iPMDS

Vu, Viet Minh; Bibal, Adrien; Frénay, Benoît

Published in:
International Joint Conference on Neural Networks

Publication date:
2021

Document Version
Peer reviewed version

iPMDS: Interactive Probabilistic Multidimensional Scaling

Viet Minh Vu
Faculty of Computer Science - NaDI
University of Namur
Namur, Belgium
vuvietminh@unamur.be

Adrien Bibal
Faculty of Computer Science - NaDI
University of Namur
Namur, Belgium
adrien.bibal@unamur.be

Benoît Frénay
Faculty of Computer Science- NaDI
University of Namur
Namur, Belgium
benoit.frenay@unamur.be

***Abstract*—Dimensionality reduction is often used for visualization without considering their understanding by users. Multidimensional scaling, for instance, provides an arbitrarily-oriented visualization. However, users can be integrated into the loop to provide clues about their understanding of the visualization. In this paper, we propose an interactive probabilistic multidimensional scaling (iPMDS) approach to compute the visualization with the lowest information loss while taking the information provided by users into account. We show that a more interpretable visualization can be obtained after interacting with the visualization while keeping a good dimensionality reduction accuracy.**

***Index Terms*—Dimensionality Reduction, Multidimensional Scaling (MDS), Probabilistic Model, User Interaction**

I. Introduction

With large datasets becoming usual today, two-dimensional data visualization becomes paramount. Visualization allows users to explore data and to get insights about them. Thanks to visualization, users can identify trends, as well as clusters.

Dimensionality reduction is a field of machine learning whose goal is to reduce the number of dimensions of a given dataset. For exploratory purposes, the dimension of the dataset can be reduced to 2 and then be visualized. Many algorithms exist to project data in a 2-dimensional space and to perform visualization, such as principal component analysis (PCA) [1], multidimensional scaling (MDS) [2], [3] and t-distributed stochastic neighbor embedding (t-SNE) [4].

Some techniques, like MDS, are heavily used in social sciences like psychology, despite some flaws that make its visualization less interpretable. Indeed, MDS provides a visualization where the pairwise distances between the instances match the dissimilarities between the instances provided as input (using an error measure called the *stress*). As the stress is invariant to the rotation of the visualization, the final visualization can have an arbitrary orientation. However, this arbitrary orientation may not be the most suitable to interpret the visualization, and the user input may be necessary.

In order to tackle the above issue, this work introduces a human-in-the-loop MDS through a new version of probabilistic MDS. By moving points in the 2-dimensional space created by a probabilistic MDS, users provide information on what would be interesting to them under the form of a prior. Our interactive probabilistic MDS (iPMDS) then recomputes the visualization while taking into account the information given by the user. In a qualitative experiment, we show that our technique makes it possible to provide a view of the data that is interpretable by users while keeping a low stress.

In order to introduce iPMDS, Sec. II first presents methods in the literature to add interaction to MDS. Sec. III then introduces iPMDS, our contribution. As we deal with a problem that is quantitatively difficult to assess, several case studies are provided in Sec. IV to show the performance of iPMDS. Finally, Sec. V concludes the paper.

II. Background and Related Work

The problem addressed by our work is to integrate information provided by users, such as their knowledge about the dataset, their needs, or feedback about their desired visualization into the MDS visualization. The background on MDS and existing probabilistic approaches are first introduced in Sec. II-A. Several related methods for incorporating user feedback into MDS are then presented in Sec. II-B.

A. Background on MDS and Probabilistic Approaches

MDS is a family of methods that position points in a target space in which the pairwise distances should reflect similarities or dissimilarities in the original space [5]. In this work, we focus on metric MDS, where the (dis)similarities are measured by a metric distance. Metric MDS preserves the global structure of the data by preserving the inter-points distances in the high-dimensional space. The distance preserving criterion is measured by the Kruskal's *stress* $\sqrt{\sum_{1\leq i,j\leq N}\left(d_{ij}-D_{ij}\right)^2}$, where N is the number of points in the dataset, and D_{ij}, d_{ij} are the distances between the i^{th} and j^{th} instances in the high and low-dimensional spaces respectively [3].

Another way to formulate the MDS problem is to consider it as a pairwise distance matching problem [6]. This problem requires a probabilistic model such that the distances generated from the model are similar to the observed input distances. This probabilistic approach consists of two separate steps. The first step is to posit a probabilistic model to characterize the distribution of distances in any metric space. Several existing methods simply model the distances by real values using a Normal distribution [7] or by non-negative values using a Log-Normal distribution [6], [8]. The second step is to infer model parameters to fit the observed input distances. Different inference algorithms are used such as like maximum a posteriori estimation (in [7]), variational inference (in [6]), or Markov Chain Monte Carlo (in [8]). Recent advances in gradient-based methods make it possible to efficiently solve the inference step and drive us to focus effort on the modeling step. In this work, we focus not only on building a model that accurately represents distances while being computationally efficient but also on modeling additionally injected information from users in an interactive setting.

B. Related Work on Interactive MDS

The main motivation for an interactive MDS method is that the MDS visualization can have any arbitrary orientation. In order to explore and understand the visual patterns in the MDS results, users can apply methods that find the best rotation for interpretation if external features are available [9], [10]. In this work, we do not consider external features to find the best rotation, but directly the user's feedback instead.

Another solution is to let users interact directly with the visualization to freely control and explore different orientations. The typical interactive strategy used in the literature and in our method is to fix several anchor points and observe the positions of other points. Users can fix the anchor points based on their prior knowledge or by observing an initial visualization and determining the desired positions for their points of interest. XGvis [11] is one of the early interactive systems designed to manipulate points in the MDS visualization. XGvis uses a custom stress function and gradient descent to optimize the positions of the non-fixed points.

