Novel Visualization Methods for Protein Data

Shahzad Mumtaz
Non-Linearity and Complexity Research Group, School of Engineering and Applied Sciences, Aston University
Birmingham, B4 7ET.
Email: mumtazs@aston.ac.uk

Ian T. Nabney
Non-Linearity and Complexity Research Group, School of Engineering and Applied Sciences, Aston University
Birmingham, B4 7ET.
Email: i.t.nabney@aston.ac.uk

Darren Flower
School of Life and Health Sciences, Aston University
Birmingham, B4 7ET.
Email: d.r.flower@aston.ac.uk

***Abstract*—Visualization of high-dimensional data has always been a challenging task. Here we discuss and propose variants of non-linear data projection methods (Generative Topographic Mapping (GTM) and GTM with simultaneous feature saliency (GTM-FS)) that are adapted to be effective on very high-dimensional data. The adaptations use log space values at certain steps of the Expectation Maximization (EM) algorithm and during the visualization process. We have tested the proposed algorithms by visualizing electrostatic potential data for Major Histocompatibility Complex (MHC) class-I proteins. The experiments show that the variation in the original version of GTM and GTM-FS worked successfully with data of more than 2000 dimensions and we compare the results with other linear/non-linear projection methods: Principal Component Analysis (PCA), Neuroscale (NSC) and Gaussian Process Latent Variable Model (GPLVM).**

Keywords: Visualization, generative topographic mapping, feature saliency, log space, expectation maximization, major histocompatibility complex, principal component analysis, neuroscale, gaussian process latent variable model.

I. Introduction

Recent advances in sciences such as astronomy, biology, weather forecasting and economics have led to the generation, collection, and storage of large high-dimensional datasets. Such datasets have not only presented new challenges for researchers but also created new openings for theoretical developments [1].

Traditional statistical methods fail partially because of the increase in number of objects but mostly due to the immense increase in the number of variables [2]. The problems that arise due to high dimensionality of data are termed the 'curse of dimensionality' [3]. In this paper we study visualization (i.e. projection of data to a low-dimensional space (usually $2D$ or $3D$)) of large high-dimensional datasets in the domain of bioinformatics.

Bioinformatics is the field of studying biological activities of macromolecules, such as carbohydrates, lipids, proteins and nucleic acids, using computational technologies. In general there are three aims of bioinformatics [4]: the first is to maintain a database (such as a protein data bank[1], for three-dimensional macromolecules, or the IMGT/HLA[2] database for maintaining HLA sequences) accessible for researchers to analyze it; the second is to develop tools that are helpful for analyzing these datasets and to understand the functions of macromolecules; and the third aim is to use these analysis tools for interpreting biologically meaningful information about the macromolecules.

Recent research in the field of bioinformatics has provided an extensive set of protein amino acid sequences available in the form of sequence databases such as Swiss-Prot[3], TrEMBL[4], IMGT/HLA[5] etc. In the February 2012 release, there are $534,695$ and $20,127,441$ and $7,274$ known sequence entries respectively. The function of very few protein sequences in these databases are known today. Therefore, predicting the functions of protein sequences is important and is often achieved by searching for the most similar (homologous) sequences with already known functionality [5].

Two sequences with high similarity in primary sequences are expected to have similar three-dimensional structure whereas two similar three-dimensional structures may not have strong similarity in their amino acid sequences [6]. For example, the three-dimensional structures of the human α-globin and myoglobin are very similar but their amino acid sequences only have 26% identity [7]. Predicting protein function from structure is known to be more successful than predicting function from amino acid sequence; there are two reasons for this. First, three-dimensional structures are more conserved than amino acid sequences [8]. Second, the regions where a protein can interact with a ligand[6] are determined by three-dimensional structure [9].

X-ray Crystallography [10], Nuclear Magnetic Resonance Spectroscopy [11] and Electron Microscopy [12] are the standard techniques for determining three-dimensional protein structures. These experimental methods are costly and time

[1] http://www.pdb.org/pdb/home/home.do

[2] http://www.ebi.ac.uk/imgt/hla/

[3] Swiss-Prot is a high quality manually annotated and non-redundant protein sequence database (http://web.expasy.org/docs/relnotes/relstat.html)

[4] TrEMBL is computationally generated annotation and large-scale functional characterization sequence database (http://www.ebi.ac.uk/uniprot/TrEMBLstats/)

[5] http://www.ebi.ac.uk/imgt/hla/

[6] A ligand is an atom or a molecule or an ion that can bind to a specific binding site of the protein. Binding is the key to protein function.

consuming [13]. Therefore, very few three-dimensional protein structures are known in comparison to the large number of known protein amino acid sequences [7]. The database that holds three-dimensional protein structures is the Research Collaboratory for Structural Bioinformatics (RCSB) protein database [14]: in the March 2012 release there are $74,151$ known protein structures and $5,890$ other macromolecules.

Computational methods such as homology (comparative) modeling have been developed for predicting the three-dimensional protein structure for an amino acid sequences using already known similar three-dimensional structures. Therefore if the amino acid sequence of the known three-dimensional structure and target protein sequence are at least 30% similar (i.e. in terms of number and similarity of amino acid residues) then the predicted three-dimensional structure based on the homology modeling is usually close to being correct [15].

We are interested identifying and clustering protein families (such as Major Histocompatability Complex (MHC)) using spatially distributed properties like electrostatic and lipophilicity around a given or predicted set of three-dimensional protein structures. Electrostatic potential is important for understanding the specificity and kinetics of proteins binding with ligands and with other proteins, and can be calculated within and around the protein three-dimensional structure. Algorithms for computing electrostatic potential are usually described as 'explicit-solvent' or 'implicit-solvent' [16]. Explicit-solvent methods treat the solvent with full atomic detail making them computationally intensive. However, implicit-solvent methods treat the solvent in its average effect on solute and are thus much faster to compute. The latter method of computing electrostatic potential has opened new horizons for the researchers in the field of drug design and computational structural biology [17].

