Observation-Level Interaction with Statistical Models for Visual Analytics

Alex Endert[+]   Chao Han[*]   Dipayan Maiti[*]   Leanna House[*]   Scotland Leman[*]   Chris North[+]

[+] Department of Computer Science   * Department of Statistics

Virginia Tech

ABSTRACT

In visual analytics, sensemaking is facilitated through interactive visual exploration of data. Throughout this dynamic process, users combine their domain knowledge with the dataset to create insight. Therefore, visual analytic tools exist that aid sensemaking by providing various interaction techniques that focus on allowing users to change the visual representation through adjusting parameters of the underlying statistical model. However, we postulate that the process of sensemaking is not focused on a series of parameter adjustments, but instead, a series of perceived connections and patterns within the data. Thus, how can models for visual analytic tools be designed, so that users can express their reasoning on observations (the data), instead of directly on the model or tunable parameters? *Observation level* (and thus "observation") in this paper refers to the data points within a visualization. In this paper, we explore two possible observation-level interactions, namely *exploratory* and *expressive*, within the context of three statistical methods, Probabilistic Principal Component Analysis (PPCA), Multidimensional Scaling (MDS), and Generative Topographic Mapping (GTM). We discuss the importance of these two types of observation level interactions, in terms of how they occur within the sensemaking process. Further, we present use cases for GTM, MDS, and PPCA, illustrating how observation level interaction can be incorporated into visual analytic tools.

KEYWORDS: observation-level interaction, visual analytics, statistical models.

INDEX TERMS: H.5.0 [Human-Computer Interaction]

1   INTRODUCTION

Visual analytics is "the science of analytical reasoning facilitated by interactive visual interfaces" [1]. The goal of visual analytics (VA) is to extract information, perform exploratory analyses, and validate hypotheses through an interactive exploration process known as sensemaking [2]. In this sensemaking loop, users proceed through a complex combination of proposing and evaluating hypotheses and schemas about their data, with the ultimate goal of gaining insight (i.e. "making sense of" the data). A wide variety of statistical models have been specifically designed for visualizations of this purpose. Thus, many visual analytic systems are fundamentally based on interaction with statistical models and algorithms, using visualization as the medium for the communication (i.e. where the interaction occurs). This communication is performed via direct interaction with the parameters of the model. For example, Interactive Principal Component Analysis, iPCA [3], allows the user to change the weight for each dimension in calculating the direction of projection using multiple sliders (one slider per dimension). Also, in an interactive visualization using MDS [4], the user can weight the dissimilarities in the calculation of the stress function through similar visual controls.

In both instances, the model is made aware of the user input through a formal and direct modification of a parameter (i.e. *parameter level interaction*). The drawback of this type of interaction is that users are expected to be experts in the underlying model that generates the visualization. Moreover, as datasets continue to increase in size and dimensionality, directly adjusting dimensions or parameters creates an issue of scalability. Both interactive MDS [4] and object-centered MDS [5] also allow interactions such as "anchoring" points to provide the algorithm with user specified starting positions, either to test the sensitivity of the current visualization or to obtain an alternate spatial layout based on the anchored observations. In both cases, the visual analytic system does not leverage the observation level interaction to obtain information about the parameters of the model.

In this paper, we reevaluate interaction with such models, moving away from parameter level interactions, and propose to focus on interacting with data (i.e. *observation level interaction*). In contrast to parameter level interactions, users are familiar and comfortable interacting directly with the data in a spatial visualization, freely organizing and relocating observations as an integral part of their sensemaking process [6]. Thus, it is necessary for us to design models that are more tightly integrated with interaction at the observation level, rather than through visual controls of parameters.

Our framework shields users from the technicalities of the model and allows them to interact freely with the data in the visual space. The typical steps in a discovery process based on such a framework will be as follows: 1) the visual analytic system provides a visualization based on initial values of model parameters, 2) users interact with observations to inject understanding and semantic reasoning of the data, 3) under a certain predefined mapping of the user's observation-level interaction to analytic reasoning, the parameters of the model are tuned or re-weighted to reflect the user's understanding of the data, and finally 4) the system regenerates an updated visualization based on the new parameter values of the model. The process continues iteratively, as does sensemaking, for the duration of the analytic process.

We show examples that such a framework can be applied to dimension reduction algorithms for visual analysis of high-dimensional data. Our framework of incorporating user interaction can be applied to either deterministic or probabilistic methods. We demonstrate this on: PPCA (a probabilistic projection-based

{aendert, chaohan, dipayanm, lhouse, leman, north}@vt.edu
Blacksburg, VA 24061

model), MDS (a deterministic stress minimization model), and GTM (a probabilistic manifold learning model). However, the fundamental framework can be applied to numerous other models. Finally, we discuss the tradeoffs between these models for observation-level interaction.

2 RELATED WORK

The three methods in this paper were chosen either because of their wide usage or flexibility in modelling non-linear data. A large and growing body of literature has shown their successful applications in visualization. For example, PCA has gained a lot of success in the area of image classification, with applications such as face recognition [7-10]. MDS has been used in graph layout for network visualization [11-13] due to its rich distance information. GTM is good at visualizing unstructured data like newsgroup text collections, web navigation datasets [14], and datasets which have complicated structure, for instance, protein sequences [15] and the standard Swiss-Roll dataset [16].

Research has gone into creating systems that allow for interaction with these algorithms. iPCA [3] allows direct interaction with the parameters of PCA, through the use of visual controls. In adjusting these parameters, users can observe the corresponding change in the visualization. Buja et al. demonstrate an interactive version of MDS in which users can define static locations of a number of observations, and the algorithm positions the remaining observations into the layout [4]. We would consider this an example of an observation-level interaction, as users can "test" the location of specific observations, and see how the layout (and thus the algorithm) responds. However, the interaction is directly on pairwise dissimilarities, instead of updating of global dimension weights based on the user's positioning of the observations.

Similarly, Broekens et al. describe an interactive MDS algorithm using "object-centric interaction", where users can explore alternative positions of observations by moving them within the spatialization [5]. This is similar to our concept of observation-level interaction, in that the interaction is occurring in the spatialization. However, the movement of an observation is to discover the proportional error contribution, and not to adjust the parameters of MDS. Another example of interacting directly in the spatialization is "Dust & Magnet", an interactive visualization allowing users to understand large, multivariate datasets [17]. It is based on the metaphor of magnets, which can attract observations that share the attributes of the magnet. Thus, in placing multiple magnets into specific locations in the space, users can gain insight into the structure of the data through seeing how observations respond to the attractors. Therefore, the interaction is performed on attractors (i.e., parameters), not on the observations.

From this work, we learn that statistical methods are widely used in visual analytics, and approaches to making these methods interactive have been proposed. However, interactivity in these cases mainly refers to direct manipulation of model parameters. With observation-level interaction, we focus on interacting with the observations within the spatial metaphor, and handle the corresponding parameter updates through our methods.