Another issue of MDS and of most dimensionality reduction (DR) methods is that it may produce different visualizations that are equivalent, up to some transformation, leading to different interpretations. Therefore, in order to select one, it is necessary to enable users to inject their domain knowledge or feedback into the visualization. More generally, the visualization needs to be adapted to the needs of users in exploratory data analysis [12]. Several frameworks are created for interactive visual analytic and are tailored to integrate users' feedback into MDS visualizations. The *Bayesian Visual Analysis* (BaVA) framework [13] and its deterministic version, the *visual to parametric interaction* (V2PI) framework [14], focus on how to transform the user's cognitive feedback into a parametric feedback used in DR algorithms like MDS or probabilistic PCA. *User-guided MDS* [15] is a concrete example of applying V2PI framework on *weighted MDS* [16], a modified version of MDS in which each distance is assigned a particular weight. When users fix the position of their points of interest, these frameworks update the corresponding weights of the related points in order to produce a new MDS embedding that reflects the relative distances of the fixed points.

Our work is based on the same idea as BaVa and V2PI of transforming the user's feedback into a parameterized term that can be optimized in MDS. However, we characterize the pairwise distances for MDS and model user's feedback in a unified probabilistic model, which is detailed in the following section. The main advantage of our modeling approach is that, with the additional information from users, the expressiveness of the distance model can be increased, i.e., the model can well explain not only the observed distances but also the injected feedback at the same time.

III. INTERACTIVE PROBABILISTIC MDS MODEL

The main contribution of this work is a latent variable probabilistic MDS model with a flexible prior to encode the need of users about their desired visualization. Sec. III-A introduces a specific model to represent the distribution of distances, which is used in the MDS problem in Sec. III-B. Sec. III-C details our proposed interactive method of integrating user knowledge into the probabilistic MDS model.

A. Hefner Distance Model

The general idea of the distance model is to model the distribution of the Euclidean pairwise distances between the data points. In general, the distances D_{ij} between two data points i and j are observed (measured or calculated), while the data points themselves are unknown. We can assume that each point lies in a r-dimensional space $\mathbf{z}_i = [\mathrm{z}_{i1}, \dots, \mathrm{z}_{ir}] \in \mathbb{R}^r$. The goal is to characterize the distribution of distances $D_{ij} = \sqrt{\sum_{k=1}^{r} (\mathrm{z}_{ik} - \mathrm{z}_{jk})^2}$.

Since the exact coordinate of each data point is unknown, Hefner proposes to represent each data point i by a specific *location parameter* $\boldsymbol{\mu}_i$ and a local *variability parameter* $\boldsymbol{\sigma}_i^2$ [17]. The data points $\mathbf{z}_i, \mathbf{z}_j$ are considered as multivariate Gaussian random variables. In this setting, the distance D_{ij} is not a scalar value but a random variable since we can sample different points from the two multivariate Gaussian distributions of $\mathbf{z}_i, \mathbf{z}_j$ and calculate the squared Euclidean distance between them. The sampled distances are illustrated by the dotted lines in Figure 1. To find the distribution of distances, the Hefner model uses the following theorem.

Theorem 1: Let $\{\mathbf{t}_1, \dots, \mathbf{t}_r\}$ be r independently and normally distributed random variables with mean μ_k and unit variance $\sigma_k^2 = 1$. A new random variable $\mathbf{t} = \sum_{k=1}^{r} \mathbf{t}_k^2$ is distributed by a noncentral chi-squared distribution $\chi^2(r, \lambda)$ with r degrees of freedom and a noncentrality $\lambda = \sum_{k=1}^{r} \mu_k^2$.

Theorem 1 can be applied for the Gaussian variables with $\sigma^2 \neq 1$, simply by standardizing the variable to obtain unit variance (i.e., dividing by the standard deviation). In the Hefner model, each component in a r-dimensional data point $\mathbf{z}_i$ is modeled by a Gaussian with a location parameter μ_{ik} and a variance σ_i^2. Applying a classical result for the difference of two Gaussian random variables, we obtain

$$\mathrm{z}_{ik} \sim \mathcal{N}(\mu_{ik}, \sigma_i^2) \quad (1a)$$

$$\mathrm{z}_{jk} \sim \mathcal{N}(\mu_{jk}, \sigma_j^2) \quad (1b)$$

$$\mathrm{z}_{ik} - \mathrm{z}_{jk} \sim \mathcal{N}(\mu_{ik} - \mu_{jk}, \sigma_i^2 + \sigma_j^2). \quad (1c)$$

Applying Theorem 1 of noncentral chi-squared distribution to the Hefner model, we obtain

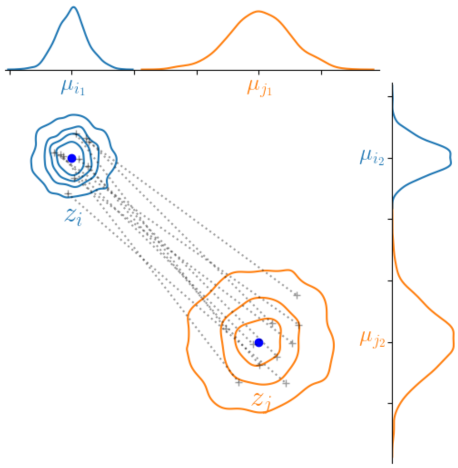


Fig. 1: Illustration for a distance model. Two points $\mathbf{z}_i$ and $\mathbf{z}_j$ are represented by their locations $\boldsymbol{\mu}_i, \boldsymbol{\mu}_j$ and their variances shown by the contours around their locations. The distance between $\mathbf{z}_i$ and $\mathbf{z}_j$ is a random variable that can be estimated by sampling the first point from the distribution of $\mathbf{z}_i$, sampling the second point from the distribution of $\mathbf{z}_j$ and calculating the Euclidean distance between the two sampled points. The dotted lines represent the sample distances. The Hefner distance model [17] characterizes the distribution of the sample distances represented by these dotted lines.

