Data Visualization with Simultaneous Feature Selection

Dharmesh M. Maniyar and Ian T. Nabney
Neural Computing Research Group
Aston University, Birmingham. B4 7ET, United Kingdom
Email: {maniyard,nabneyit}@aston.ac.uk

***Abstract*— Data visualization algorithms and feature selection techniques are both widely used in bioinformatics but as distinct analytical approaches. Until now there has been no method of deciding feature saliency while training a data visualization model. We derive a generative topographic mapping (GTM) based data visualization approach which estimates feature saliency simultaneously with the training of the visualization model. The approach not only provides a better projection by modeling irrelevant features with a separate noise model but also gives feature saliency values which help the user assess the significance of each feature. We compare the quality of the projection obtained using the new approach with the projections from traditional GTM and self-organizing maps (SOM) algorithms. The results obtained on a synthetic and a real-life chemoinformatics dataset demonstrate that the proposed approach successfully identifies feature significance and provides coherent (compact) projections.**

***Index Terms*— Data visualization, feature selection, generative topographic mapping, unsupervised learning, chemoinformatics.**

I. Introduction

Data visualization is an important means of extracting useful information from large quantities of raw data. It is difficult for people to visualize data in more than three dimensions, so high-dimensional data is projected onto lower-dimensional space. Here, we use the term *visualization* to mean any method of projecting data into a lower-dimensional space in such a way that the projected data keeps most of the topographic properties (i.e. ‘structure’) and makes it easier for the users to interpret the data to gain useful information from it.

Data visualization is extensively used in the bioinformatics and drug discovery communities. It is useful to understand “natural” grouping in a large multivariate dataset using data visualization. In a recent review on “Statistical Challenges in Functional Genomics”, Sebastiani et. al. [1], stated “The newly born functional genomic community is in great need of tools for data analysis and visual display of the results”. Dimensionality reduction methods such as principal component analysis (PCA) [2] and factor analysis [3] have been used for data visualization with moderate success for complex datasets. This is because methods based on variance, such as PCA, need not provide good clustering, as features with large variance can be independent of the intrinsic grouping of the data. Advanced projection methods such as Sammon’s mapping [4], multidimensional scaling (MDS) [5], self-organizing maps (SOM) [6] and generative topographic mapping (GTM) [7], which try to preserve topographic structure of the input space in the projection space, have been widely used in bioinformatics and drug discovery domains with more success [8] [9] [10] [11].

In many real-life problems in bioinformatics we are required to work with large multivariate datasets [12] [13]. In principle, the more information we have about each pattern, the better a visualization algorithm is expected to perform. This seems to suggest that we should use as many features as possible to represent the patterns. However, this is not the case in practice. Some features can be just “noise”. For a large multivariate dataset, feature selection is important for several reasons, the fundamental one being that noisy features can degrade the performance of most learning algorithms. Feature selection has been widely studied in the context of supervised learning and applied to many supervised learning problems in bioinformatics [14] [15] [16]. Feature selection algorithms for supervised learning problems can be broadly divided into two categories: *filters* and *wrappers*. The filter approaches evaluate the relevance of each feature (subset) using the data set alone, regardless of the subsequent learning algorithm [14]. On the other hand, wrapper approaches [17] invoke the learning algorithm to evaluate the quality of each feature.

Feature selection for unsupervised problems is more difficult and has received comparatively very little attention because, unlike in supervised learning, there are no class labels for the data and, thus, no obvious criteria to guide the search [18] [19]. Recently Law et. al. [20] proposed a solution to the feature selection problem in unsupervised learning using mixture models by casting it as an estimation problem, thus avoiding any combinatorial search. Instead of selecting a subset of features, they estimate a set of real-valued (in $[0, 1]$) quantities (one for each feature) which are called as the *feature saliencies*. They adopt a minimum message length (MML) [21] penalty for model selection. This approach can be classified as a wrapper approach.

GTM is a principled probabilistic mixture-based data visualization algorithm where each data point is modeled as having been generated by one of a set of probabilistic models. Since GTM is a mixture-based projection method, it is possible to modify the feature selection approach proposed in [20] and apply it to the training of the mixture model in GTM. We propose a GTM-based data visualization with simultaneous feature selection (GTM-FS) approach which not only provides a better visualization by modeling irrelevant features (“noise”)

D. M. Maniyar is the corresponding author. Phone: +44 784 356 7510; Fax: +44 121 204 3685; E-mail: maniyard@aston.ac.uk.

using a separate shared distribution but also gives a saliency value for each feature which helps the user to assess their significance. Such notion of feature saliency is more appropriate than a "hard" feature selection (a feature is either selected or not) for many real-life datasets [22]. Model selection is a less critical issue for density models, particularly when they are used for visualization [7].

The remainder of this paper is organized as follows: The proposed approach, GTM with feature saliency (GTM-FS), is introduced and mathematically derived in Section II. The projection evaluation methods we employed are explained in Section III. The experimental results on both synthetic and real-life chemoinformatics datasets are reported in Section IV. In Section V, we discuss computational costs for the projection algorithms. The results are discussed in detail in Section VI. Finally, we draw the main conclusions in Section VII.

