Optimization in Computational Chemistry and Molecular Biology, pp. ??-??
C. A. Floudas and P. M. Pardalos, Editors

Visualization of Chemical Databases Using the Singular Value Decomposition and Truncated-Newton Minimization

Dexuan Xie and Tamar Schlick
Departments of Chemistry and Mathematics
Courant Institute of Mathematical Sciences
New York University and The Howard Hughes Medical Institute
251 Mercer Street, New York, NY 10012
dexuan@cims.nyu.edu, schlick@nyu.edu

Abstract We describe a rapid algorithm for visualizing large chemical databases in a low-dimensional space (2D or 3D) as a first step in chemical database analyses and drug design applications. The compounds in the database are described as vectors in the high-dimensional space of chemical descriptors. The algorithm is based on the singular value decomposition (SVD) combined with a minimization procedure implemented with the efficient truncated-Newton program package (TNPACK). Numerical experiments show that the algorithm achieves an accuracy in 2D for scaled datasets of around 30 to 46%, reflecting the percentage of pairwise distance segments that lie within 10% of the original distance values. The low percentages can be made close to 100% with projections onto a ten-dimensional space. The 2D and 3D projections, in particular, can be efficiently generated and easily visualized and analyzed with respect to clustering patterns of the compounds.

Keywords: chemical databases, clustering analysis, visualization, SVD, TNPACK, optimization, drug design.

1 Introduction

The field of combinatorial chemistry was recognized by *Science* as one of nine areas of study in 1997 that have great potential to benefit society [30]. The systematic assembly of chemical building blocks to form potential biologically-active compounds and their rapid testing for bioactivity has experienced a rapid growth in both experimental and theoretical approaches [4]. As experimental synthesis techniques are becoming cheaper and faster, huge chemical databases are becoming available for computer-aided design, and the development of reliable computational tools for their study is becoming more important than ever.

The specific computational problems involved in chemical libraries can be associated with certain mathematical disciplines. **Library characterization** involves the tools of *multivariate statistical analysis* and *numerical linear algebra* (see below for specific applications). The **similarity problem** in drug design involves finding from the database a drug that binds to a specific target or a drug that is similar to another drug with known bioactive properties. This search can be performed using 3D structural and energetic searches or using the concept of molecular descriptors introduced below. In either case, *multivariate nonlinear optimization* and optionally *configurational sampling* is involved. The **diversity problem** in drug design involves defining the most diverse subset of compounds within the given library. This problem is a *combinatorial optimization* task, and is known to have a non-polynomial time complexity [8, 24].

Typically, these combinatorial optimization problems are solved by stochastic and heuristic approaches [26]. These include genetic algorithms, simulated annealing, and tabu-search variants. As in other applications, the efficiency of simulated annealing is strongly dependent of the choice of cooling schedule and other parameters. In recent years, several potentially valuable annealing algorithms such as deterministic annealing, multiscale annealing, and adaptive simulated annealing have been extensively studied.

In special cases, combinatorial optimization problems can be formulated as integer programming and mixed-integer programming problems [8, 24, 17]. In this approach, linear programming techniques such as interior methods, can be applied to the solution of combinatorial optimization problems, leading to branch and bound algorithms, cutting plane algorithms, and dynamic programming techniques. Parallel implementation of combinatorial optimization algorithms is also important in practice to improve the performance [26].

One way to analyze a database of n potential *biologically active* compounds (drugs) is to characterize each compound in the database by a list of m *chemical descriptors*. These variables reflect atom connectivity, molecular topology, charge distribution, electrostatic properties, molecular volume, and so on. These descriptors can be generated from several commercial packages such as the popular Molconnx program [1]. Assigning associated *biological activity* for each compound (e.g., with respect to various ailments or targets, which may include categories like headache, diabetes, protease inhibitors) requires synthesis and biological testing. Hence, analyses of chemical databases (such as clustering, similarity, or dissimilarity sampling) can be performed on the collection of m-dimensional real vectors in the space R^m. However, due to large size of the dataset, some database-analysis tasks (say the diversity problem) are extremely challenging in practice because exhaustive procedures are not realistic. Any systematic schemes to reduce this computing time can be valuable.