Using an implicit-solvent system, one popular method of calculating the electrostatic potential for a protein in a solvent is by numerically solving the Poisson-Boltzmann equations [18] (there is no analytical solution) using finite-element, finite-difference and boundary-element methods [16]. Software tools include: Delphi[7] and University of Houston Brownian Dynamics (UHBD)[8] use finite-difference numerical methods; the Adaptive Poisson Boltzmann Solver (APBS)[9] uses finite-elements; and Charged Particle Optics (CPO)[10] uses the boundary-element method. These tools generate large datasets containing the potentials at a fine grid of points in and around the protein.

Clustering or grouping a set of proteins based on their similarity is a valuable contribution to drug design. A web-based tool called WebPIPSA [19] allows a user to compare electrostatic potentials for a set of protein structures using the PIPSA method (Protein Interaction Property Similarity Analysis) [20]. The tool compares a pair of proteins using similarity indices and distance measures, and presents the results as a colour matrix and a tree-like diagram. This tool has number of limitations: first, being a web-based tool, it only supports the comparison of a few proteins (up to one hundred); secondly, the way the similarity indices are calculated can give a false sense of similarity in a pair of proteins. Instead, we propose to analyse the electrostatic potentials calculated at a fine grid by projecting the dataset onto a low-dimensional space combined with interactive data exploration techniques so that humans can interpret easily a large set of proteins. In the domain of pattern recognition various dimensionality reduction techniques, such as principal component analysis (PCA) [21], projection pursuit [22] and factor analysis [23], have been used in different domains with some success [24]. Dimensionality reduction approaches based on variance, such as PCA, do not provide good clustering or grouping information because certain features with large variance can dominate the actual grouping of the data. Therefore, advanced dimensionality reduction techniques such as the self-organizing map (SOM) [25], Sammon's Mapping [26] and the Generative Topographic Mapping (GTM) [27] have been applied more successfully in bioinformatics [28] [29] [30] [31] [32]. However, GTM worked better than other dimensionality reduction techniques in the field of bioinformatics as shown in [33] [34] [35]. In [24], an algorithm for GTM with simultaneous feature selection was proposed (GTM-FS), which projects the data and also computes the *saliency* of each feature to help the user determine the importance of each feature. Both GTM and GTM-FS may fail when applied to high-dimensional datasets (with more than 200 dimensions) partially due to numerical problems. In fact, in high-dimensional spaces the probability density values are sufficiently small that rounding error is very significant (i.e. values may round to zero). Here, we propose that using log-space values and re-arranging mathematical expressions at certain steps of training algorithms and visualization process of GTM and GTM-FS can avoid such numerical problems.

The structure of the rest of the paper is as follows: the proposed variant of the GTM and GTM-FS using log space and a basic description of the original algorithms are given in sections II and III respectively. A description of our visualization software tool is given in section IV and a description of the experimental validation is given in section V. Discussion of the results is contained in section VI. Finally, we present conclusions and future work in section VII.

II. GTM with Log-Space Probabilities

GTM was proposed as an alternative to the SOM which estimates a generative probability distribution [27]. It is an unsupervised learning algorithm and is a non-linear method for representing high-dimensional data in a low-dimensional space using a latent variable model. The generative model uses a mapping from latent space to data space while the inverse mapping that provides data visualization uses Bayes' theorem. GTM is based on a constrained mixture of Gaussians whose parameters are optimized using an Expectation Maximization

[7] http://wiki.c2b2.columbia.edu/honiglab_public/index.php/Software:DelPhi
[8] http://adrik.bchs.uh.edu/uhbd.html
[9] http://www.poissonboltzmann.org/apbs/
[10] http://simion.com/cpo/bem.html

(EM) algorithm. EM is an iterative procedure to compute the maximum likelihood estimate of parameters in the presence of missing or hidden data.

The primary objective of the latent variable model is to estimate the probability distribution $p(x)$ that represents data $x \in \mathbb{R}^D$ using latent variables $z \in \mathbb{R}^q$. The non-linear function $y(z;W)$ maps a point z in the latent space to a corresponding point $y(z;W)$ in data space and the mapping function is parameterized by the matrix W. A Radial Basis Function (RBF) network is used as a mapping function and the parameter matrix W represents the network weights and biases. GTM is most useful when the latent-space dimensionality is one or two (i.e. $q = 1$ or $q = 2$). Suppose we have a latent space of dimensionality $q = 2$ and data space of dimensionality $D = 3$ then the function $y(z;W)$ maps the latent space into a q-dimensional manifold S in data space as shown in Figure 1.

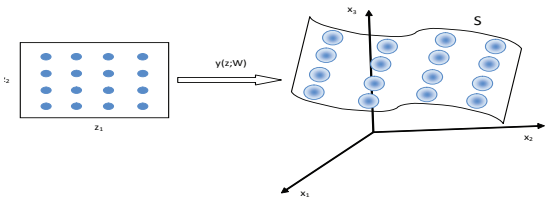


Fig. 1. Latent space to data space mapping using a non-linear mapping function y(z;W).