II. GTM with Feature Saliency (GTM-FS)

The generative topographic mapping (GTM) is a probability density model which describes the distribution of data in a space of several dimensions in terms of a smaller number of latent (or hidden) variables [7]. The map $f : \mathcal{H} \Rightarrow \mathcal{D}$ between the latent space, $\mathcal{H}$, and the data space, $\mathcal{D}$, is non-linear, which implies that the image of the (flat) latent space is a curved and stretched manifold in the data space. We use a mixture of Gaussians as a latent grid to model the data in the data space. Given a point $\mathbf{z}_m \in \mathcal{H}$ in the latent space, its image under the map f is

$$f(\mathbf{z}_m, \mathbf{W}) = \mathbf{\Phi}(\mathbf{z}_m)\mathbf{W}, \tag{1}$$

where $\mathbf{\Phi}(\mathbf{z}_m) = (\phi_1(\mathbf{z}_m), ..., \phi_K(\mathbf{z}_m))^T$ is a set of fixed non-linear basis functions, $\mathbf{W}$ is a $K \times D$ matrix of weight parameters and $f(\mathbf{z}_m, \mathbf{W})$ forms the center of the Gaussian component, m, in the data space.

In GTM, this mixture of Gaussians are made up with spherical Gaussians. To calculate feature saliency, we assume that the features are conditionally independent given the mixture component label. In the particular case of Gaussian mixtures, the conditional independence assumption is equivalent to adopting diagonal covariance matrices. So instead of having a mixture of spherical Gaussians, as in GTM, we use a mixture of diagonal Gaussians. Then the probability density function is given by,

$$p(\mathbf{x}_n|\alpha, \theta) = \sum_{m=1}^{M} \alpha_m \prod_{d=1}^{D} p(x_{nd}|\theta_{md}), \tag{2}$$

where M is the total number of components in the mixture (equal to the number of grid points in latent space), and as in GTM, we take the mixing coefficient, α_m, to be constant and equal to $\frac{1}{M}$. D is the total number of features in the input space, $\mathbf{x}_n$ is a D dimensional vector representing the input point n, and $p(\cdot\,|\theta_{md})$ is the pdf of the dth feature for the mth component, with parameters $\theta_{md} = \{f(\mathbf{z}_m, \mathbf{W}), \sigma^2{}_d\}$. $\sigma^2{}_d$ is common (same) across all the components for each feature d.

The dth feature is irrelevant if its distribution is independent of the component labels, i.e., if it follows a common density, denoted by $q(x_{nd}|\lambda_d)$ which is taken to be a diagonal Gaussian, and λ_d is the set of parameters of that Gaussian. Let $\Psi = (\psi_1, ..., \psi_D)$ be an ordered set of binary parameters, such that $\psi_d = 1$ if feature d is relevant and $\psi_d = 0$, otherwise. Now the mixture density is,

$$p(\mathbf{x}_n|\mathbf{\Delta}) = \sum_{m=1}^{M} \alpha_m \prod_{d=1}^{D} [p(x_{nd}|\theta_{md})]^{\psi_d} [q(x_{nd}|\lambda_d)]^{(1-\psi_d)}. \tag{3}$$

where $\mathbf{\Delta} = \{\{\alpha_m\}, \{\theta_{md}\}, \{\psi_d\}\}$.

The notion of feature saliency is summarized in 2 steps: 1) ψ_ds are treated as "missing variables" in the EM algorithm [23] sense and 2) the feature saliency is defined as $\rho_d = P(\psi_d = 1)$, the probability that the dth feature is relevant. Now the resulting model can be written as

$$p(\mathbf{x}_n|\mathbf{\Theta}) = \sum_{m=1}^{M} \alpha_m \prod_{d=1}^{D} (\rho_d p(x_{nd}|\theta_{md}) + (1-\rho_d) q(x_{nd}|\lambda_d)), \tag{4}$$

where $\mathbf{\Theta} = \{\{\alpha_m\}, \{\theta_{md}\}, \{\lambda_d\}, \{\rho_d\}\}$ is the set of all the parameters of the model. An intuitive way to see how (3) is obtained is to notice that $[p(x_{nd}|\theta_{md})]^{\psi_d} [q(x_{nd}|\lambda_d)]^{(1-\psi_d)}$ can be written as $\psi_d p(x_{nd}|\theta_{md}) + (1-\psi_d) q(x_{nd}|\lambda_d)$, because ψ_d is binary.

Figure 1 illustrates the notion of data visualization with simultaneous feature selection in a GTM-FS model for a three dimensional data with feature 1 (d_1) and feature 2 (d_2) as salient features and feature 3 (d_3) as an irrelevant feature ("noise"). Then the fitting of the mixture with four components (given by (2), represented as a two dimensional manifold, shown as 'Latent Space' in Figure 1) can be illustrated schematically as four oblate spheroids (flat disks) on the manifold having larger width (variance) in the directions of features d_1 and d_2 and near-zero width in the direction of the d_3 in the data space. The separate shared pdf, $q(\cdot|\lambda)$, which models the irrelevant features, d_3, is displayed as a prolate spheroid in the middle of the manifold in the data space.

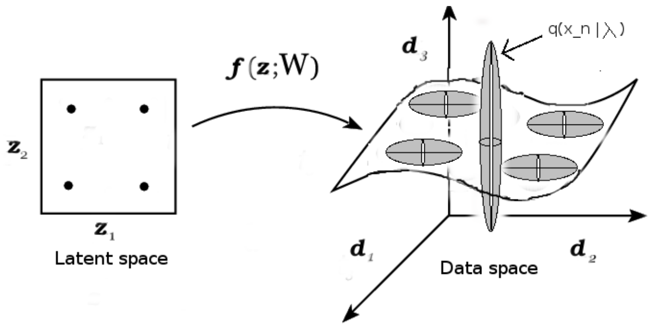


Fig. 1. Schematic representation of the GTM-FS model. d_1 and d_2 have high saliency and d_3 has low saliency.