In this paper we describe an algorithm that produces rapidly two-dimensional (2D) or 3D views of the compounds in a chemical database for clustering analysis. This visualization problem is often formulated as a distance-geometry problem: find n points in 2D (or 3D) so that their interpoint distances match the corresponding values from R^m as closely as possible. This approach was implemented by Sammon with the steepest descent (SD) minimization algorithm for clustering analysis in 1969 [28]. More recently, Sammon's method has been applied to the analysis and 2D projection mapping of molecular databases [3, 27]. The SD algorithm with a randomly chosen starting point generally suffers from slow convergence and may generate a 2D mapping that poorly approximates the original distances. As an alternative to the distance-geometry approach, a neural network procedure by Kohonen – the self-organizing map method [18] – has also been applied to the visualization of chemical databases [6, 11]. This method usually defines a mapping based on a 2D regular grid of nodes such that similar compounds are close to one other on the grid. Compared with a distance-geometry mapping method, however, it is unclear how the distance relationships of chemical database are preserved.

Our visualization algorithm consists of two parts. The first part defines a 2D projection mapping by the *singular value decomposition* (SVD) [15], a technique used for data compression in many practical applications like image processing. This factorization, in contrast to optimization, only requires the input (high-dimensional) data vectors; it has a complexity of order $O(n^2m)$ floating point operation and $O(nm)$ memory locations; no initial projection guess is needed. We find that the accuracy of the SVD mapping depends on the distribution of the singular-value magnitudes: if the first two singular values are much larger than the others, the 2D mapping has a high accuracy. This generalizes to mapping in higher dimensions as well; that is, if the first ten singular values can be largely separated from the rest, a 10D projection can be accurate. For scaled datasets as used in practice, however, two or three dominant singular values cannot generally be found.

The second part refines the SVD projection based on the distance geometry approach when the accuracy of the SVD projection is not satisfactory. Here, the SVD projection is used as a starting point for the truncated-Newton minimization iterative method. Determining a good initial guess for a minimization algorithm is an important and difficult objective in the distance-geometry approach. Our new distance error objective function is minimized with our efficient truncated-Newton program package, TNPACK [29, 32]. We call our algorithm the *SVD/TNPACK* method. This method is also described in [33], along with other applications. The applications in this paper all involve a natural scaling of the datasets rather than range-scaling as used in [33]. The projection analysis done here also illustrates an application to diversity and similarity sampling and presents 3D in addition to 2D projections.

We report numerical tests of the SVD/TNPACK procedure for two chemical datasets: an artificial dataset made of eight groups of compounds with different pharmacological activities (ARTF) and a dataset of monoaminooxidase inhibitors (MAO). ARTF and MAO contain 402 and 1623 compounds, respectively. All compounds in these datasets have been characterized with 312 topological descriptors. In addition, the MAO dataset has also been characterized by 153 binary descriptors (MAO_{01}). Since the various chemical descriptors vary drastically in their magnitudes as well as the variance within the dataset, scaling is important for proper assessment of distance relationship. Given no chemical/physical guidance, we consider a natural scaling procedure for ARTF and MAO, such that all scaled chemical descriptors have a mean of zero and a standard deviation of one.

For these scaled databases, SVD alone produced poor 2D projections (e.g., only about 0.004% of the distance segments are within 10% of the original distances for MAO_{01}), and the TNPACK minimizations that follow SVD become crucial (e.g., TNPACK increased this number 0.004% to 30% in less than one minute on an SGI R10000 processor). We also find that a larger number than three of the projection space is required to reach higher accuracy. Namely, the accuracy can be improved to 96% when the dimension number of the projection space is increased from two to ten for both scaled ARTF and MAO.

Numerical results also show that SVD is very fast: the computational time is one second for ARTF (402 compounds) and six seconds for MAO (1623 compounds) on an SGI R10000 processor; TNPACK is also very efficient (several minutes), and much more efficient than SD.

In section 2, we describe a mathematical framework for analysis of chemical datasets. Sections 3 describes the SVD/TNPACK method. Section 4 presents the numerical results and chemical structure analyses for the scaled datasets. Conclusions are summarized in Section 5.

2 Mathematical framework for analysis of chemical databases

We consider a database $\mathcal{S}$ of n potential *biologically active* compounds (drugs), where each compound is described by a list of m chemical descriptors. Thus, we can express the database $\mathcal{S}$ as a collection of n vectors

$$\mathcal{S} = \{X_1, X_2, \ldots, X_n\},$$

where vector $X_i = (x_{i1}, x_{i2}, \ldots, x_{im})^T$ denotes the i-th compound in $\mathcal{S}$, and the real numbers $\{x_{ik}\}$ are values of the associated chemical descriptors.