3 OBSERVATION-LEVEL INTERACTION

In general, observation-level interaction refers to interactions, occurring within a spatialization, that enable users to interact directly with data points (i.e., observations). A spatialization in this context refers to a two-dimensional layout calculated from high-dimensional data where the metaphor of relative spatial proximity represents similarity between documents. That is, data points placed closer together are more similar. Observation-level

interactions are therefore tightly coupled with the underlying mathematical models creating the layout, thus allowing the models to update parameters based on the interaction occurring. While numerous forms of interaction may exhibit these characteristics (e.g., moving clusters of documents, marking regions of interest within the spatialization, etc.), in this paper we will focus on one – movement of observations. From previous studies, we found that movement of observations (in those cases documents) closer together is one way for the user to externalize the analytical reasoning that those documents are somehow similar [6]. In this study, the spatial rearrangement of documents was an integral part of each intelligence analysts' sensemaking process. Further, this study points out that users perform observation-level interaction in two ways, *exploratory* or *expressive*, based on the particular analytical reasoning associated with the interaction, and also how the system responds.

During an exploratory interaction, users utilize the algorithm to explore the data and the space. For example, through dragging one observation within the layout, users gain insight into the structure of the data by observing how other data reacts given the algorithm. While an observation is dragged through the layout, the algorithm adjusts the layout of the remaining data according to how the algorithm computes similarity. Thus, when the observation is dragged towards a cluster of data, similar data points attract, while dissimilar ones repel. Additional information such as a list of similar and dissimilar parameters can also be displayed. Through this process, users learn about a single observation, and how it relates to the other observations in the dataset.

An expressive interaction is different, in that it allows users to "tell" the model that the criteria (i.e. the parameters, weights) used for calculating the similarity need to be adjusted globally. For example, as a user reads two documents, she denotes they are similar by dragging them close together. If this were exploratory, the two documents would repel again. However, in an expressive form of this interaction, it is the responsibility of the underlying mathematical model to calculate and determine why these documents are similar, and update the model generating the spatial layout accordingly. Using the methods below, we illustrate how both expressive and exploratory forms of observation-level interaction are enabled through modifications made to three common statistical methods (PPCA, MDS, and GTM).

4 METHODS INTEGRATING OBSERVATION-LEVEL INTERACTION

A probabilistic model assumes a sampling distribution for the observed data and an uncertainty over the model parameters (e.g. PPCA and GTM discussed in Section 4.1 and 4.3 respectively). A deterministic method makes no such assumptions about the data or the parameters (e.g. Weighted MDS, discussed in Section 4.2). House et al. describe in detail the underpinnings of the probabilistic framework, termed as "Bayesian Visual Analytics" (BaVA) [18]. The BaVA process begins with an initial display of the data. In turn the user may assess the display and decide if it matches her mental model of the data. If it does not, the user may convey her cognitive feedback $f^{(c)}$ by adjusting the locations of two observations to convey her mental model about the two observations. The user might also explore an alternative spatial location of an observation and see how the other observation responds to such an interaction. In short, iterations of user interaction and subsequent regeneration of the visualization are modelled as sequential updating of maximum a posteriori estimates of parameters. The deterministic version of the framework, termed as "Visual to Parametric Interaction" (V2PI), also starts with an initial display and upon obtaining a user feedback sequentially updates the parameters, but the updated

values of the parameters are such that they minimize some measure of discrepancy between the expected configuration of the data under the user's reasoning and the original data [19].

For each of the models discussed in this paper, we present an overview of the model, describe the modifications made to allow observation-level interaction, and show a use case demonstrating how an end-user can interact with each model. Given that each of these models is designed for different types of data (varying in structure, size, and nature of the data), the example use cases below each use different datasets to match the intended use of the models with the use case. The use cases are performed in prototype visualizations to show a proof of concept, and we are actively working to incorporate these models into more fully featured tools.

4.1 PPCA

4.1.1 Overview

Principal Component Analysis (PCA) [20-22] is a common, deterministic method used to summarize data in a reduced dimensional form. The summary is a projection of a high-dimensional dataset in the directions with the largest variance. When only two directions are chosen, PCA may produce a spatial representation or map of the data that is easy to visualize. One problem with PCA is that important structures (e.g., clusters) in data may not correlate with variance. Thus, PCA spatializations may mask information in the data that analysts may find useful.

Probabilistic PCA [23] is, simply, a probabilistic form of PCA. This means that PPCA is not a deterministic algorithm, but a statistical modeling approach (specifically, a factor modeling approach) that *estimates* low-dimensional representations of high-dimensional data. Let $d=[d_1,...,d_n]$ represents a $p \times n$ high-dimensional data matrix, where n represents the number of observations, p represents the number of columns, and d_i (for $i \in \{1,...,n\}$) represents a $p \times 1$ vector for observation i. Also, let $r=[r_1,...,r_n]$ represent a low-dimensional analogy of d, such that r is $q \times n$ and $q < p$. For our purposes, we set $q=2$. PPCA models d as a function of r,

$$d_i | W, r_i, \mu, \sigma^2 = No(Wr_i + \mu, I_p \sigma^2)$$

where, No(.,.) represents the Multivariate Normal Distribution; μ represents a $p \times 1$ mean-vector of d; W is a $p \times q$ transformation matrix known as the factor loadings of d; I_p is a $p \times p$ identity matrix; and σ^2 represents the variance of each dimension in d. By convention, PPCA models each r_i with a Multivariate Normal distribution centered at zero and with unit variance: $r_i \sim No(\mathbf{0}_2, \mathbf{I}_2)$. In turn, the conditional posterior distribution for r_i is $No(\eta, \Sigma_r)$, where

$$\eta = (W'W + I_2 \sigma^2)^{-1} W' (d_i - \mu)$$
$$\Sigma_r = (W'W \sigma^{-2} + I_2 \sigma^2)^{-1} \qquad (1)$$

A spatialization of data d that relies on PPCA plots the posterior expectation η. Similar to PCA, the coordinates η rely on the variability observed in d. To see this, let Σ_d represent the marginal variance of d_i, ($\Sigma_d = V[d_i | W, \mu, \sigma^2]$). Since $\Sigma_d = W'W + I_2 \sigma^2$, we can rewrite η as $\eta = \Sigma_d^{-1} W(d_i - \mu)$ which shows that the relationship between Σ_d and η is well defined.

The final step in PPCA is to estimate the model parameters, $\{W, \mu, \sigma^2, \Sigma_d\}$. We take a Bayesian approach. We specify either reference or flat priors for each unknown (as suggested by [23] and use Maximum A Posteriori (MAP) estimators to assess (and plot) η. For example, when we assign $\pi(\Sigma_d) \propto 1$, the posterior distribution for Σ_d is an Inverse Wishart (IW) distribution,

$$\pi(\Sigma_d | d) \propto \text{IW}(nS_d, p, n - p - 1) \qquad (2)$$

Where S_d represents the empirical variance of d. The MAP estimate of Σ_d is S_d.

4.1.2 User Guided PPCA

To enable analysts to guide PPCA via the data visualization, we take advantage of the relationship between Σ_d and η. Namely, changes in Σ_d will effect η, and changes in η will effect Σ_d, when we invert Equation (1).

