost. We suggest that for cytometry applications α parameter may remain unchanged and set to 12, as suggested in van der Maaten 2014, or reverted to α = 4, as originally proposed in van der Maaten and Hinton (2008) since based on our experience, any value between 4 and 20 leads to comparable results.

Increased perplexity has been proposed as an intuitively beneficial method for visualization improvement since it translates to larger number of considered nearest neighbors and hence more accurate approximation of attractive forces, while decreased perplexity can completely fail the visualization (Wattenberg 2016). KLD values of runs with varying perplexity cannot be compared since KLD value is related to perplexity, but it does not appear that increased perplexity results in faster resolution of clusters (Fig 4C) or cleaner data visualization (Fig 4D). However, while changing α does not affect t-SNE computation time, perplexity is linearly related to the time and memory required to create the embedding (data not shown). Although we and others have seen some benefits of perplexity increase for map quality in otherwise suboptimal t-SNE runs, optimizing the EE step as described above and further in this work does not leave much space for improvement with perplexity tuning (Fig 4E).

3.6. Learning step size is a key parameter to ensure t-SNE visualization of large datasets

The step size in t-SNE gradient descent is updated at each iteration per Jacobs adaptive learning rate (Jacobs 1988). This method increases the learning rate in directions in which the gradient is stable. A conservative initial value of 200 is hard-coded into most platforms. We hypothesized that larger datasets may stay longer on KLD plateau due to the number of iterations it takes to build up

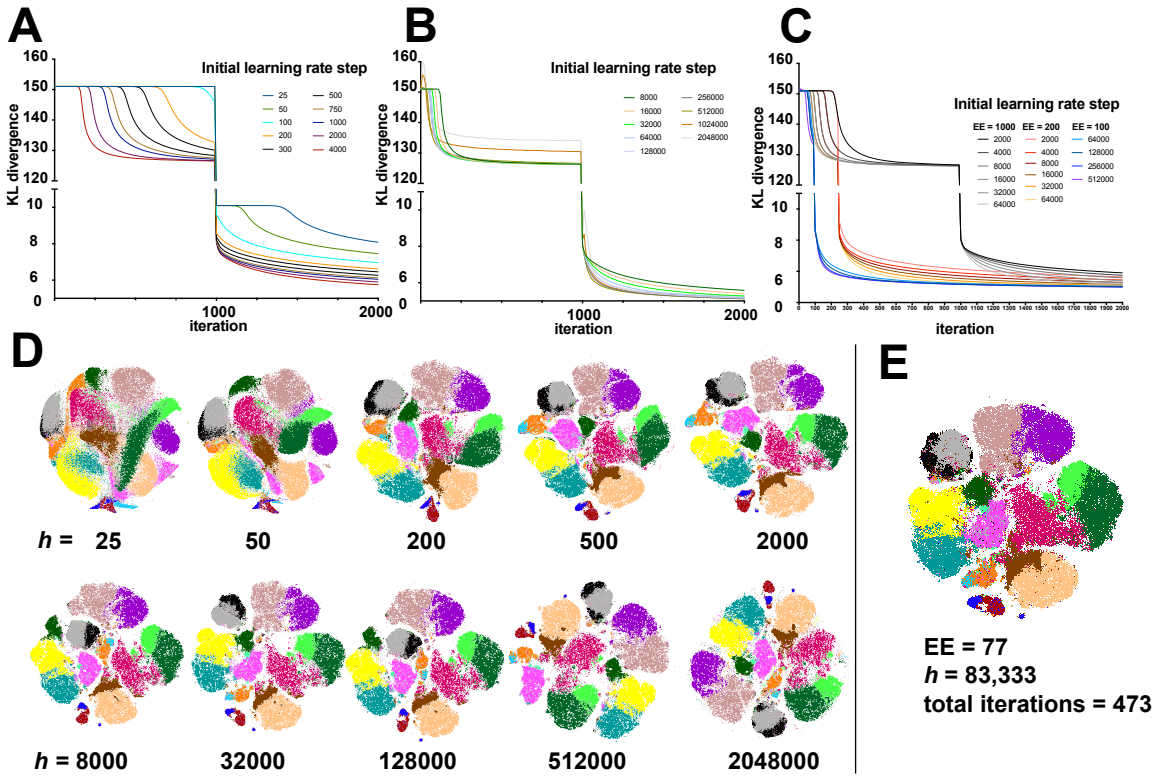

Figure 4. Learning step size optimization for t-SNE visualization of large datasets. KLD reports generated with initial learning rate step size values. A, EE = 1000 iterations, learning rate step = 25-4000; B, EE = 1000 iterations, learning rate step = 8K-2048K; C, EE = 100-1000 iterations, learning rate step = 2K - 512K. D, representative t-SNE plots of embeddings graphed on A. E, t-SNE plot of an optimal embedding.

