

# t-viSNE: A Visual Inspector for the Exploration of t-SNE

Angelos Chatzimpampas\*

Rafael M. Martins†

Andreas Kerren‡

Department of Computer Science and Media Technology, Linnaeus University, Sweden

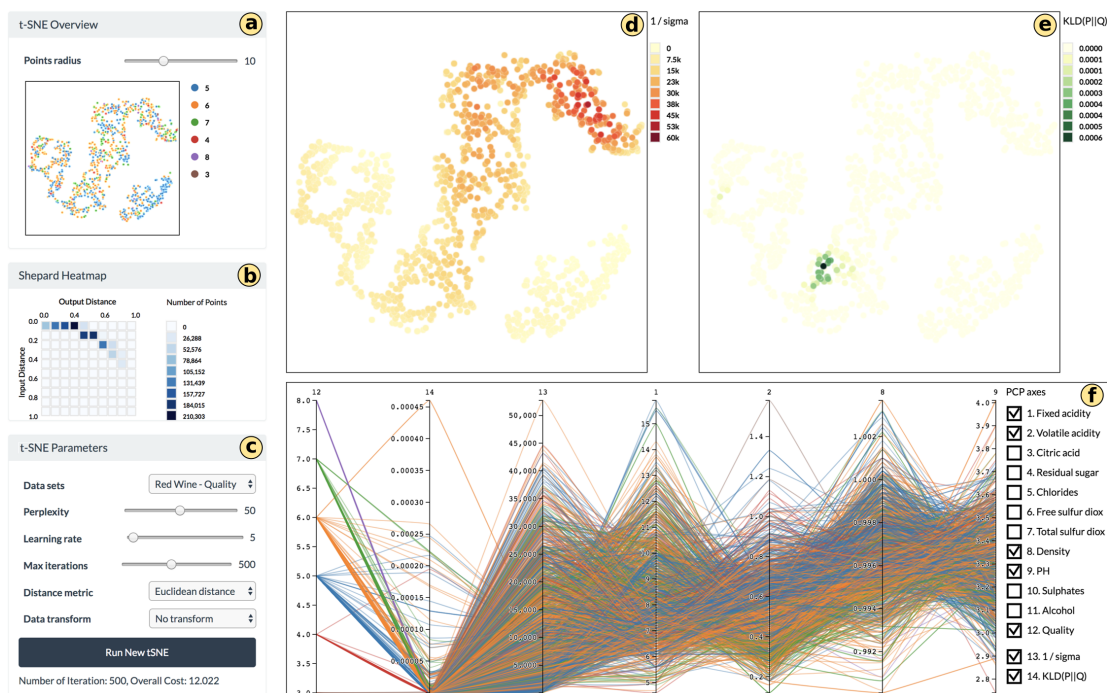


Figure 1: Inspection of t-SNE results with our tool, t-viSNE: (a) overview of the results with data-specific labels encoded with categorical colors; (b) the Shepard Heatmap of all pairwise distances; (c) t-SNE parameters and input data; (d) scatterplot showing the density of neighborhoods in the original high-dimensional space; (e) scatterplot showing the final cost (Kullback-Leibler Divergence) of each point; (f) parallel coordinates plot (PCP) of data features, density of neighborhoods, and cost for every point.

## ABSTRACT

The use of t-Distributed Stochastic Neighborhood Embedding (t-SNE) for the visualization of multidimensional data has proven to be a popular approach, with applications published in a wide range of domains. Despite their usefulness, t-SNE plots can sometimes be hard to interpret or even misleading, which hurts the trustworthiness of the results. By opening the black box of the algorithm and showing insights into its behavior through visualization, we may learn how to use it in a more effective way. In this work, we present t-viSNE, a visual inspection tool that enables users to explore anomalies and assess the quality of t-SNE results by bringing forward aspects of the algorithm that would normally be lost after the dimensionality reduction process is finished.