After obtaining an initial PPCA display, the user adjusts the locations of two observations; i.e., adjusts two columns in η. If the two observations are moved close to one another, the analyst is conveying that in her mental map, the observations are more similar than what they appear in the display; and, if the observations are dragged apart, the analyst is conveying that the observations differ more than what they appear.

The challenge in BaVA is to parameterize the cognitive feedback and update the visualization [18]. First, we determine the dimensions of the data d for which the adjusted observations are similar and different. Second, we transform the adjustments to η into a hypothetical $p \times p$ variance matrix. We denote this matrix by $f^{(p)}$, as it is a quantified version of $f^{(c)}$. In $f^{(p)}$, the dimensions for which the adjusted observations are similar have small variances and the dimensions for which adjusted observations differ have large variances. Third, we consider the hypothetical variance $f^{(p)}$ to be a realization of a Wishart distribution that has an expectation equal to Σ_d. Finally, we apply Bayesian sequential updating [24, 25] to adjust Equation (2) by the parametric feedback $f^{(p)}$,

$$\pi(\Sigma_d | d, f^{(p)}) = \text{IW}(pS_d + vf^{(p)}, p, n + v - p - 1)$$

where, v is solved from a specification $\kappa (\kappa \in [0,1])$ made by the analyst that states how much weight to place on the feedback relative to the data. Namely, the updated MAP estimate for Σ_d is a weighted average of the empirical variance S_d and feedback $f^{(p)}$

$$MAP(\Sigma_d) = \frac{v}{v + n} f^{(p)} + \frac{n}{v + n} S_d$$

thus $v = n\kappa/(1-\kappa)$. Now, the PPCA projection of the data d that is based on MAP(Σ_d) will portray both information in the data and expert feedback.

4.1.3 Example

A sensitive issue for taxpayers, parents, children, educators, and policy makers is whether an increase in money devoted to education will increase education quality. Money provides a means to buy modern textbooks, employ experienced teachers, and provide a variety of classes and/or extra curricular activities. Although, do the students who benefit from these high-priced resources actually improve academically?

In 1999, Dr. Deborah Guber compiled a dataset for pedagogical purposes to address this question [26]. Based on the following variables, the dataset summarizes the academic success, educational expenses, and other related variables in 1997 for each U.S. state: the average exam score on the Standard Aptitude Test

(SAT); the average expenditure per pupil (EXP); the average number of faculty per pupil (FAC); the average salary for teachers (SAL); and the percentage of students taking the SAT (PER). To increase the complexity of the dataset slightly, we added two variables from the National Center for Education Statistics(www.http:nces.ed.gov): the number of high school graduates (HSG) and the average household income (INC). We hypothesize that states that spend more on education will cluster with states with high SAT averages.

To assess the hypothesis and explore the data, we implement the BaVA process using PPCA. Figure 1a), displays our initial view of the data. Notice that the visualization does not present any structure in the data. Analysts in the field of education, notice that two states with different expectations for SAT scores are displayed close to one another. Thus, we select the appropriate observations and drag them apart as an expressive interaction to obtain an updated view that is displayed in Figure 1b). There are two clusters in 1b). These clusters correspond with SAT scores above and below the national median.

Based on our hypothesis, we suspect that the clustering structure in SAT relates to EXP. However, when we re-plot 1b) and label the upper and lower EXP 50% quantiles in Figure 1c), EXP does not explain the clusters. Thus, we used a bi-plot to identify which variables explain the structure we see in Figure 1b). When we mark the observations above and below the empirical PER median in Figure 1d), we see that PER and SAT clearly relate to the formation of clusters in the dataset. Thus,

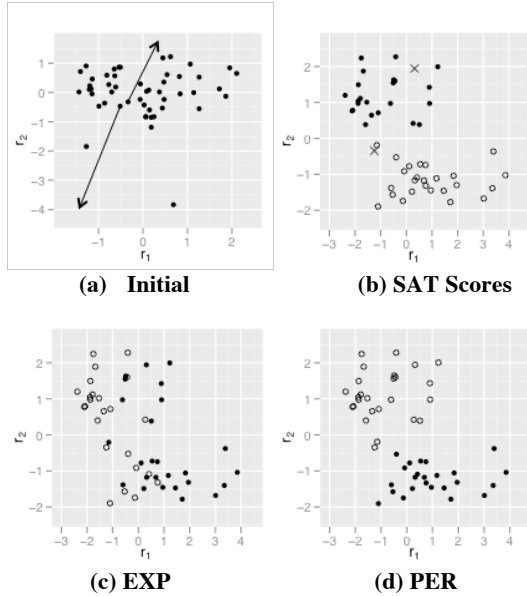(a) **Initial**   (b) **SAT Scores**

(c) **EXP**   (d) **PER**

Figure 1. After injecting expert feedback into Figure 1a), we obtain Figures b)-c). For frame of reference, we marked the two points moved to inject feedback by `x' in Figure b). The configuration of points in each graph are identical, but the observations are labeled differently. In Figure b), symbols `•' and `○' mark the upper and lower 50% quantiles for SAT scores respectively; in Figure c), symbols `•' and `○' mark the upper and lower 50% quantiles for EXP scores respectively; and in Figure d), symbols `•' and `○' mark the upper and lower 50% quantiles for the percentage of students taking the SAT (PER) respectively. Notice the clusters in each graph correspond with SAT and PER, but not EXP.

further analyses of SAT and EXP must control for PER.

4.2 MDS

We extend our framework to another deterministic method, which forms the basis for a large number of visualization techniques: Multi-Dimensional Scaling (MDS).

4.2.1 Overview

All complex data visualizations are based on high-dimensional datasets, which contain features corresponding to dimensions, and the relative importance of such features through a set of weights (w_i). Classically weighted multidimensional scaling deals with mapping a high dimensional dataset $d=[d_1,...,d_n]$ into a low dimensional (in our case two-dimensional) space r, by preserving pairwise distances between observations in the low dimensional representation. Let w represent the p-vector of feature weights: $w=\{w_1,...,w_p\}$. Given a set of feature weights, the low dimensional spatial coordinates are found by solving:

$$\min_{r_1,...,r_n} \sum_{i<j\leq n} \left| \left\| r_i - r_j \right\| - \delta_{i,j}^{(w)} \right|,$$

where

$$\delta_{i,j}^{(w)} = \sum_{k=1}^{p} w_k \, \text{dist}(d_{ik} - d_{jk})$$

such that $\sum_k w_k=1$. dist() represents any distance function for measuring individual features in the high dimensional space. Because it is not possible to estimate weights and the set r simultaneously, we provide a uniform weighting of the space $w_i=1/p$ for our first iteration.