The database $\mathcal{S}$ can also written as a rectangular matrix X by listing, in rows, the m descriptors of the n compounds:

$$X = (X_1, X_2, \ldots, X_n)^T = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix}. \tag{1}$$

This rectangular matrix typically has $n \gg m$ for large databases, where n may be of order million.

To measure the similarity or diversity for each pair of compounds X_i and X_j, we define distance quantities δ_{ij} on the m-dimensional vector space $\mathcal{R}^m$. The simplest one is the *Euclidean distance*:

$$\delta_{ij} = \|X_i - X_j\| = \sqrt{\sum_{k=1}^{m} (x_{ik} - x_{jk})^2}, \tag{2}$$

where $\|\cdot\|$ denotes the Euclidean norm. There are $n(n-1)/2$ distance segments $\{\delta_{ij}\}$ in $\mathcal{S}$ for pairs $i < j$.

Scaling may be important for proper assessment of distance quantities because the various chemical descriptors vary drastically in their magnitudes. Generally, scaled descriptors $\{\hat{x}_{ik}\}$ can be defined by the following formula: For $k = 1, 2, \ldots, m$,

$$\hat{x}_{ik} = a_k(x_{ik} - b_k), \qquad 1 \le i \le n, \tag{3}$$

where a_k and b_k are real numbers, and $a_k > 0$. They are called the scaling and displacement factors, respectively.

In practice, however, it is very difficult to determine the appropriate scaling and displacement factors for the specific application problem [34]. Given no chemical/physical guidance, the following two scaling procedures are often used. The first modifies each column of X by setting

$$b_k = \min_{1 \le i \le n} x_{ik} \quad \text{and} \quad a_k = 1/(\max_{1 \le i \le n} x_{ik} - b_k) \quad \text{for } k = 1, 2, \ldots, m. \tag{4}$$

This makes each column in the range $[0, 1]$. The second sets

$$b_k = \frac{1}{n} \sum_{i=1}^{n} x_{ik} \quad \text{and} \quad a_k = 1/\sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_{ik} - b_k)^2} \quad \text{for } k = 1, 2, \ldots, m, \tag{5}$$

so that each scaled column of X has a mean of zero and a standard deviation of one. The scaling procedure with (4) is also referred to as a standardization of descriptors. Both scaling procedures (4) and (5) assume that no one descriptor dominates the overall distance measures.

The distances $\{\delta_{ij}\}$ can be used in performing similarity searches among the database compounds and between these compounds and a particular target. This task can be formulated as finding:

$$\min_{\substack{1 \le i \le n \\ i \ne j}} \delta_{ij},$$

where $\delta_{ij} = \|X_i - X_j\|$, and X_j is a given target. Note that each distance segment δ_{ij} requires $\mathrm{O}(m)$ floating-point operations (flops) to compute, an exhaustive calculation over all n candidates requires a total of $\mathrm{O}(nm)$ flops. An effective scheme is sought when n and m are large.

More difficult and computationally-demanding is the diversity problem. Namely, we seek to reduce the database of the n compounds by selecting a "representative subset" of the compounds contained in $\mathcal{S}$, that is one that is "the most diverse" in terms of potential chemical activity. This problem naturally arises since pharmaceutical companies must scan huge databases each time they search for a specific pharmacological activity. This molecular diversity problem can be formulated as determining:

$$\max_{\mathcal{S}_0 \subset \mathcal{S}} \sum_{\substack{X_i, X_j \in \mathcal{S}_0 \\ X_i \ne X_j}} \|X_i - X_j\|,$$

where each $\mathcal{S}_0$ contains n_0 representative compounds ($n_0 \ll n$, a fixed integer number). This is a *combinatorial optimization* problem, an example of a very difficult computational task (NP-complete). An exhaustive search of the most diverse subset $\mathcal{S}_0$ requires a total of $\mathrm{O}(C_n^{n_0} n_0^2 m)$ flops because there are $C_n^{n_0}$ possible subsets of $\mathcal{S}$ of size n_0 and each subset takes $\mathrm{O}(n_0^2 m)$ flops. Here $C_n^{n_0} = n(n-1)(n-2)\cdots(n-n_0+1)/n_0$.

As a first step in solving such similarity and diversity problems, methods that produce a *low*-dimensional projection view of the compounds can be used for clustering analysis.

