JOURNAL OF CHEMICAL INFORMATION AND MODELING

ARTICLE

pubs.acs.org/jcim

Visualization of Molecular Fingerprints

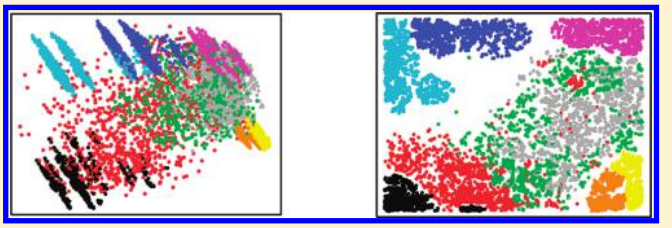
John R. Owen,[†] Ian T. Nabney,[*,†] José L. Medina-Franco,[‡] and Fabian López-Vallejo[‡]

[†]Nonlinearity and Complexity Research Group (NCRG), Aston University, Aston Triangle, Birmingham B4 7ET, United Kingdom

[‡]Torrey Pines Institute for Molecular Studies, 11350 SW Village Parkway, Port St. Lucie, Florida 34987, United States

Supporting Information

ABSTRACT: A visualization plot of a data set of molecular data is a useful tool for gaining insight into a set of molecules. In chemoinformatics, most visualization plots are of molecular descriptors, and the statistical model most often used to produce a visualization is principal component analysis (PCA). This paper takes PCA, together with four other statistical models (NeuroScale, GTM, LTM, and LTM-LIN), and evaluates their ability to produce clustering in visualizations not of molecular descriptors but of molecular fingerprints. Two different tasks are addressed: understanding structural information (particularly combinatorial libraries) and relating structure to activity. The quality of the visualizations is compared both subjectively (by visual inspection) and objectively (with global distance comparisons and local *k*-nearest-neighbor predictors). On the data sets used to evaluate clustering by structure, LTM is found to perform significantly better than the other models. In particular, the clusters in LTM visualization space are consistent with the relationships between the core scaffolds that define the combinatorial sublibraries. On the data sets used to evaluate clustering by activity, LTM again gives the best performance but by a smaller margin. The results of this paper demonstrate the value of using both a nonlinear projection map and a Bernoulli noise model for modeling binary data.



INTRODUCTION

With the recent advances in combinatorial chemistry and computational chemistry, pharmaceutical chemists frequently need to analyze very large data sets of molecular properties. A visualization plot of a data set of molecular descriptors is a useful tool for gaining insight into a set of molecules.[1] Let M be a statistical model that produces a visualization plot by projecting a high-dimensional data set (molecular descriptors) to a space of lower dimensionality (usually two-dimensional space), while maintaining as much geometric structure as it can. In two dimensions, the projected data set can easily be plotted (using a scatter plot) and visualized. A typical data set is a set of n structurally related molecules, each molecule represented by a vector of p descriptors. These descriptors are assembled into an $(n \times p)$ data array, and this is acted on by the model. M then projects the data array to a two-column array of points; these points are plotted to produce a visualization of the data set. If the points are colored by molecular structural class, the scientist can inspect the visualization for structure clustering in the hope of gaining insight into the molecular structures of the data set. Similarly, if the points are colored by binned-activity value, the visualization may reveal new structure−activity relationships.

Principal Component Analysis (PCA) is the statistical model most commonly used to produce visualizations. Several good examples of its use as a research tool have appeared over the past decade.[2−6] PCA is a linear mapping. Over the past 15 years, statistical models more complex than PCA, based on nonlinear mappings, have been developed by the machine learning community; Yin[7] is a good overview of some of these models. This paper takes four of these models, together with PCA, and evaluates their ability to produce both structure and activity clustering in visualizations of drug-like molecules. The four non-PCA models evaluated are NeuroScale, the Generative Topographic Mapping (GTM), the Latent Trait Model (LTM), and the Linear Latent Trait Model (LTM-LIN). Visualizations produced by these models are evaluated, and the model that performs best is identified.

In chemoinformatics, PCA is commonly used with molecular descriptors; most descriptors are continuous (as opposed to discrete) variables (e.g., logP, molecular weight). In this paper, molecules are represented by molecular fingerprints, which as bitstrings are collections of binary variables. (Two of the models, LTM and LTM-LIN, were specifically designed to model discrete variables.) To be processed by the models, the fingerprints are assembled into a "fingerprint array" in which each element is either 0 or 1. In this approach, the fingerprints must be of fixed length; the models cannot be directly used with variable length fingerprints. In all of the analyses herein, the MDL 166-key[8] fingerprint has been used; this is a nonhashed fingerprint consisting of 166 bits. (The MDL 166-key fingerprint is also known as the "Molecular Access System (MACCS)" key in the literature.) The ideal fingerprint would have a 1−1 mapping between each bit and the corresponding molecular feature. In practice, the MDL 166-key fingerprint is one of the very few available that offers a 1−1 mapping. Most other fingerprints map

Received: October 15, 2010
Published: June 22, 2011

several molecular features to a single bit (many−1). This will not invalidate the models' projections, but it will produce some "noise" in that molecules with different properties may map to the same bitstring and hence be projected to the same location. Other fingerprints that are suitable for use with the models include the MDL 960-key[8] (960 bits) and PubChem's CACTVS[9] (881 bits). These clearly provide much greater detail than the MDL 166-key. The use of high-resolution fingerprints will provide more details in the plots, but it may also weaken broad-scale clustering by splitting large clusters into several smaller ones.

PCA and a nonlinear mapping were used by Clark et al.[10] to visualize data sets of around 300 fingerprints. Their nonlinear mapping was essentially a modified version of Sammon[11] mapping, an algorithm related to the NeuroScale model described herein.

In the clustering-by-activity analyses herein, the "similarity principle" is assumed to hold viz. that structurally similar molecules have similar biological activities. Martin et al.[12] reported that this principle is generally true, and that the greater the structural similarity, the greater the biological similarity. They also reported that the similarity in biological activity values between molecules is not as close as many researchers had previously thought.

This paper is structured as follows. First of all, PCA and the four non-PCA models are described. The next section evaluates the ability of the five models to cluster by structure and has the following subsections: description of the data sets; performance measurement and results; and visualizations produced by PCA and LTM. Similarly, the next section evaluates the ability of the five models to cluster by activity and includes a similar set of subsections. Finally, conclusions are drawn from the results.

MODELS EVALUATED

Five models were evaluated for clustering by molecular structure and activity; a brief overview of each of these models is given in the following subsections. Little, if any, of the mathematics underlying each model has been included, for that the reader is referred to the cited papers. Essentially, all five models were used to reduce the dimensionality of the fingerprint array from 166-dimensional space to two-dimensional space where it could be plotted and visualized.

Principal Component Analysis (PCA). PCA[13] has been widely applied in scientific research for many years. The main advantage of PCA is its robustness; the algorithm will successfully map almost any data set without fail. The main disadvantage of PCA is that it is a linear mapping; the other models herein, apart from LTM-LIN, are nonlinear and can map more complex structures than PCA. Tipping and Bishop[14] reformulated PCA as Probabilistic PCA (PPCA); PPCA is essentially PCA but with a probabilistic model for the observed data. The use of a probabilistic model provides a direct link to GTM (described below). The fundamental model structure is that the data has a Gaussian distribution.

Assume that PCA is to be performed on an $(n \times p)$ matrix $\mathbf{X}$ of n molecules with p descriptors. Let the covariance matrix of $\mathbf{X}$ be $\mathbf{C} = \mathbf{X}\mathbf{X}^T$. Let the eigenvalues of $\mathbf{C}$ be $\lambda_i (i = 1,...,n)$, and assume they have been sorted so that $\lambda_1 \geq ... \geq \lambda_n$; let the corresponding eigenvectors of $\mathbf{C}$ be $\mathbf{v}_i (i = 1,...,n)$. These eigenvectors are the principal components of the PCA mapping. $\mathbf{v}_1$ captures the largest variance in the data set; $\mathbf{v}_2$ captures the second largest variance in a direction orthogonal to $\mathbf{v}_1$; etc. By projecting onto the first two principal components $\mathbf{v}_1$ and $\mathbf{v}_2$, the data set $\mathbf{X}$ can be mapped to two-dimensional space while maintaining a lot of geometric structure. PCA is a nonparametric algorithm as its only input is $\mathbf{X}$ and it has no parameters that can be adjusted.

