See Without Looking: Joint Visualization of Sensitive Multi-site Datasets*

Debbrata K. Saha,[1,2] **Vince D. Calhoun,**[1,2] **Sandeep R. Panta,**[2] **Sergey M. Plis**[1,2]
[1] University of New Mexico
[2] The Mind Research network
debbrata@unm.edu, vcalhoun@mrn.org, spanta@mrn.org, s.m.plis@gmail.com

Abstract

Visualization of high dimensional large-scale datasets via an embedding into a 2D map is a powerful exploration tool for assessing latent structure in the data and detecting outliers. There are many methods developed for this task but most assume that all pairs of samples are available for common computation. Specifically, the distances between all pairs of points need to be directly computable. In contrast, we work with sensitive neuroimaging data, when local sites cannot share their samples and the distances cannot be easily computed across the sites. Yet, the desire is to let all the local data participate in collaborative computation without leaving their respective sites. In this scenario, a quality control tool that visualizes decentralized dataset in its entirety via global aggregation of local computations is especially important as it would allow screening of samples that cannot be evaluated otherwise. This paper introduces an algorithm to solve this problem: decentralized data stochastic neighbor embedding (dSNE). Based on the MNIST dataset we introduce metrics for measuring the embedding quality and use them to compare dSNE to its centralized counterpart. We also apply dSNE to a multi-site neuroimaging dataset with encouraging results.

1 Introduction

Large-scale datasets have proven to be unreasonably effective in facilitating solutions to many difficult machine learning problems (Halevy *et al.*, 2009). High tolerance to mistakes and possible problems with individual data samples in applications relevant to internet businesses,[1] together with advances in machine learning methodologies (such as deep learning (Goodfellow *et al.*, 2016)) are able to effectively average out problems with individual samples lead to improved performance in recognition tasks. The story is different in the domains working with biomedical data, such as neuroimaging, where a data sample is a magnetic resonance image (MRI) of the entire brain containing on the order of 100,000 volumetric pixels (voxels). There, the data collection process for each sample is expensive, considerations of data privacy often prevent pooling data collected at multiple places, thus the datasets are not as large. Yet they are large enough to be difficult to manually vet each sample. An incorrect data sample may still lead to wrong conclusions and quality control is an important part of every analysis. Methods for simultaneous embedding of multiple samples are welcomed, as it is very difficult to scan through each and are used in practice for quality control (Panta *et al.*, 2016).

A common way of visualizing a dataset consisting of multiple high dimensional data points is embedding it to a 2 or 3-dimensional space. Such embedding can be an intuitive exploratory tool for quick detection of underlying structure and outliers. Although linear methods such as principal component analysis (PCA) provide the functionality they are not usually useful when there is a need to preserve and convey hidden nonlinear structure in the data. Many methods were developed for the task on nonlinear data embedding and visualization including Sammon mapping (Sammon Jr, 1969), curvilinear components analysis (Demartines and Hérault, 1997), Stochastic Neighbor Embedding (Hinton and Roweis, 2002), Isomap (Tenenbaum *et al.*, 2000), Maximum Variance Unfolding (Weinberger and Saul, 2006), Locally Linear Embedding (Roweis and Saul, 2000), Laplacian Eigenmaps (Belkin and Niyogi, 2003). The problem with these approaches is in their inability to retain local and global structure in a single map. A method to handle this situation efficiently was alternatively proposed: t-distributed stochastic neighbor embedding (t-SNE) (Van der Maaten and Hinton, 2008). The embeddings resulting from t-SNE applications are usually intuitive and interpretable which makes it an attractive tool for domain scientists (Bushati *et al.*, 2011; Panta *et al.*, 2016).

All of these methods, however, are built on the assumption that the input dataset is locally available. t-SNE, for example, needs computation of the pairwise distances between all samples (points) in the dataset. Yet, in many situations it is impossible to pull the data together into a single loca-

*This work was supported by NIH R01DA040487; SP was in part supported by NSF IIS-1318759

[1] It is expected that the mistakes average out and even if they do not the cost of displaying an image that the user did not request is low.

tion due to legal and ethical considerations. This is especially true for biomedical domain, where the risk of identifiability of anonymized data often prevents open sharing. Meanwhile, many systems allow virtually pooling datasets located at multiple research sites and analyzing them using algorithms that are able to operate on decentralized datasets (Carter *et al.*, 2015; Gaye *et al.*, 2014; Plis *et al.*, 2016). The importance of operating on sensitive data without pooling it together and thus generating truly large-scale neuroimaging datasets is so high that researchers successfully engage into manually simulating a distributed system (Thompson *et al.*, 2014). For all of the applications of the above systems quality control is essential and intuitive visualization of the complete virtual dataset physically spread across multiple locations is an important and much needed tool for filtering out participating sites with bad data, incorrect processing or simply mistakes in the input process.