$$\sum_{k=1}^{r} \frac{(\mathrm{z}_{ik} - \mathrm{z}_{jk})^2}{\sigma_i^2 + \sigma_i^2} \sim \chi^2\Big(r, \lambda = \sum_{k=1}^{r} \frac{(\mu_{ik} - \mu_{jk})^2}{\sigma_i^2 + \sigma_i^2}\Big) \tag{2a}$$

$$\frac{D_{ij}^2}{\sigma_{ij}^2} \sim \chi^2\Big(r, \frac{d_{ij}^2}{\sigma_{ij}^2}\Big), \tag{2b}$$

where $d_{ij}^2 = \sum_{k=1}^{r}(\mu_{ik} - \mu_{jk})^2$ and $\sigma_{ij}^2 = \sigma_i^2 + \sigma_j^2$. Hence, the standardized squared pairwise distances $\frac{D_{ij}^2}{\sigma_{ij}^2}$ follow a noncentral chi-squared distribution with r degrees of freedom and a noncentrality parameter $\lambda = \frac{d_{ij}^2}{\sigma_{ij}^2}$. This distribution is parameterized by $\boldsymbol{\mu} = [\mu_{ik}]$ with $i = 1..N$ and $k = 1..r$, and $\boldsymbol{\sigma}^2 = [\sigma_i^2]$ with $i = 1..N$.

After obtaining the distribution of $\frac{D_{ij}^2}{\sigma_{ij}^2}$, the goal is to characterize the distribution of D_{ij}. Let us denote G the cumulative distribution function (CDF) of a noncentral chi-squared distribution of $\frac{D_{ij}^2}{\sigma_{ij}^2}$, and F the CDF of our target quantity D_{ij}. Classical results for the distribution function give $F(D_{ij}) = G\left(\frac{D_{ij}^2}{\sigma_{ij}^2}\right)$. Taking the derivative of these CDFs gives us the probability density function (PDF) function

$$f(D_{ij}) = \frac{2D_{ij}}{\sigma_{ij}^2} g\left(\frac{D_{ij}^2}{\sigma_{ij}^2}\right), \tag{3}$$

where g is the PDF function of a noncentral chi-squared random variable. The PDF of a random variable $\mathbf{x}$ distributed by a noncentral chi-squared distribution (with a degree of freedom k and a noncentrality parameter λ) is defined as

$$\begin{aligned} g(\mathbf{x}; k, \lambda) &= \chi^2(r, \lambda) \\ &= \frac{1}{2} \exp \frac{-(\mathbf{x}+\lambda)}{2} \left(\frac{\mathbf{x}}{\lambda}\right)^{\frac{k}{4}-\frac{1}{2}} I_{\frac{k}{2}-1}(\sqrt{\lambda \mathbf{x}}), \end{aligned} \tag{4}$$

where $I_\nu(\mathbf{y})$ is a modified Bessel function of the first kind of degree ν

$$I_\nu(\mathbf{y}) = \left(\frac{\mathbf{y}}{2}\right)^\nu \sum_{i=0}^{\infty} \frac{(\mathbf{y}^2/4)^i}{i!\,\Gamma(\nu+i+1)}. \tag{5}$$

In summary, the Hefner model represents the distribution of Euclidean distances between points in an r-dimensional space by the density function defined in (3). This representation is used to address the distance preserving problem, which is detailed in the following section.

B. Probabilistic Distance Preserving Model

The Hefner model represents distances between data points in an r-dimensional space. If r is equal to the original dimension of the data, the pairwise distances D_{ij} can be reconstructed exactly. However, if r is smaller than the original dimension, the original pairwise distances can only be approximated in this lower-dimensional space. In the case of dimensionality reduction, the data points in the high dimensional (HD) space are considered unknown, and we only observe the pairwise distances $\boldsymbol{D} = [D_{ij}]$. The goal is to reduce the dimensionality of the data to r (to represent the data points in an r-dimensional space), where r is usually very small. This is a typical metric MDS problem, i.e., finding a configuration of data points in a low-dimensional (LD) space where pairwise distances d_{ij} match distances D_{ij} in the HD space.

Probabilistic MDS (PMDS) is a probabilistic approach to solve this problem. PMDS uses the Hefner model to represent the pairwise distances in a LD space ($r = 2$ or 3). In PMDS, the distances $d_{ij} = \sqrt{\sum_{k=1}^{r}(\mu_{ik} - \mu_{jk})^2}$ in the LD space are used to approximate the true input distances D_{ij}. PMDS uses the likelihood function $p(D_{ij}|d_{ij})$ to measure how well we can approximate the original distance D_{ij} of points in the HD space using the distance d_{ij} in the LD space. The locations and variances $(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$ are parameters of the model and are estimated by maximum likelihood.

C. Our Interactive Latent Variable Probabilistic MDS Model

Our goal is to integrate prior knowledge, feedback, or constraints of the user into an MDS visualization. With the probabilistic formulation of MDS problem, our idea is to encode these kinds of user's information into the prior distribution of the position of the points in the embedding. We propose a latent variable PMDS model in which the locations and variances are latent variables to be inferred. With this new hierarchical model shown in Figure 2, we can introduce a prior distribution on $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ and use maximum a posteriori (MAP) estimation to infer these latent variables. In a simplified version of our model, we impose an assumption of equal variance for all points $\sigma_i = \sigma_j = \text{constant}, \forall i, j \in [1, N]$