To use GTM for visualization, the inverse mapping from data space to latent space is performed using Bayes' theorem to compute the posterior probability density $p(z|x_n)$ for a data point x_n. Using the prior distribution as in [27], the posterior probability density is a sum of delta functions centered at the lattice points z_j with weights given by R_{jn} (i.e. $p(j|x_n)$ the posterior probability of the jth component in the latent space). For visualizing a set of data points, the posterior distribution for each data point gives too much information (since this would require a distinct 2D plot for each data point), and it is therefore necessary to use a summary statistic, usually the mean

$$\langle z|x_n, W, \sigma\rangle = \sum_{j=1}^{M} R_{jn} z_j, \tag{1}$$

where M is the number of latent space components. More details of the GTM can be found in [27].

We shall focus on the main step where numerical problem arise while using GTM for high-dimensional data. We propose that a log-space version of a mixture of spherical Gaussians (as shown in equation (2)) can be used with a GTM model to compute the probability that a point x is generated by the jth component.

$$\begin{aligned} p^{log}(x|j) &= \log\left(\frac{1}{(2\pi\sigma_j^2)^{\frac{D}{2}}}\exp\left\{-\frac{(x-\mu_j)^2}{2\sigma_j^2}\right\}\right) \\ &= -\frac{(x-\mu_j)^2}{2\sigma_j^2} - \sum_{i=1}^{D/2}\log(2\pi\sigma_j^2), \end{aligned} \tag{2}$$

where the *log* superscript is used to denote values in the log space. After calculating the log probabilities, the resultant values are converted back to real space to compute the component responsibilities $p(j|x) = R_{jn}$ that are used in the M-step of the EM algorithm.

III. GTM-FS using Log-Space Probabilities

To calculate feature saliency with GTM, it is assumed that features are conditionally independent given the mixture component label [24]. Specifically for a mixture of Gaussians such independence can be achieved using diagonal covariance matrices. Therefore, GTM-FS uses a mixture of diagonal Gaussians and the probability density function can be expressed as

$$p(x_n|\alpha,\theta) = \sum_{m=1}^{M}\alpha_m \prod_{d=1}^{D} p(x_{nd}|\theta_{md}), \tag{3}$$

where M represents the number of components, as in GTM, α_m is a mixing coefficient that is assumed to be equal to $\frac{1}{M}$, D represents the number of variables, x_n represents the nth point (in $\mathbb{R}^D$), and $p(x_{nd}|\theta_m)$ represents the probability density function of the dth feature for the mth component with the mean and variance parameters $\theta_{md} = \{y(z_m;W), \sigma_d^2\}$. As can be seen from the notation, it is also assumed that σ_d^2 is same for all the components in the mixture. The dth feature is considered as irrelevant only if the distribution of the feature is independent of the mixture component labels and is then modeled by the Gaussian distribution $q(x_{nd}|\lambda_d)$ with a diagonal covariance. We use $\psi = (\nu_1, \ldots, \nu_D)$ to denote a set of binary values where ν_d is equal to 1 for a relevant feature and 0 for an irrelevant feature. With these definitions, the probability density function is defined as

$$p(x_n|\Omega) = \frac{1}{M}\sum_{m=1}^{M}\prod_{d=1}^{D}[p(x_{nd}|\theta_{md})]^{\nu_d}[q(x_{nd}|\lambda_d)]^{(1-\nu_d)}, \tag{4}$$

where $\Omega = (\theta_{md}, \nu_d)$. The concept of feature saliency is represented as follows.

- The ν_ds are treated as missing variables in the EM algorithm.
- The probability of the relevant feature is represented as ρ_d.

Now the resultant model can be written as

$$p(x_n|\Upsilon) = \frac{1}{M}\sum_{m=1}^{M}\prod_{d=1}^{D}(\rho_d p(x_{nd}|\theta_{md})] + [(1-\rho_d)q(x_{nd}|\lambda_d)], \tag{5}$$

where $\Upsilon = \theta_{md}, \lambda_d, \rho_d$ represents all of the parameters of the model. The complete-data log-likelihood of the data is defined as

$$L(x_n, \Upsilon) = \ln \prod_{n=1}^{N} p(x_n|\Upsilon), \tag{6}$$

where N represents total number of input points. For estimating parameter of the GTM-FS, here we present a variant of the EM algorithm that uses the log space.

A. An EM Algorithm for GTM-FS using Log-Space

We propose a variant of the EM training algorithm for GTM-FS that uses log space and is able to deal high-dimensional data both for visualization and feature saliency purposes. We use real to log-space transformations [36]. The product of real-space values is equivalent to the sum of log-space values

$$\prod_i x_i \rightarrow \sum_i \log x_i, \tag{7}$$

and the sum of real-space can be computed in log-space by

$$\sum_i x_i \rightarrow \eta + \log\left(\sum_i \exp(\log x_i - \eta)\right) = \underset{i}{S}(x_i), \tag{8}$$

where $\eta = \max_i \log x_i$. In the following derivation of the EM algorithm the log superscripts are used to represent the values in the log space.

In the EM algorithm of GTM-FS, the dth feature is considered to be relevant with probability ρ_d: in that case, a mixture component $p(.|\theta_{md})$ is used to generate its value; otherwise a common density represented by $q(.|\lambda_d)$ is used.