Assume we have a mapping from $\mathcal{R}^m$ to $\mathcal{R}^{low}$ that takes each point $X_i \in \mathcal{R}^m$ to $Y_i \in \mathcal{R}^{low}$, where $low \ll m$. Typically the integer low is 2 or 3 but we use $low > 3$ in some cases discussed below; the projection cannot be easily visualized for $low > 3$, but the compressed matrix from X ($n \times low$ instead of $n \times m$) can be useful in reducing computer time for database applications. The intercompound distances for the vectors Y_i and Y_j is denoted as $d(Y_i, Y_j)$. We define it as

$$d(Y_i, Y_j) = \sqrt{\sum_{k=1}^{low} (y_{ik} - y_{j_k})^2}.$$

An ideal projection mapping will generate points $\{Y_i\}$ such that their distance values match the original values, i.e., satisfy

$$d(Y_i, Y_j) = \delta_{ij} \tag{6}$$

for all $\{i, j\}$ pairs. However, no such a mapping exists in general because the problem is typically over-determined — finding $n \times low$ unknowns $\{y_{ik}\}$ satisfying $n(n-1)/2$ equations of form (6). An optimal approximate mapping is thus sought based on the distance geometry approach [25]. Specifically, an objective error function E to describe the discrepancy between $\{\delta_{ij}\}$ and $\{d(Y_i, Y_j)\}$ is constructed, and then we find a minimum point $Y^* = (Y_1^*, Y_2^*, \ldots, Y_n^*)$ with $Y_i^* \in R^{low}$ for $i = 1, 2, \ldots, n$ such that

$$E(Y_1^*, Y_2^*, \ldots, Y_n^*) = \min_{Y_i \in R^{low},\ 1 \leq i \leq n} E(Y_1, Y_2, \ldots, Y_n), \tag{7}$$

where each $Y_i = (y_{i1}, y_{i2}, \ldots, y_{ilow})^T$. The objective function E can be formulated in many different ways [3, 25, 27]. Here we use the following expression:

$$\begin{aligned} E(Y_1, Y_2, \ldots, Y_n) &= \frac{1}{4} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \omega_{ij} \left(d(Y_i, Y_j)^2 - \delta_{ij}^2 \right)^2, \\ \omega_{ij} &= \begin{cases} 1/\delta_{ij}^4 & \text{if } \delta_{ij}^4 \geq \eta, \\ 1 & \text{if } \delta_{ij}^4 < \eta, \end{cases} \end{aligned} \tag{8}$$

where $\{\omega_{ij}\}$ denote weights, and the parameter η is a small positive number such as 10^{-12}. The first and second derivatives of E are well defined, and an efficient second-derivative method like Newton-type algorithms [12] can be applied.

Various error measures can be used to assess the agreement between the original and projected pairwise distances. Besides the value of the objective function E, we use the following percentage ρ to measure the quality of the approximation of $d(Y_i, Y_j)$ to δ_{ij} for all pairs $i < j$:

$$\rho = \frac{T_d}{n(n-1)/2} \cdot 100. \tag{9}$$

The variable T_d is the total number of the distance segments $d(Y_i, Y_j)$ satisfying

$$|d(Y_i, Y_j) - \delta_{ij}| \leq \epsilon \delta_{ij} \quad \text{when } \delta_{ij} > d_{min}, \tag{10}$$

or

$$d(Y_i, Y_j) \leq \tilde{\epsilon} \quad \text{when } \delta_{ij} \leq d_{min}, \tag{11}$$

where $\epsilon, \tilde{\epsilon}$, and d_{min} are given small positive numbers less than one. For example, we set $\epsilon = 0.1$ to specify a 10% accuracy ($d_{min} = 10^{-12}$ and $\tilde{\epsilon} = 10^{-8}$). The second case above (very small original distance) may occur when two compounds in the datasets are similar highly. The greater the ρ values, the better the mapping and the more information can be inferred from the projected views of the complex data.

3 The SVD/TNPACK method

The SVD decomposition of the database rectangular matrix X (defined in (1)) as $U\Sigma V^T$ can be written as the sum of rank-1 matrices [15]:

$$X = \sum_{k=1}^{r} \sigma_k u_k v_k^T, \tag{12}$$

where r is the rank of matrix X ($r \leq m$), $u_k \in R^n$ and $v_k \in \mathcal{R}^m$, respectively, are left and right singular vectors, and σ_k is the singular value. All singular values are arranged in decreasing order:

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0 \quad \text{and} \quad \sigma_{r+1} = \ldots = \sigma_m = 0.$$