NeuroScale. NeuroScale uses a nonlinear Radial Basis Function (RBF) neural network to produce a mapping from p-dimensional space to d-dimensional space (for a visualization $d = 2$). The model was introduced by Tipping and Lowe[15] and was designed to overcome some weaknesses in the Sammon mapping.[11]

A nonlinear optimization algorithm (named "shadow targets" by Tipping and Lowe) is used to estimate the parameters of the RBF network. The error function for the RBF network is the Sammon stress metric

$$E_{sam} = \sum_{i=1}^{N} \sum_{j>i}^{N} (d_{ij} - d_{ij}^*)^2$$

where N is the number of molecules in the data set, d_{ij} is the distance between molecules i and j in the latent (i.e., visualization) space; and d_{ij}^* is the distance between molecules i and j in the data (i.e., descriptor) space. For molecules, the distance is usually given by the Soergel distance function: if $\mathbf{x}$ and $\mathbf{y}$ are fingerprint vectors, then the Soergel distance between $\mathbf{x}$ and $\mathbf{y}$ is defined to be the complement of the Tanimoto similarity function

$$soergel(\mathbf{x}, \mathbf{y}) = 1 - tanimoto(\mathbf{x}, \mathbf{y}) \quad (1)$$

$$tanimoto(\mathbf{x}, \mathbf{y}) = \left(\frac{\mathbf{x}.\mathbf{y}^T}{\mathbf{x}.\mathbf{x}^T + \mathbf{y}.\mathbf{y}^T - \mathbf{x}.\mathbf{y}^T} \right)$$

(In the tanimoto($\mathbf{x}$, $\mathbf{y}$) function: if the denominator is 0, then the value is 0, unless if $\mathbf{x} = \mathbf{y} = \mathbf{0}$ when the value is 1.) The Soergel distance lies in the range [0,1]; the closer it is to 1, the greater the distance between the two fingerprints. Other distance functions (e.g., Euclidean, Hamming) can also be used.[16]

Generative Topographic Mapping (GTM). GTM was introduced by Bishop et al.[17,18] and was designed to improve upon the self-organizing map (SOM) of Kohonen.[19] SOM has been widely employed, but it suffers from a number of theoretical weaknesses. These weaknesses and a detailed description of GTM can be found in Nabney.[20] GTM is nonlinear and works by training an RBF neural network to produce a mapping from two-dimensional latent (i.e., visualization) space to n-dimensional data (i.e., descriptor) space. GTM is a probabilistic model and can be viewed as a constrained mixture of Gaussians: thus, the assumption is that the data attributes are continuous variables. The parameters for the RBF network are found by using the (iterative) EM algorithm.[21,22] Once the RBF network has been trained, the reverse mapping (data space to latent space) for a visualization is produced by a straightforward application of Bayes's theorem.

Note that PPCA (described above) is a simplification of GTM in which GTM's nonlinear RBF mapping is replaced by a linear mapping and a single Gaussian output.

Latent Trait Model (LTM); Linear Latent Trait Model (LTM-LIN). Kabán and Girolami[23] modified GTM to model data sets of discrete variables. Their GTM-derived model, called LTM, models data set noise with a Bernoulli, rather than a Gaussian, noise model. So in LTM, each variable in the data set must be binary (i.e., 0 or 1); in GTM, each variable is continuous (i.e., can take any real value). Because of this property, LTM is well suited to

modeling the (binary valued) fingerprint array. A related model, called LTM-LIN, has been derived from LTM by the authors of this paper. Although LTM-LIN seems unlikely to be novel, it has not been found in the literature. LTM-LIN is the same as LTM, but with LTM's nonlinear RBF-based mapping replaced by a linear mapping.

Model Implementation. All of the models were implemented using Netlab[24,20] (a free-to-download Matlab toolbox for neural networks and pattern recognition). The following material can be downloaded from the Supporting Information for this paper: Matlab scripts for all of the models; a Matlab program called the Data Visualization and Modeling System (DVMS); and the DVMS User Guide. DVMS has an easy-to-use GUI that provides access to all of the models and a set of interactive tools[25] to help the user interpret the visualization plots.

The fingerprint array was used as the data set for all of the models. When evaluating the models' ability to cluster by activity, NeuroScale was found to work best when the fingerprint array had been normalized (i.e., rescaled so that each column had zero mean and unit variance). Normalization did not lead to better results with NeuroScale when clustering by molecular structure, and so it was not applied in the structure clustering analyses. Data normalization was also not appropriate for any other model.

Post-Training Projection of New Data. Assume a chemist has trained a model with an array of data **D**, and post-training the chemist wishes to project with this model a new array of data **E**. With PCA and the four non-PCA models herein, no additional training is required to project **E**; with post-training, all of these models are able to project new data (functions exist within the Netlab toolbox for projecting new data). However, with the Sammon mapping (related to NeuroScale above), new points can only be projected by generating an entirely new model: a new array **F** is formed by merging **E** with **D**, and a completely new model is generated by training with **F**. Despite its name, the Sammon mapping does not define a projection between two spaces. Instead, it finds points in the visualization space that correspond to its training points in the data space. Consequently, the Sammon mapping cannot project new points without retraining (making it impossible to assess generalization), and its smoothness cannot be controlled (which can lead to problems of overfitting).

CLUSTERING BY MOLECULAR STRUCTURE

Data Sets for Clustering by Molecular Structure. The data sets used to evaluate the models' ability to cluster compounds by molecular structure are given in Table 1. The Molecular Operating Environment (MOE)[26] was used to generate a set of MDL 166-key fingerprints for each of the data sets. These fingerprints were assembled into the fingerprint array, and this array was then passed to the models. Data set A was data sets A1 to A5 merged into a single data set and contained 5000 molecules. Similarly, data set B was data sets B1 to B5 merged into a single data set and contained 5000 molecules. Data set C was data sets A1 to A5 and B2 to B5 merged into a single data set and contained 9000 molecules (B1 was not merged into C as B1 was identical to A1). Data set C1 contained 5000 molecules randomly sampled from data set C; data set C1 was created as not enough memory was available to run a NeuroScale model with data set C (NeuroScale would have required a distance array with 9000 × 9000 elements). This highlights an advantage of NeuroScale over the Sammon mapping: as NeuroScale defines a projection mapping, it can be trained on a subsample of a data set and then can be used to project the entire data set; this is not possible with the Sammon mapping.

Table 1. Data Sets Used to Evaluate Structure Clustering.[a]

data set	color	diversity	no. entries	description
A1	red	0.69	1000	approved drugs from ZINC database
A2	yellow	0.12	1000	combinatorial library
A3	black	0.32	1000	combinatorial library
A4	magenta	0.22	1000	combinatorial library
A5	blue	0.24	1000	combinatorial library
A	N/A	0.49	5000	data sets A1 to A5 merged
B1	red	0.69	1000	identical to data set A1
B2	cyan	0.25	1000	combinatorial library
B3	gray	0.31	1000	MLSMR compounds from PubChem database
B4	green	0.41	1000	natural products from ZINC database
B5	orange	0.12	1000	combinatorial library
B	N/A	0.49	5000	data sets B1 to B5 merged.
C1	N/A	0.47	5000	random sample of 5000 entries from data set C
C	N/A	0.46	9000	data sets A1 to A5, and B2 to B5, merged

[a] The colors are those used in the visualization plots; the diversities were computed using eq 3.

In Table 1, six data sets were combinatorial libraries, and three were taken from the public databases PubChem[27,28] and ZINC.[29] The six combinatorial libraries used were bis-cyclic diketopiperazines (A2), *N*-methyl triamines (A3), dihydroimidazolyl methyl-diketopiperazines (A4), dihydroimidazolyl-butyl-cyclic ureas (A5), bis-cyclic guanidines (B2), and *N*-methyl-1,4,5-trisubstituted-2,3-diketopiperazines (B5).[30] The chemical structures of the core scaffolds are given in Figure 1. All of the combinatorial libraries were enumerated using the QuaSAR—Combi-Design module in MOE. Because each scaffold had three attachment points (Figure 1), diverse subsets of 1000 compounds were built by connecting a combination of 10 different *R* groups per attachment point to the selected scaffolds (giving $10^3 = 1000$ compounds). Table S1 in the Supporting Information gives the *R* groups used to build the libraries. The three data sets taken from the PubChem and ZINC databases were as follows. Data set A1 contained a random selection of 1000 drugs from DrugBank,[31] as implemented in ZINC (downloaded January 2011). Data set B3 was a diverse set of 1000 compounds from the Molecular Libraries Small Molecule Repository (MLSMR), which was retrieved from PubChem (downloaded May 2010). Data set B4 contained 1000 lead-like natural products extracted from ZINC (downloaded March 2010). The data sets were preprocessed with MOE removing salts and keeping the largest structural component.

