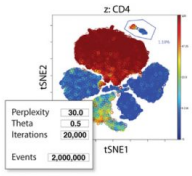
Blog

JANUARY 17, 2017 | CYTOBANK, EDUCATION, VISNE | BY CYTOBANK | 0 COMMENTS

Fine-Tune viSNE to Get the Most of Your Single-Cell Data Analysis

Guest Blog by Tyler Burns



Tyler Burns is a Cancer Biology Ph.D Candidate in Dr. Garry Nolan's lab at Stanford and a consultant for Cytobank. Tyler's work in the Nolan lab is focused on developing novel computational methods for high-parameter single cell analysis.

T-distributed stochastic neighborhood embedding (t-SNE) is a data visualization algorithm developed by Laurens Van Der Maarten and Geoffrey Hinton in 2008. It reduces complex high-parameter data into an easily understandable two-dimensional "map". In 2013, El-ad David Amir and colleagues introduced this method to mass cytometry data analysis, calling it viSNE. It has since been utilized by mass and fluorescent cytometry users in both academia and industry to visualize their data and make discoveries that they would have missed using traditional biaxial gating.

Although the default parameters in Cytobank (introduced in the 2013 Amir *et al.* paper) work well for many cytometry analyses, the algorithm has many parameters that need to be tuned for certain analyses and can be optimized to get the most out of any analysis. A wonderful interactive tool for visualizing how these parameters affect a t-SNE map was developed by Wattenberg and colleagues.

The latest version of viSNE available in Cytobank allows users to change a number of these tunable parameters to get the most out of their data. Cytobank's support article discusses in depth what these parameters are and how they affect output. This blog post is meant to efficiently provide guidance on the best parameter values for certain types of datasets and clinical/biological questions. To this end, I organize this post into a number of lessons learned from examining output and run-time after testing the viSNE parameters through Cytobank's user interface.

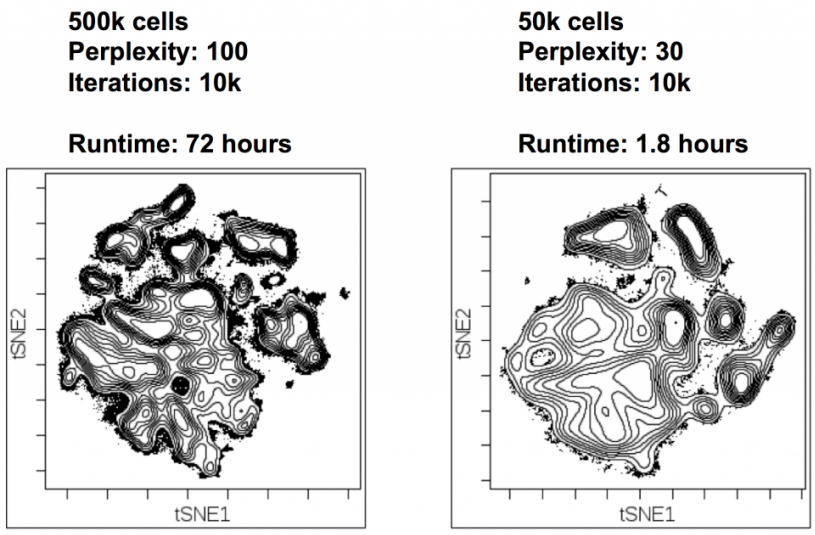
Lesson 1:
Optimizing the number of iterations and perplexity helps separate the cell populations on the viSNE map

One useful way to see the number of populations and the separation of those populations on a viSNE map is using a contour plot, in which the contours represent cell density. Different cell populations will be seen as "peaks" in the contours. viSNE maps that aren't well-resolved won't have clear separation of the populations. Separation of the populations is also useful if you're interested in manually gating or clustering the viSNE map to identify the populations automatically (*e.g.* by running SPADE on viSNE).

Iterations

viSNE in Cytobank iteratively reduces what is called the Kullback-Liebler (KL) divergence between the original high-parameter data and the 2-D map that is visualized as output. As the 2D map gets more and more similar to the high-parameter data, the KL divergence value gets lower and ultimately converges (bottoms out), but only when enough iterations have been performed. Below, using 10,000 iterations rather than the default value of 1000 leads to greater separation of cell subsets (keeping default values for everything else). In this case, the data don't look too different.