We take y (the hidden class labels) and ν_ds to be the missing variables. In the E-Step using the current parameters Υ, posterior probabilities (i.e. $R_{nm} = P(y_n = m|x_n)$) can be calculated for the mth Gaussian component for each data point as

$$\begin{aligned} R_{nm}^{log} = & \left[\alpha_m + \sum_{d=1}^{D}\left(S\left((\rho_d^{log} + p^{log}(x_{nd}|\theta_{md})),\right.\right.\right. \\ & \left.\left.\left.((1-\rho_d^{log}) + q^{log}(x_{nd}|\lambda_d))\right)\right)\right] \\ & - \underset{m}{S}\left[\alpha_m + \sum_{d=1}^{D}\left(S\left((\rho_d^{log} + p^{log}(x_{nd}|\theta_{md})),\right.\right.\right. \\ & \left.\left.\left.((1-\rho_d^{log}) + q^{log}(x_{nd}|\lambda_d))\right)\right)\right]. \end{aligned} \tag{9}$$

Some of the terms used in equation (9) are defined in equations (10) and (11).

$$\begin{aligned} p^{log}(x_{nd}|\theta_{md}) = & -\frac{\sigma_{md} * (x_{nd} - \mu_{md})^2}{2} \\ & + \log(\sqrt{\sigma_{md}}) - \log(2 * pi), \end{aligned} \tag{10}$$

$$\begin{aligned} q^{log}(x_{nd}|\lambda_d) = & -\frac{\sigma_d * (x_{nd} - \mu_d)^2}{2} \\ & + \log(\sqrt{\sigma_j}) - \log(2 * \pi). \end{aligned} \tag{11}$$

Based on the responsibility matrix R (as shown in equation (9)), the value $U_{nmd} = P(\nu_d = 1, y_n = m|X_n)$ can be calculated which shows the importance (relevance) of the nth pattern with mth component using the dth feature and $V_{nmd} = P(\nu_d = 0, y_n = m|X_n)$ that shows the irrelevance (noise) of the dth feature.

$$\begin{aligned} U_{nmd}^{log} = & R_{nm}^{log} \\ & + \frac{\rho^{log} + p^{log}(x_{nd}|\theta_{md})}{S(\rho^{log} + P^{log}(x_{nd}|\theta_{md}), ((1-\rho_d^{log}) + q^{log}(x_{nd}|\lambda_d)))}, \end{aligned} \tag{12}$$

$$V_{nmd} = \exp(R_{nm}^{log}) - \exp(U_{nmd}^{log}). \tag{13}$$

Now, during the $M-$step these posterior responsibilities are used for estimating the weight matrix W by solving the following set of linear equations for each feature,

$$\phi^T G_d \phi \hat{w_d} = \phi^T \exp(U_d^{log}) x_d, \tag{14}$$

Where ϕ represents a $M \times K$ matrix, $\hat{w_d}$ represents a $K \times 1$ weight vector, U_d^{log} is a $M \times M$ matrix calculated using equation (12), x_d is a $N \times 1$ data vector, and G_d is an $M \times M$ matrix with elements

$$g_{mmd} = \exp(\underset{n}{\overset{N}{S}}(U_{nmd}^{log})). \tag{15}$$

Where S represents a function of sum for log space values. In this framework, g_{mmd} is calculated in the log space (to reduce rounding errors) and then transformed back to real space to solve equation (14). Now, using this re-estimated $\hat{W}$, the centres of the mixture components in the data space can be calculated using the mapping function

$$\widehat{Mean\theta}_m = \mu_m = \phi_m \hat{W}, \tag{16}$$

where μ_m represents a $1 \times D$ vector. After updating the centres for the mixture components in the data space, the variances of the Gaussians for each feature can be calculated

$$\sigma_d = \exp(\underset{m}{\underset{n}{SS}}(U_{nmd}^{log} + \log[(x_{nd} - \mu_{md})^2] - \underset{m}{\underset{n}{SS}}(U_{nmd}^{log})). \tag{17}$$

The parameters λ of the common density $q(x_{nd}|\lambda_d)$ are updated using a similar formula as in the original GTM-FS algorithm

$$\widehat{Mean\lambda}_d = \frac{\sum_n (\sum_m V_{nmd}) x_{nd}}{\sum_{nm} V_{nmd}}, \tag{18}$$

$$\widehat{Var\lambda}_d = \frac{\sum_n (\sum_m V_{nmd})(x_{nd} - \widehat{Mean\lambda}_d)^2}{\sum_{nm} V_{nmd}}. \tag{19}$$

The feature saliency parameters are updated during EM training as follows:

$$\widehat{\rho_d} = \frac{\max(\sum_{nm} U_{nmd} - \frac{ML}{2}, 0)}{\max(\sum_{nm} U_{nmd} - \frac{ML}{2}, 0) + \max(\sum_{nm} V_{nmd} - \frac{s}{2}, 0)}. \tag{20}$$

IV. Software Tool

A well known framework for information visualization system is Shneiderman's mantra [37] which states: 'Overview first, zoom and filter, then details on demand'. Based on this, a framework was proposed in [38] and a MATLAB-based tool, Data Visualization and Modeling System (DVMS), was developed. It uses principled projection algorithms like PCA, GTM and hierarchical GTM for dimensionality reduction combined with information visualization techniques like scatter plots and parallel coordinates. DVMS uses the MATLAB toolbox NETLAB [39] for the machine-learning algorithms. First the proposed framework performs dimensionality reduction from high-dimensional data to two-dimensional space. The projected data can then be visualized using scatter plots to get an overview of structure of the data. The system provides interactive scatter plots by which the user can select any point from the region of interest on the plot and a number of neighbouring data points around the selected point: this group of points is visualized using parallel coordinates, providing the user with a more detailed view of data space. We recently re-designed and re-developed DVMS to improve its usability using the partially object-oriented facilities provided in MATLAB, and have released this tool on our website[11]. We have included log-space versions of GTM and GTM-FS as part of this tool.