To add some objectivity to the analysis of the visualizations, the *diversity* of the molecules within each data set was computed as follows. Let d_{uv} be the Soergel-based interdata set distance between the two data sets D_u and D_v

$$d_{uv} = \frac{1}{N_u N_v} \sum_{i=1}^{N_u} \sum_{j=1}^{N_v} soergel(\mathbf{x}_i^u, \mathbf{x}_j^v) \qquad (2)$$

dx.doi.org/10.1021/ci1004042 |J. Chem. Inf. Model. 2011, 51, 1552–1563

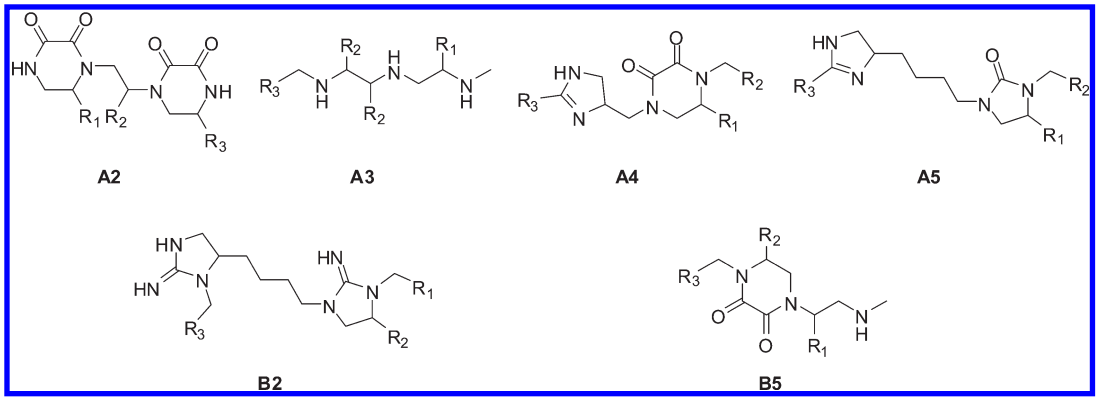


Figure 1. Combinatorial libraries A2 to A5, B2, and B5.

where N_u and N_v are the number of molecules in data sets D_u and D_v, and $\mathbf{x}_r^s$ is the fingerprint vector from row r of the fingerprint array for data set D_s. Let d_u be the diversity of the molecules within a single data set D_u. By rearranging eq 2, an expression which is much quicker to evaluate is obtained

$$d_u = \frac{2}{N_u^2} \sum_{i=1}^{N_u - 1} \sum_{j=i+1}^{N_u} soergel(\mathbf{x}_i^u, \mathbf{x}_j^u) \quad (3)$$

Equation 3 was used to compute the diversity of the data sets in Table 1. The data set of drugs A1 (= B1) has the greatest diversity, so this data set is expected to exhibit the weakest clustering when visualized. Combinatorial libraries A2 and B5 have the lowest diversities and so should exhibit the most clustering. By a chance coincidence, data sets A and B have the same diversity. As data sets C and C1 contain data sets A and B, their diversities are very similar to those of A and B.

Performance Measurement with the Γ-score. This subsection introduces the Γ-score, a score developed to measure the ability of a model to produce similar-structure clustering in a visualization. A useful visualization will contain well-grouped clusters of similar-structure molecules; the better the similar-structure clustering, the more useful is the visualization to the scientist. This score can be calculated for a data set where there is a known grouping of examples into classes and is based on the accuracy of a simple classification model.

The Γ-score is a mean of $\gamma(k)$-cluster scores, which are defined as follows. Assume that the data set being analyzed has been has been split into two arrays: a training array U and a test array V (containing N_V examples). Also assume that a model has been trained using U, and that V has been projected (using the trained model) into the two-dimensional visualization space. Then the $\gamma(k)$-cluster score is given by the following equations

$$\gamma(k) = \frac{1}{N_V} \sum_{i=1}^{N_V} G(i,k) \quad (4)$$

$$G(i,k) = \frac{1}{k} \sum_{j=1}^{k} g(v_i, j) \quad (5)$$

In eq 4, $\gamma(k)$ is the mean of $G(i,k)$. Let the molecule in the ith row of V be denoted by v_i. In eq 5, let the function $g(v_i, j)$ be 1 if the molecule that is jth nearest to v_i in the visualization space is in the same class (i.e., data set subset) as v_i, and otherwise let $g(v_i, j)$ be 0. So, for example, $g(v_i, 2)$ will return 1 if v_i and the molecule that is second closest to v_i in the visualization space have the same class; otherwise $g(v_i, 2)$ will return 0. So $G(i,k)$ is the accuracy of a k-Nearest-Neighbor (kNN) classifier defined using the projection of the test data set V. (Strictly, it is the leave-one-out accuracy by not including the test point in its own neighborhood.) k is an integer, typically set in the range [5,20]; experiments have shown that the value of $\gamma(k)$ is not very sensitive to the choice of k. The greater the value of $\gamma(k)$, the better the clustering of the molecules from V in the visualization space.

Assume that model M has been applied to data set D and that M has been evaluated using n-fold cross-validation. Let $\gamma_i(k)$ be the $\gamma(k)$-cluster score for the visualization plot of fold i. Then $\Gamma_{M,D}(k)$ is defined to be the mean

$$\Gamma_{M,D}(k) = \frac{1}{n} \sum_{i=1}^{n} \gamma_i(k) \quad (6)$$

Model Parameters for Clustering by Structure. All of the models, apart from PCA, had parameters for adjusting their architectures. This subsection gives the architectural parameters of each model, along with the values that were assigned to them when training the models to cluster by structure. NeuroScale has a single architectural parameter, the number of Radial Basis Function (RBF) centers, and this was set to 10. GTM and LTM have the same two architectural parameters: the number of nodes on the side of the square grid that defines the number of RBF centers or the number of latent-space points. R_G is the RBF-centers grid parameter and L_G is the latent-space grid parameter. These parameters were set using the following equations

$$R_G = ceil\left(\sqrt{\frac{N}{5}}\right) \quad (7)$$

$$L_G = ceil\left(\sqrt{\frac{N}{4}}\right) \quad (8)$$

where N is the number of molecules in the data set. LTM-LIN has a single architectural parameter, L_G, and this was set using eq 8. This choice of architectural parameters was found by experiment to give good results. Details of how these parameters affect the architectures of the models can be found in Nabney.[20]

The data sets used to evaluate clustering by structure were A, B, and C1 (Table 1). As each of these data sets had 5000 entries and 5-fold cross-validation was applied, each training data set had 4000 entries and so N = 4000. All of the architectural parameters and the values of eqs 7 and 8 evaluated with N = 4000 are given in Table 2.

Table 2. Architectural Parameters for Structure Clustering

model	parameter	value
NeuroScale	no. RBF centers	10
GTM	RBF-centers grid	29
GTM	latent-space grid	32
LTM	RBF-centers grid	29
LTM	latent-space grid	32
LTM-LIN	latent-space grid	32

Table 3. Γ-Scores for Each Model[a]

data set	k	PCA	NeuroScale	GTM	LTM	LTM-LIN
A	5	0.91	0.96	0.96	**0.99**	**0.99**
B	5	0.68	0.77	0.82	**0.87**	0.79
C1	5	0.69	0.71	0.83	**0.91**	0.82
mean (k = 5)	N/A	0.76	0.81	0.87	**0.93**	0.87
A	15	0.90	0.95	0.92	**0.98**	**0.99**
B	15	0.67	0.76	0.75	**0.83**	0.78
C1	15	0.66	0.68	0.74	**0.87**	0.79
mean (k = 15)	N/A	0.75	0.80	0.80	**0.90**	0.86

[a] Best result in each row is in bold.

Γ-score Results. The Γ-score results are given in Table 3; these results were produced with 5-fold cross-validation. Clearly, all four non-PCA models, having greater mean Γ-scores than PCA, performed better than PCA. LTM produced the best Γ-scores across all five models and all three data sets (apart from on data set A with k = 15 where it was very slightly outperformed by LTM-LIN). LTM, with mean Γ-scores of 0.93 (k = 5) and 0.90 (k = 15), has emerged as the best model for similar-structure clustering, an encouraging result as LTM was specifically designed to model binary data sets. It is also concluded that data set A has the strongest clustering structure: LTM has achieved almost 100% accuracy, implying very clear class-separation in the latent space.