Let $u_k = (u_{1k}, u_{2k}, \ldots, u_{nk})^T$. Using (12), we can express each vector X_i as a linear combination of orthonormal basis vectors $\{v_k\}_{k=1}^m$ of $\mathcal{R}^m$:

$$X_i = \sum_{k=1}^{m} \sigma_k u_{ik} v_k = \sum_{k=1}^{r} \sigma_k u_{ik} v_k, \quad i = 1, 2, \ldots, n$$

since $\sigma_{r+1} = \ldots = \sigma_m = 0$. Hence, the compound vector X_i is expressed in terms of new coordinates

$$X_i = (\sigma_1 u_{i1}, \sigma_2 u_{i2}, \ldots, \sigma_r u_{ir}, 0, \ldots, 0)^T. \tag{13}$$

Based on (13), we define the *low* dimensional vector Y_i as the natural projection of X_i onto the subspace $\mathcal{R}^{low}$ of R^m:

$$Y_i = (\sigma_1 u_{i1}, \sigma_2 u_{i2}, \ldots, \sigma_{low} u_{ilow})^T, \quad i = 1, 2, \ldots, n. \tag{14}$$

When the percentage ρ defined in (9) is not large enough, we improve the accuracy of the SVD projection (14) by our truncated Newton program package, TNPACK [29, 32] by minimizing the objective error function E defined in (8).

The truncated Newton method [9] consists of both outer and inner loops. The outer loop defines a sequence of solution vectors $\{Y^k\}$ expressed in the form

$$Y^{k+1} = Y^k + \lambda_k P^k, \quad k = 0, 1, 2, \ldots,$$

where Y^k and P^k are vectors of $R^{low \cdot n}$, P^k is a descent direction, λ_k is the steplength, and Y^0 is an initial guess. The inner loop defines P^k by a "truncated" preconditioned conjugate gradient scheme. The steplength λ_k is generated by using a line search scheme ([21], for example).

TNPACK was first published in 1992 [29] and updated recently [32]. One of the features of TNPACK is an application-tailored preconditioner matrix (that approximates the Hessian of the objective function) used to accelerate convergence [31]. This novel preconditioner makes TNPACK an efficient tool for the minimization of molecular potential functions in comparison to other available minimizers [10, 31]. For the present applications, we used the new version of TNPACK [32] in combined with a simple preconditioner, namely, the diagonal part of the Hessian, or terms $\partial^2 E(Y_1, Y_2, \ldots, Y_n)/\partial y_{ik}^2$ (for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, low$).

We use the SVD projection (14) as the starting point Y^0, and terminate TNPACK iteration at Y^k provided that it satisfies

$$||g(Y^k)|| < \epsilon_g (1 + |E(Y^k)|), \tag{15}$$

where ϵ_g is a small positive number (we used 10^{-5}), and g is the gradient vector of E. Such an Y^k defines the SVD/TNPACK projection.

4 Numerical examples

Two datasets were used for testing our SVD/TNPACK scheme: MAO ($n = 1623$ and $m = 312$) and ARTF ($n = 402$ and $m = 312$). ARTF merges eight different groups of molecules with different types of pharmacological activities. Descriptors for ARTF and MAO were generated from the software package Molconn-X [1]. We scaled descriptors using scaling procedure (5), and deleted all zero columns of dataset matrix X. We thus obtained dense rectangular matrices with $m = 202$ for scaled ARTF and $m = 204$ for scaled MAO. We also considered MAO with binary descriptors, MAO_{01} ($m = 153$). The binary descriptors were generated from the software MACCS II [20].

We used the NAG library [2] to compute the SVD of each dataset. For simplicity, we used all default parameters of TNPACK [29, 32] for the minimization that follows the SVD projection. The target accuracy ϵ in (10) was set to 0.1. The termination rule for TNPACK is (15) with $\epsilon_g = 10^{-5}$. All computations were performed in double precision on a single R10000/195 MHZ processor of an SGI Power Challenge L computer at New York University.

Table 1 displays the performance of SVD and SVD/TNPACK in defining 2D mappings for these datasets. The accuracy of 2D mapping is indicated by the percentage ρ defined in (9) (i.e., the portion of the distance segments that are within 10% of the original distance values). From Table 1 we see that both SVD and TNPACK are efficient: computer CPU time ranges from one second to seven minutes. SVD alone yields poor accuracies in terms of distance preservation (ρ ranges from 0.004 to 25%). TNPACK greatly improves the SVD projection in this regard (ρ ranges from 30 to 46%). Table 1 Figure 1