In this paper we propose a way to embed into a 2D map a decentralized dataset that is spread across multiple locations such that the data at each location cannot be shared with others due to e.g. privacy concerns. Even the methods that are seemingly suitable to this setting (after a possible modification) do not seem to address the problem of inability to compute the distance between samples located at different sites. For example, Globerson *et al.* (2007) suggested a method of embedding multiple modalities into the same space. We could think of the modalities as our locations and modify their approach to our settings. This, however, is not straightforward as, again, the approach requires measuring co-occurrence, which transcends the borders of local sites.

We base our approach on availability of public anonymized datasets, which we use as a reference and build the overall embedding around it. This is most similar to the landmark points previously used for improving computational efficiency (De Silva and Tenenbaum, 2003, 2004). In fact, we start with a method that resembles the original landmark points approach. We show that it is not as flexible and does not produce good results. Then we introduce a dynamic modification that indeed is able to generate an embedding that reflects relations between points spread across multiple locations. Unlike the original landmark point approach, we use t-SNE as our base algorithm. We call the decentralized data algorithm dSNE. To evaluate the performance and to compare with the centralized version we use the MNIST dataset (LeCun *et al.*, 1998) and taking advantage of the known classes of the samples introduce a metric of overlap and roundness to quantify the comparisons. We evaluate and compare our algorithms in a range of various settings, establishing 4 experiments with MNIST data. Furthermore, we apply our approach to a truly multi-site neuroimaging dataset: ABIDE (Di Martino *et al.*, 2014).[2]

2 Methods

In the general problem of data embedding we are given a dataset of N points $\mathbf{X} = [\boldsymbol{x}_1 \ldots, \boldsymbol{x}_N]$, such that each point $\boldsymbol{x}_i \in \mathbb{R}^n$ with the task of producing another dataset $\mathbf{Y} = [\boldsymbol{y}_1 \ldots, \boldsymbol{y}_N]$, such that each point $\boldsymbol{y}_i \in \mathbb{R}^m$, where $m << n$. Usually $m = 2$ for convenience of visualization. Of course, this is an incomplete definition of the problem, as for any interesting results $\mathbf{Y}$ must be constrained by $\mathbf{X}$.

2.1 t-SNE

In t-SNE the distances between the points in $\mathbf{Y}$ must be as close to the distances between same points in $\mathbf{X}$ as possible given the weighted importance of preserving the relations with nearby points over those that are far. To achieve that, tSNE first converts the high dimensional Euclidean distances between datapoints into conditional probability that represent similarities (see Algorithm 1).

Algorithm 1 `PairwiseAffinities`

```
Input: $p$ (site index), $\rho$ (perplexity), $\mathbf{X} \in \mathbb{R}^{N \times C \times K_p}$
Output: $\mathbf{P}$
  Eq. (1) to compute $p_{ij}$ with perplexity $\rho$
  $\mathbf{P}_{ij} = (\mathbf{P}_{ji} + \mathbf{P}_{ij})/(2n)$
```

The similarity of datapoint x_j to datapoint x_i is the conditional probability, $p_{j|i}$, that x_i would pick x_j as its neighbor if neighbors were picked in proportion to their probability density under a Gaussian centered at x_i.

Algorithm 2 `tSNE`

```
Input:
    Data: $\mathbf{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2 \ldots \boldsymbol{x}_N], \boldsymbol{x}_i \in \mathbb{R}^n$
    Objective parameters: $\rho$ (perplexity)
    Optimization parameters: $T$ (number of iterations), $\eta$ (learning rate), $\alpha$ (momentum)
Output: $\mathbf{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N\}, \boldsymbol{y}_i \in \mathbb{R}^m, m << n$
  $\{p_{ij}\}$ = PairwiseAffinites $0, \rho, \mathbf{X}$
  $\mathbf{Y} \propto \mathcal{N}(0, 10^{-4}\mathbf{I}), \mathbf{I} \in \mathbb{R}^{m \times m}$ initialize from Gaussian
  for $i = 1$ to $T$ do
      Eq. (2) to compute low-dimensional affinities $q_{ij}$
      Eq. (3) to compute $\delta C$ / $\delta \boldsymbol{y}_i$
      $\boldsymbol{y}_i^t = \boldsymbol{y}_i^{t-1} + \eta(\delta C/\delta \boldsymbol{y}_i) + \alpha(t)(\boldsymbol{y}_i^{t-1} - \boldsymbol{y}_i^{t-2})$
  end for
```

At the same way we can compute q_{ij} from low dimensional output data. Algorithm 2 outlines the full procedure. In this algorithm, high dimensional and low dimensional pairwise affinities are formulated using equation (1) and (2) respectively. The gradient of the Kullback-Leibler divergence between $\mathbf{P}$ and the Student-t based joint probability distribution $\mathbf{Q}$ is expressed in (3).

$$p_{j|i} = \frac{\exp(-||\boldsymbol{x}_i - \boldsymbol{x}_j||^2 / 2\sigma_i(\rho)^2)}{\sum_{k \neq i} \exp(-||\boldsymbol{x}_i - \boldsymbol{x}_k||^2 / 2\sigma_i(\rho)^2)} \tag{1}$$

$$q_{ij} = \frac{(1 + ||\boldsymbol{y}_i - \boldsymbol{y}_j||^2)^{-1}}{\sum_{k \neq l} (1 + ||\boldsymbol{y}_k - \boldsymbol{y}_l||^2)^{-1}} \tag{2}$$

$$\frac{\delta C}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij})(\boldsymbol{y}_i - \boldsymbol{y}_j)(1 + ||\boldsymbol{y}_i - \boldsymbol{y}_j||^2)^{-1} \tag{3}$$

[2] http://fcon_1000.projects.nitrc.org/indi/abide/

Inspired by the overall quite satisfactory performance of t-SNE on a range of tasks, we base on it our algorithms for decentralized datasets.