a sufficient learning rate step size. To evaluate this possibility, we titrated the step size *h* while observing the KLD with fixed EE=1000 iterations in Mass41parameter dataset. In agreement with our hypothesis, *h* = 25 and *h* = 50 runs failed to resolve from KLD plateau within 1000 iterations of EE (Fig 5A) and *h* = 200 finished the plateau in ~700 iterations as previously shown. With further increase in *h,* we found that not only it takes progressively less iterations to complete the plateau, but also the final KLD of the maps scored at lower values. KLD is directly related to the quality of visualization since it reflects the faithfulness of representation of high-dimensional data in t-SNE space, therefore, lower KLD values indicate superior visualization quality.

With higher *h* values, we continued to see improvement in plateau duration and KLD values until *h* ~ 64,000, a value that is drastically far from the "default" *h* = 200 setting (note that in most platforms, *h* is restricted to ranges below 3,000) (Fig. 5B). At *h* ~ 256,000 we observed irregular peaks in KLD graph indicating that the visualization starts to fail calculating stable gradient. However, using lower values of *h* we were able to converge the map with lowest KLD values at the fraction of time when limiting the EE step to 200 and even 100 iterations despite the 1M size of the dataset (Fig. 5C). Visual inspection of the embedded maps over the range of *h* values agrees with KLD values (Fig 5D).

In a recent theoretical publication, Linderman and Steinerberger (Linderman and Steinerberger 2017) prove that generally t-SNE embedding will not converge if a product of EE factor α (which is we kept at a default value of 12) and of learning rate step size *h* is larger than the number of datapoints *n (i.e. if ah>n).* Since we employ adaptive learning rate, our selection of initial *h* value