The complete-data log-likelihood is then given by

$$\mathcal{L}(\mathbf{x}_n, \mathbf{\Theta}) = \ln \prod_{n=1}^{N} p(\mathbf{x}_n|\mathbf{\Theta}), \tag{5}$$

where N is the total number of input points.

Algorithm 1: Summary of the GTM-FS algorithm

```
Input: Training dataset.
Output: Trained GTM-FS visualization model with estimated feature saliency values for all the features.
begin
    Generate the grid of latent points $\{\mathbf{z}_m\} \in \mathcal{H}, m = 1, 2, \ldots, M$;
    Generate the grid of basis functions, $\mathbf{\Phi}(\mathbf{z_m})$, centers $\{\nu_k\}, k = 1, \ldots, K$;
    Select the basis functions, $\mathbf{\Phi}(\mathbf{z_m})$, width;
    Compute the matrix of basis function activations, $\mathbf{\Phi}$ (like in GTM [2]);
    Initialize $\mathbf{W}$, randomly or using PCA;
    Initialize width of the diagonal Gaussians in the grid (mixture);
    Initialize feature weight, $\rho_d$, for each feature $d$, to $0.5$;
    Initialize the mixing coefficient, $\alpha_m$, for each component, $m$, in the grid to $1/M$;
    Set the mean and the variance of the shared distribution, $q(\cdot|\lambda)$, as the mean and covariance of the training set;
    repeat
        Compute $\mathbf{R}$, $\mathbf{U}$ and $\mathbf{V}$ using (6), (7) and (8) respectively, using current parameters, $\mathbf{\Theta}$;
        for $d \leftarrow 1$ to $D$ do
            Reestimate the weight vector, $\mathbf{w}_d$, using $\hat{\mathbf{w}}_d = (\mathbf{\Phi}^T \mathbf{G}_d \mathbf{\Phi})^{-1} \mathbf{\Phi}^T \mathbf{U}_d \mathbf{x}_d$, derived from (9);
        end
        Obtain the center, $\mu_m$, of each component, $m$, of the mixture in the data space, using (11);
        Reestimate the width of the diagonal Gaussians, $\sigma_d$, using (12), for all the features;
        Reestimate the mean and the variance of the shared distribution using (13) and (14) respectively;
        Reestimate the feature weight, $\rho_d$, using (15), for all the features;
    until convergence;
end
```

The parameters are estimated using a variant of the EM algorithm as follows.

A. An EM Algorithm for GTM-FS

We can exploit the latent-variable structure of the model as for GTM and use the expectation maximization (EM) algorithm to estimate the parameters in the model. For each feature $d = \{1, \ldots, D\}$, we flip a biased coin whose probability of a head is ρ_d; if we get a head, we use the mixture component $p(\cdot\,|\theta_{md})$ to generate the dth feature; otherwise, the common component $q(\cdot\,|\lambda_d))$ is used.

We treat $\mathbf{y}$ (the hidden class labels) as the missing variables. In the E-step we use the current parameter set $\mathbf{\Theta}$ to evaluate the posterior probabilities (responsibilities), $R_{nm} = P(y_n = m|\mathbf{x}_n)$, of each Gaussian component m for every data point $\mathbf{x}_n$ using Bayes' theorem in the form

$$R_{nm} = \frac{\alpha_m \prod_{d=1}^{D} (\rho_d p(x_{nd}|\theta_{md}) + (1-\rho_d) q(x_{nd}|\lambda_d))}{\sum_{m=1}^{M} \alpha_m \prod_{d=1}^{D} (\rho_d p(x_{nd}|\theta_{md}) + (1-\rho_d) q(x_{nd}|\lambda_d))}. \tag{6}$$

Using the responsibilities matrix $\mathbf{R}$, we can calculate $u_{nmd} = P(\psi_d = 1, y_n = m|\mathbf{x}_n)$, which measures how important the nth pattern is to the mth component, when the dth feature is used, and $v_{nmd} = P(\psi_d = 0, y_n = m|\mathbf{x}_n)$ as follows

$$u_{nmd} = \frac{\rho_d p(x_{nd}|\theta_{md})}{\rho_d p(x_{nd}|\theta_{md}) + (1-\rho_d) q(x_{nd}|\lambda_d)} R_{nm}, \tag{7}$$

$$v_{nmd} = R_{nm} - u_{nmd}. \tag{8}$$

Then in the M-step we use the posterior probabilities to re-estimate the weight matrix $\mathbf{W}$ by solving the following system of linear equations for each feature (see [24] for a detail derivation of this matrix form)

$$\mathbf{\Phi}^T \mathbf{G}_d \mathbf{\Phi} \hat{\mathbf{w}}_d = \mathbf{\Phi}^T \mathbf{U}_d \mathbf{x}_d, \tag{9}$$

where $\mathbf{\Phi}$ is a $M \times K$ matrix, $\hat{\mathbf{w}}_d$ is a $K \times 1$ weight vector , $\mathbf{U}_d$ is a $M \times N$ matrix calculated using (7), $\mathbf{x}_d$ is a $N \times 1$ data vector, and $\mathbf{G}_d$ is an $M \times M$ diagonal matrix with elements

$$g_{mmd} = \sum_{n}^{N} u_{nmd}. \tag{10}$$

Then using this re-estimated $\hat{\mathbf{W}}$, it is straight forward to obtain the centers of the mixture components in data space, using (1), as follows:

$$\widehat{\text{Mean } \theta}_m = \mu_m = \mathbf{\Phi}(\mathbf{z}_m) \hat{\mathbf{W}}, \tag{11}$$

where μ_m is $1 \times D$ vector.