2.2 Single-shot dSNE

In a scenario that we consider here, no data can leave a local site and thus it seems impossible to compute distances of samples across the sites. Without those distances (see equation (1)) we will not be able to obtain a common embedding. Fortunately, in neuroimaging not all data is private and unshareable. Public repositories of MRI data are popular and they provide diverse datasets for analyses (Castellanos *et al.*, 2013; Hall *et al.*, 2012; Ivory, 2015).

First we introduce some notation for sending and receiving messages. For a matrix $\mathbf{X}$, $\mathbf{X}^{\rightarrow p}$ means that it is sent to site p and $\mathbf{X}^{\leftarrow p}$ means that it is received from site p. We assume that a shared dataset is accessible to all local sites and the sites have it downloaded.

Algorithm 3 `singleshotDSNE`

```
Input:
    Objective parameters: $\rho$ (perplexity)
    Optimization parameters: $T$, $\eta$, $\alpha$
    Shared Data: $\mathbf{X}_s = [\boldsymbol{x}_1^s, \boldsymbol{x}_2^s \dots \boldsymbol{x}_{N_s}^s], \boldsymbol{x}_i^s \in \mathbb{R}^n$
    Data at site $p \forall p$: $\mathbf{X}_p = [\boldsymbol{x}_1^p, \boldsymbol{x}_2^p \dots \boldsymbol{x}_{N_p}^p], \boldsymbol{x}_i^p \in \mathbb{R}^n$
Output: $\mathbf{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_N\}, \boldsymbol{y}_i \in \mathbb{R}^m, m << n, N = \sum_p N_p + N_s$
$\mathbf{Y}_s \leftarrow$ tSNE $\mathbf{X}^s$, $\rho$, $T$, $\eta$, $\alpha$        ▷ At the master node
for $p = 0$ to $P$ do
    $\mathbf{Y}_s^{\rightarrow p}$
    Run tSNE on $[\mathbf{X}_p, \mathbf{X}_s]$        ▷ At local site $p$
    At each iteration only update $\mathbf{Y}_p$        ▷ At local site $p$
end for
$\mathbf{Y} \leftarrow []$        ▷ At the master
for $p = 0$ to $P$ do        ▷ At the master
    $\mathbf{Y}_p^{\leftarrow p}$
    $\mathbf{Y} \leftarrow [\mathbf{Y}, \mathbf{Y}_p]$
end for
$\mathbf{Y} \leftarrow [\mathbf{Y}, \mathbf{Y}_s]$
```

For Single shot d-SNE (Algorithm 3) we at first pass the reference data from centralized site C to each local site.

Now each local site's data consists of two portions. One is its local dataset, for which we need to preserve privacy, and another one is the shared reference dataset both comprise the combined datasets. Each local site runs the t-SNE algorithm on this combine data and produces an embedding into a low dimensional space. However, while computing each iteration of tSNE a local site computes gradient based on combined data, but it only updates the embedding vectors $\boldsymbol{y}$ for local datasets. The embedding for the shared data has been precomputed at the master node and shared with each local site. Similarly to the landmark points approach of De Silva and Tenenbaum our method uses reference points to tie together data from multiple sites. In practice the samples in the shared dataset are not controlled by the researchers using our method, and it is hard to assess the usefulness of each sample in the shared data in advance. In the end each local site obtains an embedding of its data together with the embedding of the shared dataset. Since the embedding points of the shared dataset did not change, all local embeddings are easily combined by aligning the points representing the shared data.

2.3 Multi-shot dSNE

Algorithm 4 `GradStep`

```
Input:
    Data embeddings: $\mathbf{Y}_p$ (local), $\mathbf{Y}_s$ (shared), $\mathbf{P}$
    Optimization parameters: $\eta$, $\alpha$
Output: $\hat{\mathbf{Y}}_p$ (local), $\hat{\mathbf{Y}}_s$ (shared)
Eq. (2) to compute low-dimensional affinities $q_{ij}$
Eq. (3) to compute $\delta C$ / $\delta \boldsymbol{y}_i$
$\hat{\boldsymbol{y}}_i = \eta(\delta C / \delta \boldsymbol{y}_i) + \alpha(\boldsymbol{y}_i^{t-1} - \boldsymbol{y}_i^{t-2})$
group $\hat{\boldsymbol{y}}_i$ into $\hat{\mathbf{Y}}_p$ (local) and $\hat{\mathbf{Y}}_s$ (shared)
```

Singeshot dSNE, however, tends to produce results with significant overlap as the data from different sites does not affect each other and often end up in the same location despite belonging to different classes. To overcome this problem we developed an iterative algorithm: multishot d-SNE. Unlike the single shot version we allow the embedding of the shared dataset to change. However, we change it exactly the same way for each site. This is achieved by averaging the updates to shared data embedding based on the updates that each of the local sites requires to be made. This allows information about desirable embedding of the local data points to flow across sites and thus communicate preferred location through the influence on the shared data. Algorithm 6 details all of the steps.