V. Experiments

A. Dataset

The dataset we used for experiments is related to MHC class-I. We downloaded a sequences of gene-A, gene-B and gene-C alleles from the IMGT/HLA database [40] (releases July 2011 for gene-A and November 2011 respectively for gene-B and gene-C). According to the IMGT/HLA database nomenclature[12], the HLA allele name has six parts of which the first is HLA prefix, the second is gene name, the third is allele group, the fourth is specific protein id, the fifth is synonymous DNA substitution in the coding region and the sixth contains codes to represent the differences in the non-coding region along with a suffix to express changes in the expression. At first we excluded all those sequences which have 'N'[13] or 'L'[14] or 'Q'[15] as suffix at the end of the sixth part of the allele name. Secondly, from the rest of the allele set we have considered only those protein sequences that either have only one known DNA substitution within the coding region or if there is more than one DNA substitution, only the sequence with maximum length was considered. So, we modeled $1,236$ proteins for gene-A, $1,779$ for gene-B and 929 for gene-C, using homology modeling with three reference proteins, as in [41], retrieved from the RCSB protein database. For polymorphic residues, side chain placement was performed using SCWRL4 [42]. All structures of gene-B and gene-C were super-positioned on one of the structures of gene-A based on the C-Alpha carbon atom. For computing electrostatic potentials using the APBS tool, protein structures (in PDB format) were surrounded by a three-dimensional grid box with 17^3 grid points placed on the target region. We used electrostatic potentials calculated at the top region of $\alpha 1$ and $\alpha 2$ chains with $9 \times 17^2 = 2601$ grid points (as shown in Figure 2). We are interested to analyze electrostatic potential values outside the van der Waals surface of proteins and therefore we ignored electrostatic potential values of all points which were inside the van der Waals surface of all the target set of proteins. The number of dimensions we considered for a set of alleles of gene-A, gene-B, gene-C, and all three genes combined datasets are given in Table I. Figure 2 shows a protein structure with the bounded box on the target region. We considered electrostatic potential at all those grid points which are outside the top surface of all the proteins in a given set, resulting in different numbers of points in different datasets (see Table I). The purpose of our analysis is to identify group of similar proteins based on the similarities of electrostatic potential around the top surface of MHCs that is exposed to T cell Receptor (TCR), and to identify supertypes.

Gene	No of Protein Structures	Target Region Grid Points
A	1,236	2,369
B	1,799	2,382
C	929	2,388
A, B and C	3,944	2,418

TABLE I
SUMMARY OF PREDICTED STRUCTURES WITH TARGET REGION DIMENSION.

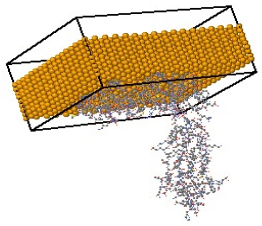
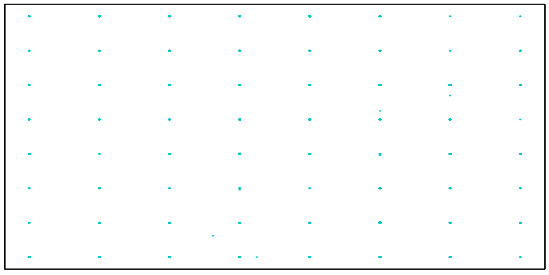


Fig. 2. Three-dimensional structure of protein HLA-A-0001-01-01-01-01 with grid points (orange dots) around the target region of protein. Grid points shown are outside the van der Waals surface.

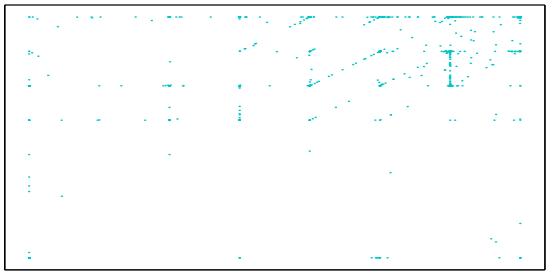
We normalized the data using a linear transformation (Z-score transformation) to maintain similar ranges for all variables and both the variants of the GTM and GTM-FS algorithm were trained using $M = 64$ latent-space grid points. For projection purpose we used proposed variants of GTM (log-space) and GTM-FS (log-space) instead of using standard GTM and GTM-FS to avoid numerical problems that can raise due to the high dimensionality of the data. Standard GTM and GTM-FS have shown numerical problems partially on

[11] http://www.aston.ac.uk/ncrg
[12] http://hla.alleles.org/announcement.html
[13] 'Null', representing an allele which is not expressed.
[14] representing a sequence with low cell surface expression.
[15] representing sequences that are questionable.

the dataset of dimension greater than 200 and fully on the dimensions greater than 500. Projection of gene-A, gene-B and gene-C datasets are shown in Figure 3, Figure 4 and Figure 5 using both proposed variants (GTM (log-space version) and GTM-FS (log-space version)) respectively. Projections of the combined dataset of gene-A, gene-B, and gene-C are shown in Figure 6(a) and Figure 6(b) using GTM (log-space version) and GTM-FS (log-space version) respectively. A feature saliency plot for the combined dataset of gene-A, gene-B and gene-C is shown in Figure 7(a) which explains saliency of each feature in the dataset. The plot is bit cluttered due to the large number of dimensions and it can be observed from the plot that most of the features have high saliency for the dataset of MHC class I. For further exploration of these plots, interactive functions are supported by DVMS to help users to select dense areas to identify clusters or outliers either by drawing a polygonal region of interest or selecting the k-nearest-neighbours (based on Euclidean distance in the latent space) around a user-selected point. The interactivity function is shown in Figure 7(b) to explain both the region selection methods (i.e. polygon region and K-nearest neighbours region).



