Spontaneous Symmetry Breaking in Data Visualization

Anonymous Author(s)
Affiliation
Address
email

Abstract

Data visualization tools are designed for creating low-dimensional representations of data that emphasize structure and suppress noise. However, such non-linear amplifications of structural differences can have side effects like spurious clustering in the widely used t-SNE (Amid and Warmuth [2018]). We present a more general class of spurious structure, namely broken symmetry, defined as visualizations that lack symmetry present in the underlying data. We develop a simple workflow for detection of broken symmetry and give examples of spontaneous symmetry breaking in t-SNE and other well-known algorithms such as GPLVM and kPCA. Our extensive, quantitative study shows that these algorithms frequently break symmetry, thereby highlighting new shortcomings of current visualization tools.

1 Motivation

Data visualization is a core tool in the machine learning toolbox. Data sets are visualized for exploration, to formulate hypotheses and to make modeling decisions. Visualization is commonly used for interpretation of learned models, e.g. visualization of latent variables of a generative model to understand representations. Data visualization is also very useful for debugging. For these applications *faithfulness* is a concern - can we trust the structure revealed in a visualization?

Most data of interest is high dimensional, hence can not be directly visualized. Rather, some form of dimensionality reduction is required, which inevitably will lead to loss of information. Popular schemes such as t-SNE [van der Maaten and Hinton, 2008], aim at two or three-dimensional representations that capture both local and global structure in data. Fig. 1 shows a two-dimensional t-SNE visualization of images from the COIL-20 dataset [Nene et al., 1996]; the given example concerns a wooden object on a turntable that is viewed from multiple, equidistant angles forming full 360° rotation. Such an incremental physical rotation leads to a set of images with a simple topological structure which can be quantified by the neighborhood graph. More specifically, we form a graph with the images as nodes and connect neighboring nodes along the rotation path to obtain the graph of a circle. The neighborhood graph presents us with a strong physical symmetry and we naturally expect a visualization of the data to reveal this pattern by a structure which is topologically equivalent to a circle. Evidently, this does not happen: The visualization has broken the symmetry and "invented" a difference between neighboring points that is non-physical.

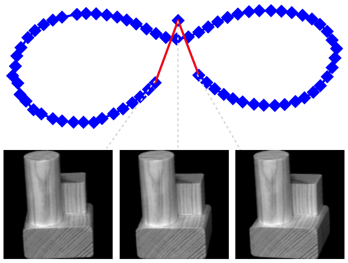

Figure 1: We analyse a set of images of an object subject to a 360° rotation on a turntable. The nearest neighbor graph forms a simple circle, however when the set is visualized using t-SNE the symmetry is lost.

The significance of transformations and the ensuing question of symmetry preservation goes beyond the physical rotations of the COIL data set. Parameterized transformations are key to modern data augmentation strategies. In this setting, the question of preservation of symmetries in augmented data sets is related to whether given symmetries are successfully represented by the learning algorithm.

Our contribution is to identify a new, general class of spurious structures in data visualization, namely spontaneously broken symmetry, defined as representations that lack symmetry present in the underlying data. Furthermore, we provide a topological, quantitative measure to detect broken symmetry (Sec. 2) allowing for a systematic study. Our empirical studies (Sec. 3) show that widely used visualization techniques break simple symmetries like rotations, hence, challenging the notion that they conserve global structure.

2 Symmetries, Graphs, and Persistent Homology

Symmetry groups. We consider symmetries. Here, a symmetry refers to a property of a system that remains unchanged under a given transformation. The images of the wooden toy in Fig. 1 are formally *equivariant* when the toy is rotated physically on the turn-table, while the outputs of a deep network for image based object classification ideally would be invariant (symmetric) under rotation.

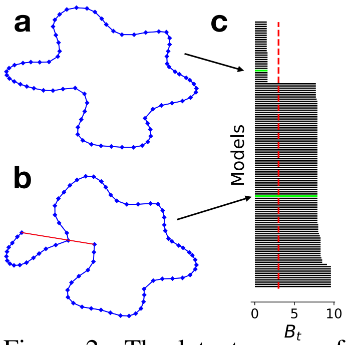

Figure 2: The latent space of a model that preserves symmetry (*a*) and one that does not (*b*). (c) A *barcode* as a function of thresholds B_t.

Mathematically, such transformations and symmetries are described by Lie groups [Hall, 2015]. A real Lie group is a smooth differentiable manifold on which points are connected through a group operation and its inverse. For instance, rotation matrices form a smooth group with the matrix multiplication group operation. The unit circle can then be generated by a single unit vector and its multiplication with all members of the group of rotation matrices. If the rotation group governs a physical phenomenon then we expect to observe data along a path that topologically is a circle, disregarding observation noise.