Algorithm 5 `UpdateStep`

```
Input:
    Data embeddings: $\mathbf{Y}_p$, $\mathbf{Y}_s$, $\hat{\mathbf{Y}}_p$, $\hat{\mathbf{Y}}_s$
Output: $\mathbf{Y}_p$, $\mathbf{Y}_s$
$\mathbf{Y}_p = \mathbf{Y}_p + \hat{\mathbf{Y}}_p$
$\mathbf{Y}_s = \mathbf{Y}_s + \hat{\mathbf{Y}}_s$
```

2.4 Comparison Metrics

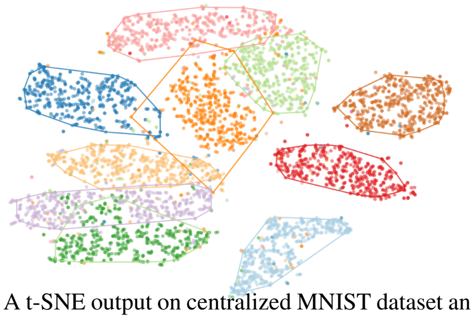


Figure 1: A t-SNE output on centralized MNIST dataset and outlier-free convex hull boundaries for each digit.

An objective comparison of inherently subjective visualization algorithms is difficult. We quickly found that the k-

Algorithm 6 `multishotDSNE`

```
Input:
    Objective parameters: $\rho$ (perplexity)
    Optimization parameters: $T$, $\eta$, $\alpha$
    Shared Data: $\mathbf{X}_s = [\boldsymbol{x}_1^s, \boldsymbol{x}_2^s \dots \boldsymbol{x}_{N_s}^s], \boldsymbol{x}_i^s \in \mathbb{R}^n$
    Data at site $p \forall p$: $\mathbf{X}_p = [\boldsymbol{x}_1^p, \boldsymbol{x}_2^p \dots \boldsymbol{x}_{N_p}^p], \boldsymbol{x}_i^p \in \mathbb{R}^n$
Output: $\mathbf{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_N\}, \boldsymbol{y}_i \in \mathbb{R}^m, m << n, N = \sum_p N_p + N_s$
$\mathbf{Y}_s \propto \mathcal{N}(0, 10^{-4}\mathbf{I}), \mathbf{I} \in \mathbb{R}^{m \times m}$ initialize from Gaussian
for $p = 0$ to $P$ do                                    ▷ Initialize at sites
    $\mathbf{Y}_s^{\rightarrow p}$
    $\mathbf{P}_p \leftarrow$ PairwiseAffinities $p, \rho, [\mathbf{X}_p, \mathbf{X}_s]$
    $\mathbf{Y}_p \propto \mathcal{N}(0, 10^{-4}\mathbf{I}), \mathbf{I} \in \mathbb{R}^{m \times m}$
end for
for $i = 0$ to $T$ do
    for $p = 0$ to $P$ do                                ▷ At local sites
        $\hat{\mathbf{Y}}_p, \hat{\mathbf{Y}}_s \leftarrow$ GradStep $[\mathbf{Y}_p, \mathbf{Y}_s, \mathbf{P}_p]$
    end for
    $\hat{\mathbf{Y}} \leftarrow 0$                                          ▷ At the master
    for $p = 0$ to $P$ do                                ▷ At the master
        $\hat{\mathbf{Y}}_s^{\leftarrow p}$
        $\hat{\mathbf{Y}} \leftarrow \frac{1}{P}\hat{\mathbf{Y}}_s$                          ▷ Average local $\hat{\mathbf{Y}}_s$
    end for
    for $p = 0$ to $P$ do                                ▷ At local sites
        $\hat{\mathbf{Y}}^{\rightarrow p}$
        $\mathbf{Y}_p, \mathbf{Y}_s \leftarrow$ UpdateStep $[\mathbf{Y}_p, \mathbf{Y}_s, \hat{\mathbf{Y}}_p, \hat{\mathbf{Y}}]$
    end for
end for
```

means criterion is only weakly correlated with the usefulness of produced embeddings:

$$\alpha = \frac{\sum_{d=0}^{9} \sum_{S \epsilon X_d} ||\boldsymbol{\mu}_d - \boldsymbol{x}_s^d||_2}{\sum_{(i,j),(i>j),(i \neq j)} ||\boldsymbol{\mu}_i - \boldsymbol{\mu}_j||_2} \qquad (4)$$

In an attempt to quantify perceptional quality of the resulting embeddings we have developed two additional metrics: overlap and roundness. To remove sensitivity to noise we first remove the outliers [see Figure 1] in each digit's cluster (Liu *et al.*, 2012). Then compute the convex hull for each digit and use them to compute the measure of the overlap (the sum of all polytope areas minus the area of the union of all the polytopes normalized by this union's area) and the roundness (ratio of the area of each polytope to the area of the circumscribed circle).