Performance in Interpoint Distance Preservation. This subsection measures the ability of each projection model to preserve interpoint distances in the visualization space. An array of Soergel distances (eq 1) between each pair of points in the data space was computed, together with the corresponding array of Euclidean distances in the visualization space. Pearson's (linear) correlation coefficient was computed between the corresponding elements of these two arrays; the results are given in Table 4. In Table 4, NeuroScale with the highest mean coefficient has emerged as the model most able to preserve interpoint distances. This not a surprising result, as NeuroScale uses the Soergel distance array as input, and its algorithm was specifically designed to preserve distances. PCA, GTM, and LTM all produced a very similar performance; LTM-LIN's performance was the weakest. These results confirm the theory underlying each model. GTM, LTM, and LTM-LIN are designed to produce a latent space where the data is as close to being uniformly distributed as possible (within the constraints of the model's architecture). These models tend to be more informative than NeuroScale in that they give structures between and within clusters (as shown by the Γ-score results); NeuroScale is better for finding outliers because it preserves the global distance structure of its data set. PCA is similar to GTM but with a Gaussian distribution in latent space; this tends to produce visualizations where there is a heavy concentration of points in the middle of the plot.

Table 4. Correlation Coefficient for Distance Preservation[a]

data set	PCA	NeuroScale	GTM	LTM	LTM-LIN
A	0.55	**0.80**	0.51	0.58	0.50
B	0.38	**0.78**	0.47	0.54	0.39
C1	0.58	**0.72**	0.58	0.41	0.42
mean	0.50	**0.77**	0.52	0.51	0.44
rank	4	1	2	3	5

[a] Best result in each row is in bold.

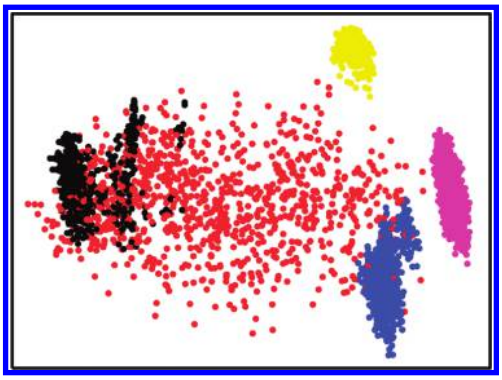


Figure 2. PCA visualization of data set A.

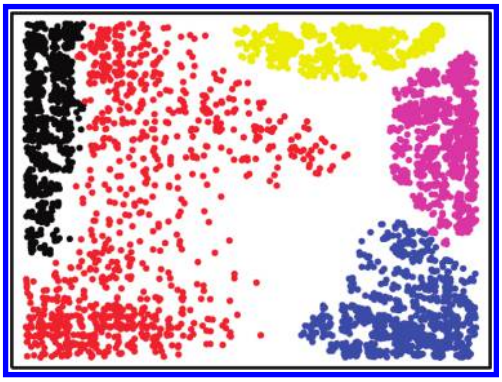


Figure 3. LTM visualization of data set A.

PCA and LTM Visualizations Compared for Clustering by Structure. In practice, a scientist will often want to train and project with the same (single) data set; let this approach be called "entire-data set" training. Entire-data set PCA and LTM visualizations of data sets A, B, and C are given in Figures 2−7. The data set (subset) colors used in these plots are given in Table 1; the LTM architectural parameters used to generate these plots were given by eqs 7 and 8.

LTM (but not PCA) often produces an effect known as "gridding" in visualizations. Several molecules may be projected to a single latent-space grid-point: this produces a regular grid of molecules in the visualization and causes molecules to be plotted directly on top of each other. To reveal gridded molecules and improve the ability of the user to interact with LTM visualizations, random jitter was added to each molecule in all of the LTM visualizations herein. (A jittered point is one that has had small random offsets $(\delta x, \delta y)$ added to its (x, y) coordinates: $x' = x + \delta x$, $y' = y + \delta y$.)

dx.doi.org/10.1021/ci1004042 |J. Chem. Inf. Model. 2011, 51, 1552–1563

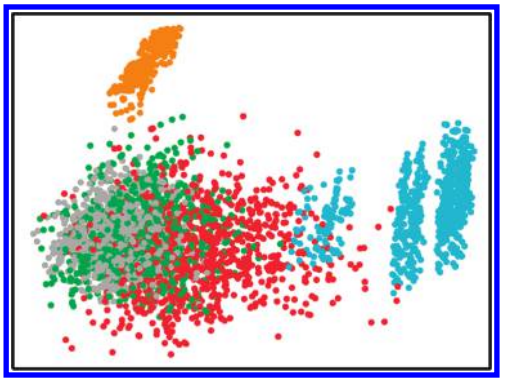


Figure 4. PCA visualization of data set B.

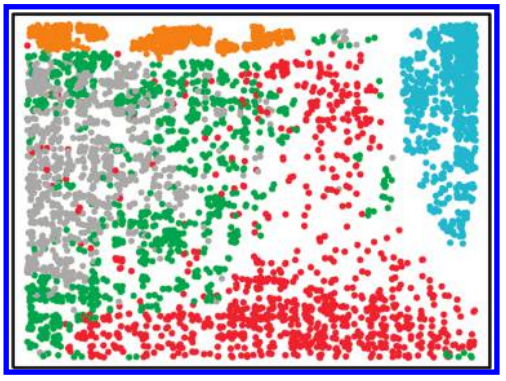


Figure 5. LTM visualization of data set B.

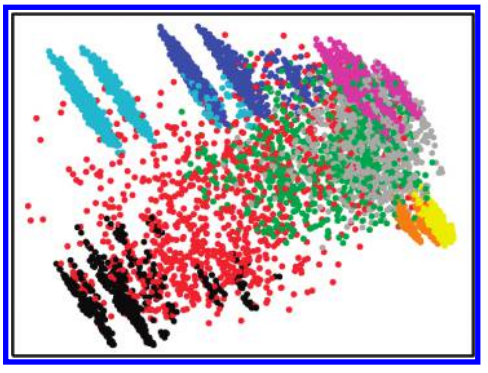


Figure 6. PCA visualization of data set C.

In Figures 2−7, the data set of drugs A1 (red) is the most dispersed, and in Figures 6 and Figure 7, the combinatorial libraries A2 (yellow) and B5 (orange) are the most tightly clustered, as expected from the diversity scores in Table 1.

Comparing Figures 2 and 3, LTM has produced a better separation of the combinatorial libraries A3 (black) and A5 (blue) from the drugs A1 (red) than PCA. Comparing Figures 4 and 5, LTM has again produced better subset separation than PCA, with LTM doing well on the difficult-to-separate gray, green, and red subsets. Comparing Figures 6 and 7, LTM has clearly done far better than PCA in separating the subsets.

In all three LTM plots, LTM has dispersed the molecules over the two-dimensional visualization space much better than PCA and thus has better separated the subsets. This results from the properties of LTM's latent-space grid (in particular, the uniform

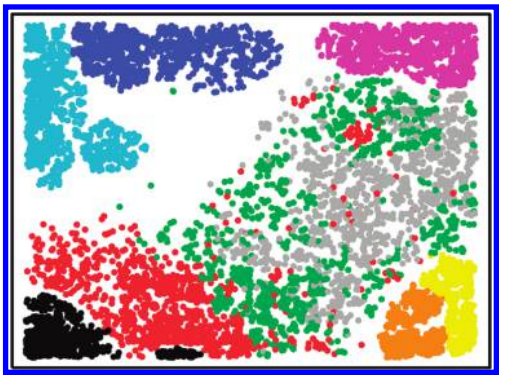


Figure 7. LTM visualization of data set C.

distribution of the projected data) and from the appropriateness of LTM's Bernoulli noise model for binary data. This dispersal ability will be particularly effective in any visualization that contains a large number (>1000) of molecules and will enable the chemist to see subclusters inside clusters that will not be produced by PCA.

Interpretation of LTM's Clustering by Structure. The core scaffolds of the six combinatorial libraries are shown in Figure 1. These combinatorial libraries are part of the ongoing efforts by Medina-Franco et al.[30] to identify active compounds for a number of therapeutic targets. In Figure 3, the subset A1 of drugs (red) is less clustered than the four combinatorial libraries (A2−A5). This result is in full agreement with the diversity of each data set (Table 1) and with the nature of the chemical structure of the libraries. The drug subset contains a wide variety of chemical scaffolds increasing the chemical diversity.[2] In contrast, the combinatorial libraries share a common scaffold (Figure 1). LTM has also successfully captured the structural relationships between the combinatorial libraries A2−A5: in Figure 3, combinatorial libraries A2, A4, and A5 are relatively close to each other, whereas library A3 (black) is in a different region of the plot. This relative distribution of the combinatorial libraries is consistent with the structure of the scaffolds. The core scaffolds of A2, A4, and A5 have at least one ring in common (diketopiperazine, dihydroimidazolyl, or cyclic urea). In sharp contrast, the core scaffold of A3 has no rings, which makes this library significantly different to the other three (Figure 1). Furthermore in Figure 3, A4 (magenta) is located between A2 and A5. This is also in agreement with the structure of the core scaffolds because A4 has one ring in common with A2 (diketopiperazine) and one ring in common with A5 (dihydroimidazolyl). A2 and A5 have no common rings in their core scaffolds and are far apart in the plot. Thus, LTM has separated the subsets in Figure 3 and has also displayed relationships between the combinatorial libraries on the basis of structural relations between some of the core scaffolds.