Using the updated center locations of the components of the mixture in the data space, width of the diagonal Gaussians in each direction, corresponding to one feature each, is re-estimated as below

$$\sigma_d = \frac{\sum_m \sum_n u_{nmd} (x_{nd} - \mu_{md})^2}{\sum_m \sum_n u_{nmd}}. \tag{12}$$

Note that the width, is common across all the components in the mixture.

Parameters of the common density, λ_d, are updated as follows:

$$\widehat{\text{Mean }\lambda_d} = \frac{\sum_n (\sum_m v_{nmd}) x_{nd}}{\sum_{nm} v_{nmd}}, \tag{13}$$

$$\widehat{\text{Var }\lambda_d} = \frac{\sum_n (\sum_m v_{nmd}) x_{nd}}{\sum_{nm} v_{nmd}}. \tag{14}$$

It is natural that the estimates of the mean and the variance in, λ_d, are weighted sums with weight v_{nmd}.

The feature saliency variable, ρ_d, is updated as follows:

$$\hat{\rho}_d = \frac{\max(\sum_{nm} u_{nmd} - \frac{ML}{2}, 0)}{\max(\sum_{nm} u_{nmd} - \frac{ML}{2}, 0) + \max(\sum_{nm} v_{nmd} - \frac{S}{2}, 0)}, \tag{15}$$

where L and S are the number of parameters in θ_{md} and λ_d, respectively. The term $\sum_{nm} u_{nmd}$ can be interpreted as how likely it is that ψ_d equals one, explaining why the estimate of ρ_d is proportional to $\sum_{nm} u_{nmd}$.

Summary of the GTM-FS algorithm is presented in Algorithm 1. Readers interested in a detailed derivation of the EM algorithm for GTM-FS are directed to [24].

III. Evaluation Methods

Though visually we can observe the effectiveness of a projection, it is hard to compare projections obtained using different methods. We employed the following three evaluation methods to compare different aspects of the projections.

A. Kullback-Leibler (KL) divergence

It is useful to get an analytical measurement of the separation between different data classes in the projections. To obtain such a measurement, first we fit a Gaussian mixture model (GMM) [2] to each class in the projection space and then we calculate the Kullback-Leibler (KL) divergence [25] between the fitted GMMs:

$$D_{KL}(p_a \parallel p_b) = \sum_x p_a(x) \log \frac{p_a(x)}{p_b(x)}, \tag{16}$$

where p_a and p_b are the GMMs for classes a and b respectively. The greater the value of KL divergence, the greater the separation between classes.

B. Magnification Factors (MF) sum

One of the main advantages of using GTM–based models is that it is possible to analytically calculate the Magnification Factors (MF) of the projection manifold. MFs of a GTM–based projection manifold, Ω, are calculated as the determinant of the Jacobian of the visualization map f [26]. MF plots are helpful to observe the amount of stretching in a manifold at different parts of the latent space, which helps in understanding the data space, outlier detection, and cluster separation. Small MF values correspond to less stretch in the manifold and hence a more coherent (compact) mapping in the data space.

The magnification factor is represented by color shading in the projection manifold (e.g., see Figure 2(a)). The lighter the color, the more stretch in the projection manifold.

C. Nearest-Neighbor (NN) classification error

Though data visualization is an unsupervised learning problem, it can be useful to objectively evaluate the quality of a classifier based on the visualization output. We calculate Nearest-Neighbor (NN) classification error for which we classify each data point according to the class of its nearest neighbor in the two dimensional latent space obtained by the visualization algorithms.

IV. Experiments

We tested GTM-FS on a synthetic dataset and a real life chemoinformatics dataset used in [11]. Projection results using GTM-FS are compared with the results from traditional GTM and SOM algorithms. The experiments were carried out for 5 times with different random seeds in the training algorithm to calculate standard deviations for the estimated feature saliency values. Label information was used for better presentation of the distribution of data points from different classes in the projections. Label information was also used to calculate KL divergence and NN classification error.

A. Synthetic dataset

The synthetic dataset consists of 800 data points from a mixture of four equiprobable Gaussians $\mathcal{N}(\mathbf{m}_i, \mathbf{I}), i = 1, 2, 3, 4$, where $\mathbf{m}_1 = \left(\begin{smallmatrix}0\\3\end{smallmatrix}\right), \mathbf{m}_2 = \left(\begin{smallmatrix}1\\9\end{smallmatrix}\right), \mathbf{m}_3 = \left(\begin{smallmatrix}6\\4\end{smallmatrix}\right), \mathbf{m}_4 = \left(\begin{smallmatrix}7\\10\end{smallmatrix}\right)$. Eight independent “noisy” features (sampled from a $\mathcal{N}(0, 1)$ density) are then appended to this data, yielding a set of 800 10-dimensional patterns.