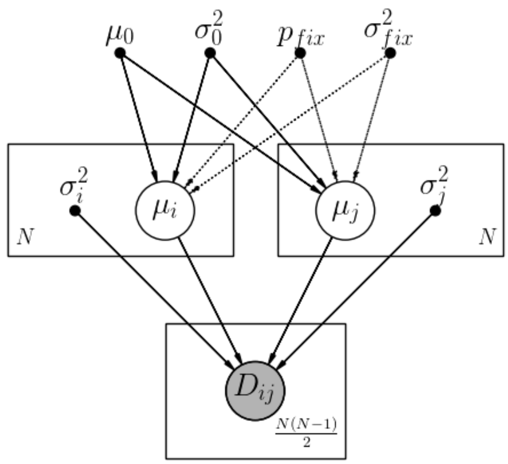


Fig. 2: Graphical model for the proposed latent variable interactive probabilistic MDS model. The latent variables $\boldsymbol{\mu}_i, \boldsymbol{\mu}_j$ are Gaussian random variables with mean $\boldsymbol{\mu}_0$ and variance σ_0^2. The fixed position of points indicated by the user are also encoded in the prior of $\boldsymbol{\mu}$ using the specific positions $\boldsymbol{p}_{fix}$ and σ^2_{fix}. The variance parameters of each point σ_i, σ_j are hyperparameters of the model. The observed variable D_{ij} models the distance between the points sampled from two Gaussian distributions defined above. In a dataset of N instances, $N(N-1)/2$ unique pairs are considered.

and consider it as an hyperparameter for the model. This formulation allows us to model the variance of points in the LD space based on the uncertainty of the users' feedback, which is detailed below.

Given the dataset of all observed pairwise distances $D = \{(i, j, D_{ij})\}$, the goal is to maximize the log-posterior $\log p(\boldsymbol{\mu}, \boldsymbol{\sigma}^2 \mid \boldsymbol{D}) \propto \log p(\boldsymbol{D} \mid \boldsymbol{\mu}, \boldsymbol{\sigma}^2) p(\boldsymbol{\mu}, \boldsymbol{\sigma}^2)$. The loss function on $\boldsymbol{\mu}$, with $\boldsymbol{\sigma}^2$ as hyperparameters, is expanded as

$$\mathcal{L}(\boldsymbol{\mu}) = \sum_{1 \le i < j \le N} \log p(D_{ij} \mid \boldsymbol{\mu}_i, \boldsymbol{\mu}_j, \sigma_i^2, \sigma_j^2) + 2(N-1) \sum_{i=1}^{N} \log p(\boldsymbol{\mu}_i \mid \boldsymbol{\mu}_0, \sigma_0), \tag{6}$$

where $\boldsymbol{\mu}_0, \sigma_0$ are the hyperparameters for the global prior distribution of $\boldsymbol{\mu}$. In general, $\boldsymbol{\mu}_i \sim \mathcal{N}(0, \mathbb{1}) \forall i$ (i.e., $\sigma_0 = 1$). Each of N location variable $\boldsymbol{\mu}_i$ appears in $N-1$ pairs connecting to other points $\boldsymbol{\mu}_j$. For each observed variable D_{ij}, there are two related location variables $\boldsymbol{\mu}_i, \boldsymbol{\mu}_j$. That gives the factor of $2(N-1)$ for the prior term in (6).

The log-likelihood term (the first term in (6)) is derived from (3) and (4) in the case where $r = 2$ as

$$\log p(D_{ij} \mid \boldsymbol{\mu}_i, \boldsymbol{\mu}_j, \sigma_i^2, \sigma_j^2) = \log\Big(\frac{D_{ij}}{\sigma_{ij}^2}\Big) - \frac{1}{2}\frac{(D_{ij} - d_{ij})^2}{\sigma_{ij}^2} + \log IE_0\Big(\frac{D_{ij}\, d_{ij}}{\sigma_{ij}^2}\Big), \tag{7}$$

where $IE_0(.)$ is the exponentially-scaled modified Bessel function of degree 0, which ensures the numerical stability. When $\sigma_{ij}^2 \to 0$, maximizing the log likelihood in (7) is similar to minimizing the stress $(D_{ij} - d_{ij})^2$ of MDS. As such, MDS is a special case of our method iPMDS when σ tends to zero and the interaction is not used.

The interactive part of our work makes it possible for users to manipulate (rotate, translate, flip) the visualization and, by doing so, to improve its interpretability. This manipulation can simply be done by fixing some points of interest to the desired position. This type of user feedback helps to create the *anchor points* that play the role of anchors to guide the other points. In our model, the user feedback is directly encoded into the local prior of the corresponding indicated points as

$$\boldsymbol{\mu}_i \sim \begin{cases} \mathcal{N}(\boldsymbol{p}_i, \sigma^2_{fix}\mathbb{1}) & \text{if the } i^{th} \text{ point is indicated,} \\ \mathcal{N}(0, \mathbb{1}) & \text{otherwise,} \end{cases} \tag{8}$$

where σ^2_{fix} is a hyperparameter used to model the uncertainty of the users' feedback and $\boldsymbol{p}_i$ is the desired position of the i^{th} point indicated by the user. σ^2_{fix} is set to a small value (less uncertainty) when the user is certain about their feedback. The prior only encodes the user's desired positions, while the actual position of these indicated points will be inferred. In the optimization process, all the points are firstly initialized randomly. Points that are not indicated by the user have a large variance, which gives the optimizer the freedom to move them freely in the LD space to maximize the objective function. On the contrary, the points indicated by the user (with some level of uncertainty controlled by σ^2_{fix}) will be moved towards the indicated positions specified in the prior. One should note that the positions of these anchor points are not hard-fixed but are learned (inferred) by the model. The inference step maximizes the log-likelihood in (7) and can be performed by a stochastic gradient descent method like Adam [18].