The LTM visualization of data set B in Figure 5 is also in full agreement with the diversities of the libraries in Table 1. B1 (red), B3 (gray), and B4 (green) are data sets with large diversities and are the most dispersed in this LTM plot. In contrast, data sets B2 (cyan) and B5 (orange) have low diversities and exhibit the tightest clustering. This LTM plot also has some overlap between B1, B3, and B4; this overlap is consistent with the partial overlap of drugs, MLSMR, and natural products in chemical space that was reported by Medina-Franco et al.[2,32]

Table 5. Data Sets Used to Evaluate Activity Clustering

data set	no. molecules	activity	activity range
D1	197	inhibition of tuberculosis	0−100%
D2	1093	toxicity against *T. Pyriformis*	−2.67 to 3.34 $pIGC_{50}$
D3	328	blood-brain distribution in rats	−182 to 1.44 log BB
D4	238	absorption in humans	0−100%
D5	275	oral bioavailability in humans	0−100%

In the LTM visualization of data set C given in Figure 7, combinatorial library A5 (blue) is very close to B2 (cyan). This is in full agreement with the chemical structure of their core scaffolds, as both libraries have two 5-membered rings containing two nitrogen atoms connected by a 4-atom linker. Also, subset A2 (yellow) is close to B5 (orange); this is caused by the common diketopiperazine ring in their core scaffolds.

Some data sets such as A3, A5, B2, and B5 are split into a few subclusters in both the PCA and LTM plots. Perhaps the most pronounced examples are A3 (black in Figures 6 and 7) and B2 (cyan in Figures 6 and 7). The main structural feature of these two libraries is the absence of oxygen atoms in their core scaffolds (Figure 1). In these libraries, the molecules that have hydroxyl groups in their side chains (*R*-group substitutions) have a bit set in their MDL 166-key fingerprints that distinguishes them from the molecules in which no hydroxyl-group side-chain is present. This results in A3 and B2 being split. A similar effect can be seen in the other libraries, but it is less pronounced, as these other libraries have oxygen atoms in their core scaffolds. (There are five bits in the MDL 166-key fingerprint that have a large impact on the subclusters: no. 97 (NAAAO), N, O with 3-atom spacer; no. 131 (QH > 1), >1 H donor; no. 139 (OH), hydroxyl; no. 146 (O > 2); no. 157 (C−O).)

CLUSTERING BY ACTIVITY

Data Sets for Clustering by Activity. The data sets used to evaluate the models' ability to cluster by activity are given in Table 5. These data sets contained only drug-like molecules and were taken from a number of different sources. This group of dissimilar data sets was chosen to improve the generality of the results; the model identified as the best for activity clustering is likely to be effective for other real-life data sets of drug-like molecules.

All of the data sets contained molecules represented by SMILES strings and were obtained either from a Web site or direct from a research group. The SMILES strings were converted to MDL 166-key fingerprints. The fingerprints were assembled into the fingerprint array, and this array was then passed to the models. References for each of the data sets are as follows. Data set D1, a set of antituberculosis drug candidates, was compiled from eight papers.[33−40] Data set D2 was downloaded from the Environmental Toxicity Prediction Challenge 2009 Web site[41], an online challenge that invited researchers to predict the toxicity of molecules against *T. Pyriformis*. Data set D3 was used by Abraham et al.[42] in a paper on the in vivo distribution of drugs from blood, plasma, or serum, to rat brain. Data set D4 was used by Zhao et al.[43] in a paper on the descriptor-based prediction of the absorption of drugs; this data set can be downloaded from QSAR World[44] (data set no. 12). Data set D5 was used by Veber et al.[45] in a paper on the oral bioavailability of drug candidates; this data set can be downloaded from QSAR World[44] (data set no. 43).

Table 6. Soergel-Based Interdata Set Distances

	D1	D2	D3	D4	D5
D1	0.41	0.84	0.76	0.70	0.69
D2	0.84	0.79	0.84	0.82	0.82
D3	0.76	0.84	0.77	0.74	0.73
D4	0.70	0.82	0.74	0.65	0.65
D5	0.69	0.82	0.73	0.65	0.65

To measure the structural diversity of the data sets, the interdata set distances were calculated using eq 2. As none of the values in Table 6 are <0.41, all of the data sets can be considered to be structurally diverse from each other. d_{ii} can be taken to measure the structural diversity within data set d_i. As all of the d_{ii} values are ≥0.41, all of the data sets can be considered to contain a structurally diverse set of molecules. Being diverse, it is expected all of the data sets will produce only weak clustering when visualized.

Performance Measurement with the Λ-score. This subsection introduces the Λ-score, a score used to measure the ability of a model to produce similar-activity clustering in a visualization. This score was used to compare the performance of the models on each of the data sets in Table 5. A useful visualization will contain well-grouped clusters of similar-activity molecules; the better the similar-activity clustering, the more useful the visualization to the scientist.

The Λ-score is a variation of the Γ-score (eqs 4−6) and is a mean of λ-cluster scores. The λ-cluster score for a visualization (introduced by Lawrence[46] as a nearest-neighbor classifier) is defined as follows. Assume the data set has been split into two arrays: a training array (A) and a test array (B). Assume the model has been trained using A, a visualization has been produced, and B has been projected (using the trained model) into the visualization space. The λ-cluster score, λ, is given by the following equations

$$\lambda = \frac{1}{k_2 - k_1 + 1} \sum_{k=k_1}^{k_2} L(k) \tag{9}$$

$$L(k) = \sqrt{\frac{1}{n_B} \sum_{i=1}^{n_B} [\psi(b_i, 0) - \Psi(b_i, k)]^2} \tag{10}$$

$$\Psi(b_i, k) = \frac{1}{k} \sum_{j=1}^{k} \psi(b_i, j) \tag{11}$$

Let the ith row of B be denoted by b_i. In eq 11, let the function $\psi(b_i, j)$ return the activity value of the molecule from A that is jth nearest in the visualization space to b_i (so, for example, $\psi(b_i, 2)$ is the activity value of the molecule from A that is second nearest in the visualization space to the molecule in row b_i). So $\Psi(b_i, k)$ is the k-Nearest-Neighbor regression (kNNR) activity prediction for molecule b_i. In eq 10, n_B is the number of molecules in B; and $\psi(b_i, 0)$ is the actual activity value for molecule b_i. So $L(k)$ is the root mean squared error (RMSE) over B of actual versus predicted activity values. Equation 9 defines the λ-cluster score, λ, to be the mean of $L(k)$ for a range of values of k. The lower the

value of λ, the better the clustering of B in the visualization space. In eq 9, k_1 and k_2 are positive integers ($k_2 > k_1$); typically these were set in the range [1,10]. A range of values of k was used in the calculation of λ (rather than a single value, e.g., $k = 3$) as there was found to be some fluctuation in $L(k)$ for $k = 1,...,10$. Taking the mean in eq 9 smoothed out these fluctuations. (The values of k_1 and k_2 used herein were $k_1 = 3$ and $k_2 = 8$. These values were found by experimentation to smooth out the fluctations in $L(k)$ well; they should work well with most data sets.)

Assume that model M has been applied to data set D; let the Λ-score for this pair be Λ_{MD}. Also assume that M has been evaluated using n-fold cross-validation; let λ_i be the λ-cluster score for the visualization plot of fold i. Then $\Lambda_{M,D}$ is taken to be the mean

$$\Lambda_{M,D} = \frac{1}{n}\sum_{i=1}^{n}\lambda_i$$

Optimization of Model Parameters. All of the models, apart from PCA, had some architectural parameters that were optimized to minimize the Λ-score (as the lower the Λ-score, the greater the clustering of similar-activity molecules in the visualization). NeuroScale has a single parameter, the number of Radial Basis Function (RBF) centers. GTM and LTM have the same two architectural parameters: the number of nodes on the side of the square grid that defines the number of RBF centers or the number of latent-space points. R_G is the RBF-centers grid parameter, and L_G is the latent-space grid parameter. LTM-LIN has a single parameter L_G. Nabney[20] gives details of how these parameters affect the architectures of the models.