In this paper, we focus on situations where the governing group is known and investigate if the group structure is preserved by common visualization techniques. This is achieved by verifying if the topology of the group remains intact under visualizations.

Discrete approximations. In practice, we only observe a finite number of data points, rather than the entirety of a group. We can, however, approximate the path spanned by the observations with a graph, where points are connected if their generating group elements are close under the group metric. For instance, we may connect rotated images in a graph if their rotation angles are similar as also described in Sec. 1.

Measuring broken symmetry. For visualization, we map data to a low-dimensional space (typically $\mathbb{R}^2$); we let $X = \{x_i\}$ denote data coordinates in this low-dimensional space. We can now determine if a symmetry has been preserved under visualization by asking if the associated graph can be recovered from the low-dimensional coordinates. As the graph informs as to which points should be neighbors, we measure for each set of neighbors the radius of the ball needed to include one in the other's neighborhood graph. To compare across methods, we scale all distances by their median

$$B_{\text{median}} = \operatorname*{median}_{(x_i, x_j) \in G} (\|x_i - x_j\|), \tag{1}$$

where G denotes the graph associated with the generating group. We rely on the median due to its high breakdown point [Huber, 2004]. We, thus, measure

$$B_{ij} = \frac{\|x_i - x_j\|}{B_{\text{median}}}. \tag{2}$$

We can then threshold this measure such that, we say that a symmetry has been broken if $B_{ij} > B_t$ for any pair or equivalently, $\max(B_{ij}) > B_t$. We define $B_{\text{max}} := \max(B_{ij})$. Note that this measure does not distinguish between one or multiple instances of broken symmetry.

Persistent homology. The measure above is linked with *persistent homology* [Hatcher, 2005]. This is a key mathematical tool in topological data analysis that has been shown robust to perturbations of the input data [Cohen-Steiner et al., 2007]. Following Carlsson [2009], we may place balls on each data point with radius ϵ and points falling within this ball defines a neighborhood. This defines a topological space Ω_ϵ. By varying ϵ, we can create multiple topological spaces and let the Betti numbers $b_i(\Omega_\epsilon)$ quantify the structure of the topological space. The number b_0 represents the approximate number of connected components and b_1 the number of circles or holes.

In persistent homology, we study a spectrum of neighborhood sizes. For a *known* generating group, we would know its Betti numbers, and may ask which (if any) ϵ yield the given Betti numbers in the visualization point set. This allows us to consider multiple thresholds of our measure (2) of symmetry.

Barcodes. A broken symmetry is defined by the maximum of the normalized pairwise distances B_{max} being greater than a threshold B_t. This we can represent by a bar ranging from zero to B_{max} that visualizes the birth and death of symmetry. Stacking such bars (as in Fig. 2) yields a *barcode*. This lets us inspect the sensitivity of a chosen threshold for multiple models visually as each bar corresponds to a model [Ghrist, 2008]. The 'sharper' the transition from short bars to long bars is, the more robust the conclusion is. The barcode in Fig. 2 suggests that a choice of $B_t = 3$ is quite robust as choosing values in the range $B_t = 2$ and $B_t = 8$ yields the same conclusions.

For quantitative comparisons across experiments and for simplicity, we have opted for a consistent threshold $B_t = 3$. In all cases, associated barcodes will be provided to quantify the robustness of the chosen threshold in supplements.

3 Experiments

In our empirical study, we consider four methods representing the spectrum of data visualization techniques:

t-SNE matches an exponential distribution of pairwise distances in data space with a t-distribution of pairwise distances in the latent space [van der Maaten and Hinton, 2008]. The visualization is controlled by a *perplexity* parameter that quantifies the effective number of neighbors used in the exponential distribution over pairwise distances. This is a randomized model and experiments were performed using scikit learn [Pedregosa et al., 2011].

TriMap [Amid and Warmuth, 2019] is a recent method that relies on an elaborate triplet weighting scheme such that point triplets are weighted with their pairwise distance before obtaining the final triplet weight $\omega_{ijk} = \zeta_\gamma\left(\delta + \tilde{\omega}_{ijk}/\omega_{\text{max}}\right)$. Here $\zeta_\gamma(u) = \log(1 + \gamma u)$, where the γ *parameter* is said to place focus on either local or global structure. The method is randomized and experiments were performed using software provided by Amid and Warmuth [2019].