(a)

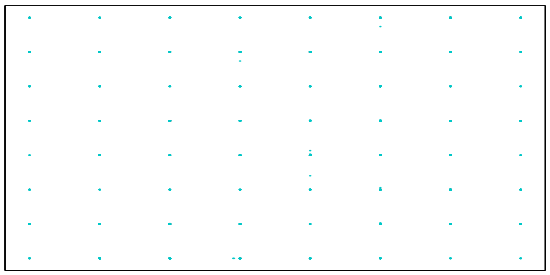


(b)

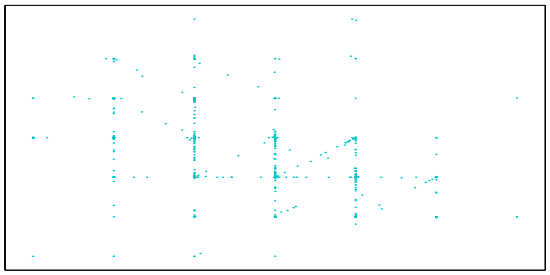
Fig. 3. Projection of Gene-A dataset. (a) GTM (Log-space version). (b) GTM-FS (Log-space version).

B. Kullback-Leibler (KL) divergence

We compared the proposed variants of GTM and GTM-FS on the combined dataset of gene-A, gene-B and gene-C using Kullback-Leibler (KL) divergence as a measure of class dissimilarity. We prefer projections in which these gene classes are separated. To compute this dissimilarity measure, we first built a Gaussian mixture model [43] (GMM) with 18 Gaussian mixture components on the projected data of each three classes separately and then calculated the Kullback-Leibler (KL) divergence [44] between classes using these GMMs

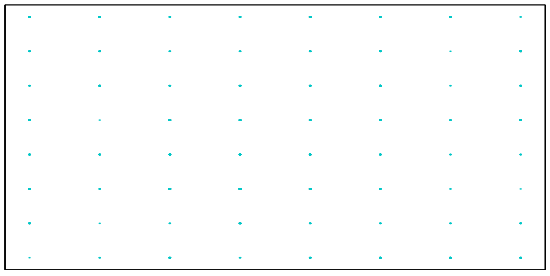


(a)



(b)

Fig. 4. Projection of Gene-B dataset. (a) GTM (Log-space version). (b) GTM-FS (Log-space version).



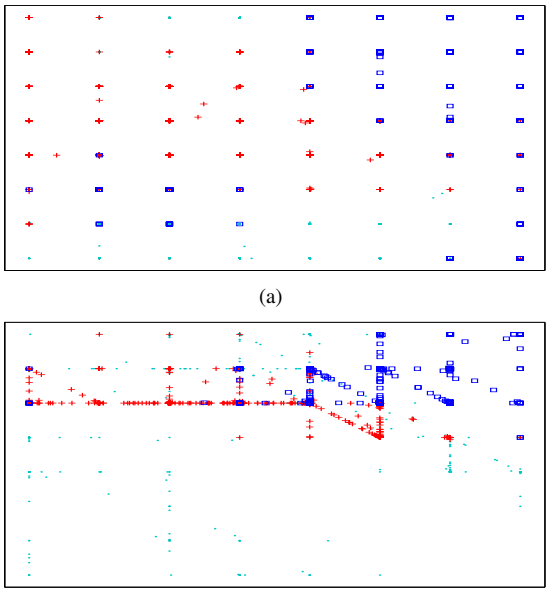
(a)

(b)

Fig. 5. Projection of Gene-C dataset. (a) GTM (Log-space version). (b) GTM-FS (Log-space version).

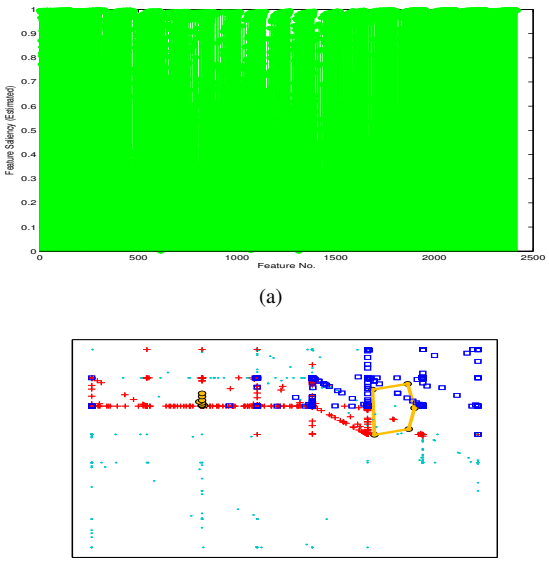
$$D_{KL}(p_a||p_b) = \sum_x p_a(x) \log \frac{p_a}{p_b}, \tag{21}$$

where p_a and p_b are GMMs for the classes A and B. We sum up the KL divergences for all pairs of classes and the results are shown in Table II. Higher values of KL divergence represents better separation between classes. The novel variances of GTM are also compared with other linear/non-linear



(a) (b)

Fig. 6. Projection of combined dataset (Cyan color dots ('.') for gene-A, red positive sign ('+') for gene-B and blue squares ('□') for gene-C). (a) GTM (Log-space version). (b) GTM-FS (Log-space version).



(a) (b)

Fig. 7. Feature Saliency and Projection of combined dataset. (a) Feature saliency plot (b) GTM-FS (Log-space version) showing selection on the plot using interactive tool.

projection algorithms: PCA, Neuroscale [45] (NSC), Gaussian Process Latent Variable Model [46] (GPLVM).

We also computed other projection quality measures such as trustworthiness and continuity [47], range from 0 to 1, taking 20 nearest neighbours. Higher values represents better neighbourhood preservation and the results in Table II show that GTM gives better neighbourhood preservation than of GTM-FS and other algorithms.