**Index Terms:** Human-centered computing—Visualization—Visualization application domains—Visual analytics; Computing methodologies—Machine learning—Learning paradigms—Unsupervised learning

\*e-mail: angelos.chatzimpampas@lnu.se

†e-mail: rafael.martins@lnu.se

‡e-mail: andreas.kerren@lnu.se

## 1 INTRODUCTION

Machine learning approaches are widely used to perform various tasks, such as the detection of clusters from abstract data or the analysis of multivariate data. However, understanding their implementation details and interpreting their results are not always trivial tasks. The visual analysis of machine learning techniques and models is a recent and rapidly growing topic in visualization research, with results showing that it is a viable and efficient method [3, 4].

In this extended abstract, we present our ongoing work on a visualization tool, called t-viSNE, which is designed to allow the investigation of the t-Distributed Stochastic Neighbor Embedding (t-SNE), a well-known machine learning algorithm that has been very popular since its proposal in 2008 [6]. We bring forward some of the hidden internal workings of the algorithm which, when visualized, may provide important insights about the characteristics of the multidimensional data set. Our visualization approach supports the following tasks: (i) quality check of distance preservation with a Shepard Heatmap, (ii) exploring the density of multidimensional neighborhoods, (iii) highlighting badly-optimized cases by showing the remaining cost for each point, and (iv) the presentation of data set features using a Parallel Coordinates Plot (PCP).

## 2 BACKGROUND AND RELATED WORK

*Dimensionality Reduction (DR)* techniques reduce the original dimensions of the data set maintaining—as much as possible—its

original structure. When used for visualization, the output is set to two or three dimensions, and the results are commonly visualized with scatterplots, where similar objects are modeled by nearby points, and dissimilar objects are modeled by distant points. Classic DR techniques include Principal Components Analysis (PCA) and Multidimensional Scaling (MDS), but more recent non-linear DR techniques have shown promise in the visualization of complex real-world data sets, with t-Distributed Stochastic Neighbor Embedding (t-SNE) being among the most popular (for a comprehensive review, see [7]).

In t-SNE, each item  $i$  of a data set is initially modeled as a neighborhood-based probability distribution  $P_{ij}$ , where a high probability means that  $j$  is a close neighbor of  $i$  (in the original multidimensional space). Each  $P_{ij}$  follows a Gaussian distribution centered at  $i$ , with the item-specific variances  $\sigma_i$  obtained by searching for a value that results in a target entropy controlled by the *perplexity* parameter [6]. In practice, that means the  $P_{ij}$  of an item  $i$  located in a denser region (in the multidimensional space) will use a small  $\sigma_i$ , while one in a sparser region will use a larger value. Similarly to  $P_{ij}$  (but not identically), another probability distribution  $Q_{ij}$  is obtained from a candidate low-dimensional representation of the data set, and a total cost is computed as the sum of the Kullback-Leibler Divergences between  $P_{ij}$  and  $Q_{ij}$  for each  $i$ . This cost is then minimized by iteratively improving the low-dimensional representation of each point  $i$ , so that the newly updated  $Q_{ij}$  matches—as best as possible—its original distribution  $P_{ij}$ .

The issues with interpreting and assessing the quality and trustworthiness of DR-produced scatterplots have been recognized and tackled in different ways. One of the most common ways to get insights into the differences between pairwise distances of points in the original and final spaces is by using a Shepard Diagram [2], a scatterplot where each point represents a pair of data set elements, and the two axes map their distances in the original and final spaces. A good embedding would result in all points being close to the diagonal of the scatterplot since the original and final pairwise distances would be as similar as possible. Other works have proposed to extract different types of distance- or neighborhood-preservation measures to compare the low-dimensional embedding to the original data, often mapping these measures over the scatterplot itself in different ways (e.g., [1, 5]). Our proposal is similar to these in intent and method, but while previous works focused on DR-independent measures (i.e., measures that work with any embedding), we propose with t-viSNE to extract the measurements/values directly from t-SNE itself, making it a technique-specific tool.