Kernel principle component analysis (kPCA) [Schölkopf et al., 1998] extends classic PCA through the kernel trick. We use the squared exponential kernel $k(x_i, x_j) = \exp\left(-||x_i - x_j||^2/\lambda\right)$, which is controlled by the *scale* parameter λ. The model is deterministic and experiments were performed using scikit learn [Pedregosa et al., 2011].

Gaussian process latent variable model (GPLVM) [Lawrence, 2005] visualizes data using a latent representation with a Gaussian process prior with covariance function $k_{ij} = \theta \exp\left(-1/2\|x_i - x_j\|^2\right) + \sigma^2\delta_{ij}$, where x_i, x_j denote latent points. The model is deterministic for a given *initial condition* of the hyperparameters θ and σ^2, θ_0 and σ_0^2. Experiments were performed using Pyro [Bingham et al., 2019].

In all experiments, we vary method parameters over a large range, and randomized methods are repeated multiple times and reported numbers are averages. Experimentally, we focus on the most elementary symmetry of interest: *the rotation group*. We consider images from (1) *COIL-20* where objects are rotated 360° in 72 steps and (2) *MNIST* where we synthetically rotate images with up to 360° and 5% Gaussian noise is added. We perform a detailed analysis of each model's behavior, and quantitatively compare and summarize in Sec. 3.5.

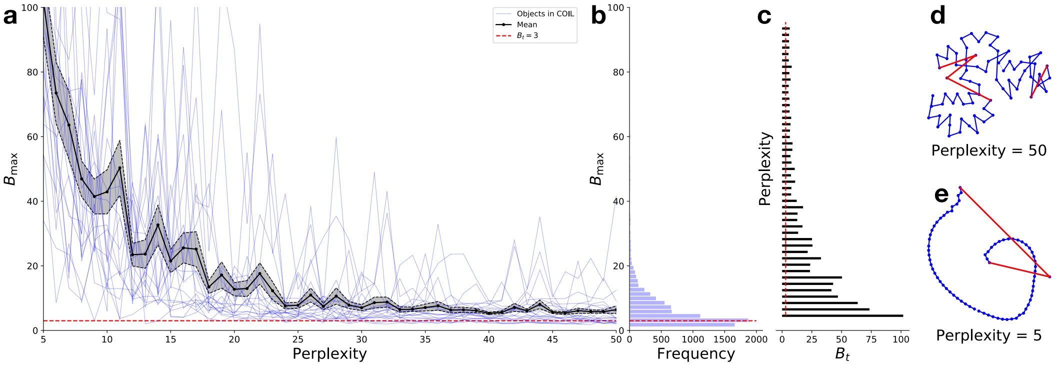

Figure 3: (*a*) Perplexity parameter space for t-SNE. Each blue line represents the mean over 30 repeats for an object in COIL-20. The dotted, red line marks $B_t = 3$ and the black lines represent mean and standard error over all objects. (*b*) Histogram of models. (*c*) Barcode for the mean (black lines in (*a*) of objects. (*d* & *e*) Latent space in models with perplexities 50 and 5.

3.1 t-Distributed Stochastic neighborhood Embedding (t-SNE)

To investigate possible symmetry breaking in t-SNE, we fit 30 t-SNE models to images of each COIL-20 object over a large span of *perplexity* parameters. We measure $B_{\max} = \max B_{ij}$ using (2) and report averages over the 30 models. These are the blue lines in Fig. 3(a). We see that as perplexity increase, $B_{\max}$ becomes smaller. This is to be expected as perplexity controls the smoothness of the t-SNE model. In 73% of all models, we observe broken symmetry ($B_{\max} > 3$). The barcodes reveal that this percentage is not particular sensitive to the choice of threshold (the red dotted line correspond to $B_t = 3$). On MNIST, we observe a similar pattern (omitted due to space constraints) with 96.5% of all models having a broken symmetry.

In our experience, t-SNE tends to amplify small gaps in the data, leading to broken symmetry. This is linked to the 'spurious clustering' effect observed by Amid and Warmuth [2018]. We generally observe that random initialization of t-SNE seems to better preserve symmetries than initialization by other methods such as PCA or Isomap. This former approach requires multiple restarts and choosing the embedding with lowest KL divergence.