The projections obtained using GTM, GTM-FS and SOM algorithms are presented in Figure 2. Background color shading in Figure 2(a) and Figure 2(b) displays the corresponding magnification factors for those projection manifolds. A comparative evaluation of these projections is presented in Table III.

The estimated saliencies of all the 10 features, together with standard deviations (error bars), are shown in Figure 2(d).

The results are further discussed in Section VI.

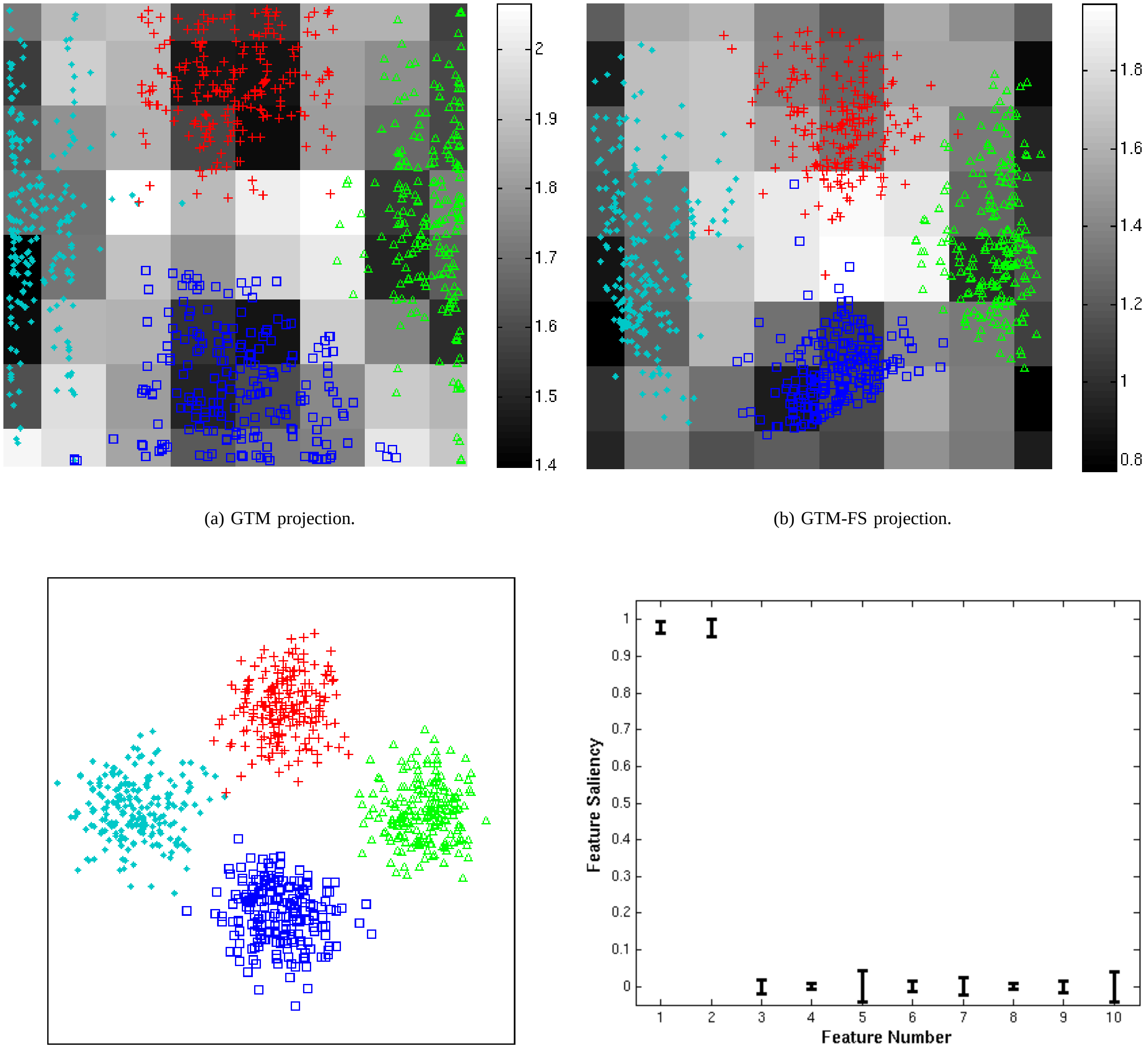
B. Chemoinformatics dataset

The chemoinformatics dataset we used is composed of 11,799 compounds; biological activity is measured for five different biological targets and there are 11 whole-molecule physicochemical properties. Thus, the dataset has, in total, 16 variables (dimensions) in the data space and we want to visualize it effectively on a 2-dimensional manifold.

Out of these five biological targets, two are peptidergic G-Protein coupled receptor (GPCR) targets, two are aminergic GPCR targets, and one is a kinase target. The four GPCR targets are of related receptor types whilst the kinase is a completely unrelated enzyme target class. Table I lists the label information and distribution of compounds in different labels.

In addition to the biological activity values, 11 whole–molecule physiochemical properties were included for each compound in the dataset (Table II).

Since different input variables in the dataset have different ranges, before the development of visualization models we



(a) GTM projection.

(b) GTM-FS projection.

(c) SOM projection.

(d) Feature saliencies.

Fig. 2. GTM, GTM-FS and SOM projections for the synthetic dataset. Background in the GTM and GTM-FS plot is their corresponding magnification factors on a $\log_{10}$ scale.