Algorithm	KL divergence	Trustworthiness	Continuity
GTM log-space version	**124.4432**	**0.8306**	**0.8407**
GTM-FS log-space version	87.7051	0.7321	0.7957
GPLVM	16.0828	0.7171	0.7972
Neuroscale	20.3281	0.7061	0.8197
PCA	16.6686	0.6964	0.7961

TABLE II
VISUALIZATION QUALITY MEASURES.

VI. DISCUSSION

Both proposed variants of GTM and GTM-FS have yielded similar visual representations for all the datasets. Figures 3(a), 4(a), 5(a) and 6(a) show all the latent-grid centres with tight clusters around latent space grid points. On the other hand, Figures 3(b), 4(b), 5(b) and 6(b) show some diagonal structures in all the datasets.

The advantage of using GTM-FS is that the feature saliencies can be calculated during the training process. The KL divergence, trustworthiness and continuity measures of GTM (log-space version) are higher (as shown in Table II) than the GTM-FS (log-space version) and other algorithms, which shows that GTM gives much better separation between classes and maintains better neighbourhood preservation. In both the plots (Figure 6) there are three major groups (or clusters): the first (the top right corner) where alleles of gene-C are shown as separate cluster(s) from gene-A and gene-B alleles, the second (the bottom area from centre to left) where gene-A alleles have a separate cluster from gene-B and gene-C alleles and the third (the diagonally central region from top left to bottom right) where gene-B alleles are maximum in number with few regions with some alleles of gene-A and gene-C. Interactivity on these plots helps users to identify and select regions to generate list of IDs for the alleles in the specified region.

VII. CONCLUSION

In this paper we proposed variants of the non-linear projection methods of GTM and GTM with simultaneous feature saliency (GTM-FS) to visualize high-dimensional datasets in a low-dimensional space. Our proposed variants use a transformation from real space to log space and vice versa at certain steps of the EM algorithm for training the parameters and during visualization process in order to avoid numerical problems that arise due to the high dimensionality of the data. We successfully tested both the proposed variants on the MHC class-I dataset (with more than 2000 dimensions). Both the proposed algorithms have been incorporated into a visualization tool (DVMS) that can be accessed freely online. We will extend this approach to hierarchical GTM [48], a probabilistic mixture-based hierarchical visualization algorithm.

ACKNOWLEDGMENT

Shahzad Mumtaz is thankful to Prof. Dr. Belal A. Khan (Rector, Foundation University) for arranging funds to do his PhD studies.

REFERENCES

[1] D. L. Donoho, "Aide-memoire. high-dimensional data analysis: The curses and blessings of dimensionality," 2000.

[2] I. Fodor, "A survey of dimension reduction techniques," tech. rep., Lawrence Livermore National Laboratory, 2002.

[3] R. Bellman and R. Corporation, *Dynamic programming*. Rand Corporation research study, Princeton University Press, 1957.

[4] N. M. Luscombe, D. GreenBaum, and D. Gerstein, "What is bioinformatics? an introduction and overview," *Methods of Inoformation in Medicine*, vol. 40, no. 4, pp. 346–358, 2001.

[5] J. M. Thornton, A. E. Todd, D. Milburn, N. Borkakoti, and C. A. Orengo, "From structure to function, applications and limitations," *Nature Structur to function: Approaches and Limitations*, vol. 7, pp. 991–994, 2000.

[6] K. Gupta, D. Thomas, S. V. Vidya, and S. Ramakimar, "Detailed protein sequence alignment based on spectral similarity score(sss)," *BMC Bioinformatics*, vol. 6, no. 105, pp. 1–16, 2005.

[7] U. Langel, B. F. Cravatt, A. Graslund, G. V. Heijne, T. Land, S. Niessen, and M. Zorko, *Introducntion to Proteins and Peptides*. CRC Press Taylor and Francis Group, 2010.

[8] C. Chothia and A. M. Lesk, "The relation between the divergence of sequence and structure in proteins," *The EMBO journal*, vol. 5, no. 4, pp. 823–826, 1986.

[9] R. A. Laskowski, J. D. Watson, and J. M. Thornton, "Protein function prediction using local 3d templates," *Journal of Molecular Biology*, vol. 351, no. 3, pp. 614–626, 2005.

[10] M. Smyth and J. Martin, "X-ray crytallography," *Clin Pathol:Mol Pathol*, vol. 53, pp. 8–14, 2000.

[11] W. Gronwald and H. Kalbitzer, "Automated structure determination of proteins by NMR spectroscopy," *Biological Cybernetics*, vol. 44, pp. 33–96, 2004.

[12] R. A. Meyers, *Protein: Electron Microscopy of Biomolecules*. Wiley VCH, 2007.

[13] D. Lee, O. Redfern, and C. Orengo, "Predicting protein function from sequence and structure," *Nature Reviews Molecular Cell Biology*, vol. 8, pp. 995–1005, 2007.

[14] F. C. Bernsten, T. F. Koetzle, G. F. Williams, E. E. Meyer, M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi, and M. Tasumi, "The protein data bank: A computer-based archival file for macromolecular structures," *Journal of Molecular Biology*, vol. 112, no. 535, 1977. http://www.rcsb.org/pdb/.

[15] E. Krieger, E. B. Nabuurs, G. Vriend, E. Philip, E. Bourne, and H. Weissig, "Homology modeling," *Methods of Biochemical Analysis*, vol. 44, 2003.

[16] F. Dong, B. Oslen, and N. A. Baker, "Computation methods for biomolecular electrostatics," *Mehtods in Cell Biology*, vol. 84, pp. 843–870, 2008.