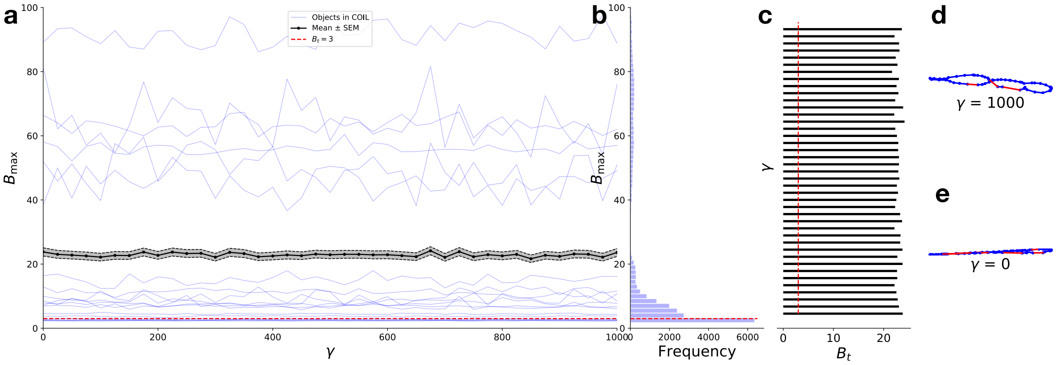

Figure 4: (*a*) Parameter space for TriMap. Each blue line represents the mean over 30 repeats for an object in COIL-20. The dotted, red line marks $B_t = 3$ and the black lines represent mean and standard error over all objects. (*b*) Histogram of models. (*c*) Barcode for the mean (black lines in (*a*) of objects. (*d*) Latent space in model with $\gamma = 1000$. (*e*) Latent space in model with $\gamma = 0$

3.2 TriMap

Amid and Warmuth [2019] developed TriMap motivated by the spurious clustering effect in t-SNE, and we hypothesized that TriMap would lead to less symmetry breaking. However, the evidence in

Fig. 4 does not support that conclusion. As before, each blue line shows the average B_{max} for 30 randomly initialized models for each object in COIL-20 over a wide span of the γ parameter. Here 77% of all models are estimated to show broken symmetry, which is roughly on par with t-SNE. The barcode indicates that the choice of threshold is robust, though we find some inter-object variability (see supplements for details). Our findings for MNIST are similar with 93.32% estimated symmetry breaking.

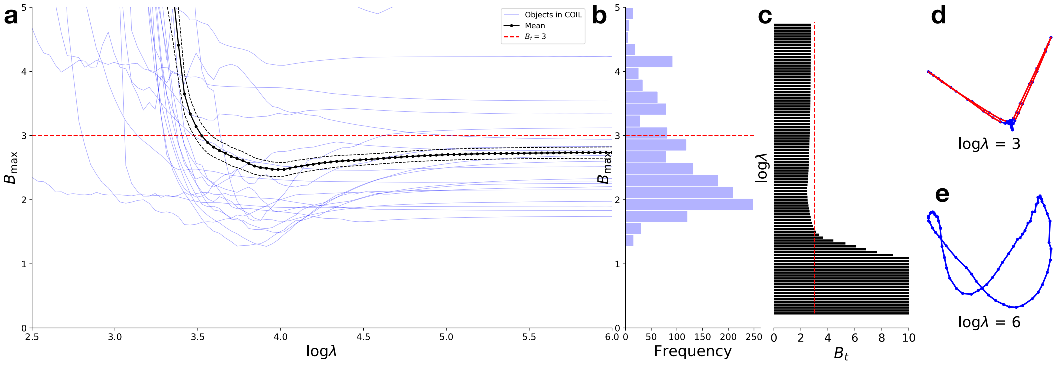

Figure 5: (a) Scale (λ) parameter space for kPCA. Each blue line represents an object in COIL-20. The dotted, red line marks $B_t = 3$ and the black lines represent mean and standard error over all objects. (b) Histogram of models. (c) Barcode for the mean of objects (black lines in (a)). (d) Latent space of model with $\log\lambda = 3$. (e) Latent space of model with $\log\lambda = 6$.

3.3 Kernel Principal Component Analysis (kPCA)

In kPCA, we examine symmetries as a function of the kernel scale parameter λ. The barcode (Fig. 5) shows the robustness of the conclusion of preserved symmetry for the mean across COIL-20 objects. For large values of the scale parameter, the conclusion is robust as B_t can vary, but for smaller values, our conclusions become sensitive to the specific choice of B_t.

In the non-linear regime (small values of λ), B_{median} (1) is driven to a very small values (see example in Fig. 5d) and thus B_{max} diverges. In the linear regime (large values of λ), the model approaches PCA which explains the flattening (see example in Fig. 5e).

In 42% of models, we observe broken symmetry and we note that five objects in COIL-20 give rise to broken symmetries: Object 2 (wooden toy), object 16 (round bottle), object 16 (ceramic vase), object 18 (tea cup) and object 20 (round container). Of these, four are rotationally symmetric in the plane of rotation, supporting our hypothesis that the additional symmetry can induce symmetry breaking. More details can be found in the supplements.