3 Results

We base our experiments in this section on two datasets: MNIST (LeCun *et al.*, 1998) for handwritten images of all digits in the 0 to 9 range, and Autism Brain Imaging Data Exchange (ABIDE) for fMRI data (Di Martino *et al.*, 2014). MNIST data were taken from a Kaggle competition[3] which has $28,000$ gray-scale images of handwritten digits. Each image contains $28 \times 28 = 784$ pixels. Among these data, we randomly (but preserving class balance) pick 5,000 different samples from the data set for our needs. At first, we reduce dimension of the data from 784 to 50 using PCA. Then, dSNE and tSNE are used to generate (x, y) coordinates in two dimensional space. The second dataset is the ABIDE fMRI set, which contains data of 1153 subjects. The ABIDE data has been pre-processed down to multiple spatial and temporal quality control (QC) measures[4]. For ABIDE, because of the low dimension of QC measures, we do not use dimensionality reduction but directly run our tSNE and dSNE to produce the embeddings.

3.1 MNIST Experiments

Experiment 1 (No diversity in the reference data): In this experiment, the reference dataset contains a single digit that is also present at all of the three (3) local sites. We run an experiment for each of the 10 digits. Each site contains 400 samples for each of its corresponding digits. Reference dataset contains 100 samples of its digit.

Experiment 2 (Effect of the sample size): Often centralized stores accumulate more data than any separate local site can contain. Our goal is to check the adaptability of our algorithm on this case. In this experiment, every local site contains only one digit and the reference dataset contains all digits (0-9). We consider 2 cases: when each site contains 400 samples and each digit in the reference consists of 100 samples; and the inverse case, when sites only have 100 samples, while the reference digits are represented by 400.

Experiment 3 (Missing digit in reference data): In this experiment we investigate the effect of the case when a digit is missing from shared data. This approximates the case of unique conditions at a local site. Each local site out of 10 contains a single digit. We run 10 experiments; in each, the reference dataset is missing a digit. For each of the experiments we have 2 conditions: in one the reference dataset is small (100 samples for each digit but the missing one) while the sites are large (400 samples per site); in another the reference data is large (400 samples per digit) and the sites are small (only 100 samples).

Experiment 4 (Effect of the number of sites): In this experiment, we investigate whether the overall size of the virtual dataset affects the result. Every local site, as well as the reference data, contains all digits (0-9) 20 samples per digit. We continuously increase the number of sites from 3 to 10. As a result, the total number of samples across all sites increases.

3.2 Single-shot

Figure 2 shows examples of the output generated by the single shot dSNE algorithm together with tSNE layout of the centralized data for comparison. We run single shot for the datasets of Experiment 2, Experiment 3, and Experiment 4. For all experiments, we are able to correctly group and embed same digits from different sites. However, the problem

[3] `https://www.kaggle.com/c/digit-recognizer`

[4] The full list is available here: `https://github.com/preprocessed-connectomes-project/quality-assessment-protocol/tree/master/normative_data`

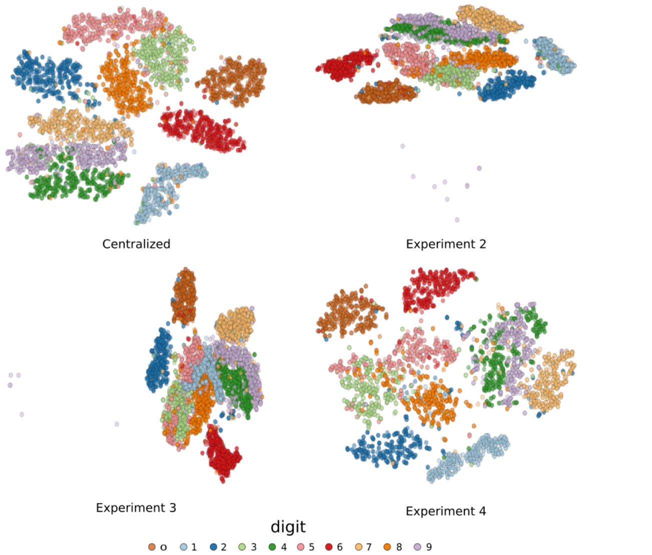


Figure 2: Single Shot dSNE examples. Digits are correctly grouped into clusters that tend to heavily overlap.

was that the digit clusters tended to heavily overlap. Even for experiment 4, where we found the best results, one can still observe heavy overlap of digit clusters.