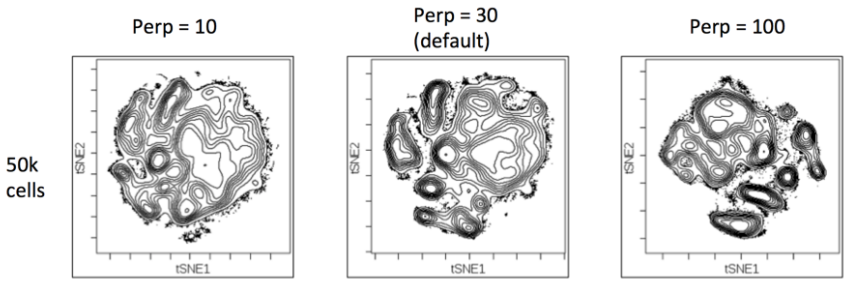
There are still four “islands” to the northeast of the “mainland.” It does look cleaner, but the viSNE run takes longer. In many applications, though, more iterations are a necessity (see lesson #2 below). To make this easier, Cytobank’s interface will report how long every 50 iterations take, so a user can predict how long it will take to run 10,000 iterations rather than 1000.



Increasing the number of iterations used by ten-fold separates cell populations on the final viSNE map into more distinct islands

Perplexity

Perplexity is effectively the number of nearest neighbors a given cell will be compared to in high-dimensional space in order to construct the final viSNE map (https://lvdmaaten.github.io/tsne/). Cytobank sets the default value to 30. Similar to iterations, increasing the perplexity beyond the default value of 30 allows for greater separation of populations and can be seen on a contour plot.



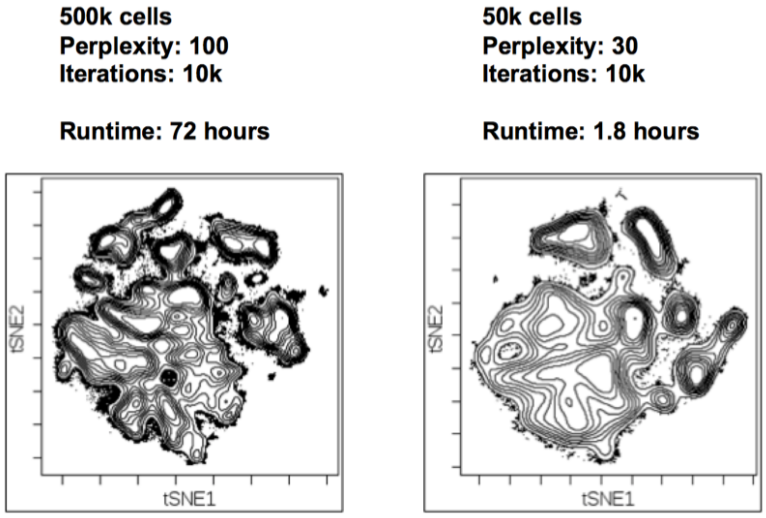
Altering the perplexity values above and below the default value of 30 affects the separation of cells on the viSNE map

Like iterations, higher perplexity (above the default value of 30) may be of benefit if one intends to manually gate or cluster cell subsets on a viSNE map, or in cases where the viSNE map doesn’t converge at lower settings (see lesson #2). However, a value of 30 may be just fine for some questions. Like iterations, higher perplexity slows down the algorithm. One of the benefits of Cytobank’s viSNE implementation in the cloud is that when more iterations or higher perplexity are needed, these lengthy viSNE runs don’t tie up your laptop like they would if you ran them on a local desktop solution.

Lesson 2:
Larger experiments may require more iterations and higher perplexity in order to separate cell populations when many samples are combined or rare cell populations are a target.

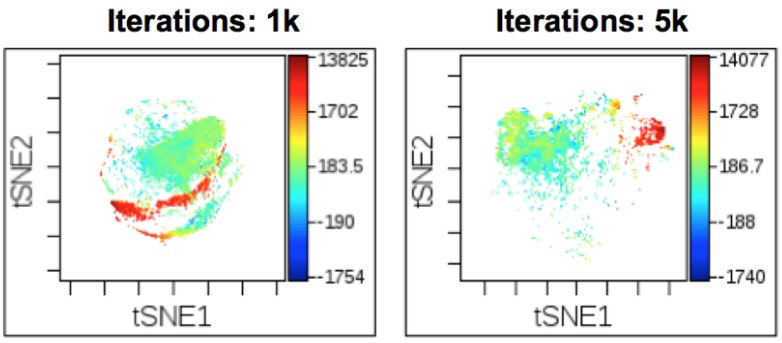
Given what I have shown above, there may be instances where it might be necessary to maximize the perplexity and number of iterations for a dataset, particularly when you have a lot of files and need to make sure an adequate number of cells are included from each, or when you’re interested in rare cell populations.

One example of this is shown on the contour plots below, where you can see many more cell sub-populations (as shown by “peaks” in the contour plot, as well as islands) when I use the total 500,000-cell dataset that the 50,000 cells in lesson 1 were sampled from with the maximum perplexity (100) and 10-times the default number of iterations (10,000). Bear in mind this is an extreme example just to see what the “best” possible t-SNE output is (the run took 72 hours), but it shows what is possible with these very high parameters, and you can imagine that this level of resolution might help identify rare cell populations.