On MNIST data, the rate of broken symmetries was 7.23%. One possible explanation for this reduction, is that if PCA on the MNIST data does not induce symmetry breaking then fewer models will break the symmetry because kPCA converges to PCA in the linear regime.

3.4 Gaussian Process Latent Variable Model (GPLVM)

We investigated the GPLVM design space by varying the initial values of the kernel hyperparameters, θ_0 and σ_0^2 all with identical initialization of the latent space (isomap [Tenenbaum, 2000]). In Fig 6a, θ_0 is fixed and σ_0^2 is varying and in Fig 6b, σ_0^2 is fixed while θ_0 varies. An interesting thing to notice is while we mostly get consistent results, sometimes a small change in the initial condition induces a large change in the B_{max} leading to somewhat complex behavior.

We notice that the loss is often an indicator of broken symmetry as we saw with the KL divergence for t-SNE. If the parameter space contains symmetry-preserving models and symmetry-breaking models then these generally have lower loss than models that break symmetry.

The hyperparameters θ and σ^2 converge to the final values independent of the model preserving the symmetry. This means that the difference in loss between symmetry-preserving and symmetry-breaking models must be accounted for by the latent variables. It also means that it is not possible to

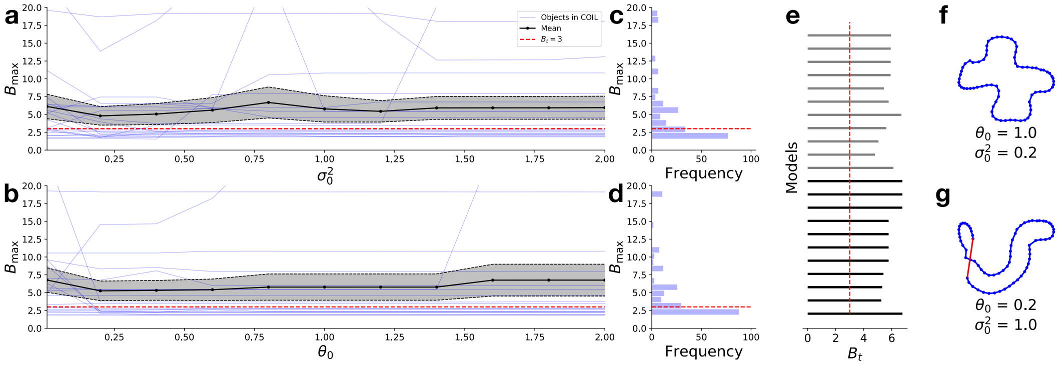

Figure 6: (*a-b*) Parameter space formed by the initialization values of the hyperparameters, θ_0 and σ_0^2 for GPLVM. Each blue line represents an object in COIL-20. The dotted, red line marks $B_t = 3$ and the black lines represent mean and standard error over all objects. (*a*) Parameter space in σ_0^2 with fixed θ_0. (*b*) Parameter space in θ_0 with fixed σ_0^2. (*c*) Histogram of models in *a*. (*d*) Histogram of models in *b*. (*e*) Barcode for the mean of objects (black lines in (*a*) and (*b*)). (*f*) Latent space of model with $\theta_0 = 1$ and $\sigma_0^2 = 0.2$. (*g*) Latent space of model with $\theta_0 = 0.2$ and $\sigma_0^2 = 0.2$.

detect a broken symmetry from the optimized hyperparameters but rather, one have to consider the latent variables to detect a broken symmetry.

Like in kPCA, we find broken symmetries in the most symmetric objects. In the GPLVM, this is linked to the choice of initialization of the latent space. More in supplements.

Overall, we found broken symmetries in 65.48% of the models and similarly, in MNIST, the rate was 46.32%. The supplements contains parameter spaces for additional datasets.