4.2.2 User Guided MDS

Once a <mark>visualization is generated, the user may either agree with the display and learn from certain aspects of the visualization, or disagree</mark>, based on their domain expertise. Hence, the user may wish to interact and rearrange a few of the observations in the visualization. Given a spatial interaction in the form of adjusting the relative position of a set of points, we compute a set of feature weights, which are consistent with both, the users adjustment and the underlying mathematical model. <mark>These are computed by inverting the optimization,</mark> by fixing the locations of the adjusted points and finding an optimal set of weights, which are consistent with the visualization. Explicitly, we solve for w such that
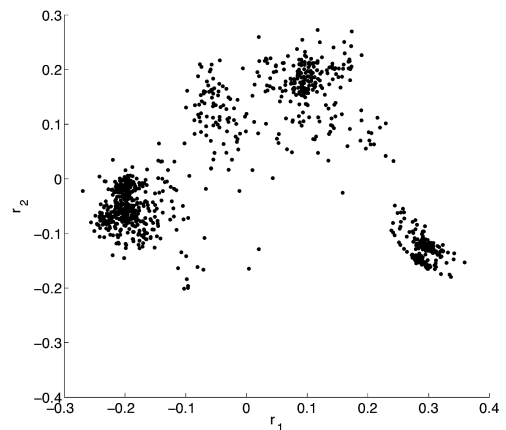

Figure 2. Visualization of the 1990 census dataset using classical MDS.

$$\min_{w_1,\ldots,w_p} \sum_{i<j\leq l} \left| \sum_{k=1}^{p} w_k \, \mathrm{dist}\left(d_{(i)k} - r_{(j)k}\right) - \delta_{i,j}^{(\beta)} \right|$$

and $\sum_k w_k = 1$, where $d_{(i)k}$ represents the kth element in the observation of d that maps to r_i in the adjusted visualization and l is the total number of manipulated observations. It should be noted that computing the new weights is extremely fast, and is then followed by a full MDS step. Thus, the entire generation of a new view can be performed in real time, depending on the size of the dataset and the specific hardware used.

4.2.3   Example

Consider for example a visualization produced by a standard MDS technique. In this example we focus on the 1990 census dataset [27] under a Classical Metric Scaling (CMS) [28], using a Hamming distance (due to the categorical nature of the dataset) for measuring features in the high dimensional space. Figure 2 illustrates results obtained under a Classical Metric Scaling (CMS).

Given this visualization, a user may distinguish 3-5 main clusters, and inquire what they mean. We see two major ways a user can interact with the visualization, in order to explore the space, and learn about the underlying dataset. The first of these is by highlighting a subset of the data, based on some question the user seeks to answer, and then rearranging the visualization based on inconsistencies with their mental model (expressive interactions).

The second approach is to hone in on visual structure, and move points in the visual space in order to learn what the structure relates to in terms of the feature space (exploratory interactions). Both of these interactions are nearly identical, however the motivation for the interactions will differ. We will illustrate both types of visual reasoning through an example based on the 1990 census dataset.

The user may wish to interact expressively and identify points in the space that pertain to high and low income groups. The user highlights individuals with incomes below 15K and over 60K, as shown by ■ and × in leftmost panel of Figure 3, respectively. Because of the close proximity of the highlighted groups in the main clusters, the user drags (denoted by ⊗) a few representative low and high-income individuals into sets of groups in each of the 3 main sub-clusters. The system reports back a set of weights, which explain how much a particular feature explains the arrangement of points suggested by the user. High weights relate to important features, while low weights suggest their

Figure 4. A user performing an exploratory interaction to learn what distinguishes two clusters.

corresponding features do not relate to the user's visual rearrangement. For our example, we learn not only that income level (29%), but also by their means of transportation to work (20%), whether or not they worked the full year (25%), and their level of education (10%) are related to the user's repositioning of points. Given this information, the system updates the visualization, as shown in center panel of Figure 3. We notice that in the resulting visualization, the income groups are clearly separated. The resulting visualization displays a much richer spatialization than simply showing clusters relating to the income groups. For example, we highlight individuals that actually worked in the right most panel of Figure 3, and notice these individuals are shown in distinct sub-clusters. 2 of the 4 clusters in which individuals work pertain to low-income groups, and the other 2 pertain to high-income groups (as illustrated by the ■ and × symbols).

Figure 4 shows how the user might perform an exploratory interaction in order to learn what explains the clustering structure between the working/low income groups. To suggest the clusters could be moved further away from each other than they appear in the current visualization, the system reports back the weights, which explains the differences in the groups. For this example, the user learns that one of these clusters contains individuals that have a reliable mode of transportation to work (93% explained). The visualization could be updated based on this information, or the user could simply document this fact and proceed by explaining other areas of the spatialization. As always, there are an endless
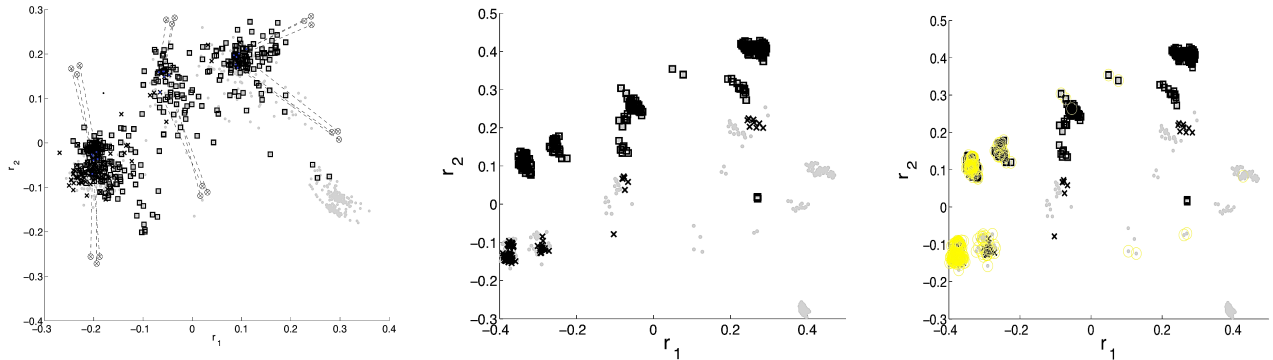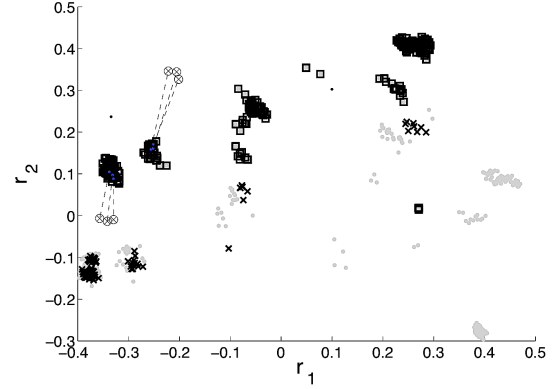

Figure 3. A sequence of visualizations derived through observation-level interaction with a modified MDS method. (Left) The user moves a set of points into new locations, communicating his intuition that there may be additional structure within each cluster. (Middle) The updated visualization showing new clusters. (Right) Highlighting showing the separation of income groups in the updated visualization.

number of possibilities for learning about a high dimensional dataset via visual expression/exploration. <mark>Another example of an exploratory interaction with MDS is demonstrated by Buja et al. in which users can constrain observations to specific spatial locations [4].</mark>