A higher number of cells required a higher perplexity and more iterations to effectively separate cell subsets topologically

Another example is shown on the viSNE maps below, which are colored by a surface marker that isolates a specific cell population. These viSNE maps are for a single sample that was part of a large experiment using fluorescent data. To ensure the inclusion of an adequate number of cells from each sample, 800,000 cells were included in the viSNE. With only 1000 iterations, the cell population that expresses this marker was not grouped on the viSNE map. However, with 5000 iterations, the population is nicely condensed. This can be seen easily in Cytobank by coloring the viSNE map by this channel.



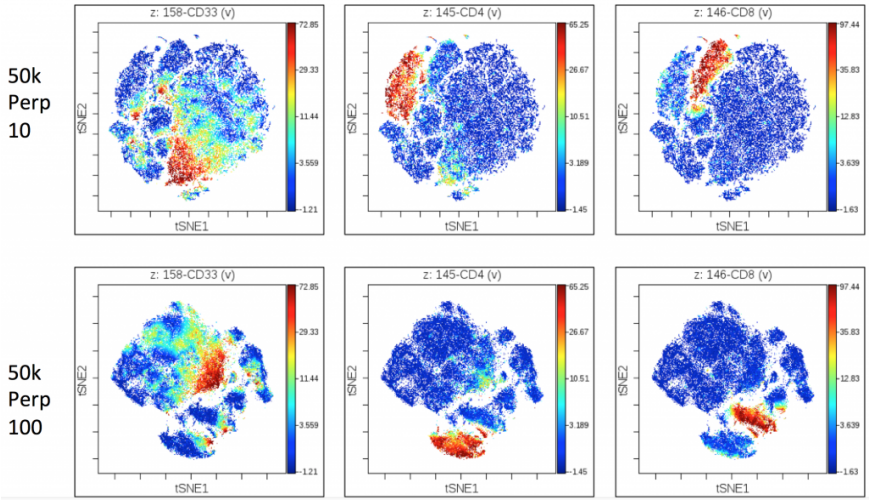
A single sample run as part of a combined viSNE across many samples. Fluorescent data colored by a surface marker that isolates a cell population. Left: 800k cells with 1000 iterations. Right: 800k cells with 5000 iterations

This 5000-iteration run with 800,000 cells took 18 hours on Cytobank’s cloud. I could do both of these lengthy analyses because Cytobank runs viSNE in the cloud. My computer was entirely unaffected, and I got an automated email letting me know when my viSNE analysis was complete. Local desktop viSNE implementations would require a user to keep the computer open and running, and might negatively impact the processing speed of any other program the user is running in parallel. Cytobank’s cloud was also important to my analyses because I was running several of these viSNE analyses in parallel, which is currently not possible on a desktop solution.

Lesson 3:

Even when your viSNE hasn’t completed converging, you can still see the cell populations grouping together if you color by channel.

The perplexity and number of iterations needed for a given dataset may not be very high depending on your question or intent. Here, I performed t-SNE with a perplexity of 10 and a perplexity of 100. In this example, I show that while the perplexity of 100 separates cell populations better on the map, the lower perplexity values nonetheless group cells together in terms of surface marker expression. In other words, even if the map looks less organized, cell subsets are still grouped together in this dataset, which can be seen coloring by channel. If all the user cares about is seeing grouping of cells by marker expression, then high perplexity and iterations may not be needed. On the other hand, if separation of cell subsets on the viSNE map does matter (*e.g.* when the user wants to gate or cluster populations from the viSNE map), then higher perplexity and iterations may be needed. It all depends on the user's goals.



Lower perplexity values produce less physical separation between cell subsets on the t-SNE map, but cell subsets are still close together, as seen by surface marker expression

Putting it all together

I have shown that increasing perplexity and iterations can make viSNE maps cleaner, and better group cell populations as "islands" on the map. This comes at the expense of slowing down the algorithm, however! Cytobank's cloud solution is critical for these longer and more complex viSNE runs, especially in situations where higher perplexity and iterations are necessary to achieve even an adequate amount of resolution. I have also shown that even when your viSNE doesn't converge enough to separate these islands, coloring by surface marker expression lets you visualize the cell populations.

So now you have a cytometry dataset in front of you, and you're trying to figure out how to most effectively utilize viSNE for your analysis. I would think about what you want to get out of your viSNE map. With that in mind, I would pilot a couple runs with some different inputs for perplexity and iterations and see what it does to your data. Remember, this all can be done in parallel and in the cloud! Choose the input values that lead to the best-looking t-SNE map with the minimal necessary run time, and scale up!