IV. Experiments

Since it is hard and subjective to compare interactive methods, we demonstrate the usefulness and several applications of our proposed method via case studies. For each case study, the goal and task that the user has to perform are presented, as well as how to use iPMDS to accomplish this task. The task can be to move several points in the visualization to achieve a pre-defined goal. Since different users can move the points in different ways, our case studies use task-based scenarios to guide the users to move the points intentionally to achieve the goal. After each task, we assess if iPMDS can give the desired visualization in different interactive scenarios. The new visualization is assessed using both quantitative and qualitative criteria like the MDS stress, the visual quality, and the interpretability of the visualization. The experiments are performed with various datasets of different types and different sizes. For each dataset, the input distances are normalized in the interval $[0, 1]$. All the experiments are run using the Adam optimizer with the exponential decay rate for the first and second moments set to 0.9 and 0.999, respectively. The learning rate of Adam is manually tuned for each dataset by observing the value of the loss function to make sure the log-likelihood and log-prior in (6) increase consistently and keep stable when converged (after 150 iterations for all datasets). The hyperparameter $\boldsymbol{\sigma}^2_{fix}$ for the fixed points is set to a small value of 10^{-3} to indicate the level of certainty in the user feedback. This small variance also keeps the fixed points close to the indicated position to help identify these points before and after the interaction easily. Therefore, one simple way to verify the correctness of our interactive model is to assess if the indicated points are placed closely to the user's desired positions in the visualization of iPMDS.

A. *Case Study 1: Alignment of US Cities in a Map*

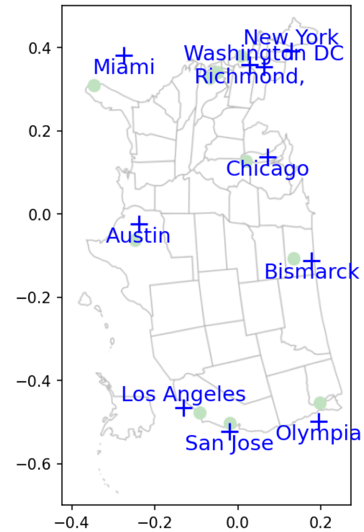
Given the geographical locations (latitude and longitude) of 10 US cities, the corresponding positions in a 2D plane should be found so as to preserve pairwise distances between cities. The distances between the 10 cities are calculated on the spherical Earth by the Haversine (great circle) distance, what gives 45 pairs to be used as input data. The original non-probabilistic metric MDS (SMACOF algorithm) gives the solution (best stress among 10 different runs) shown in Figure 3a. The map of the US with the correct position of the cities as green dots is plotted in the background as ground truth. The result of metric MDS can be in any arbitrary orientation and, as a matter of fact, the visualization in Figure 3a is unintuitive. It should be manually flipped and rotated to correspond to the reality. Therefore, even though this visualization has a low stress of 0.003, it is not optimal for users to understand.

Our goal is to produce a 2D map that is as intuitive as possible. The task is to solve the arbitrary orientation issue of MDS and to place the cities in 2D such as it is understandable by users. This can be accomplished with iPMDS by fixing the position of 2 points corresponding to, e.g., *Olympia* (on the West coast) and *Washington, D.C.* (on the East coast). iPMDS is run with a learning rate of 1.0. Figure 3b shows the new result of iPMDS with two points fixed by the user. Small errors can still be observed in the visualization, but the visualization is now understandable. A similar case study performed on *weighted MDS* with six fixed points can be found in [14].

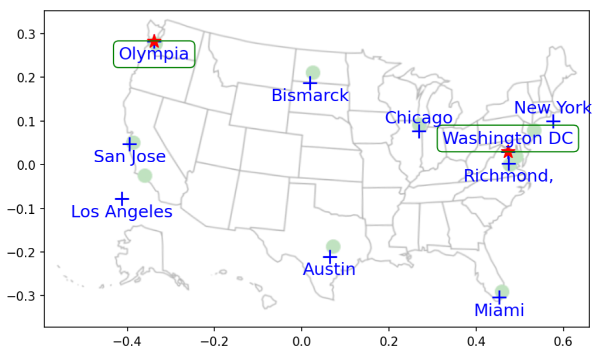
B. *Case Study 2: Interaction in the Incomplete Data Setting*

One of the main advantages of the probabilistic approach for DR methods is that it can handle missing data. In the case of iPMDS, the missing data setting is also called *incomplete data* [19]. In general, the input data for MDS is a $N \times N$ matrix of pairwise distances. In most cases, when the distance measure is not corrupted by noise, the distance is symmetric, and we have $N(N-1)/2$ distinct pairs. However, if less than $N(N-1)/2$ pairs are known, we have incomplete data.

iPMDS does not use the pairwise distance matrix directly, but processes the list of pairs and can thus handle missing pairs. In this case study, iPMDS model is evaluated in different settings where $p\%$ of the input pairwise distances are missing. The experiments are performed on a subset of 250 points of the first five classes of the Digits dataset [20]. Each data point is an 8×8 gray-scale image of a hand-written digit. With 250 samples, the complete data consists of 31,125 unique pairs.



(a) The 2D map found by the original metric MDS with stress = 0.003.



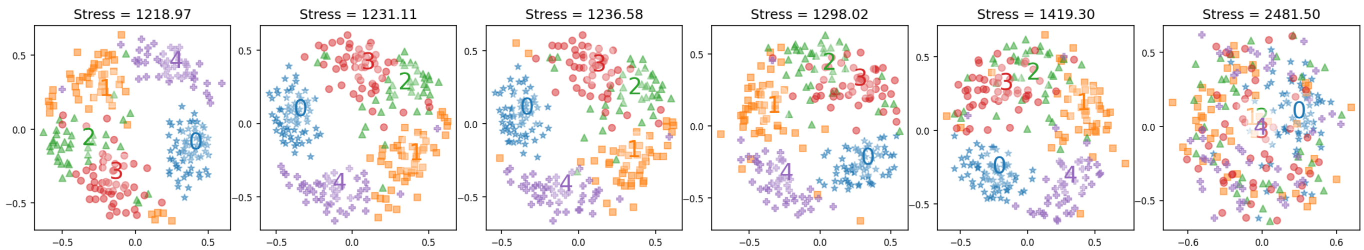
(b) The 2D map result of iPMDS with stress = 0.041. The position of *Olympia* and *Washington, D.C.* are indicated by the user.