4.3 GTM

4.3.1 Overview

Introduced by Bishop et al, [29] Generative Topographic Mapping (GTM) is a nonlinear latent variable modeling approach for high-dimensional data clustering and visualization. It is considered to be a probabilistic alternative for both the Self-Organizing Map (SOM) algorithm [30] and Nonlinear PCA. Similar to PPCA, GTM estimates a latent variable $r=[r_1,\dots,r_n]$ ($q{\times}n$ matrix) that is a low-dimensional representation of high-dimensional data $d=[d_1,\dots,d_n]$ ($p{\times}n$ matrix such that $p>q$). However, unlike PPCA, the q-dimensional coordinates r in GTM map nonlinearly to a complex manifold $m=[m_1,\dots,m_n]$ that is embedded in the high-dimensional space. This manifold, ideally, characterizes important structure in data d and represents geometrically the expected value for d in the Gaussian model,

$$d_i : N(W\Phi(r_i), I_p\beta^{-1}) \qquad (3)$$

To estimate a coordinate m_i, GTM takes a weighted average of J radial basis functions $\{\Phi_1(),\dots,\Phi_J()\}$ ($\Phi_j()$ represents a radially symmetric Gaussian kernel) given r_i and parameters there in,

$$m_i = W\Phi(r_i), \qquad (4)$$

$$\Phi_j(r_i) = \exp(-\frac{\|r_i - \mu_j\|^2}{2\sigma^2}), \qquad (5)$$

where W is a $p{\times}J$ transformation matrix; $\Phi(r_i)$ is a $J{\times}1$ vector such that $\Phi(r_i)=[\Phi_1(r_i),\Phi_2(r_i)\dots,\Phi_J(r_i)]'$; and μ_j is a $q{\times}1$ vector that centers the basis functions. The center coordinates $\mu=[\mu_1,\dots,\mu_J]$ cover the q-dimensional latent space uniformly. Model parameters are estimated using the EM algorithm [31].

One advantage of GTM is that, by construction, it lacks sensitivity to outliers. For tractability, the coordinates of each r_i are limited a priori to a finite set g of K possibilities, $r_i \in g=\{g_1,\dots,g_K\}$ that covers the q-dimensional latent space uniformly. To decide which value for r_i generates d_i, GTM estimates the posterior probability, i.e., *responsibility*, that $r_i{=}g_k$. Given a prior probability that $r_i{=}g_k$ is $1/K$ for all k $\in\{1,\dots,K\}$, let R_{ik} represent the posterior responsibility that latent variable r_i generates d_i, when $r_i{=}g_k$,

$$R_{ik} = \frac{\pi(d_i|r_i = g_k, W, \Phi())}{\sum_{l=1}^{K}\pi(d_i|r_i = g_l, W, \Phi())}, \qquad (6)$$

In turn, GTM plots the posterior mode, expectation, or any quantile of r_i given specifications g and estimates for $\{R_{i1},\dots R_{iK}\}$.

4.3.2 User Guided GTM

GTM is a complex modeling approach that relies on many tunable parameters that are hard to interpret. User Guided GTM (ugGTM) will allow analysts to both take advantage of the benefits of GTM and guide the complicated GTM parameterization. Specifically, analysts may label, i.e., *tag* clusters, *tag* regions of the visualization space, and query differences in documents.

Here, we illustrate ugGTM within the context of an example. We have a collection of 54 abstracts from proposals funded by the National Institute for Health (NIH). After standard preprocessing, we apply a ranking system that we will call an Importance Index (ImpI), which is based on the Gini coefficient. ImpI considers both the frequency and uniqueness of words that are shared across documents and assigns a metric between 0 and 1. Entities that occur equally frequently in all the documents have ImpI=0 and entities that occur in only one document has ImpI=1. We selected the 1000 entities with the highest ImpI. One advantage of ImpI is that we can measure document similarity using Euclidean distance between proposals. Pairs of documents with small Euclidean distances have comparable terms with similar frequency; and pairs of documents with large Euclidean distances have few, if any, words in common.

We apply GTM for J=16 and K=400 to obtain an initial display of the proposals, shown in Figure 5. Notice four clusters appear in Figure 5 that we labeled A, B, C, and D.

Tagging the Clusters and the Space. To understand the meaning of the clusters, we determine the words that both overlap the least within each cluster and have the highest ImpI's. Specifically, we apply k-means [32] to the low-dimensional data coordinates to determine cluster memberships. For each cluster we sum the ImpI vectors across the documents and rank the entities based on the ImpI sum. Entities ranked highest are those that 1) have importance in the corpus (as determined by the ImpI) and 2) have occurred most frequently. Given top rankings from each cluster, we delete those shared by all four clusters. Table 1 lists the unique key words that describe each cluster. Group A represents proposals that include brain related cancer studies and their clinical applications. Group B represents proposals related to human neural systems. Group C represents proposals that address genomic and transcriptomic research problems. Group D represents proposals about infectious diseases, such as tuberculosis, and immunity.

Table 1. Cluster tags (top 10 keywords) for NIH abstract groups.

Group A	tumors, brains, stem, treatments, patients, generations, drugs, ordering, controlling, therapeutics
Group B	stem, neuronal, brains, proteins, deliveries, regulations, neural, patients, differentiation, expression, treatments
Group C	stem, genetically, regulations, drugs, structurally, proteins, genomics, epigenetics, RNAs, complexities
Group D	Infections, treatments, tuberculosis, expression, patients, drugs, strains, resistance, vaccination, immunity
Shared by All Groups	cells, functionalization, diseases, developments, genes, cancerous , studying, researchers, proposing, mechanisms, specification

As described previously in Equation (3), GTM characterizes high-dimensional data as random perturbations from a complex manifold m; $E[d_i] = m_i$ for all $i \in [1,\dots,n]$. To tag the visualization space, we select any spot, r^+, in the visualization and use Equation (4) to estimate its corresponding location on the manifold, m^+. The estimate m^+ will be a 1000$\times$1 vector of ImpI's that we may use to rank the entities. We report the top ranked entities to tag the space. For example, in Figure 5, we pick up a spot r^+ (represented

by a pink circle) that locates roughly at the center of cluster D. Several of the tagged top keywords overlap with the words describing cluster D.

Document-Based Query and Cluster Reorganization. It is common for users to assess documents by searching for keywords. However, keyword searching may be a tedious task and fail to reveal document clusters of interest. For example, keyword searches may identify documents with similar keywords, but used in different contexts; miss documents that contain combinations of the keywords; or prioritize words that have little relative importance for the user. In response to the challenges of keyword searching, many analysts rely on document matching. For document matching, entire documents can be used to identify which of the remaining documents in the corpus are most similar (to the chosen document). Hence such a matching algorithm is a document-based query of a corpus.

In our ugGTM, users may query documents in the corpus by dragging a document of interest directly in the visualization and watching how the remaining documents respond; e.g., similar documents will follow the document being dragged and dissimilar documents will repel. The behaviour of the documents is similar in spirit to Dust and Magnets (DnM) [17]. In DnM, analysts may drag or shake magnets that represent variables in the dataset and watch as relevant documents follow the magnets. However, a major difference between DnM and ugGTM is that when users drag documents (not variables) and watch how the remaining react, they are comparing documents based on all of the variables in the dataset simultaneously. In turn, users may learn which variables are important for comparisons, based on tags within the visualization space.