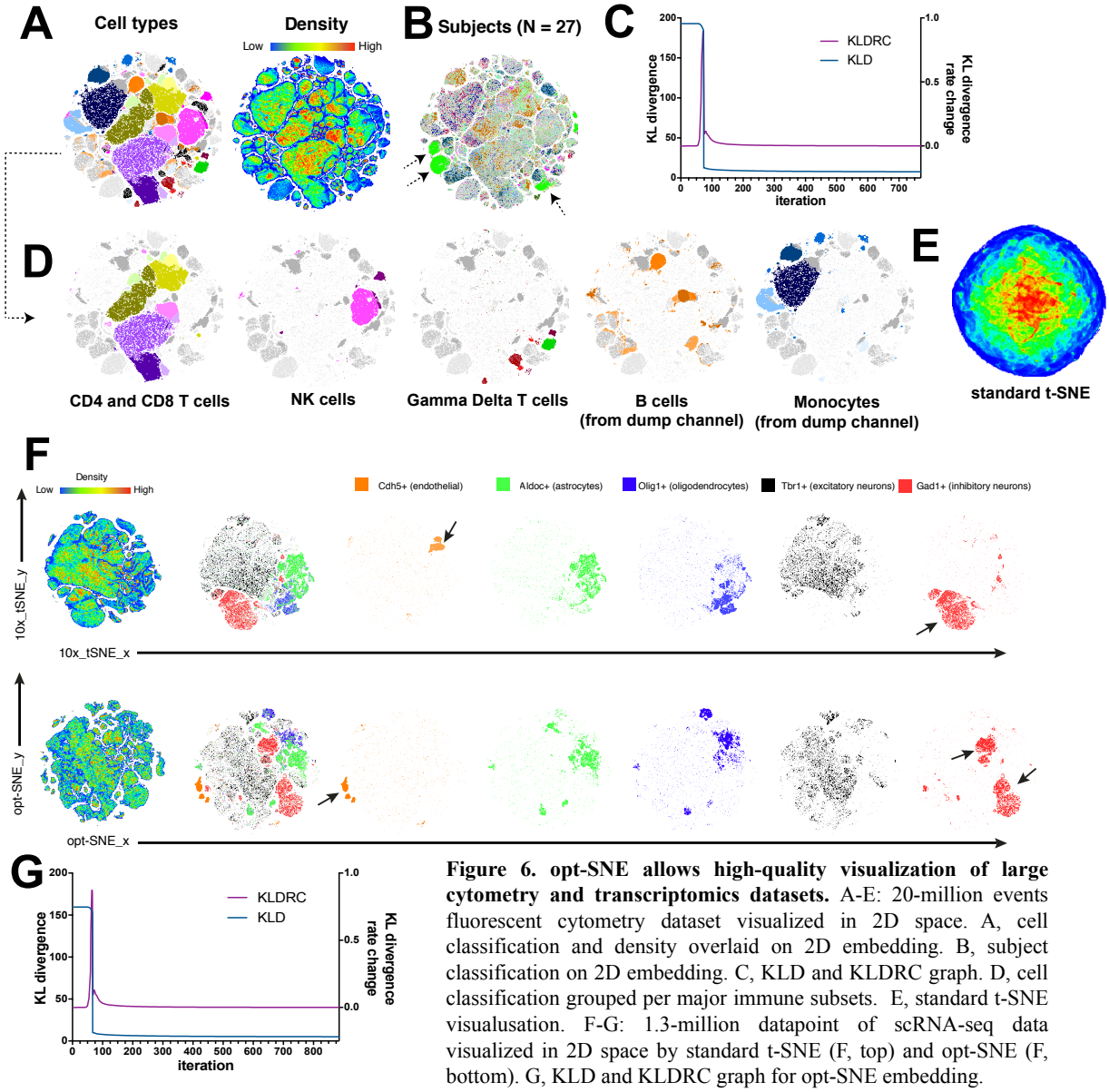

Figure 6. opt-SNE allows high-quality visualization of large cytometry and transcriptomics datasets. A-E: 20-million events fluorescent cytometry dataset visualized in 2D space. A, cell classification and density overlaid on 2D embedding. B, subject classification on 2D embedding. C, KLD and KLDRC graph. D, cell classification grouped per major immune subsets. E, standard t-SNE visualusation. F-G: 1.3-million datapoint of scRNA-seq data visualized in 2D space by standard t-SNE (F, top) and opt-SNE (F, bottom). G, KLD and KLDRC graph for opt-SNE embedding.

is more forgiving, however, in our experiments we found the optimal settings of h to be close to $h = n / 12$. Therefore, we propose to initiate the gradient descent with $h = n / \alpha$ to create optimal t-SNE visualization (Fig 4E).

3.7. Opt-SNE allows successful embedding of large datasets

We implemented all proposed techniques including dataset-specific automated early exaggeration step controlled by KLDRC sensor and calculated optimal learning rate step size, and KLDRC-driven embedding termination, in a single workflow that we labeled ‘opt-SNE’, or ‘optimized t-Stochastic neighbor embedding’. To test opt-SNE performance, we chose a >20,000,000 event fluorescent cytometry dataset concatenated from 27 individual PBMC samples stained with a variation of the OMIP-037 fluorescent cytometry panel that allow assessment of naïve and memory T cell subsets and deeper gamma delta T cell analysis (Fig 6A). The embedding completed in 770

iterations with only 73 iterations required to pass the EE step (with h = 1,537,700) and resulted in clear separation of cell clusters as evaluated by marker annotation (Fig 6C). Embedding was also annotated per sample to control for batch effects in clustering. The majority of populations appear to be evenly represented in all subjects with the exceptions of several populations that contained sample-unique debris features (Fig 6B, arrows). A detailed breakdown of identified populations is presented in Fig 6D that shows subsets of CD4+ and CD8+ T cells, NK cells, gamma delta T cells, B cells and monocytes. Interestingly, B cells and monocytes were labeled together with dead cell marker in a dump channel in this panel and therefore cannot be gated accurately via traditional analysis. However, t-SNE has been able to identify them in high-dimensional space by the combination of other markers and light scatter characteristics and cluster them into populations minimally mixed with dead cells. Remarkably, running a standard t-SNE algorithm over several thousands of iterations completely failed to reveal the structure of the multi-million flow cytometry data (Fig 6E).