Fig. 3: Placement of US cities on the map using MDS and iPMDS. The US map with the correct positions of 10 cities in green dots are shown in background as ground truth.

The pairwise distances are calculated, and $p\%$ of them are randomly removed to create an incomplete data setting.

When evaluating the effect of missing pairs, we run iPMDS without any interaction. Figure 4 shows the iPMDS embeddings for the subset of the Digits dataset with different values of percentage of missing pairs. It can be seen in Figure 4 that visualizations are still readable even when 70% of pairs are missing. As expected, it can also be observed that the error when preserving the input distances increases with the percentage of missing pairs. Figure 5 shows the average stress of iPMDS embeddings corresponding to 20 values of missing percentage $p \in [0, 95]$. For each value of p, iPMDS is run 20 times with different random initialization. The mean and standard deviation of the stress values are reported.

iPMDS can handle incomplete data and produces reliable and stable results (readable visualization, low stress, and small variance in the stress score) with up to 70% of missing pairs. Results are only presented for the Digits dataset, but they hold



(a) Original non-probabilistic MDS (b) PMDS with p=0% (c) PMDS with p=20% (d) PMDS with p=50% (e) PMDS with p=70% (f) PMDS with p=90%

Fig. 4: iPMDS visualizations with different setting of incomplete data for a subset of Digits dataset. (a): Visualization of the original non-probabilistic metric MDS. (b): Visualization of iPMDS with complete data. (c) - (f): Visualizations of iPMDS in incomplete data setting with 20%, 50%, 70% and 90% missing pairs, respectively. iPMDS is run without interaction.

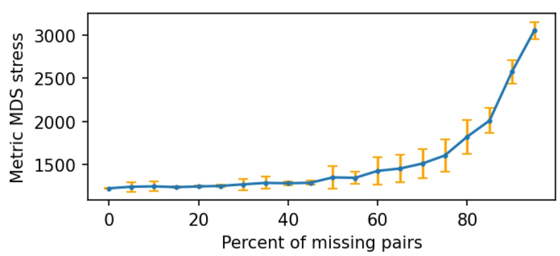


Fig. 5: Evolution of the iPMDS stress for a subset of the Digits dataset with 20 increasing values of the percentage of missing pairs $p \in [0, 95]$. For each value of p, iPMDS is run 20 times, the mean value of stress score is shown in blue and the standard deviations are shown with orange bars.

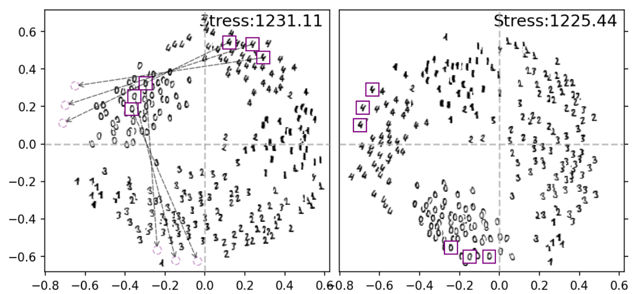
true for other datasets experimented in this paper. Moreover, we can always integrate the user's feedback in incomplete data setting with iPMDS. Figure 6 shows an interaction scenario in which the user moves several points in a visualization to achieve different effects, like a rotation effect in Figure 6a, or a flipping effect in Figure 6b. The interaction is applied in iPMDS with complete input data (Figure 6a), and with only 70% of input pairs (Figure 6b). In all experiments of this case study, iPMDS is run with a learning rate of 2.25.

C. Case Study 3: User-steering Interpretable Axes

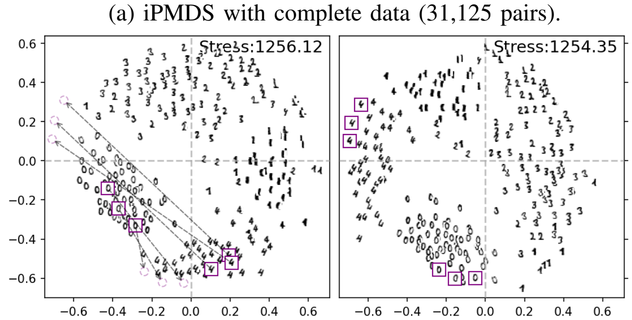
As shown in the previous case studies, users can easily manipulate visualization by fixing the position of several points of interest. The MDS result can be in any arbitrary orientation, which can make it difficult to interpret the two coordinate axes of the MDS visualization. This case study shows how to apply iPMDS to create visualizations with interpretable axes in two scenarios with two different datasets.

1) Meaningful Axes for a Fashion-MNIST Visualization: This example uses a subset of 250 gray-scale 28×28 images from the Fashion-MNIST dataset [21]. Figure 7a shows an embedding of iPMDS, which reveals patterns of objects with different shapes (like shoes, trousers, bags, and T-shirt/pulls) or different zones of low/high-density images. However, it is not clear what is the meaning of the two coordinate axes.

iPMDS allows users to implicitly propose the meaning of axes using examples. Based on the initial visualization of iPMDS without interaction in Figure 7a, users can use images to describe the axes: three images of thin, long-shaped trousers are moved to the left; three images of full rectangular-shaped bags are moved to the right; three low-density images of sandals and shoes are moved to the bottom; and three high-density images of pulls are moved to the top. The first two changes indicate that the horizontal axis should represent the shape, while the two last changes indicate that the vertical axis should represent the pixel density of images. Figure 7c summarizes the conceptual axes implied in the feedback. Figure 7b shows the iPMDS result where the global structure is similar to the initial visualization while reflecting the desired axes. iPMDS is run with a learning rate of 2.25.



(a) iPMDS with complete data (31,125 pairs).



(b) iPMDS with 30% missing pairs (only 21,787 pairs are available).

Fig. 6: iPMDS in complete (a) and incomplete (b) settings with a subset of the Digits dataset of 250 instances. The positions of 6 instances of two groups are fixed and highlighted in the plots on the left. The result of iPMDS are shown on the right.