### 3 T-VISNE: VISUALIZATION AND INTERACTION DESIGN

After the user loads a data set and runs t-SNE with the selected parameters (Fig. 1c), different perspectives of the obtained results are shown in the various views of the tool. The *Overview* (Fig. 1a) shows the points and their data-specific labels using a categorical colormap (and a slider for setting the radius of the embedded data points). The *Shepard Heatmap* is an aggregated version of the Shepard Diagram, where each cell shows the density, i.e., the number of points, in each region of the diagram. This was done to avoid clutter and to increase the readability of the Shepard Diagram for large data sets. Both axes were scaled between 0.0 (minimum distance) and 1.0 (maximum distance). The *Sigma Plot* (Fig. 1d) shows the value of  $1/\sigma_i$  (i.e., the inverse of  $\sigma_i$ , so that high values mean more density) of each point color-encoded over the points themselves. As discussed in Sec. 2, this value represents the different densities of the high-dimensional neighborhoods of each point, which is a valuable piece of information from the original data space. By mapping it over the points themselves, we allow the visual comparison of original and final neighborhood arrangements. The *KLD Plot* (Fig. 1e) shows the final value of  $KLD(P_i||Q_i)$ , i.e., the remaining cost after the last iteration, for each point (with a different colormap). This allows the

user to investigate which points (or groups of points) were positioned by t-SNE in good configurations regarding their neighbors (low remaining cost) and which were not well optimized even after all the iterations (high remaining cost). This information affects the local trustworthiness of different areas of the plot. Finally, the *Parallel Coordinates Plot* (Fig. 1f) provides a way for the user to explore the actual features of the data set in more detail and to correlate them to the patterns found in the previously-described plots. It offers two main interactions: (i) linked brushing with the other plots, such that only selected points are highlighted; and (ii) filtering and rearranging of axes, such that the user may choose which dimensions to see (and how they will be shown) at any given time.

### 4 EXAMPLE OF APPLICATION: WINE QUALITY

We illustrate our tool with a data set composed of red wine samples from the north of Portugal described by physicochemical dimensions such as acidity and residual sugar, and a sensory classification of their quality<sup>1</sup>. From the *Overview* (Fig. 1a), we can observe that the labels are not well-separated by the t-SNE layout and are mostly randomly distributed throughout the plot. The user might think that, due to this, there would be large values for the remaining costs all over the plot, but the reality is the opposite: the remaining KLD values are very low in most of the plot, except for a hot spot in the middle, see Fig. 1e. Comparing the hot spot to the overview, there is no apparent correlation with the label distribution. Thus, we have found an area of the plot that has not been well-optimized and must either be investigated further or removed from the analysis; that area would have been impossible to differentiate from the others without the visualization. Furthermore, there is apparently no drastic change in density anywhere in the overview. However, the *Sigma Plot* (Fig. 1d) shows that there is a gradient of increasing density that follows the layout roughly from left to right. Such an information might be important to the analyst, since denser neighborhoods (in the original space) may indicate more cohesive groups of data items. In fact, by looking at the overview, there is one apparent cluster of blue points in the bottom-right corner that might be hypothesized as a dense cluster of points. Comparing this cluster to the *Sigma Plot*, it appears that this hypothesis might not be correct, because the plot actually shows low densities in that area.

### 5 CONCLUSION AND FUTURE WORK

In this poster, we presented a visual inspector that helps the user to explore t-SNE's behavior and avoid potentially wrong interpretations of the results. As future work, we will improve t-viSNE by extending the information provided to the user and performing an evaluation.

### REFERENCES

- [1] M. Aupetit. Visualizing distortions and recovering topology in continuous projection techniques. *Neurocomputing*, 70(7-9):1304–1330, 2007.
- [2] I. Borg and P. J. Groenen. *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media, 2005.
- [3] J. Krause, A. Perer, and E. Bertini. Using visual analytics to interpret predictive machine learning models. *arXiv preprint arXiv:1606.05685*, 2016.
- [4] S. Liu, X. Wang, M. Liu, and J. Zhu. Towards better analysis of machine learning models: A visual analytics perspective. *Visual Informatics*, 1(1):48–56, 2017.
- [5] R. M. Martins, D. B. Coimbra, R. Minghim, and A. C. Telea. Visual analysis of dimensionality reduction quality for parameterized projections. *Computers & Graphics*, 41:26–42, 2014.
- [6] L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.
- [7] L. Van Der Maaten, E. Postma, and J. Van den Herik. Dimensionality reduction: A comparative. *J Mach Learn Res*, 10:66–71, 2009.

<sup>1</sup><https://archive.ics.uci.edu/ml/datasets/wine+quality>