To illustrate the reason why the 2D SVD mapping is poor for the scaled datasets, Figure 1 presents the distributions of the normalized singular values $\hat{\sigma}_i$ on seven intervals: $(10^{-k}, 10^{-(k-1)}]$ for $k = 1$ to 6 and $[0, 10^{-6}]$. Here the normalized singular values are defined by

$$\hat{\sigma}_i = \sigma_i / \max_{1 \le j \le r} \sigma_j \quad \text{for } i = 1, 2, \ldots, r.$$

From Figure 1 we see that most normalized singular values are not small for the scaled datasets, implying that the first two singular values are not significantly larger than the others. Hence, the 2D mapping is poor for the scaled datasets. Figure 2

Figure 2 shows that the accuracy (i.e., the percentage ρ defined in (9)) of the SVD and SVD/TNPACK projections for the scaled datasets can be improved sharply when the number of dimensions (low) of the projection space is increased from two to ten. We also found it useful to use higher-order SVD mappings for the purpose of selecting initial points for minimization refinement.

Table 2 compares the performance of TNPACK with that of the steepest descent (SD) method since SD has been used in similar applications [3, 27]. Here both TNPACK and SD used the same termination rule (15) and the same SVD starting point. Table 2 shows that TNPACK is more efficient (a factor of three) to find a minimum point. This efficiency will likely become more significant as the database size n increases.

Table 2

Table 3 compares the performance of TNPACK using the SVD projection as a starting point with that using a randomly selected starting point. It shows that the SVD starting point helps accelerate the minimization process significantly, and generate better 2D mappings (smaller values of E). Again, the improvements are likely to be more more significant as n increases.

Table 3

Figure 3 displays the 2D mappings of the scaled ARTF, the scaled MAO, and the binary MAO_{01}. These figures also compare the plots of the 2D mappings generated by SVD alone and SVD/TNPACK (blue vs. red symbols). The SVD plots have been significantly changed by TNPACK so as to improve the distance values in 2D with respect to the original values.

Figure 3

Figure 4

Figure 4a displays the distribution of eight chemical/pharmacological classes of compounds in ARTF as a result of the 2D SVD/TNPACK mapping. The number of compounds in each class is indicated in the figure next the class name. One selected chemical structure for each class is marked by a black circle and shown in Figure 5.

Figure 5

Noting that the 2D mapping has several small subclusters and a few singletons, we selected six spatially distant points (marked as A1 to A6) from different pharmacological classes on Figure 4b. See Figure 6 for their chemical structures. This is an application of the projection to the diversity sampling problem. Note that even within one family the chemical structures may differ. As an application to the similarity problem, we also selected three spatially close points (B1 to B3) from the same H1 ligand class on Figure 4b. Their similar chemical structures are presented in Figure 6.

Figure 6

Figure 7

Finally, we generated the 3D SVD/TNPACK mapping for the scaled ARTF. As expected, the accuracy of the 3D mapping is higher than the 2D mapping ($\rho = 63.46\%$ for 3D while $\rho = 46$ for 2D with $\eta = 0.1$). Four different views of the 3D mapping are displayed in Figure 7; a single point corresponding to A1 in Figure 4b was removed for better resolution. From these figures we see that the 3D mapping is quite similar to the 2D mapping: the ecdysteroids (red spheres in 3D and red triangles in 2D) and the AChE inhibitors (green spheres in 3D and green squares in 2D) classes continue to appear separate from the rest and a strong overlap between D1 agonists, D1 antagonists, H1 ligands, and 5HT ligands persists.

5 Conclusions

We have presented a mathematical framework for analysis of chemical databases. Our SVD/TNPACK method is easy to implement and efficient to use in visualizing large chemical databases in a low-dimensional space (2D or 3D).

The scaled databases make it difficult to calculate 2D/3D projections that approximate well the original distance distributions. This is because all scaled descriptors lie within the same range and there are in general no dominant singular values. However, we showed that higher-accuracy projections can be obtained for these scaled datasets when the projection dimension is increased from two to ten or so. Though these higher-dimensional projections are not easily visualized, the compression of the dataset descriptors can be advantageous in further applications of the compound library (e.g., diversity sampling) as shown here.