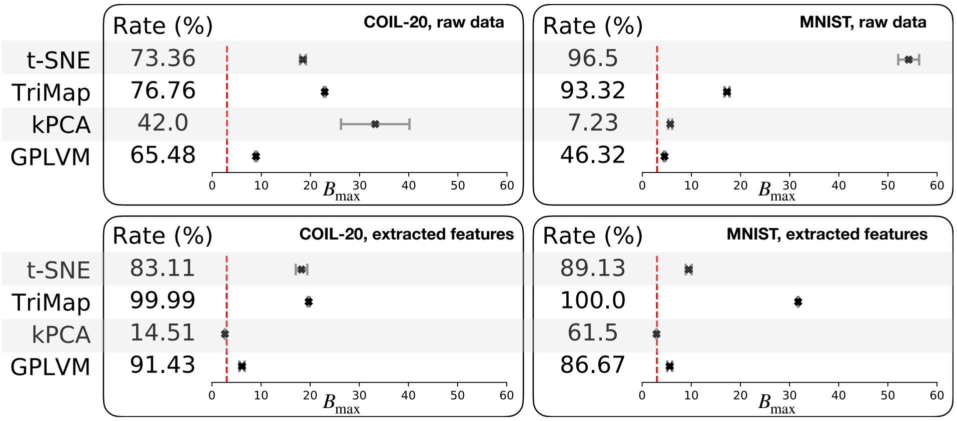

Figure 7: Each pane shows the rate of broken symmetries in percent at $B_t = 3$ with the mean and standard error plotted displayed on the axis for t-SNE, TriMap, kPCA, and GPLVM. *Top, left pane*) Summary of results on COIL-20. *Top, right pane*) Summary of results on MNIST. *Bottom, left pane*) Summary of results on features extract from COIL-20 using ResNet-18. *Bottom, right pane*) Summary of results on features extracted from MNIST using ResNet-18.

3.5 Summary of experiments

We found broken symmetry in all models with a high prevalence as summarized in Fig. 7. The details are found in Sec. 3.1-3.4. Note that we did not tune the parameters but varied important parameters across large ranges and used default parameters for others.

We note that all objects in COIL-20 are indeed symmetric in data space according to our estimator. One may expect that high-level features may be less susceptible to broken symmetry than raw data. We investigate this using ResNet18 [He et al., 2015] to extract features and investigated the prevalence here. We found no broken symmetries in the extracted features and no consistent, significant difference when looking at symmetry in the models trained on extracted features. We noticed that the most symmetric objects generally experienced more broken symmetry across models.

4 Related Works

Data visualization is important at many steps in the machine learning process. Visualization is used exploratively to form hypotheses [Arora et al., 2018], for understanding latent representations in supervised learning [Esteva et al., 2017] and generative models [Frid-Adar et al., 2018].

The desiderata of visualization are discussed by Kaski et al. [2003] and Venna et al. [2010], who argue that visualizations should be trustworthy, meaning that samples appearing similar (e.g., neighbors) in the visualization should be similar in a physical sense. Also, they point out that data points close in a physical sense should be close in visualization. They noted the similarity with the concepts of precision and recall in information retrieval. Our concept of broken symmetry is related to the "recall" dimension, i.e., data that are physical neighbors, should also be visualized as such. The precision and recall criteria together measure the faithfulness of the visualization, see also Najim [2014] for a related quantitative measure of the preservation of neighborhood relations in visualizations.

The immensely popular visualization scheme t-SNE [van der Maaten and Hinton, 2008] is constructed with the aim of representing both global and local structure. The original motivation for t-SNE included a critique of its predecessor SNE [Hinton and Roweis, 2003] for creating crowded visualizations, i.e., visualizations that did not show a clear separation of known clusters. Crowding is closely related to the trustworthiness concept of Kaski et al. [2003], Venna et al. [2010]. By using a long-tailed distribution of the representations, t-SNE aims to fix the crowding problem. However, this emphasis of local dissimilarity comes at a price as noted in Linderman and Steinerberger [2017], simple manifolds like lines and sheets are broken apart in clusters. These clustering problems are examples of broken symmetry in our definition. Motivated by the problem of over-fitting cluster structure Amid and Warmuth [2019] proposed TriMap. We observed, however, that TriMap cannot heal the problem of broken symmetry.

For the detection of symmetries, we employed topological data analysis [Carlsson, 2009], specifically persistent homology. Using this, we examined all values of thresholds simultaneously rather than study just a single threshold. Conveniently, Cohen-Steiner et al. [2007] showed that the persistent homology tool is robust under pertubations of the data. Pokorny et al. [2012] used persistent homology in its classical form whereas we have adapted it slightly as we knew which Betti numbers were required to preserve the symmetry. Our work exploits the coordinate and deformation invariances in topology and these properties aid in detecting symmetries as various deformations of the "circle" graph.

5 Discussion

We have investigated to which extend common visualization techniques are able to preserve simple symmetries, and have largely found the answer to be negative.