To test the applicability of opt-SNE for applications beyond flow and mass cytometry, we analyzed the 1.3M cell single-cell RNA-seq dataset of mouse embryonic brain cells published by 10X/Chromium. We used pre-calculated PCA projections reported by 10X to generate our opt-SNE maps that we compare with 10X t-SNE embedding (Fig 6F). In their visualization, 10X used EE = 1000/total 4000 iterations of standard t-SNE while we used opt-SNE settings (h = 97,959, EE = 66/total 885 iterations, Fig 5G). Clean annotation of non-immune transcriptomics analysis is not as straightforward as with cytometry data, since fewer scRNAseq markers can be interpreted for population identification. We applied the same labeling approach as suggested by 10X, however, opt-SNE embedding allowed us to resolve several cell clusters that were not clearly separated in 10X visualization (Fig. 6F, arrows) despite suggestive transcription profiles (data not shown). Therefore, opt-SNE allowed equivalent or superior resolution of single cell transcriptomics data as with standard t-SNE but with ~5x smaller iteration time.

4. Discussion

Similarly to other types of biological data, cytometry data carries a structure that is difficult to project because of its mixed nature often comprising cluster-like, manifold-like and/or hierarchical components (Finn et al. 2008; Mazza et al. 2018). In this paper we propose several techniques that are essential for optimal t-SNE data projection and are focused on fine-tuning of the early exaggeration stage of t-SNE embedding. EE was designed to ensure cluster formation on a 2D plane (van der Maaten and Hinton 2008). We propose an efficient measure to ensure that the cluster-like global structure of the data is fully revealed during the EE stage by monitoring the KLD output of the embedding in real time. However, it is possible that prolonged amplification of attractive forces that drives tight cluster formation in EE is detrimental for manifold-like local data structure that is often represented by so called ‘continuously-expressed’ markers. These molecules include major hallmarks of cell functional state and guide cell activation and exhaustion, as well as indicate disease phenotypes. The non-exaggerated stage of t-SNE allows to reveal local data structures (van der Maaten and Hinton 2008) if they are preserved during the EE stage. Therefore, cytometry data analysis would be missing valuable information if we reduced t-SNE applicability to clusterization only, especially since other techniques would perform that task better and faster. However, some workflows call for t-SNE pre-processing to facilitate extraction of cluster features

from multidimensional data (Becher et al. 2014; Diggins et al. 2015). In those cases, it may be helpful to adapt opt-SNE toolkit to terminate the embedding calculation immediately at the EE stop iteration and assess local structure after a clustering algorithm has identified populations in the t-SNE space.

Conversely, t-SNE is sometimes used to structure cytometry data in which the cluster-like structure is not prominent simply because of the tool's accessibility. It is advisable to note that certain data structures cannot be revealed with t-SNE (Im et al 2018) (Amid and Warmuth 2018) and, more generally, that features identified from t-SNE embedding must be verified with alternative methods.

Im and colleagues also suggest that if a continuous manifold structure exists in the data, large perplexity values may cause artificial breaks (overclustering) in the data. Perplexity values commonly used in cytometry analysis are on the lower end of suggested range for efficient clustering since it is often advised to scale the number of nearest neighbors to the average cluster size (Cao and Wang 2017), however, that might facilitate feature preservation for markers whose expression is not bimodally distributed.