When the intercompound distances in 2D/3D approximate the original distance relationships well, the 2D/3D projection offers a simple visualization tool for analyzing the compounds in a large database. We emphasize that these analyses depend on the quality of the original descriptors, an area of research on its own [7]. These clustering analyses can serve as a first step in the study of related combinatorial chemistry questions dealing with large chemical databases, and we hope to examine these possibilities in future work. It will also be important to compare our SVD/TNPACK method to the neural network procedure of Kohonen, both in terms of resulting projection accuracy of clustering and computing performance. Figure 8 shows a mapping of 32 5D-vectors by our SVD/TNPACK vs. Kohonen map, where we used the same data set and Kohonen map figure as given in [18], page 114. For comparison, a reference tree, the so called *minimal spanning tree* (where the most similar pairs of points are linked) [18], is also displayed. The SVD/TNPACK and Kohonen maps have similar clusters with different patterns. However, the SVD/TNPACK map appears more similar to the reference tree. Figure 8

Further work is also needed on extending our SVD/TNPACK approach to large chemical datasets. The huge database might be subdivided as dictated by computer memory, and the SVD/TNPACK procedure applied to each data subset. To properly assemble these sub-2D-mappings for the purpose of defining a global 2D-mapping, techniques to overlap the database segments will have to be devised. We intend to discuss this extension scheme in detail in our subsequent work. We invite interested readers to contact us about experimenting with our projection software SIEVER (SIngular Values and Error Refinement).

Acknowledgments

We are indebted to Dr. Alexander Tropsha at University of North Carolina for providing the original chemical datasets and for helpful discussions. Support by the National Science Foundation (ASC-9157582 and BIR 94-23827EQ) and the National Institutes of Health (R01 GM55164-01A2) is gratefully acknowledged. T. Schlick is an investigator of the Howard Hughes Medical Institute.

References

[1] Hall Associates Consulting (1995), *Molconn-X version 2.0*, Quincy, Maryland.

[2] NAG Inc. (1995), *NAG Fortran Library, Mark 17*, Opus Place, Suite 200, Downers Grove, Illinois.

[3] Agrafiotis, D.K. (1997), "A new method for analyzing protein sequence relationships based on Sammon maps," *Protein Science, Vol. 6*, 287-293.

[4] Boyd, D.B. (1995), "Rational drug design: Controlling the size of the haystack," *Modern Drug Discovery, Vol. 1, No. 2*, 41-47.

[5] Bakonyi, M. and Johnson, C.R. (1995), "The Euclidean distance matrix completion problem," *SIAM J. Matrix Anal. Appl., Vol. 16*, 646-654.

[6] Bienfait, B. (1994), "Applications of high-resolution self-organizing maps to retrosynthetic and QSAR analysis," *J. Chem. Inf. Comput. Sci. Vol. 34*, 890-898.

[7] Brown, R.D. and Martin, Y.C. (1997), "Information content of 2D and 3D structural descriptors relevant to ligand-receptor binding," *J. Chem. Inf. Comput. Sci., Vol. 37*, 1-9.

[8] Cook, W.J., Cunningham, W.H., Pulleyblank, W.R. and Schrijver, A. (1998), *Combinatorial Optimization,* Wiley, New York.

[9] Dembo, R.S. and Steihaug, T. (1983), "Truncated-Newton algorithms for large-scale unconstrained optimization," *Math. Programming, Vol. 26*, 190-212.

[10] Derreumaux, P., Zhang, G., Brooks, B. and Schlick, T. (1994), "A truncated-Newton method adapted for CHARMM and biomolecular applications," *J. Comp. Chem., Vol. 15*, 532-552.

[11] Gasteiger, J.; Zupan, J. (1993), "Neural Networks in Chemistry," *Angew. Chem. Int. Ed. Engl., Vol. 32,* 503-527.

[12] Gill, P.E., Murray, W. and Wright, M.H. (1983), *Practical Optimization*, Academic Press, London.

[13] Crippen, G.M. and Havel, T.F. (1988), *Distance Geometry and Molecular Conformation*, Wiley, New York.

[14] Glunt, W., Hayden, T.L., Hong, S. and Wells, J. (1990), "An alternating projection algorithm for computing the nearest Euclidean distance matrix", *SIAM J. Matrix Anal. Appl., Vol. 11*, 589-600.

[15] Golub, G.H. and Van Loan, C.F. (1996), *Matrix Computations*, John Hopkins University Press, Baltimore, Maryland, third edition.

[16] Gower, J.C. (1985), "Properties of Euclidean and non-Euclidean distance matrices," *Linear Algebra Appl., Vol. 67*, 81-97.

[17] Nemhauser, G.L. and Wolsey, L.A. (1988), *Integer and Combinatorial Optimization,* John Wiley and Sons, New York.

[18] Kohonen, T. (1997), *Self-Organizing Maps, Springer Series in Information Sciences, Vol. 30,* Springer, Berlin, Heidelberg, New York.

[19] Korte, B., Lovász, L. and Schrader, R. (1991), *Greedoids,* Springer-Verlag, New York.