5.1 Empirical findings

We have investigated four popular algorithms that also represent different branches of the literature, namely t-SNE [van der Maaten and Hinton, 2008], TriMap [Amid and Warmuth, 2019], kPCA [Schölkopf et al., 1998] and the GPLVM [Lawrence, 2005]. We have performed a systematic study of the influence of parameter choices in these methods by training more than 85.000 models over a wide parameter span. To quantitatively summarize these models' performance, we have introduced a simple scheme for detecting whether known symmetries are broken. Tools from persistent homology verify that this scheme is generally reliable, with some deviations for kPCA (see below).

t-SNE was found to be particularly sensitive to local optima and generally we found a need for multiple restarts. Fortunately, we generally observe that smaller KL reported values imply less symmetry breaking. Even with such mechanisms in place, we still see an overwhelming number of broken symmetries. Symmetry breaking can, to some extend, be reduced by increasing the perplexity parameter, but this also limits the flexibility and expressivity of the model.

TriMap, which was developed in part to alleviate problems with t-SNE, overall had comparable behavior to t-SNE with regards to broken symmetry. The γ parameter, that controls the trade-off between capturing local or global structure, was found to have practically no effect with regards to symmetry breaking. We did not expect this, but have manually verified that broken symmetry is prevalent across large spans of γ.

kPCA was in a sense the most successful method according to our estimator. Kernel PCA, however, has a tendency to collapse points on to each other when mapping only two latent dimensions in the non-linear regime leading to strong symmetry breaking. On the other hand, kPCA reduces to conventional PCA in the limit of large kernel length scales, showing less symmetry breaking.

GPLVM was generally found to be sensitive to choice of initial parameters. While we have found it helpful to consider multiple restarts and choosing the model with highest likelihood, we have found broken symmetry to be rather prevalent.

High-level features. One could suspect that symmetries are broken more commonly when working with raw data than with high-level abstract features, e.g., as those extracted by deep neural networks. We found no broken symmetries directly in the high-level features though when applying visualization algorithms, the prevalence was indeed high.

Summary. Our general finding is that symmetries are broken consistently across the studied methods. It is generally possible to manually tweak parameters to enforce that a known symmetry remains intact, but such strategies are not possible when the symmetry is unknown[1], e.g. for knowledge discovery. We also note that default parameters of publicly available implementations of the studied methods generally perform poorly with regards to broken symmetry.

5.2 Faithful representations

At the heart of our study is the quest for *faithful representations*, i.e. representations that reflect the underlying physics of the data generating process. These have wider applicability than just visualization as studied here. For instance, a representation that is not faithful will most likely not result in a fair prediction. A broken symmetry can be viewed as model that violates the Lipschitz continuity condition. *Individual fairness* [Dwork et al., 2011] can then no longer be ensured as *similar individuals should be treated similarly*.

Similar statements can be made for interpretable models, where 'almost discontinuous' models are generally difficult to interpret. From a purely predictive point of view, it is strictly not required that representations are faithful, though there is some evidence in that direction [Rieger et al., 2019].

Finally, we note that visualization may be particularly sensitive to symmetry breaking as we tend to embed onto $\mathbb{R}^2$. While it is well-known that only few graphs (namely the *planar* ones) can be embedded in $\mathbb{R}^2$, then *all* graphs can be embedded in $\mathbb{R}^3$ [Nishizeki and Chiba, 1988]. This suggest that symmetries are likely to be broken when data is forced onto a two-dimensional view, and indeed our experiments indicate that symmetry breaking is less frequent when embedding into three or more dimensions (see supplements).

5.3 Concluding remarks

We have here pointed to a previously unnoticed problem in visualizations, namely *broken symmetries*. Through a systematic study of more than 85.000 trained models, we have found an alarming rate at which even the most simple symmetries are spontaneously broken during data visualizations. This suggest a need for both new methods that can reliably visualize high-dimensional data, but also for more systematic and quantitative evaluations of visualization techniques.

We have purposefully not investigated more complex symmetries as these raise complications that are beyond existing techniques; for instance, the two-dimensional torus is mathematically impossible to embed in $\mathbb{R}^2$ without breaking the underlying symmetry. This calls for visualization techniques that embed onto curved surfaces in order to preserve symmetries, just as we use a sphere when we visualize global geoinformatics patterns.

[1] It should be emphasized that while we consider known symmetries, we only do so in order to make quantitative statements.

6 Broader impact

Our work demonstrates the alarmingly high prevalence of misrepresentation of symmetries in data visualization which is integral in machine learning. Though we have focused on visualizations, our method does not restrict itself to two dimensions and our considerations are generally applicable to dimensionality reductions and latent variable models in any latent dimensionality.