3.3 Multi-shot

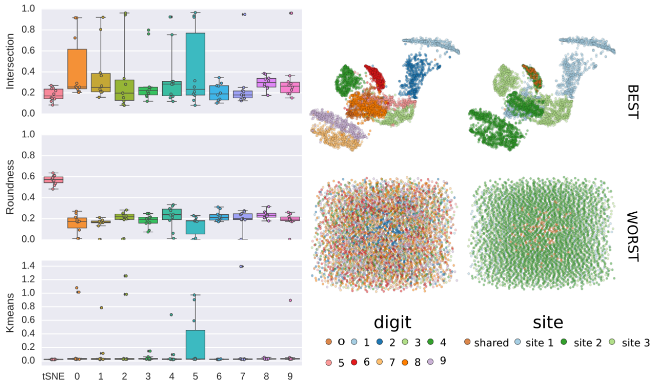


Figure 3: Experiment 1: Single digit in the reference dataset.

The multi-shot algorithm was introduced to cope with the overlap problem. Figure 3 shows its performance in experiment 1. In this figure, the plots on the left show performance metrics in comparison to those of tSNE on centralized data. For each digit we rerun the experiment 10 times with a different seed value. On the right, the top figure presents the best, and the lower figure represents the worst performing run. The layouts are the same but colored by digits and by sites. From the analysis of the results, we find that each common digit from different sites is embedded in the plots perfectly. But, for many digits, the clusters are less separable.

Figure 4 represents the result of experiment 2. The comparison metrics show that when the shared portion contains large amount of data the metrics are better than in the case of smaller number of samples in the reference dataset. However, the cluster roundness degrades with the size of the sample in the shared data. dSNE clusters are less round compared to the centralized tSNE.

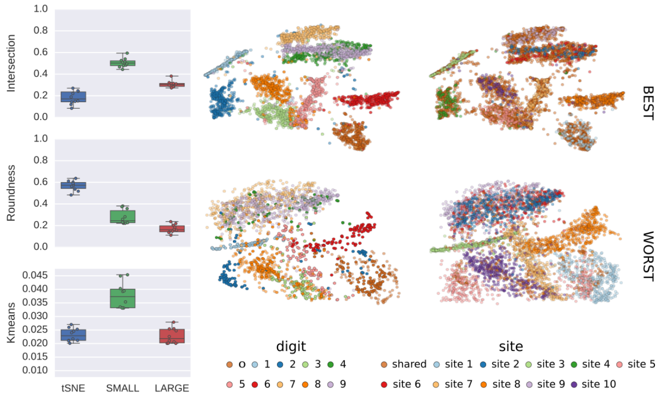


Figure 4: Experiment 2: Reference data contains samples of all digits but it is either small or large (details in the text).

Figure 5 depicts the results of Experiment 3. Although there is some variability with respect to which digit is missing from the reference dataset, this is not the largest effect. Here, we observe as similar behavior as in experiment 2: larger size of the reference dataset leads to better results. Note, when the reference dataset contains a large number of samples, we always get better results for Kmeans and Intersection Ratio compared to the case of fewer samples in the reference. Nevertheless, a visual inspection of the best results in the case of smaller reference dataset size does not lead to conclusions that the results are unusable.

Figure 6 depicts the results of Experiment 4. The effect of sample size is the most pronounced for tSNE output, while the metrics remain fairly constant on the multishot dSNE embeddings regardless of the size of the virtual dataset. This is a surprising outcome. We present best and worst embedding based on the number of different local sites in our decentralized scenario.

3.4 Real Data

We investigate performance of multishot dSNE in comparison with the embedding produced by tSNE on the pulled data using the QC metrics of the ABIDE dataset. To simulate a consortium of multiple sites we randomly split these data into ten local and one reference datsets. Results show 10 different clusters for centralized data. For three random splits of our decentralized simulation we obtained 10 different clusters as well (see Figure 7). Notably, the split into the clusters in the embedding is stable regardless of the split into sites.

4 Conclusions

Our approach enables embedding and visualization of high dimensional private data spread around multiple sites. The data does not leave the sites and only minimal gradient information from the embedding space gets transferred across the sites. We consider this approach plausibly private as most of the information about individual samples was discarded. Extensive tests on MNIST demonstrate the usefulness of the