The parameters were optimized by using exhaustive searches and cross-validation with the Λ-score as the evaluation measure. Each parameter was given a range over which it was varied. In NeuroScale, the search range for the single parameter that was optimized was [3,15]. In GTM, LTM, and LTM-LIN, the lower limit (L) and upper limit (U) of the search range for all the optimized parameter(s) were set as

$$L = ceil\left(\sqrt{\frac{N}{10}}\right) U = floor\left(\sqrt{\frac{N}{1.5}}\right)$$

where N was number of molecules in the data set. These expressions for L and U were found through experimentation to give good search ranges for both parameters. For each model M and each data set D, a Λ-score, $\Lambda_{M,D}$, was calculated for all values of the optimized parameter(s), and the optimal parameter values were taken to be those that gave the lowest score. The optimal values are given in Table 7.

Λ-score Results. The Λ-scores are shown in Table 8. These were produced by using the following experimental setup: 5-fold cross-validation; the optimal parameters of Table 7; and $k_1 = 3$ and $k_2 = 8$ in eq 9 (a range found to average out variations in $L(k)$). For ease of interpretation, the scores have been scaled to $\Lambda_{PCA} = 100.00$. Clearly, all four non-PCA models having smaller Λ-scores performed better than PCA. LTM with a mean Λ-score of 81.69 is the model most able to cluster by activity.

PCA and LTM Visualizations Compared for Clustering by Activity. PCA and LTM visualizations of data sets D1−D5 are given in Figures 8−17. All of these visualizations were produced by entire-data set training. The LTM models were trained using the optimized parameters given in Table 7. The visualizations were plotted using the following method. The data set to be plotted was first sorted by activity value into ascending order. Each molecule was then assigned a color according to the activity

Table 7. Optimal Parameter Values Found by the Exhaustive Searches

model	parameter	D1	D2	D3	D4	D5
NeuroScale	no. RBF centers	12	3	5	6	3
GTM	RBF-centers grid	5	20	6	7	11
GTM	latent-space grid	8	21	12	10	8
LTM	RBF-centers grid	9	18	9	8	8
LTM	latent-space grid	8	23	7	7	11
LTM-LIN	latent-space grid	9	14	8	9	8

Table 8. Λ-Scores (Scaled) for Each Model[a]

data set	PCA	NeuroScale	GTM	LTM	LTM-LIN
D1	100.00	95.33	85.76	**84.32**	95.25
D2	100.00	92.11	81.69	**75.33**	88.41
D3	100.00	84.07	76.49	**74.93**	83.20
D4	100.00	92.09	85.40	**81.31**	100.14
D5	100.00	**89.14**	90.38	92.56	95.02
mean	100.00	90.55	83.94	**81.69**	92.41

[a] Best result in each row is in bold.

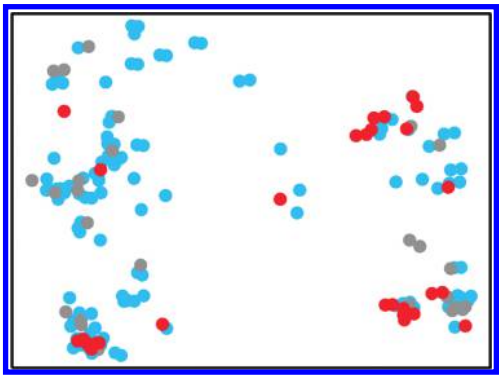


Figure 8. PCA visualization of data set D1.

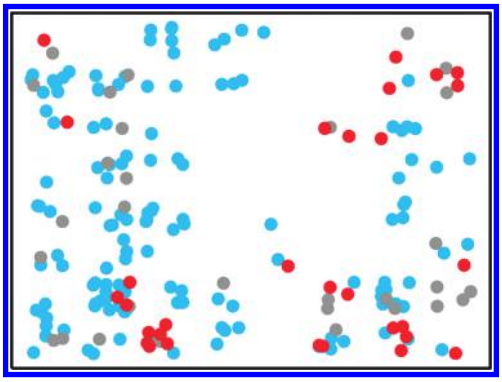


Figure 9. LTM visualization of data set D1.

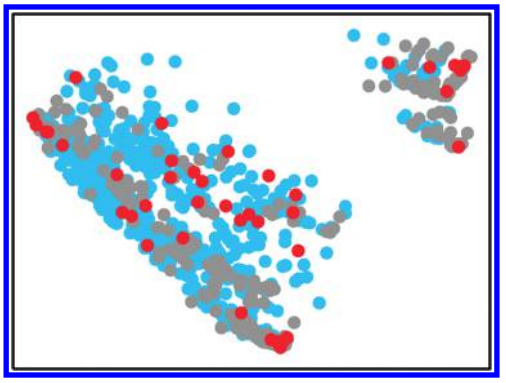


Figure 10. PCA visualization of data set D2.

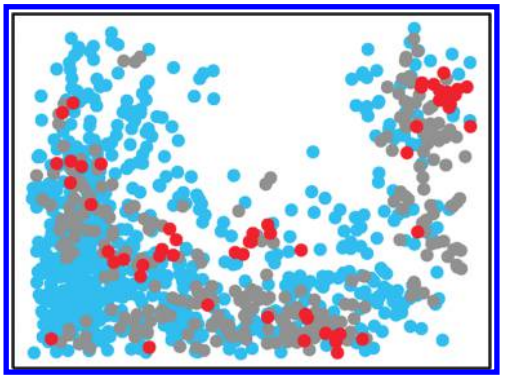


Figure 11. LTM visualization of data set D2.

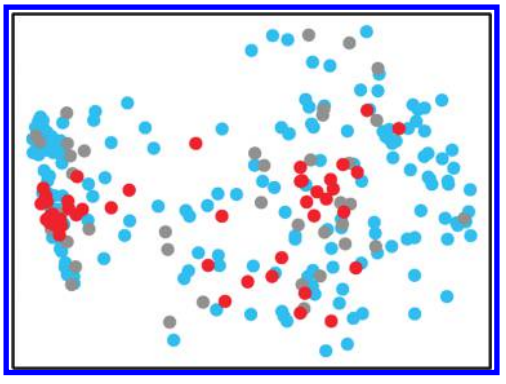


Figure 12. PCA visualization of data set D3.

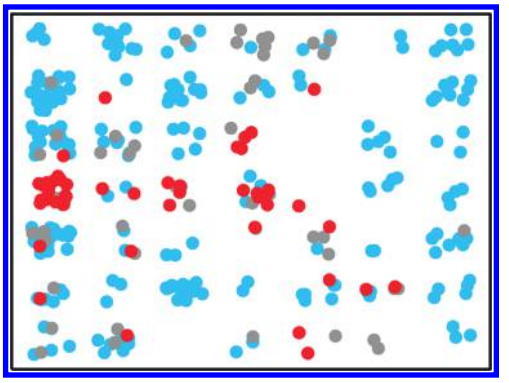


Figure 13. LTM visualization of data set D3.

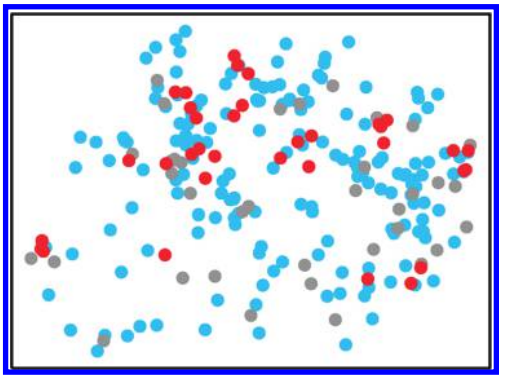


Figure 14. PCA visualization of data set D4.

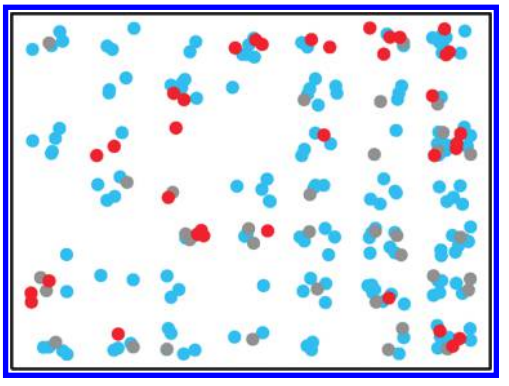


Figure 15. LTM visualization of data set D4.

bins given in Table 9. So, for example, the first 70% of molecules in the sorted data set were colored blue. The molecules were then plotted in their sorted order, so the higher-activity molecules were plotted last resulting in them appearing on top of the lower-activity molecules. In Table 9, the activity-bin ranges were set slightly differently for data set D2 as this data set contained a large number of molecules.