Our main consideration concerns how the broken symmetry might affect down-stream decision-making. As also hinted to in the main body of paper, individual fairness (*similar individuals should be treated similarly* [Dwork et al., 2011]) might be breached in models that do not preserve the symmetry. Both a broken symmetry and breached individual fairness are essentially violations of the Lipschitz continuity condition.

The consequence of this is that two similar individuals get mapped to different areas of the latent space. This is a one to many mapping from data space to latent space and might lead to an overrepresentation if the latent space is sampled from. This solely emphasizes the importance of visualizations and models being faithful in general.

References

Ehsan Amid and Manfred K. Warmuth. A more globally accurate dimensionality reduction method using triplets. *arXiv:1803.00854 [cs]*, March 2018.

Ehsan Amid and Manfred K. Warmuth. TriMap: Large-scale Dimensionality Reduction Using Triplets. *arXiv:1910.00204 [cs, stat]*, October 2019.

Sanjeev Arora, Wei Hu, and Pravesh K Kothari. An analysis of the t-sne algorithm for data visualization. *arXiv preprint arXiv:1803.01768*, 2018.

Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D. Goodman. Pyro: Deep universal probabilistic programming. *The Journal of Machine Learning Research*, 20(1):973–978, January 2019. ISSN 1532-4435.

Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, January 2009. ISSN 0273-0979. doi: 10.1090/S0273-0979-09-01249-X.

David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of Persistence Diagrams. *Discrete & Computational Geometry*, 37(1):103–120, January 2007. ISSN 1432-0444. doi: 10.1007/s00454-006-1276-5.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Rich Zemel. Fairness Through Awareness. *arXiv:1104.3913 [cs]*, November 2011.

Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, 2017.

Maayan Frid-Adar, Idit Diamant, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan. Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. *Neurocomputing*, 321:321–331, 2018.

Robert Ghrist. Barcodes: The persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1):61–75, 2008. ISSN 0273-0979, 1088-9485. doi: 10.1090/S0273-0979-07-01191-3.

Brian C. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*. Graduate Texts in Mathematics. Springer International Publishing, second edition, 2015. ISBN 978-3-319-13466-6. doi: 10.1007/978-3-319-13467-3.

Allen Hatcher. *Algebraic topology*. Cambridge University Press, 2005.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*, December 2015.

Geoffrey E. Hinton and Sam T. Roweis. Stochastic neighbor embedding. In *Advances in Neural Information Processing Systems*, pages 857–864, 2003.

Peter J Huber. *Robust statistics*, volume 523. John Wiley & Sons, 2004.

Samuel Kaski, Janne Nikkilä, Merja Oja, Jarkko Venna, Petri Törönen, and Eero Castrén. Trustworthiness and metrics in visualizing similarity of gene expression. *BMC bioinformatics*, 4:48, October 2003. ISSN 1471-2105. doi: 10.1186/1471-2105-4-48.

Neil D. Lawrence. Probabilistic Non-linear Principal Component Analysis with Gaussian Process Latent Variable Models. *Journal of Machine Learning Research*, page 34, 2005.

George C. Linderman and Stefan Steinerberger. Clustering with t-SNE, provably. *arXiv:1706.02582 [cs, stat]*, June 2017.

Safa A Najim. Information visualization by dimensionality reduction: a review. *Journal of Advanced Computer Science & Technology*, 3(2):101, 2014.

Sameer A Nene, Shree K Nayar, and Hiroshi Murase. Columbia Object Image Library (COIL-20). *Technical Report CUCS-006-96*, page 6, 1996.

Takao Nishizeki and Norishige Chiba. *Planar graphs: Theory and algorithms*. Elsevier, 1988.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Florian T. Pokorny, Hedvig Kjellström, Danica Kragic, and Carl Ek. Persistent Homology for Learning Densities with Bounded Support. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1817–1825. Curran Associates, Inc., 2012.

Laura Rieger, Chandan Singh, W James Murdoch, and Bin Yu. Interpretations are useful: penalizing explanations to align neural networks with prior knowledge. *arXiv preprint arXiv:1909.13584*, 2019.

Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, 10(5):1299–1319, July 1998. ISSN 0899-7667, 1530-888X. doi: 10.1162/089976698300017467.

J. B. Tenenbaum. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319–2323, December 2000. ISSN 00368075, 10959203. doi: 10.1126/science.290.5500.2319.

Lauren van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of machine learning research*, pages 2579–2605, 2008.

Jarkko Venna, Jaakko Peltonen, Kristian Nybo, Helena Aidos, and Samuel Kaski. Information Retrieval Perspective to Nonlinear Dimensionality Reduction for Data Visualization. *Journal of Machine Learning Research*, 11(13):451–490, 2010. ISSN 1533-7928.