The interaction is possible because ugGTM gives control to the users of some parameters in the model via the visualization. Let r^* represent the low-dimensional coordinates for a document that an analyst has chosen to drag. Given r^*, we add to the model described in Equations (3)-(6) by expanding sets g and Φ so that $g=\{g_1,\ldots, g_K, g^*\}$ and $\Phi=\{\Phi_1,\ldots,\Phi_J,\Phi^*\}$, where $\Phi^* = \exp\{-\|r_i-\mu^*\|^2/2\sigma^2\}$ and $g^*=\mu^*=r^*$. In turn, we assign the posterior responsibility (Equation 6) that r^* generates d^* via m^* to 1 (where, m^* is defined by Equation (4) so that the mapping between the low- and high- dimensional coordinates for the moving observations is deterministic.

To propagate the effect of moving r^* to the remaining visualization, we take a local regression approach [33] to characterize high-dimensional data $d_i \mid \{ r_i=g^*, m^* \}$ in that we scale d_i-m^* by the square-root of function V given scaled distance $\Delta_i = \|d^*-d_i\|/c$ so that,

$$\pi(d_i \mid r_i = g^*, W, \Phi) = \left(\frac{\beta}{2\pi}\right)^{-p/2} \exp\{-\frac{\beta V(\Delta_i)}{2}\|d_i - m^*\|^2\}$$

where c is user-defined; e.g., $V(\Delta_i)=\Delta_i^2$ and $c=0.5$. In turn, both posterior responsibility estimates (Equation 6) and estimates for m (Equation 4) change. Let $m_i^{(c)}$ and $m_i^{(u)}$ represent the current and user-adjusted manifold estimates for observation i. We define the BaVA-GTM estimate for the manifold, $m_i^{(c+1)}$, by

$$m_i^{(c+1)} = \delta_i m_i^{(c)} + (1-\delta_i)m_i^{(u)},$$

where $\delta_i=\|r_i-r^*\|/b$ and $b=\max\{\|r_1-r^*\|,\ldots,\|r_n-r^*\|\}$ so that $\delta_i\in [0,1]$. This definition for $m_i^{(c+1)}$ controls the visualization so that only the

regions of interest respond to user interactions; areas that are distant from the dragged observations do not change.

Parameters g^*, Φ^*, $V(\Delta_i)$, δ and $m^{(c+1)}$ in ugGTM work together in the following way. When a data point d_i is far from d^*, $V(\Delta_i)$ will be large and thus decrease the posterior responsibility (Equation 6) that $r_i=g^*$ generates d_i. Similarly, when d_i is near d^*, the corresponding responsibility will increase. Increases in the responsibility for $r_i=g^*$ will cause the coordinates for r_i to gravitate toward r^*. Thus, analysts may specify constant c in our definition Δ_i, depending upon how many document matches they seek for the moving document. Also, the degree to which the observations gravitate toward r^* is determined by δ and $m^{(c+1)}$. When the manifold shifts from $m^{(c)}$ to $m^{(c+1)}$, the meaning of the visualization space changes, as we demonstrate in our example.

4.3.3   Example

For our NIH example, we apply ugGTM. We display an initial GTM view of the documents (the 54×1000 dataset) in Figure 5. Suppose a user identifies a specific document of interest, e.g., Doc 7 (highlighted in yellow in Figure 5) to investigate. A preliminary investigation might involve a sequence of non-spatial interactions, such as, searching of multiple keywords, reading all or part of the document etc. However, a comprehensive assessment of the document may require spatial interactions as well. The user might explore space tags across the screen and determine a more appropriate location for the document of interest. In this case, Doc 7 is closer to Group A, and is about developing new brain tumor therapies and tumor stem cell quiescence. The keywords this document shares with group A include tumors, brains, cancerous, therapeutics and chemotherapy. However, since Doc 7 relates to therapy developments for disease, it shares some keywords with Group D; e.g., treatments, strategies, patients, drugs, resistance, clinically.

As an exploratory spatial interaction, the user drags Doc 7 to the lower left corner of the display and watches how the remaining documents react. By repositioning Doc 7, the user redefines the spatialization of the screen, i.e., modifies the space tag corresponding to a location. For example, when we tag the same coordinates r^+ in Figure 6 (r^+ are the coordinates of the space tagged in Figure 5), we learn that the top keywords include treatments and tumors as well as those that were there earlier. Recall that ugGTM uses every variable in the dataset to compare documents. For this reason, documents that mention stem cells and other important keywords in Doc 7 follow Doc 7. As expected, many documents in Group D gravitate toward Doc 7. However, a few documents in Group B also followed. Future work will allow users to weight the keywords in Doc 7, if desired. Also Documents with ID 20, 22, 32 and 39 change locations. Important keywords for these documents include the following: Doc 20 discusses diagnosis of HIV infection in patients who live with limited access to therapeutic treatments; Doc 22 discusses expression characteristics of a drug-resistant gene; Docs 32 relates to varying yeast strains; and Doc 39 relates to Lymphocyte Homing. Docs 20 and 22 repelled against Doc 7 because the redefined-manifold down-weighted their important entities in the lower left corner and up-weighted the entity tumor. Thus, Doc 20 and Doc 22 shifted to Groups A and C respectively. Docs 32 and 39 are separated slightly from Group D and gravitated toward Group C because they have a few words in common with each group, but not enough to place them in either corner.

An interesting note about the updated manifold is the change in shape or magnification factor [34]. The colour in the background is plotted based on the logarithm of the magnification factor
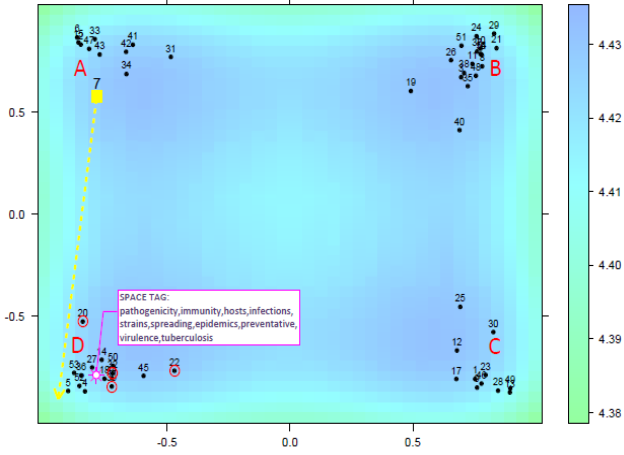
Figure 5. GTM display of the NIH abstracts. Black dots mark documents and labeled by their document ID.

Figure 6. The updated view after moving doc 7 from top left to bottom left.

evaluated on a fine grid that covers the visualization space. Due to the nonlinear mapping from r_i to m_i, equal distances in the visualization do not necessarily imply equal distances in the high-dimensional space. The magnification factor describes the rate of change between distance or area in the latent space and the corresponding distance or area on the manifold and can be interpreted as a description of how wiggly the manifold is. Overall, the magnification factor is lower in Figure 6 than in Figure 5 and the clusters formed in Figure 6 are mainly in low magnification areas. This means the clusters in Figure 6 are in flat, stable regions of the estimated manifold. Thus, observations in these clusters are closer to one another than observations shown in clusters within Figure 5.