[17] C. Azuara, E. Lindahl, P. Koehl, H. Orland, and M. Delarue, "Pdbhydro: Incorporating dipolar solvents with variable density in the poisson-boltzmann treatment of macromolecule electrostatics," *Nuclein Acids Research*, vol. 34, pp. 38–42, 2006.

[18] R. V. Polozov, V. S. Sivozhelezov, V. V. Ivanov, and Y. B. Melnikov, "On a classification of E. coli promoters according to their electrostatic potential," *Particles and Nuclei Letters*, vol. 2, no. 4(127), pp. 82–90, 2005.

[19] S. Ritcher, A. Wenzel, M. Stein, R. R. Gabdoulline, and R. C. Wade, "WebPIPSA: A web server for the comparison of protein interaction propoerties," *Nucleic Acid Research*, vol. 36, pp. 276–280, 2008.

[20] N. Blomberg, R. R. Gabdoulline, N. Michael, and R. C. Wade, "Classification of protein sequences by homology modeling and quantiative analysis of electrostatic similarity," *Proteins: Structure, Function and Genetics*, vol. 37, pp. 379–387, 1999.

[21] I. Jolliffe, *Principal Component Analysis*. 2nd Edn, Springer Series in Statistics, 2002.

[22] J. H. Friedman and J. W. Tukey, "A projection pursuit algorithm for exploratory data analysis," *IEEE Trans. Comput.*, vol. 23, pp. 881–890, September 1974.

[23] R. B. Cattell, "Factor analysis: An introduction to essentials II. the role of factor analysis in research," *Biometrics*, vol. 21, no. 2, pp. 405–435, 1965. http://www.jstor.org/stable/2528100.

[24] D. M. Maniyar and I. T. Nabney, "Data visualization with simultaneous feature selection," in *Computational Intelligence and Bioinformatics and Computational Biology, 2006. CIBCB '06. 2006 IEEE Symposium on*, pp. 1–8, 2006.

[25] T. Kohonen, *Self Organizing Maps*. Springer, third ed., 2001.

[26] J. W. Sammon, "A nonlinear mapping for data structure analysis," *IEEE Transactions on Computers*, vol. 18, pp. 401–409, 1969.

[27] C. M. Bishop and M. Svensen, "GTM: The generative topographic mapping," *Neural Compuatation*, vol. 10, no. 1, pp. 215–234, 1998.

[28] S. Mahony, D. Hendrix, A. Golden, T. J. Smith, and D. S. Rokhsar, "Transcription factor binding site identification using the self-organizing map," *Bioinformatics*, vol. 21, no. 9, pp. 1807–1814, 2005.

[29] N. Fankhauser and P. Mser, "Identification of gpi anchor attachment signals by a kohonen self-organizing map," *Bioinformatics*, vol. 21, no. 9, pp. 1846–1852, 2005.

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

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

[32] Y. h. Taguchi and Y. Oono, "Relational patterns of gene expression via non-metric multidimensional scaling analysis," *Bioinformatics*, vol. 21, no. 6, pp. 730–740, 2005.

[33] 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 Modeling*, vol. 46, no. 4, pp. 1806–1818, 2006.

[34] J. Qiu, J. Ekanayake, T. Gunarathne, J. Y. Choi, S.-H. Bae, Y. Ruan, S. Ekanayake, S. Wu, S. Beason, G. C. Fox, M. Rho, and H. Tang, "Data intensive computing for bioinformatics," tech. rep., Indiana University, Bloomington, IN, 12/29/2009 2009.

[35] A. Gisbrecht and B. Hammer, "Relevance learning in generative topographic mapping," *Neurocomputing*, vol. 74, no. 9, pp. 1358–1351, 2011.

[36] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[37] B. Shneiderman, "The eyes have it: A task by data type taxonomy for inofrmation visualization," in *IEEE Symposium on Visual Languages*, pp. 336–343, 1996.

[38] D. M. Maniyar and I. T. Nabney, "Visual data mining using principled projection algorithms and information visualzation techniques," in *12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 643–647, 2006.

[39] I. T. Nabney, *Netlab: Algorithms for pattern Recognition*. UK, Springer, 2002.

[40] J. Robinson, M. J. Waller, N. Parham, D. Groot, H. R. Kalbitzer, L. J. Bontrop, P. Kennedy, P. Stoehr, and S. G. E. Marsh, "IMGT/HLA and IMGT/MHC: Sequence databases for the study of the major histocompatibility complex," *Nucleic Acids Res*, vol. 31, no. 311, 2003.

[41] I. A. Doytchinova, P. Guan, and D. R. Flower, "Identifying human mhc supertypes uisng bioinformatics methods," *The Journal of Immunology*, vol. 172, pp. 4314–4323, 2004.

[42] M. J. Bower, F. E. Cohen, and J. Dunbrack, "Sidechain prediction from a backbone-dependent rotamer library: A new tool for homology modeling," *J. Mol. Biol*, vol. 267, pp. 1268–1282, 1997.

[43] C. M. Bishop, *Neural Networks for pattern recognition*. Oxford University Press, New York, 1st ed., 1995.

[44] T. M. Cover and J. A. Thomas, *Elements of Information theory*. Springer, 1st ed., 1991.

[45] D. Lowe and M. E. Tipping, "Neuroscale: Novel topographic feature extraction using rbf networks," 1997.

[46] N. D. Lawrence, "Gaussian process latent variable models for visualisation of high dimensional data," in *In NIPS*, p. 2004, 2004.

[47] J. Venna and S. Kaski, "Neighborhood preservation in nonlinear projection methods: An experimental study," 2001.

[48] P. Tino and I. T. Nabney, "Hierarchical GTM: Constructing localized nonlinear projection manifolds in a principled way," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 639–656, 2002.