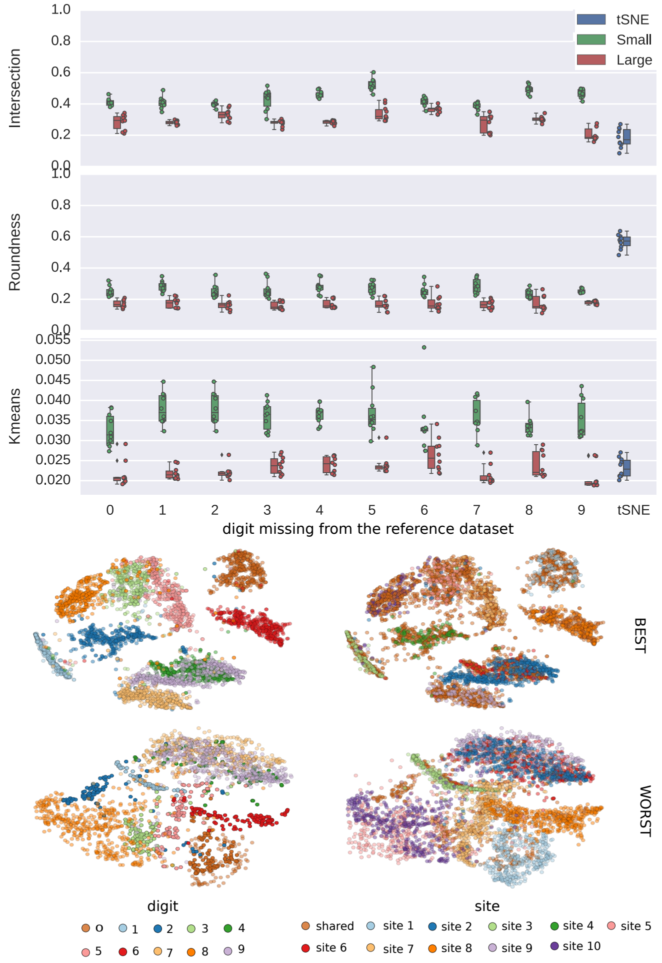


Figure 5: Experiment 3: the reference dataset is missing a digit that is present at one of the local sites.

approach and high quality of the obtained embeddings over a variety of settings. Although multi-shot dSNE is quite robust to various conditions, such as changes in the number of sites, rare or missing data etc., the best performance is achieved when the reference dataset is dense. Notably, the single-shot dSNE, which is mostly an implementation for tSNE of the previously existed landmark point method, tends to ignore the differences across the sites and only respects the reference dataset. An alternative solution—an average of the gradients weighted by the quality of their respective local tSNE—may even further bias results toward good local groupings but unacceptable overall embedding. In contrast, our multi-shot approach provides enough information propagation to warrant better embeddings. Yet, all that is being exchanged is pertinent to the already public reference data. We conclude that dSNE is a valuable quality control tool for virtual consortia working with private data in decentralized analysis setups.

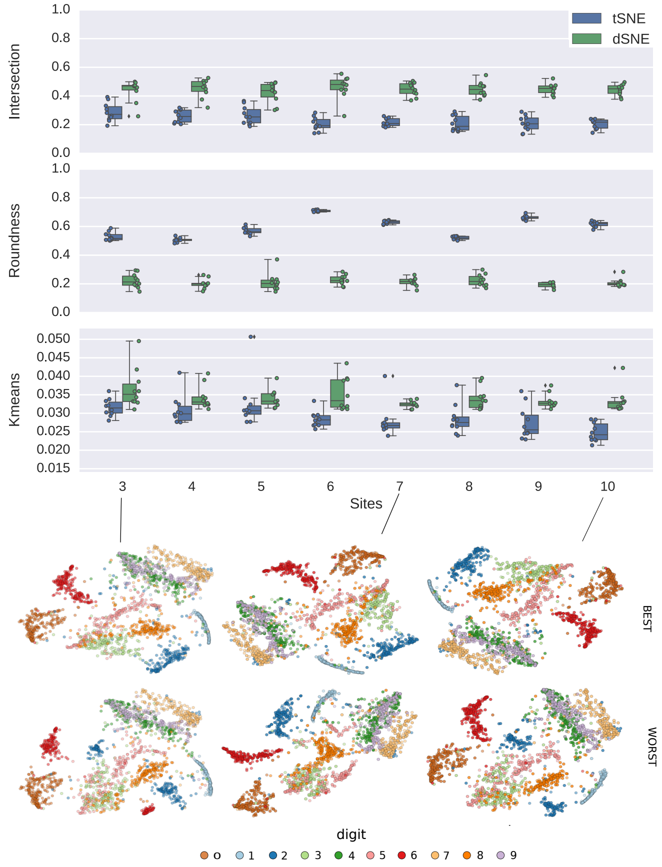
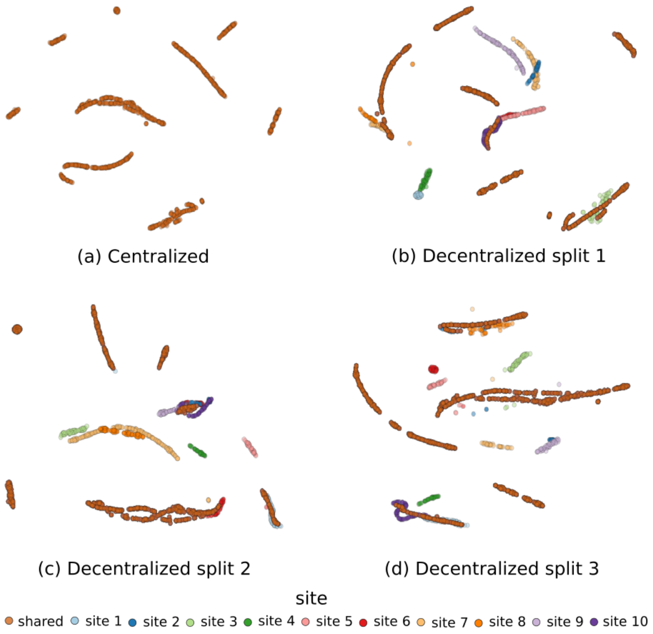


Figure 6: Experiment 4: Gradual increase of number of sites.