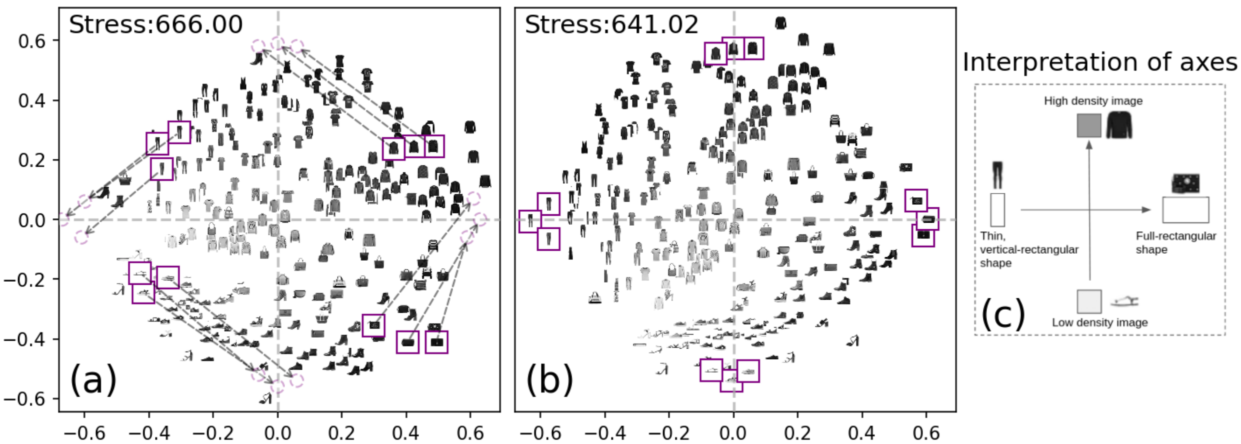


Fig. 7: iPMDS with the subset of 250 images of the Fashion-MNIST dataset. (a): iPMDS result without interaction. The global structure can be explored in this visualization, but the axes do not have any meaning. The user can thus describe the desired axes by fixing several examples highlighted in purple squares. The arrows show the change toward the new positions. (b): New visualization of iPMDS with the fixed points guided by the users shown in purple squares. (c): Interpretation of the axes defined by the user. The x-axis represents *shape* with images of thin shape on the left and images of full rectangular shape on the right. The y-axis represents *pixel density* with dark, high-density images on the top and low-density images on the bottom.

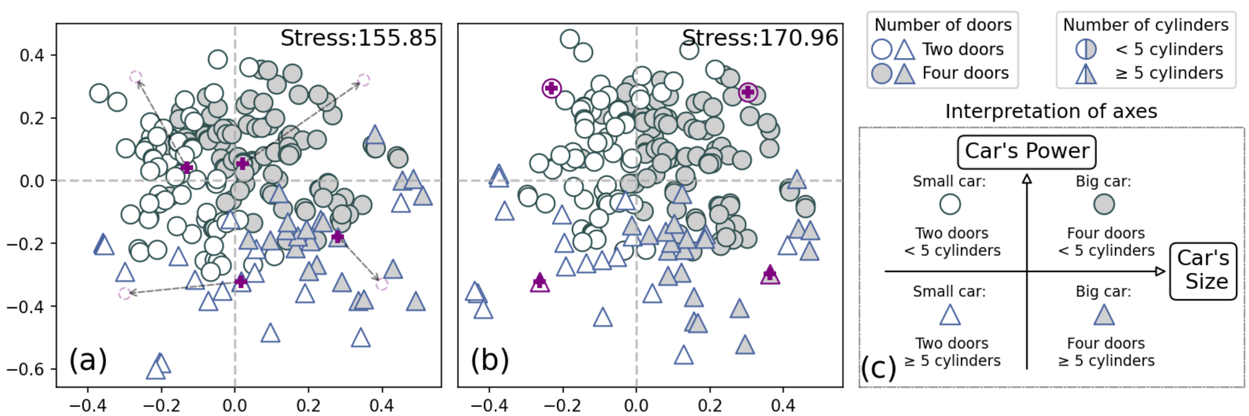


Fig. 8: iPMDS with the Automobile dataset of 203 instances. (a): Initial embedding of iPMDS without interaction. Four groups of cars characterized by two features (number of doors and number of cylinders) can be distinguished. However, the boundary between these groups is not well aligned with the coordinate axes, making hard to understand the axes. The position of four cars from the four groups are fixed in the four quadrants. The arrows show the change towards new positions. (b): New visualization produced by iPMDS in which the boundary between groups aligns with the coordinate system. (c): Interpretation of axes. The x-axis represents *car's size* with small cars of two doors on the left and larger cars of four doors on the right. The y-axis represents *car's power* with normal cars with few cylinders on the top and more powerful cars on the bottom.

2) Meaningful Axes for Automobile Visualization: The Automobile dataset consists of 203 cars characterized by 26 features[1]. Among them, two characteristics are chosen to distinguish the cars in Figure 8a: the number of doors (represented by colors) and the number of cylinders (represented by markers). Four different groups are easily revealed, however, the coordinate axes are not easy to understand.

Using iPMDS, users can indicate different groups of cars with the interaction denoted in Figure 8a: a car with four doors and four cylinders is placed in the first quadrant on the top right; a small car with two doors and two cylinders is placed in the second quadrant on the top left; a small sportive car with two doors and six cylinders is placed in the third quadrant on the bottom left; and a big wagon with four doors and eight cylinders is placed in the fourth quadrant on the bottom right. The resulting visualization of iPMDS that takes the user constraints into account is shown in Figure 8b. Axes can be interpreted as shown in Figure 8c: small cars of two doors on the left and larger cars of four doors on the right. Thus, the x-axis represents the size of the car. Cars with two or four

[1]https://archive.ics.uci.edu/ml/datasets/automobile

cylinders on the top and cars with more than four cylinders on the bottom, which makes the y-axis represent the power of the car. Therefore, the user has defined the axes implicitly by using only four examples placed in the four quadrants of the coordinate system. iPMDS is run with a learning rate of 1.0.