TABLE I

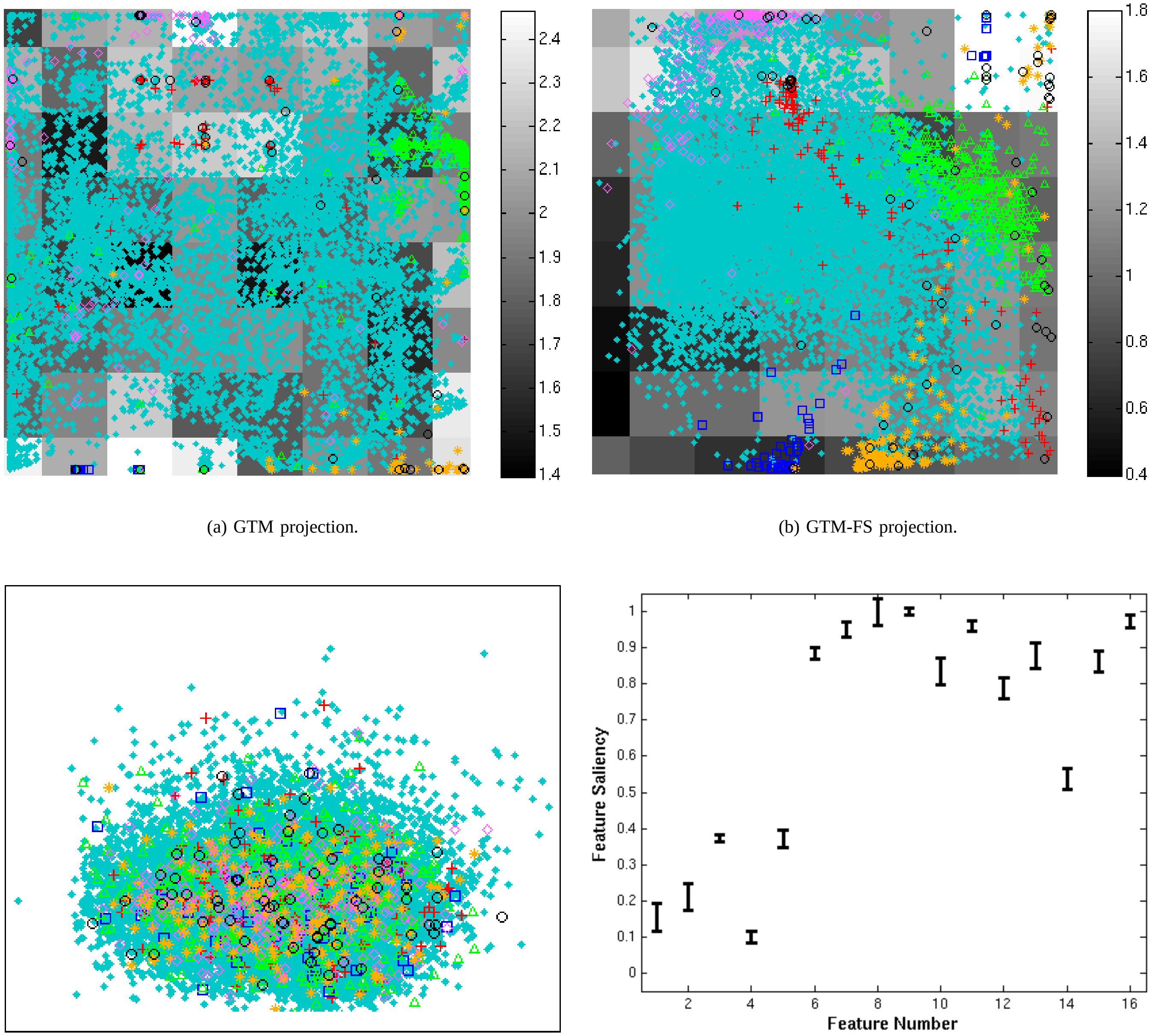
CHEMOINFORMATICS DATASET: LABEL INFORMATION AND DISTRIBUTION.

Label Description	Marker	Compounds
Not active in any screen	●	10769
Active for peptidergic type1	+	118
Active for peptidergic type2	∗	181
Active for aminergic type1	□	50
Active for aminergic type2	△	409
Active for kinase	◇	206
Active for more than 1 screen	○	66

TABLE II

CHEMOINFORMATICS DATASET: MOLECULAR PHYSICOCHEMICAL PROPERTIES.

AlogP
Molecular solubility
Number of atoms
Number of bonds
Number of Hydrogens
Number of ring bonds
Number of rotatable ring bonds
Number of Hydrogen acceptors
Number of Hydrogen donors
Molecular polar surface area
Molecular weight



(a) GTM projection.

(b) GTM-FS projection.

(c) SOM projection.

(d) Feature saliencies.

Fig. 3. GTM, GTM-FS and SOM projections for the chemoinformatics dataset. Background in the GTM and GTM-FS plot is their corresponding magnification factors on a $\log_{10}$ scale. Please refer to Table I for legend.

TABLE III

EVALUATION OF VISUALIZATION MODELS.

Method	Dataset	GTM	GTM-FS	SOM
KL divergence	Synthetic	15.31	**19.43**	12.34
	chemoinformatics	128.17	**167.56**	56.37
MF sum	Synthetic	111.63	**82.32**	-
	chemoinformatics	125.92	**71.18**	-
NN error (%)	Synthetic	0.75	0.75	**0.62**
	chemoinformatics	**38.32**	41.24	92.40

normalized the data by apply a linear transformation (Z-score transformation) to have similar ranges for all variables.