5   DISCUSSION

We present a comparison of key characteristics of the methods used in this paper in Table 2. Again, the purpose of this work is not to make a direct comparison of these three methods, but rather to present how to apply observation-level interaction to each method, and summarise our findings in the table.

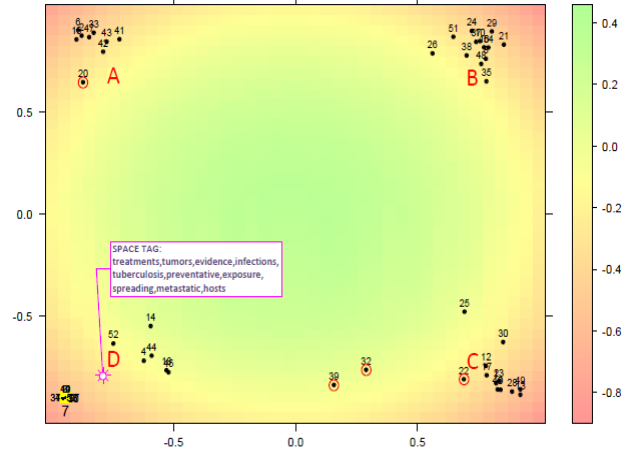**Mappings.** The three methods discussed in the paper provide us with a spatialization of the data within the bounds of their algorithmic complexity. Points that are close in the higher dimensional space remain close to each other in the visualization in all the algorithms although the concept of proximity varies depending on the algorithm. As an artifact of the algorithms, in both PPCA and MDS, the high dimensional data is assumed to be a linear mapping of the visualized representation while GTM is a non-linear mapping of the same. Hence, the same dataset might provide widely disparate visualizations for different algorithms. Spatially this might translate to the fact that based on the algorithm, the user's spatial interaction might target different sets of observations. Each algorithm can potentially have its own set of diagnostics overlaid with the visualization that might aid the user in understanding the proximity of the data in the higher dimensions; e.g. visualizing the magnification factor along with the data in GTM indicates the level of distortion. The goal of the user is to obtain a view in multiple steps that matches with his mental model irrespective of the algorithm used to visualize the data. The specific steps that the user goes through should be immaterial in so far as the final visualization is concerned and all the algorithms discussed here have the flexibility to provide that.

PPCA relies on the assumption that a single linear projection exists that can reveal useful structure. MDS provides a two-dimensional representation of the observations via penalization of any distance distortion that happens in the two-dimensional

Table 2. Comparison of the methods used in this paper.

	PPCA	MDS	GTM
Mapping Type	Linear	Linear	Non-linear
Method Characterisation	Variance	Similarity	Manifold
Distribution Assumption	Probabilistic	Deterministic	Probabilistic
Scalability (Observations)	★★★	★	★
Scalability (Dimensions)	★	★★★	★★
Conceptual Clarity	★★★	★★★	★
Running Time	★★	★★★	★
Outlier Robustness	★★	★★	★★★

★★★ = Good   ★★ = Average   ★ = Poor

representation using a stress function. However, the linear projection assumption may not hold for complex datasets or, the visualization based on minimizing stress in MDS might not reveal all the information in the data. In PPCA, using variance to select the direction in which to project data makes sense for datasets with a global linear structure [23]; the projection will minimize the number of observations that overlap so that they are as visible as possible.

However, variance estimates and hence PPCA visualizations are sensitive to outliers and it is not uncommon for PPCA to display one or two outliers and a cloud of occluded points. Under Euclidean distance, MDS is algorithmically the same as PPCA and will suffer from the same sensitivity to outliers. Assessing such a visualization and making appropriate adjustments would be, at best, challenging. Thus, a more complex methodology is often needed to summarize datasets, e.g., mixture PPCA or GTM. GTM being a topographic mapping places the outliers at one end of the screen or at a position that is distant from the region that has more structure. In our interactive framework, outliers can be brought closer to existing user defined clusters through redefining the principal components in PPCA, reweighting of the dimensions in MDS and constraining *responsibilities* in GTM; in all the cases the user's observation-level interaction initiates the parameter update.

Scalability. In terms of time complexity, GTM is O(KND) (K number of latent points, N number of observations, D data dimensionality), PPCA is O(qND) (q is the dimension of the latent space, usually equals 2) and MDS varies from O(qND) to $O(N^3)$. The effect of high dimensionality (i.e. the number of columns for every observation) on the run-time will be similar for all three algorithms. The challenge in scalability (large N) is also of the same order for the three algorithms when Euclidean distance is used.

However in the design of a visual analytic system that incorporates user interaction in the framework, the choice of the algorithm should be based not only on the run-time of the algorithm but also on the cost incurred in converting the observation-level interaction or feedback to updated values of the parameters for the method. In PPCA, it is the cost of evaluating the feedback matrix $f^{(p)}$; in MDS it is the cost of obtaining optimal feature weights w based on pair-wise distances of the observations that the user has moved; and in GTM, it is the cost of computing distances between data points and reference vectors. Under such considerations, we think MDS provides the quickest and easiest two-dimensional visualization of the data, followed by PPCA and GTM.

We maintain a probabilistic framework in PPCA and GTM. Specifically for PPCA, computation is quick since the primary parameter of interest Σ_d has a posterior distribution and a conjugate feedback distribution, and MAP(Σ_d) can be computed without MCMC. Thus, analysts can explore the data in real time. GTM (although being most flexible in handling more complicated data occlusion issues that challenge MDS or PPCA) is based on an expectation-maximization algorithm and hence needs more run time to converge to the optimal parameter value.

Sensitivity. The methods described in this paper will respond based on the interaction performed (i.e., number of observations moved, distance the observations were moved, etc.). For example, moving a single observation will generally result in a less drastic change in the layout compared to a similar interaction performed on a cluster of observations. Thus, the sensitivity of the models in terms of responding to the user's intuition is dependent on how large the change or update is provided by the user's interaction,

the size of the dataset, as well as if the data supports the suggested updated layout. The methods will attempt to find the "best fit" given the user feedback, but will maintain mathematical validity (i.e., users cannot force the layout if the data does not support it). The result is such that the system balances the user's intuition with the structure of the data to reduce bias. The goal of these techniques is not to converge on a single structure or layout, but rather to allow exploration of many possible structures.

Interaction. The examples of how observation-level interaction can occur within spatializations in this paper show only one form of interaction available to users within spatializations – movement of individual observations. The methods are expandable to allow more complex interactions, such as moving clusters of observations, annotating a region of the spatialization, and other interactions used for communicating the intuition of the user to the system. In a fully implemented visual analytics system, these interactions may include queries, highlighting, and other interactions from which analytical reasoning of users can be interpreted.