V. Discussion and Conclusion

In this paper, we propose iPMDS, an interactive probabilistic MDS model to find, with the help of users, a visualization that is more understandable for them. iPMDS uses latent variables to represent the locations of each data point and MAP for inference. The original non-probabilistic MDS is a special case of our model when the specific variance parameter of each data point towards zero and the interaction is not used. The latent variable allows us to encode the user knowledge or feedback about the positions of the points directly in terms of prior. Thanks to this prior, users can easily manipulate the visualization and create interpretable axes even with incomplete input data. This approach of using a prior distribution to encode position constraints also works with traditional probabilistic PCA [22] and could be applied to other probabilistic DR methods. The model is implemented using *automatic differentiation* components and optimized using a gradient-based optimizer, which makes our method scale well with the number of pairs in the input data [2].

However, our model has two technical limitations. First, iPMDS only produces the visualization in a 2-dimensional space. For visualizations in a 3-dimensional space, it requires a noncentral chi-squared distribution with 3 degrees of freedom and an approximation of the modified Bessel function of degree 0.5, which is slower and requires an extended work on the computation of gradient. Second, iPMDS is not a fully hierarchical Bayesian model since we consider only the latent variables for the mean $\boldsymbol{\mu}_i$, and we fix the variance $\boldsymbol{\sigma}_i$ as hyperparameters. We thus do not obtain the uncertainty about the estimated location for each point in the embedding. Besides, we assume that users have prior knowledge about their data and have hypotheses about the desired visualizations. When the user attempts to apply transformations that do not modify low-dimensional distances, our method will be able to integrate the feedback seamlessly. However, for other transformations or complex types of feedback, iPMDS will try to find a compromise between the conservation of high-dimensional distances and the feedback. In some extreme cases, users must remain cautious with the output if they provide many potentially inconsistent feedback.

In future work, a focus will be put on two aspects to improve the model and to better evaluate the visualization. First, building a probabilistic model is usually an iterative process of modeling, criticizing, and revising. We plan to focus more on the model criticism step to measure how well the model performs for a specific data exploration task [23]. Second, the visualization assessment in an interactive context with different users is still an open problem. An evaluation of the visualization quality produced by our interactive method can be performed with the feedback of real users.

[2]Our implementation using *jax* (https://github.com/google/jax) is available at https://github.com/vu-minh/probabilistic-mds.

Acknowledgments

The authors would like to thank the reviewers and our colleges Géraldin Nanfack, Valentin Delchevalerie, and Jérôme Fink for their comments and fruitful discussions. Viet Minh Vu is supported by a UNamur-CERUNA institutional PhD grant.

References

[1] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.
[2] W. S. Torgerson, "Multidimensional scaling: I. theory and method," *Psychometrika*, vol. 17, no. 4, pp. 401–419, 1952.
[3] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, pp. 1–27, 1964.
[4] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
[5] J. A. Lee and M. Verleysen, *Nonlinear Dimensionality Reduction*. Springer Science & Business Media, 2007.
[6] H. Soh, "Distance-preserving probabilistic embeddings with side information: Variational bayesian multidimensional scaling gaussian process," in *Proc. IJCAI*, 2016, p. 2011–2017.
[7] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," in *Proc. NIPS*, 2008, pp. 1257–1264.
[8] R. Bakker and K. T. Poole, "Bayesian metric multidimensional scaling," *Political Analysis*, vol. 21, pp. 125–140, 2013.
[9] A. Bibal, R. Marion, and B. Frenay, "Finding the most interpretable mds rotation for sparse linear models based on external features," in *Proc. ESANN*, 2018, pp. 537–542.
[10] R. Marion, A. Bibal, and B. Frénay, "Bir: A method for selecting the best interpretable multidimensional scaling rotation using external variables," *Neurocomputing*, vol. 342, pp. 83–96, 2019.
[11] A. Buja *et al.*, "Data visualization with multidimensional scaling," *Journal of Computational and Graphical Statistics*, vol. 17, no. 2, pp. 444–472, 2008.
[12] D. Sacha *et al.*, "Visual interaction with dimensionality reduction: A structured literature analysis," *IEEE TVCG*, vol. 23, pp. 241–250, 2017.
[13] L. House, S. Leman, and C. Han, "Bayesian visual analytics: Bava," *Statistical Analysis and Data Mining*, vol. 8, pp. 1–13, 2015.
[14] S. C. Leman, L. House, D. Maiti, A. Endert, and C. North, "A bi-directional visualization pipeline that enables visual to parametric interation (v2pi)," *PLOS One*, 2011.
[15] A. Endert, C. Han, D. Maiti, L. House, and C. North, "Observation-level interaction with statistical models for visual analytics," in *Proc. VAST*, 2011, pp. 121–130.
[16] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, 1970.
[17] R. A. Hefner, "Extensions of the law of comparative judgment to discriminable and multidimensional stimuli," Ph.D. dissertation, University of Michigan, 1958.
[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.
[19] J. L. Zinnes and D. B. MacKay, "Probabilistic multidimensional scaling: Complete and incomplete data," *Psychometrika*, vol. 48, pp. 27–48, 1983.
[20] C. Kaynak, "Methods of combining multiple classifiers and their applications to handwritten digit recognition," Master's thesis, Bogazici University, 1995.
[21] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv:1708.07747*, 2017.
[22] V. M. Vu and B. Frénay, "User-steering interpretable visualization with probabilistic principal components analysis," in *Proc. ESANN*, 2019, pp. 349–354.
[23] D. M. Blei, "Build, compute, critique, repeat: Data analysis with latent variable models," *Annual Review of Statistics and its Application*, vol. 1, pp. 203–232, 2014.