The λ-cluster scores for each visualization are given in Table 10 (for ease of interpretation, the scores have been scaled to $\lambda_{PCA} = 100.00$). In every case, LTM has produced better clustering than PCA, with particularly good scores for data sets D2 and D3. The visualizations are now compared subjectively by visual inspection. Comparing Figures 8 and 9, the amount of clustering looks to be about the same in both figures. Comparing Figures 10 and 11, the clustering looks better in the LTM figure. Notice in particular the cluster of high-activity molecules in the upper right of the LTM figure. Comparing Figures 12 and 13, the clustering is better in the LTM figure; notice the tight clustering of some of the high-activity molecules in the LTM figure and the group of medium-activity molecules near the top of this figure. Comparing Figures 14 and 15, the clustering looks very slightly better in the LTM figure; notice the high-activity molecules in the top right of the LTM figure. Comparing Figures 16 and 17, the amount of clustering looks to be about the same in both figures. Overall, out of the five LTM figures, clustering has visibly improved in two figures, very slightly improved in one figure, and stayed about the same in two figures. The marked visual improvements in clustering occurred in Figurse 11 and 13, which from Table 10

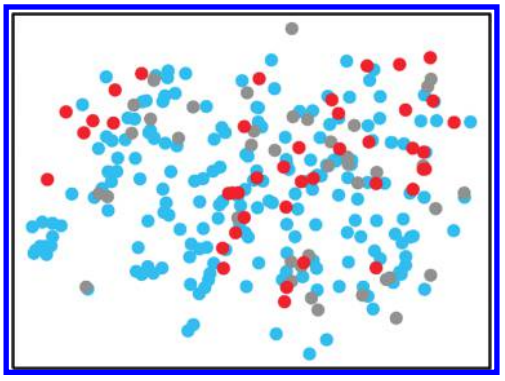


Figure 16. PCA visualization of data set D5.

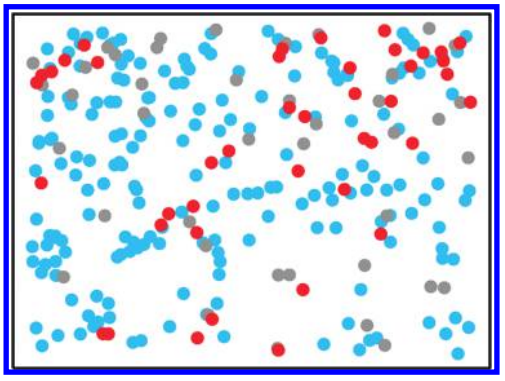


Figure 17. LTM visualization of data set D5.

Table 9. Activity-Bin Colors

color	activity	activity-bin range (%) (for data sets D1, D3, D4, D5)	activity-bin range (%) (for data set D2)
red	high	>85 to 100	>95 to 100
gray	medium	>70 to 85	>70 to 95
blue	low	≥0 to 70	≥0 to 70

Table 10. λ-Cluster Scores (Scaled) for Each Visualization

data set	λ_{PCA}	λ_{LTM}
D1	100.00	89.24
D2	100.00	79.58
D3	100.00	80.51
D4	100.00	82.83
D5	100.00	94.87

had λ-cluster scores of 79.58 and 80.51, respectively. So, a λ-cluster score of approximately <80 is required to achieve a marked visual improvement over PCA.

CONCLUSIONS

Broad Conclusions. The broad conclusions are that (1) LTM will produce significantly better visualizations than all the other models evaluated herein when clustering by molecular structure, (2) LTM will also produce better visualizations than the other models when clustering by activity, but the difference is less marked, and (3) LTM will be a particularly useful tool for the visualization of combinatorial libraries. The results herein demonstrate the value of using both a nonlinear projection map and a Bernoulli noise model for modeling binary data.

A Note on PCA versus LTM. PCA is a well-established technique. It is used every day, around the world, by scientists in many disciplines. Against this background, it is recommended that LTM is used in conjunction with PCA for the visualization of molecular fingerprints; LTM should not be taken as a replacement for the well-established PCA. The advantages of PCA are that (1) PCA is a very stable algorithm and will project almost any data set without fail, and (2) PCA is nonparametric and so is easy to use. The advantages of LTM are that (1) LTM has been specifically designed to model binary data sets, and so is well suited to projecting fingerprints, and (2) LTM disperses its projected points in the visualization space better than PCA, which will enable chemists to see subclusters inside clusters, subclusters that will not be produced by PCA. So, LTM is recommended above PCA for the visualization of fingerprints when clustering by structure. When clustering by activity, LTM is recommended above PCA for data sets with a large number (>1000) of molecules because of its ability to disperse points in the visualization space.

Ideas for Future Research. PCA was used by van Deursen et al.[47] to produce visualizations of several million molecules from the PubChem database. As an area for future research, LTM could be used to produce similar visualizations of PubChem, using the MDL 166-key fingerprint or possibly a higher-resolution fingerprint, such as PubChem's CACTVS. The LTM visualizations could be compared to those in the van Deursen et al. paper in the hope of finding better or different structure clustering. LTM could also be used to produce visualizations of both the PubChem and ChEMBL[48] databases to visually compare how they differ. Another research idea is to run LTM with customized (i.e., user designed) fingerprints. A program could be written that would enable the user to design fingerprints from perhaps a database of molecular features; the design parameters could include fingerprint length and bit positions for the molecular features. If visualizing combinatorial libraries, the core scaffolds of these libraries could be analyzed to select molecular features for the fingerprints.

ASSOCIATED CONTENT

Supporting Information. Matlab scripts for all five models (PCA, NeuroScale, GTM, LTM, LTM-LIN); Matlab source code for the Data Visualization and Modeling System (DVMS); DVMS User Guide (PDF file); all data sets; Table S1. (Two versions of this material are available to download — one for Windows, and one for Linux.) This material is available free of charge via the Internet at http://pubs.acs.org.

AUTHOR INFORMATION

Corresponding Author
*E-mail: i.t.nabney@aston.ac.uk.

ACKNOWLEDGMENT

J.L.M.F. thanks the State of Florida, Executive Office of the Governor's Office of Tourism, Trade, and Economic Development, for supporting his research. J.R.O. thanks the BBSRC for

his CASE studentship (CASE code: BBS/S/N/2006/13090A), and the CASE sponsor, Pfizer Sandwich U.K., for its support. Gaia Paolini, Jo Mulgrew, and Phil Laflin of Pfizer Sandwich are thanked for their help and advice. Dan Rathbone, a chemist at Aston University, is thanked for his advice on molecular structures and for supplying data set D1. Michael H. Abraham of UCL's Department of Chemistry is thanked for supplying data set D3. The reviewers are thanked for their helpful comments on this paper.

REFERENCES

(1) Medina-Franco, J. L.; Martínez-Mayorga, K.; Giulianotti, M. A.; Houghten, R. A.; Pinilla, C. Visualization of the chemical space in drug discovery. *Curr. Comput.-Aided Drug Des.* **2008**, *4*, 322–333.

(2) Singh, N.; Guha, R.; Giulianotti, M. A.; Pinilla, C.; Houghten, R. A.; Medina-Franco, J. L. Chemoinformatic analysis of combinatorial libraries, drugs, natural products, and Molecular Libraries Small Molecule Repository. *J. Chem. Inf. Model.* **2009**, *49*, 1010–1024.

(3) Takahashi, Y.; Konji, M.; Fujishima, S. MolSpace: A computer desktop tool for visualization of massive molecular data. *J. Mol. Graphics Modell.* **2003**, *21*, 333–339.

(4) Bajorath, J. Integration of virtual and high-throughput screening. *Nat. Rev. Drug Discovery* **2002**, *1*, 882–894.

(5) Bayada, D. M.; Hamersma, H.; van Geerestein, V. J. Molecular diversity and representativity in chemical databases. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 1–10.

(6) Yuan, H.; Lu, T.; Ran, T.; Liu, H.; Lu, S.; Tai, W.; Leng, Y.; Zhang, W.; Wang, J.; Chen, Y. Novel strategy for three-dimensional fragment-based lead discovery. *J. Chem. Inf. Model.* **2011**, *51*, 959–974.

(7) Yin, H. Nonlinear multidimensional data projection and visualisation. *Lect. Notes Comput. Sci. Eng.* **2003**, *2690*, 377–388.

(8) Durant, J. L.; Leland, B. A.; Henry, D. R.; Nourse, J. G. Reoptimization of MDL keys for use in drug discovery. *J. Chem. Inf. Comput. Sci.* **2002**, *42*, 1273–1280.

(9) PubChem Fingerprints. ftp://ftp.ncbi.nlm.nih.gov/pubchem/specifications/pubchem_fingerprints.txt (accessed July 2, 2010).