Visual exploration of data drives hypothesis formation and human serendipity, therefore t-SNE is an extremely valuable tool for data comprehension. It is often used to facilitate data perception when hypothesis generation is automated by robust computational methods (Butler et al. 2018; Van Gassen et al. 2015). It is also valuable for quality assessment of data, when abnormal clustering could be traced back to sample preparation, data acquisition and preprocessing artifacts (Mazza et al. 2018). For these applications, batch embedding of multiple experimental points is essential for sample comparison and can only be enabled when t-SNE accommodates large datasets. Several approaches to large scale t-SNE have been recently reported including LargeVis (Jian et al. 2016), net-SNE (Cho et al. 2018) and HSNE (van Unen et al. 2017), however, these improved methods often require considerable computational resources (for instance, LargeVis results were generated on a 512Gb RAM, 32 core station). With proper RAM allocation and multicore adaptations, Barnes-Hut t-SNE can be routinely run on a data analysis station in an immunology lab. In this work, we have not focused on computation time and efficiency since we were benchmarking the algorithm against itself and our improvements in computation time occurred due to less iterations required to complete the data embedding. However, all our analyses were performed using several personal computers with the exception of 20M embedding that required ~60Gb RAM at its peak. Nevertheless, we expect our conclusions to be applicable for existing or future adaptations of t-SNE even if they utilize alternative methods of computation (Chan et al. 2018; Linderman et al. 2017) as long as they retain the core principles of t-SNE embedding.

In cytometry, t-SNE was first introduced as a tool to visualize CyTOF data as fluorescence-based high-parameter datasets were less common. With recent advances in instrument and reagent availability, flow cytometry datasets with > 20-25 parameters are quickly becoming prevalent and even standard in the field, while the data assessment tools available for a general userbase are still lacking. Recently, DNA-barcoded antibodies have been used to allow simultaneous protein-epitope and transcriptome measurements in single cells (Stoeckius et al. 2017) thus expanding the repertoire of traditional cytometry methods that could employ t-SNE as a staple method of data visualization and presentation. We believe that opt-SNE is a simple to

implement and powerful optimization that removes some of the major limitations in t-SNE use in cytometry and can potentiate multiple data-driven findings in single cell research.

Acknowledgements

The authors would like to thank Yvan Saeys, El-Ad Amir, Gary Kazantsev, Jonathan Irish, and Katherine Drake for helpful discussions, and Geoff Kraker for technical assistance. We thank Robert Balderas and Keefe Chee from BD Biosciences and Sean Bendall from Stanford University for sharing data and Brian Tilton from BUSM Flow Cytometry Core Facility for assistance with data collection. Anna C. Belkina is an ISAC (International Society for Advancement of Cytometry) SRL Emerging Leader 2015-2019 and thanks ISAC organization and members for continuous support and encouragement. Josef Spidlen is an ISAC Marylou Ingram Scholar .

Data and software Availability

This preprint will be updated with repository information for C++ implementation of opt-SNE. Cytometry datasets are immediately available upon request.

To facilitate availability to flow cytometry and scRNA-seq data analysts, opt-SNE has been incorporated into FlowJo version ≥ 10.5.2 and SeqGeq version ≥ 1.4. This option was considered experimental and therefore hidden, but users can enable it by adding `<DRPlatform showAutoLearning="1" />` to the FlowJo10.prefs (or SeqGeq.prefs) XML file.

References

Amid E, Warmuth MK (2018) A more globally accurate dimensionality reduction method using triplets. arXiv:180300854

Amir el AD, Davis KL, Tadmor MD, et al. (2013) viSNE enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia. Nat Biotechnol 31(6):545-52 doi:10.1038/nbt.2594

Becher B, Schlitzer A, Chen J, et al. (2014) High-dimensional analysis of the murine myeloid cell system. Nat Immunol 15(12):1181-1189 doi:10.1038/ni.3006

Belkina AC, Snyder-Cappione JE (2017) OMIP-037: 16-color panel to measure inhibitory receptor signatures from multiple human immune cell subsets. Cytometry A 91(2):175-179 doi:10.1002/cyto.a.22983

Bendall SC, Simonds EF, Qiu P, et al. (2011) Single-cell mass cytometry of differential immune and drug responses across a human hematopoietic continuum. Science (New York, NY) 332(6030):687-96 doi:10.1126/science.1198704

Butler A, Hoffman P, Smibert P, Papalexi E, Satija R (2018) Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nature Biotechnology 36:411 doi:10.1038/nbt.4096

Cao Y, Wang L (2017) Automatic Selection of t-SNE Perplexity. arXiv:170803229

Chan DM, Rao R, Huang F, Canny JF (2018) t-SNE-CUDA: GPU-Accelerated t-SNE and its Applications to Modern Data. arXiv:180711824

Chen H, Lau MC, Wong MT, Newell EW, Poidinger M, Chen J (2016) Cytofkit: A Bioconductor Package for an Integrated Mass Cytometry Data Analysis Pipeline. PLoS Comput Biol 12(9):e1005112 doi:10.1371/journal.pcbi.1005112