The projections obtained using GTM, GTM-FS and SOM are presented in Figure 3. Background color shading in Figure 3(a) and Figure 3(b) displays the corresponding magnification factors for those projection manifolds. It is useful to see the plots in color.

Comparative evaluation of these projections is presented in Table III. As can be seen from the distribution presented in Table I, the non-active compounds are dominant. A biologist or chemist is interested in increased accuracy of prediction for active compounds, and thus the NN classification error for active compounds is reported in Table III for the chemoinformatics dataset instead of overall NN classification error.

The estimated saliencies of all the 16 features, together with their standard deviations (error bars), are shown in Figure 3(d). The results are further discussed in Section VI.

V. Computational Cost

The distance calculation between data points and mixture components of reference vectors (used in calculation of $p(\mathbf{x}_n|\mathbf{\Theta})$), respectively, is identical in GTM, GTM-FS and SOM training algorithms. Updating the parameters in SOM training depends on the neighborhood function. In the experiments presented here it was continuous on the latent space so the parameter updating scales as $\mathcal{O}(M^2ND+M^2)$, where M is the number of grid points in the SOM map and D is the dimension of the data space. When updating parameters, the GTM and GTM-FS require a matrix inversion of an $K\times K$ matrix, where K is the number of basis functions, followed by a set of matrix multiplications. The matrix inversion scales as $\mathcal{O}(K^3)$, while the matrix multiplications scales as $\mathcal{O}(MND)^1$, where M is the number of grid points in the latent space. GTM-FS requires an extra loop over the number of features, D, to reestimate the weight vector, $\hat{\mathbf{w}}_d$, in the EM algorithm.

Table IV shows the time taken to train different projection models on the chemoinformatics dataset using an Intel Pentium 4 - 2.4GHz machine with 2GB of RAM. An implementation of the algorithms in C/C++ instead of MATLAB would further improve the speed.

TABLE IV
Training time for different projection models for the training set (N = 11800, D = 16, 20 iterations).

The model	Time (seconds)	Architecture
GTM	33	$M=64, K=36$
GTM-FS	34	$M=64, K=36$
SOM	26	$M=64$

Once the models are trained, the computational cost to project data for the subsequent test set scales in the number of data points (N) in the test set.

VI. Discussion

As expected, all three projection algorithms gave four well separated cluster for the synthetic dataset. GTM-based algorithms create a uniform distribution so they spread the data more than SOM projection. This is also revealed from their higher KL divergence sum value compared to SOM. MF sum of the GTM-FS manifold is smaller than MF sum of the GTM manifold which indicates that the GTM-FS manifold is comparatively less stretched. Close observation of Figure 2(a) and Figure 2(b) also reveals that the GTM-FS manifold is more coherent (compact). This is because in GTM-FS the irrelevant features ("noise") are modeled using the separate shared distribution, $q(\cdot|\lambda)$, and thus the actual manifold is less stretched. From the estimated feature saliency values using the GTM-FS model (Figure 2(d)) we can conclude that, in this case, the GTM-FS algorithm not only provided a good projection but also correctly estimated the feature saliencies.

The projection in Figure 3(c), obtained using SOM, is like a blob and does not help us to understand the 'structure' of data in data space. The GTM-based models projections, in Figure 3(a) and Figure 3(b), show clear clusters for the compounds active for different biological targets. We get better KL divergence and MF sum values for GTM-FS which indicates the manifold obtained using GTM-FS is more coherent. GTM and GTM-FS provided much better NN classification error rate for active compounds than SOM where the projection itself is random. The estimated feature saliency values for the chemoinformatics dataset, presented in Figure 3(d), confirms with the general consensus in the pharmaceutical domain that physicochemical properties such as, molecular solubility, number of atoms, molecular weight, etc., are responsible for compounds grouping in the chemical space [27]. Chemists at Pfizer[2] also confirmed that they would have expected higher feature saliency values for these physicochemical properties.

Recently, we introduced a flexible visual data mining tool which combines advanced projection algorithms developed in the machine learning domain and visual techniques developed in the information visualization domain [28]. Although the rapid development of high-performance computing has to some extent altered our perception of computational complexity, this issue cannot be ignored in a visual data mining framework where user interaction is important. Computational complexity of GTM-FS algorithm is similar to GTM, thus it can be directly used in such an interactive data mining framework.

VII. Conclusions

Deriving useful information from a real-life large multivariate dataset is difficult due to the inherent noise and the sheer amount of data. Data visualization and feature selection are both individually important topics in bioinformatics/chemoinformatics domain. Addressing both these problems jointly is not only logical but also synergistic as each endeavor could benefit from advances in the other when they are addressed jointly.

We successfully modified a feature selection for unsupervised learning solution and applied it to the training of a probabilistic mixture-based data visualization algorithm. The new algorithm, GTM-FS, not only provided a better projection by modeling irrelevant features ("noise") using a separate shared distribution but also estimated the feature saliency values correctly which helps the user assess the significance of each feature. The usefulness of the algorithm was demonstrated on both synthetic and real-life chemoinformatics datasets.

Since the estimation of feature saliencies is conveniently integrated with the training of a probabilistic mixer-based data visualization model using a variant of EM algorithm, the computational complexity of the new algorithm remains tractable.

[1] To be exact, the matrix multiplications scales as $\mathcal{O}(MKD+MND)$, but normally the number of data points, N, exceeds the number of basis functions, K.