Implementation. The prototype visualizations shown in this paper are intended to provide working examples of the modified methods. Through the use cases, we highlighted how an end-user might interact with such systems. We plan to integrate these methods into more fully functional visual analytics tools. That will allow us to perform a series of user studies to evaluate the usability and effectiveness of observation-level interaction in terms of providing insight to users, and supporting the sensemaking process.

6 CONCLUSION

In this paper, we described how modifications of powerful statistical methods allow user interaction at the observation-level. By interacting within the visualization through movement of observations, users are able to perform exploratory and expressive interactions. Thus, users are able to perform sensemaking tasks, such as hypothesis validation, directly within the spatial metaphor. By keeping the interaction at the observation level, users are not required to transform their sensemaking into a combination of statistical parameter updates.

In particular, we modified PPCA, MDS, and GTM using BaVA [18] and V2PI [19] approaches, so that users can focus on their spatial analysis of data rather than directly updating statistical parameters of models. We present three examples (one for each modified method) that illustrate the effectiveness of these new models. Based on the positive results in this paper, as well as the lessons learned, coupling interaction with statistical models provides an opportunity to explore additional forms of spatial interaction for visual analytic applications.

REFERENCES

[1] Thomas, J. J., Cook, K. A., National, V. and Analytics, C. *Illuminating the path*. IEEE Computer Society, 2005.
[2] Pirolli, P. and Card, S. Sensemaking Processes of Intelligence Analysts and Possible Leverage Points as Identified Though Cognitive Task Analysis *Proceedings of the 2005 International Conference on Intelligence Analysis, McLean, Virginia,* 2005.
[3] Jeong, D. H., Ziemkiewicz, C., Fisher, B., Ribarsky, W. and Chang, R. iPCA: An Interactive System for PCA-based Visual Analytics. *Computer Graphics Forum*, 28, 2009.

[4] Buja, A., Swayne, D. F., Littman, M., Dean, N., Hofmann, H. and Chen, L. Interactive Data Visualization with Multidimensional Scaling. *Journal of Computational and Graphical Statistics*, 17, 2, 2008.

[5] Broekens, J., Cocx, T. and Kosters, W. A. Object-centered interactive multi-dimensional scaling: Ask the expert. *In Proceedings of the Eighteenth Belgium-Netherlands Conference on Artificial Intelligence,* 2006.

[6] Andrews, C., Endert, A. and North, C. Space to Think: Large, High-Resolution Displays for Sensemaking. In *Proceedings of the CHI*, 2010.

[7] Conde, C., Ruiz, A. and Cabello, E. PCA vs Low Resolution Images in Face Verification. In *Proceedings of the 12th International Conference on Image Analysis and Processing* (2003). IEEE Computer Society.

[8] Gottumukkal, R. and Asari, V. K. An improved face recognition technique based on modular PCA approach. *Pattern Recogn. Lett.*, 25, 4, 2004.

[9] Imran, S., Bajwa, S. and Hyder, I. PCA based Image Classification of Single-layered Cloud Types. *Journal of Market Forces*, 1, 2, 2005.

[10] Du, Q. and Fowler, J. E. Low-Complexity Principal Component Analysis for Hyperspectral Image Compression. *Int. J. High Perform. Comput. Appl.*, 22, 4, 2008.

[11] Battista, D., Eades, P., Tamassia, R. and Tollis, I. Algorithms for Drawing Graphs: An Annotated Bibliography. *Computational Geometry*, 1994.

[12] Zigelman, G., Kimmel, R. and Kiryati, N. Texture Mapping Using Surface Flattening via Multidimensional Scaling. *IEEE Transactions on Visualization and Computer Graphics*, 8, 2, 2002.

[13] Chen, L. *Local multidimensional scaling for nonlinear dimension reduction, graph layout and proximity analysis*. ScholarlyCommons@Penn, 2006.

[14] Kaban, A. *A Scalable Generative Topographic Mapping for Sparse Data Sequences*. 2005.

[15] Olier, I., Vellido, A. and Giraldo, J. Kernel generative topographic mapping. In *Proceedings of the European Symposium on Artificial Neural Networks – Computational Intelligence and Machine Learning*, 2010.

[16] Cruz-Barbosa, R. and Vellido, A. Unfolding the Manifold in Generative Topographic Mapping. In *Proceedings of the 3rd international workshop on Hybrid Artificial Intelligence Systems* (Burgos, Spain, 2008). Springer-Verlag.

[17] Yi, J. S., Melton, R., Stasko, J. and Jacko, J. A. Dust & magnet: multivariate information visualization using a magnet metaphor. *Information Visualization*, 4, 4, 2005.

[18] House, L., Leman, S. C. and Han, C. Bayesian Visual Analytics (BaVA). *In revision, Technical Report: FODAVA-10-02, http://fodava.gatech.edu/node/34*2010).

[19] Leman, S. C., House, L., Maiti, D., Endert, A. and North, C. *A Bi-directional Visualization Pipeline that Enables Visual to Parametric Interation (V2PI)*. NFS FODAVA Technical Report (FODAVA-10-41), 2011.

[20] Pearson, K. *On Lines and Planes of Closest Fit to Systems of Points in Space*. City, 1901.

[21] Jolliffe, I. *Principal Component Analysis*. John Wiley and Sons, Ltd, 2002.

[22] Torokhti, A. and Friedland, S. *Towards theory of generic Principal Component Analysis*.

[23] Tipping, M. E. and Bishop, C. M. Probabilistic Principal Component Analysis. *Journal of the Royal Statistical Society, SeriesB: Statistical Methodology*, 61, 1999.

[24] Spiegelhalter, D. and Lauritzen, S. Sequential updating of conditional probabilities on directed graphical structures. *Networks*, 20, 1990, 275-605.

[25] West, M. and Harrison, J. *Bayesian Forecasting and Dynamic Models (Springer Series in Statistics)*. Springer, 1997.

[26] Guber, D. Getting What You Pay For: The Debate Over Equity in Public School Expenditures. *Journal of Statistics Education*, 7, 2, 1999.

[27] Blake, C. and Merz, C. J. *{UCI} Repository of machine learning databases*. 1998.

[28] Schiffman, S., Reynolds, L. and Young, F. *Introduction to Multidimensional Scaling: Theory, Methods, and Applications*. Academic Press, 1981.

[29] Christopher, M. B. *GTM: The generative topographic mapping*. 1998.

[30] Kohonen, T., Kaski, S., Lagus, K., Salojarvi, J., Honkela, J., Paatero, V. and Saarela, A. Self Organization of a Massive Document Collection. *Transactions on Neural Networks*, 11, 3 2000.

[31] Dempster, A. P., Laird, N. M. and Rubin, D. B. *Maximum likelihood from incomplete data via the EM algorithm*. City, 1977.

[32] MacQueen, J. Some methods for classification and analysis of multivariate observations. *Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability*, 1, 281-297, 1967, 14.

[33] Hastie, T., Tibshirani, R. and Friedman, J. H. *The Elements of Statistical Learning*. Springer, 2003.

[34] Svensen, J. F. M. *GTM: the generative topographical mapping*. Aston University, Birmingham, 1998.