(10) Clark, R. D.; Patterson, D. E.; Soltanshahi, F.; Blake, J. F.; Matthew, J. B. Visualizing substructural fingerprints. *J. Mol. Graphics Modell.* **2000**, *18*, 404–411.

(11) Sammon, J. W., Jr. A nonlinear mapping for data structure analysis. *IEEE Trans. Comput.* **1969**, *C-18*, 401–409.

(12) Martin, Y. C.; Kofron, J. L.; Traphagen, L. M. Do structurally similar molecules have similar biological activity? *J Med. Chem.* **2002**, *45*, 4350–4358.

(13) Jolliffe, I. T. *Principal Component Analysis*, 2nd ed.; Springer: New York, 2002.

(14) Tipping, M. E.; Bishop, C. M. Probabilistic principal component analysis. *J. R. Statist. Soc. B* **1999**, *61*, 611–622.

(15) Tipping, M. E.; Lowe, D. Shadow targets: A novel algorithm for topographic projections by radial basis functions. *Neurocomputing* **1998**, *19*, 211–222.

(16) Leach, A. R.; Gillet, V. J. *An Introduction to Chemoinformatics*, revised ed.; Springer: Dordrecht, The Netherlands, 2007.

(17) Bishop, C. M.; Svensén, M.; Williams, C. K. I. GTM: The Generative Topographic Mapping. *Neural Comput.* **1998**, *10*, 215–234.

(18) Bishop, C. M.; Svensén, M.; Williams, C. K. I. Developments of the Generative Topographic Mapping. *Neurocomputing* **1998**, *21*, 203–224.

(19) Kohonen, T. Self-organized formation of topologically correct feature maps. *Biol. Cybern.* **1982**, *43*, 59–69.

(20) Nabney, I. T. *NETLAB: Algorithms for Pattern Recognition*; Springer: London, 2002.

(21) McLachlan, G. J.; Peel, D. *Finite Mixture Models*; Wiley-Interscience: New York, 2000.

(22) Bishop, C. M. *Neural Networks for Pattern Recognition*; Oxford University Press: Oxford, 1995.

(23) Kabán, A.; Girolami, M. A combined latent class and trait model for the analysis and visualization of discrete data. *IEEE Trans. Pattern Anal. Mach. Intell.* **2001**, *23*, 859–872.

(24) NETLAB. http://www1.aston.ac.uk/eas/research/groups/ncrg/resources/netlab/downloads/ (accessed July 2, 2010).

(25) Maniyar, D. M.; Nabney, I. T. In *Visual Data Mining using Principled Projection Algorithms and Information Visualization Techniques*, Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, August 20−23, 2006; Ungar, L., Craven, M., Gunopulos, D., Eliassi-Rad, T., Eds.; ACM Press: New York, 2006; pp 643−648.

(26) *Molecular Operating Environment (MOE)*, version 2009.10; Chemical Computing Group, Inc.: Montreal, Canada, 2009.

(27) Austin, C. P.; Brady, L. S.; Insel, T. R.; Collins, F. S. NIH Molecular Libraries Initiative. *Science* **2004**, *306*, 1138–1139.

(28) Wang, Y.; Xiao, J.; Suzek, T. O.; Zhang, J.; Wang, J.; Bryant, S. H. PubChem: A public information system for analyzing bioactivities of small molecules. *Nucleic Acids Res.* **2009**, *37*, W623–W633.

(29) Irwin, J. J.; Shoichet, B. K. ZINC − A free database of commercially available compounds for virtual screening. *J. Chem. Inf. Model.* **2005**, *45*, 177–182.

(30) Houghten, R. A.; Pinilla, C.; Giulianotti, M. A.; Appel, J. R.; Dooley, C. T.; Nefzi, A.; Ostresh, J. M.; Yu, Y.; Maggiora, G. M.; Medina-Franco, J. L.; Brunner, D.; Schneider, J. Strategies for the use of mixture-based synthetic combinatorial libraries: Scaffold ranking, direct testing in vivo, and enhanced deconvolution by computational methods. *J. Comb. Chem.* **2008**, *10*, 3–19.

(31) Wishart, D. S.; Knox, C.; Guo, A. C.; Cheng, D.; Shrivastava, S.; Tzur, D.; Gautam, B.; Hassanali, M. DrugBank: A knowledgebase for drugs, drug actions and drug targets. *Nucleic Acids Res.* **2008**, *36*, D901–D906.

(32) López-Vallejo, F.; Nefzi, A.; Bender, A.; Owen, J. R.; Nabney, I. T.; Houghten, R. A.; Medina-Franco, J. L. Increased diversity of libraries from libraries: chemoinformatic analysis of bis-diazacyclic libraries. *Chem. Biol. Drug Des.* **2011**, *77*, 328–342.

(33) Rathbone, D. L.; Parker, K. J.; Coleman, M. D.; Lambert, P. A.; Billington, D. C. Discovery of a potent phenolic N^1-benzylidene-pyridinecarboxamidrazone selective against Gram-positive bacteria. *Bioorg. Med. Chem. Lett.* **2006**, *16*, 879–883.

(34) Coleman, M. D.; Tims, K. J.; Rathbone, D. L. The use of computational QSAR analysis in the toxicological evaluation of a series of 2-pyridylcarboxamidrazone candidate anti-tuberculosis compounds. *Environ. Tox. Pharmacol.* **2003**, *14*, 33–42.

(35) Coleman, M. D.; Rathbone, D. L.; Chima, R.; Lambert, P. A.; Billington, D. C. Preliminary in vitro toxicological evaluation of a series of 2-pyridylcarboxamidrazone candidate anti-tuberculosis compounds III. *Environ. Tox. Pharmacol.* **2001**, *9*, 99–102.

(36) Billington, D. C.; Coleman, M. D.; Ibiabuo, J.; Lambert, P. A.; Rathbone, D. L.; Tims, K. J. Synthesis and antimycobacterial activity of some 2-heteroarylcarboxamidrazones. *Drug Des. Discov.* **1998**, *15*, 269–275.

(37) Mamolo, M. G.; Vio, L.; Banfi, E.; Predominanto, M.; Fabris, C.; Asaro, F. Synthesis and antimycobacterial activity of some 2-pyridinecarboxyamidrazone derivatives. *Farmaco* **1992**, *47*, 1055–1066.

(38) Mamolo, M. G.; Vio, L.; Banfi, E.; Predominanto, M.; Fabris, C.; Asaro, F. Synthesis and antimycobacterial activity of some 4-pyridinecarboxyamidrazone derivatives. *Farmaco* **1993**, *48*, 529–538.

(39) Mamolo, M. G.; Vio, L. Synthesis and antimycobacterial activity of some indole derivatives of pyridine-2-carboxamidrazone and quinoline-2-carboxamidrazone. *Farmaco* **1996**, *51*, 65–70.

(40) Banfi, E.; Mamolo, M. G.; Vio, L.; Predominanto, M. In-vitro antimycobacterial activity of new synthetic amidrazone derivatives. *J. Chemother.* **1993**, *5*, 164–167.

(41) Environmental Toxicity Prediction Challenge 2009. http://www.cadaster.eu/node/65 (accessed July 2, 2010)

(42) Abraham, M. H.; Ibrahim, A.; Zhao, Y.; Acree, W. E., Jr. A data base for partition of volatile organic compounds and drugs from blood/plasma/serum to brain, and an LFER analysis of the data. *J. Pharm. Sci.* **2006**, *95*, 2091–2100.

(43) Zhao, Y. H.; Abraham, M. H.; Le, J.; Hersey, A.; Luscombe, C. N.; Beck, G.; Sherborne, B.; Cooper, I. Rate-limited steps of human oral absorption and QSAR studies. *Pharm. Res.* **2002**, *19*, 1446–1457.

(44) QSAR World. http://www.qsarworld.com/qsar-datasets.php?mm=5 (accessed July 2, 2010).

(45) Veber, D. F.; Johnson, S. R.; Cheng, H.-Y.; Smith, B. R.; Ward, K. W.; Kopple, K. D. Molecular properties that influence the oral bioavailability of drug candidates. *J. Med. Chem.* **2002**, *45*, 2615–2623.

(46) Lawrence, N. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *J. Mach. Learn. Res.* **2005**, *6*, 1783–1816.

(47) Van Deursen, R.; Blum, L. C.; Reymond, J.-L. A searchable map of PubChem. *J. Chem. Inf. Model.* **2010**, *50*, 1924–1934.

(48) Brooksbank, C.; Cameron, G.; Thornton, J. The European Bioinformatics Institute's data resources. *Nucleic Acids Res.* **2010**, *38*, D17–D25.