[2] Pfizer Central Research, Sandwich, UK

One of the avenues for future work is to extend the approach for a probabilistic mixture-based hierarchical visualization algorithm, such as hierarchical GTM [29].

Acknowledgment

D. M. Maniyar is grateful to Pfizer Central Research and the Overseas Research Scholarship award for funding his PhD. We thank Bruce Williams of Pfizer for the chemoinformatics dataset and useful discussions on the results.

References

[1] P. Sebastiani, E. Gussoni, I. Kohane, and M. Ramoni, "Statistical challenges in functional genomics", *Statistical Science*, vol. 18, no. 1, pp. 33–70, 2003.

[2] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, New York, 1st edition, 1995.

[3] H. .H Harman, *Modern Factor Analysis*, Univ. of Chicago Press, 1967.

[4] J. W. Sammon, "A nonlinear mapping for data structure analysis", *IEEE Tran. on Comp.*, vol. C-18, pp. 401–409, 1969.

[5] T. F. Cox and M. A. A. Cox, *Multidimensional Scaling*, Chapman and Hall, London, 2 edition, 2001.

[6] T. Kohonen, *Self-Organizing Maps*, Springer-Verlag, Berlin, 1995.

[7] C. M. Bishop, M. Svensén, and C. K. I. Williams, "GTM: The generative topographic mapping", *Neural Computation*, vol. 10, pp. 215–234, 1998.

[8] R. M. Ewing and J. M. Cherry, "Visualization of expression clusters using sammon's non-linear mapping", *Bioinformatics*, vol. 17, no. 7, pp. 658–659, 2001.

[9] K .Torkkola, R. M. Gardner, T. Kaysser-Kranich, and C. Ma, "Self-organizing maps in mining gene expression data", *Information Sciences*, vol. 139, no. 1-2, pp. 79–96, 2001.

[10] F. Azuaje, H. Wang, and A. Chesneau, "Non-linear mapping for exploratory data analysis in functional genomics", *BMC Bioinformatics*, vol. 6, no. 1, pp. 13–34, 2005.

[11] D. M. Maniyar, I. T. Nabney, B. S. Williams, and A. Sewing, "Data visualization during the early stages of drug discovery", *Journal of Chemical Information and Modelling*, vol. 46, no. 4, pp. 1806–1818, 2006.

[12] P. Baldi and G. Hatfield, *DNA microarrays and gene expression*, Cambridge University Press, Cambridge, 2002.

[13] Y. Liu, "A comparative study on feature selection methods for drug discovery", *J Chem Inf Comput Sci.*, vol. 44, no. 5, pp. 1823–1828, 2004.

[14] A. Blum and P. Langley, "Selection of relevant features and examples in machine learning", *Artificial Intelligence*, vol. 97, no. 1-2, pp. 245–271, 1997.

[15] T. Li, C. Zhang, and M. Ogihara, "A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression", *Bioinformatics*, vol. 20, no. 15, pp. 2429–2437, 2004.

[16] E. P. Xing, M. I. Jordan, and R. M. Karp, "Feature selection for high-dimensional genomic microarray data", in *Proc. 18th International Conf. on Machine Learning*. 2001, pp. 601–608, Morgan Kaufmann, San Francisco, CA.

[17] R. Kohavi and G. H. John, "Wrappers for feature subset selection", *Artificial Intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.

[18] P. Mitra, C. A. Murthy, and S. K. Pal, "Unsupervised feature selection using feature similarity", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 301–312, 2002.

[19] J. G. Dy and C. E. Brodley, "Feature selection for unsupervised learning", *The Journal of Machine Learning Research*, vol. 5, pp. 845–889, 2004.

[20] M. H. C. Law, M .A. T. Figueiredo, and A. K. Jain, "Simultaneous feature selection and clustering using mixture models", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1154–1166, 2004.

[21] C. S. Wallace and D. L. Dowe, "Minimum message length and kolmogorov complexity", *Computer Journal*, vol. 42, no. 4, pp. 270–283, 1999.

[22] D. Modha and S. Spangler, "Feature weighting in k-means clustering", *Machine Learning*, vol. 52, no. 3, pp. 217–237, 2003.

[23] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", *Journal of the Royal Statistical Society*, vol. B 39, pp. 1–38, 1977.

[24] D. M. Maniyar and I. T. Nabney, "EM algorithm for GTM-FS", *NCRG Technical Report, Aston University*, 2006, Web: http://www.ncrg.aston.ac.uk/publications/search.html.

[25] T. Cover and J. Thomas, *Elements of Information Theory*, Wiley, New York, 1st edition, 1991.

[26] C. M. Bishop, M. Svensén, and C. K. I. Williams, "Magnification factors for the GTM algorithm", *Proceedings IEE Fifth International Conference on Artificial Neural Networks*, pp. 64–69, 1997.

[27] C. A. Lipinski, F. Lombardo, B. W. Dominy, and P. J. Feeney, "Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings", *Adv. Drug Delivery Rev.*, vol. 23, pp. 3–25, 1997.

[28] D. M. Maniyar and I. T. Nabney, "Visual data mining using principled projection algorithms and information visualization techniques", in *Proc. 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2006, Philadelphia, USA, to appear.

[29] P. Tiňo and I. T. Nabney, "Constructing localized non-linear projection manifolds in a principled way: hierarchical generative topographic mapping.", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 639–656, 2002.