(a) Centralized

(b) Decentralized split 1

(c) Decentralized split 2

(d) Decentralized split 3

Figure 7: Experiment for QC metrics of the ABIDE datasets.

References

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Natascha Bushati, James Smith, James Briscoe, and Christopher Watkins. An intuitive graphical visualization technique for the interrogation of transcriptome data. *Nucleic acids research*, page gkr462, 2011.

Kim W Carter, Richard W Francis, KW Carter, RW Francis, M Bresnahan, M Gissler, TK Grønborg, R Gross, N Gunnes, G Hammond, et al. Vipar: a software platform for the virtual pooling and analysis of research data. *International journal of epidemiology*, page dyv193, 2015.

F Xavier Castellanos, Adriana Di Martino, R Cameron Craddock, Ashesh D Mehta, and Michael P Milham. Clinical applications of the functional connectome. *Neuroimage*, 80:527–540, 2013.

Vin De Silva and Joshua B Tenenbaum. Global versus local methods in nonlinear dimensionality reduction. *Advances in neural information processing systems*, pages 721–728, 2003.

Vin De Silva and Joshua B Tenenbaum. Sparse multidimensional scaling using landmark points. Technical report, Technical report, Stanford University, 2004.

Pierre Demartines and Jeanny Hérault. Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on neural networks*, 8(1):148–154, 1997.

Adriana Di Martino, Chao-Gan Yan, Qingyang Li, Erin Denio, Francisco X Castellanos, Kaat Alaerts, Jeffrey S Anderson, Michal Assaf, Susan Y Bookheimer, Mirella Dapretto, et al. The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular psychiatry*, 19(6):659–667, 2014.

Amadou Gaye, Yannick Marcon, Julia Isaeva, Philippe LaFlamme, Andrew Turner, Elinor M Jones, Joel Minion, Andrew W Boyd, Christopher J Newby, Marja-Liisa Nuotio, et al. DataSHIELD: taking the analysis to the data, not the data to the analysis. *International journal of epidemiology*, 43(6):1929–1944, 2014.

Amir Globerson, Gal Chechik, Fernando Pereira, and Naftali Tishby. Euclidean embedding of co-occurrence data. 8(Oct):2265–2295, 2007.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT Press, 2016.

Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8–12, 2009.

Dan Hall, Michael F Huerta, Matthew J McAuliffe, and Gregory K Farber. Sharing heterogeneous data: the national database for autism research. *Neuroinformatics*, 10(4):331–339, 2012.

Geoffrey Hinton and Sam Roweis. Stochastic neighbor embedding. In *NIPS*, volume 15, pages 833–840, 2002.

Mary Ivory. Federal interagency traumatic brain injury research (fitbir) bioinformatics platform for the advancement of collaborative traumatic brain injury research and analysis. In *143rd APHA Annual Meeting and Exposition (October 31-November 4, 2015)*. APHA, 2015.

Yann LeCun, Corinna Cortes, and Christopher JC Burges. The mnist database of handwritten digits, 1998.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 6(1):3, 2012.

Sandeep R Panta, Runtang Wang, Jill Fries, Ravi Kalyanam, Nicole Speer, Marie Banich, Kent Kiehl, Margaret King, Michael Milham, Tor D Wager, et al. A tool for interactive data visualization: Application to over 10,000 brain imaging and phantom mri data sets. *Frontiers in neuroinformatics*, 10, 2016.

Sergey M Plis, Anand D Sarwate, Dylan Wood, Christopher Dieringer, Drew Landis, Cory Reed, Sandeep R Panta, Jessica A Turner, Jody M Shoemaker, Kim W Carter, et al. Coinstac: a privacy enabled model and prototype for leveraging and processing decentralized brain imaging data. *Frontiers in Neuroscience*, 10, 2016.

Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, (5500):2323–2326, 2000.

John W Sammon Jr. A nonlinear mapping for data structure analysis. *Computers, IEEE Transactions on*, 100(5):401–409, 1969.

Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

Paul M Thompson, Jason L Stein, Sarah E Medland, Derrek P Hibar, Alejandro Arias Vasquez, Miguel E Renteria, Roberto Toro, Neda Jahanshad, Gunter Schumann, Barbara Franke, et al. The ENIGMA consortium: large-scale collaborative analyses of neuroimaging and genetic data. *Brain imaging and behavior*, 8(2):153–182, 2014.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

Kilian Q Weinberger and Lawrence K Saul. An introduction to nonlinear dimensionality reduction by maximum variance unfolding. In *AAAI*, volume 6, pages 1683–1686, 2006.