Cho H, Berger B, Peng J (2018) Generalizable and Scalable Visualization of Single-Cell Data Using Neural Networks. Cell Syst 7(2):185-191 e4 doi:10.1016/j.cels.2018.05.017

Diggins KE, Ferrell PB, Jr., Irish JM (2015) Methods for discovery and characterization of cell subsets in high dimensional mass cytometry data. Methods 82:55-63 doi:10.1016/j.ymeth.2015.05.008

DiGiuseppe JA, Tadmor MD, Pe'er D (2015) Detection of minimal residual disease in B lymphoblastic leukemia using viSNE. Cytometry Part B, Clinical cytometry 88(5):294-304 doi:10.1002/cyto.b.21252

Donnenberg AD, Donnenberg VS (2007) Rare-event analysis in flow cytometry. Clin Lab Med 27(3):627-52, viii doi:10.1016/j.cll.2007.05.013

Finn WG, Carter KM, Raich R, Stoolman LM, Hero AO (2008) Analysis of clinical flow cytometric immunophenotyping data by clustering on statistical manifolds: Treating flow cytometry data as high-dimensional objects. Cytometry Part B: Clinical Cytometry 76B(1):1-7 doi:10.1002/cyto.b.20435

Jacobs RA (1988) Increased rates of convergence through learning rate adaptation. Neural Networks 1(4):295-307 doi:https://doi.org/10.1016/0893-6080(88)90003-2

Jian T, Jingzhou L, Ming Z, Qiaozhu M (2016) Visualizing Large-scale and High-dimensional Data Proceedings of the 25th International Conference on World Wide Web %@ 978-1-4503-4143-1. International World Wide Web Conferences Steering Committee, Montréal, Québec, Canada, p 287-297

Levine JH, Simonds EF, Bendall SC, et al. (2015) Data-Driven Phenotypic Dissection of AML Reveals Progenitor-like Cells that Correlate with Prognosis. Cell 162(1):184-97 doi:10.1016/j.cell.2015.05.047

Lin L, Frelinger J, Jiang W, et al. (2015) Identification and visualization of multidimensional antigen-specific T-cell populations in polychromatic cytometry data. Cytometry A 87(7):675-82 doi:10.1002/cyto.a.22623

Linderman GC, Rachh M, Hoskins JG, Steinerberger S, Kluger Y (2017) Efficient Algorithms for t-distributed Stochastic Neighborhood Embedding. arXiv:171209005

Linderman GC, Steinerberger S (2017) Clustering with t-SNE, provably. arXiv:170602582

Mazza EMC, Brummelman J, Alvisi G, et al. (2018) Background fluorescence and spreading error are major contributors of variability in high-dimensional flow cytometry data visualization by t-distributed stochastic neighboring embedding. Cytometry Part A 93(8):785-792 doi:10.1002/cyto.a.23566

Stoeckius M, Hafemeister C, Stephenson W, et al. (2017) Simultaneous epitope and transcriptome measurement in single cells. Nature Methods 14:865 doi:10.1038/nmeth.4380

Van Der Maaten L (2014) Accelerating t-SNE using Tree-Based Algorithms. The Journal of Machine Learning Research 15(1):3221-3245

van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. The Journal of Machine Learning Research 9(2579-2605):85

Van Gassen S, Callebaut B, Van Helden MJ, et al. (2015) FlowSOM: Using self-organizing maps for visualization and interpretation of cytometry data. Cytometry A 87(7):636-45 doi:10.1002/cyto.a.22625

van Unen V, Hollt T, Pezzotti N, et al. (2017) Visual analysis of mass cytometry data by hierarchical stochastic neighbour embedding reveals rare cell types. Nat Commun 8(1):1740 doi:10.1038/s41467-017-01689-9

Wattenberg MV, Fernanda; Johnson, Ian (2016) How to Use t-SNE Effectively. Distill doi:10.23915/distill.00002

Weber LM, Robinson MD (2016) Comparison of clustering methods for high-dimensional single-cell flow and mass cytometry data. Cytometry Part A 89(12):1084-1096 doi:10.1002/cyto.a.23030