[20] Molecular Design Ltd. *Maccs-II,* 14600 Catalina St., San Leandro, California.

[21] Moré, J.J. and Thuente, D.J. (1994), "Line search algorithms with guaranteed sufficient decrease," *ACM Trans. Math. Softw., Vol. 20*, 286-307.

[22] Moré, J.J. and Wu, Z. (1997), "Distance geometry optimization for protein structures," *Technical Report MCS-P628-1296*, Argonne National Laboratory, Argonne, Illinois.

[23] Oxley, J.G. (1992), *Matroid Theory,* Oxford University Press, New York.

[24] Papadimtriou, C.H. and Steiglitz, K. (1982), *Combinatorial Optimization: Algorithms and Complexity,* Prentice-Hall, Englewood Cliffs, New Jersey.

[25] Pinou, P., Schlick, T., Li, B. and Dowling, H.G. (1996), "Addition of Darwin's third dimension to phyletic trees," *J. Theor. Biol., Vol. 182*, 505-512.

[26] Reeves, C. (1993), *Modern Heuristic Techniques for Combinatorial Problems,* Halsted Press, New York.

[27] Robinson, D.D., Barlow, T.W. and Richard, W.G. (1997), Reduced dimensional representations of molecular structure. *J. Chem. Inf. Comput. Sci., Vol. 37*, 939-942.

[28] Sammon Jr, J. W. (1969), "A nonlinear mapping for data structure analysis," *IEEE Trans. Comp. C-18*, 401-409.

[29] Schlick, T. and Fogelson, A. (1992), "TNPACK — A truncated Newton minimization package for large-scale problems: I. Algorithm and usage," *ACM Trans. Math. Softw., Vol. 14*, 46–70.

[30] Science and Business (1998), "New partnerships for biology and business," *Science, Vol. 282, No. 18*, 2160-2161.

[31] Xie, D. and Schlick, T. (1999), "Efficient implementation of the truncated-Newton algorithm for large-scale chemistry applications," *SIAM J. Optim., Vol. 9.*

[32] Xie, D. and Schlick, T. (1999), "Remark on Algorithm 702 — the updated truncated Newton minimization package," *ACM Trans. Math. Softw., Vol. 25, No. 1.*

[33] Xie, D., Tropsha, A. and Schlick, T. (1999), "An efficient projection protocol for chemical databases: the singular value decomposition combined with truncated Newton minimization," preprint, submitted.

[34] Willett, P. (1998), "Structural similarity measures for database searching." In von Ragué Schleyer, P. (Editor-in Chief), Allinger, N.L., Clark, T., Gasteiger, J., Kollman, P.A. and Schaefer, III, H.F., editors, *Encyclopedia of Computational Chemistry, Vol. 4*, John Wiley & Sons, West Sussex, UK, 2748-2756.

List of Tables

List of Figures

Table 1: Performance of the 2D SVD and SVD/TNPACK (TN) mappings. Here E is the minimization objective function defined in (8), and ρ the percentage defined in (9), which measures the quality of the approximation of the 2D mapping

Datasets	E		ρ		TN	CPU time	
	SVD	TN	SVD	TN	Itn.	SVD (sec.)	TN (min.)
ARTF	7.06×10^3	2.77×10^3	25.91	45.95	31	1.18	0.45
MAO	1.31×10^5	5.41×10^4	5.51	43.94	33	6.24	7.49
MAO_{01}	2.4×10^5	9.79×10^4	0.004	29.10	11	3.65	0.77

Table 2: Comparison of TNPACK versus SD for minimizing E

Method	Final E	Final $\|g\|$	Iterations	CPU time (min.)
ARTF				
SD	2.77×10^3	2.77×10^{-3}	1375	1.17
TNPACK	2.77×10^3	1.05×10^{-4}	31	0.45
MAO				
SD	5.42×10^4	5.42×10^{-1}	1768	26.35
TNPACK	5.41×10^4	2.21×10^{-1}	33	7.49

Table 3: Comparison of TNPACK minimization using the SVD mapping as a starting point (SVD) versus a randomly selected starting point (RAN)

Starting point X^0	Final E	Final $\|g\|$	Iterations	CPU time (min.)
ARTF				
SVD	2.77×10^3	1.05×10^{-4}	31	0.45
RAN	2.87×10^3	1.44×10^{-2}	61	0.91
MAO				
SVD	5.41×10^4	2.21×10^{-1}	33	7.49
RAN	5.55×10^4	3.14×10^{-1}	133	25.83

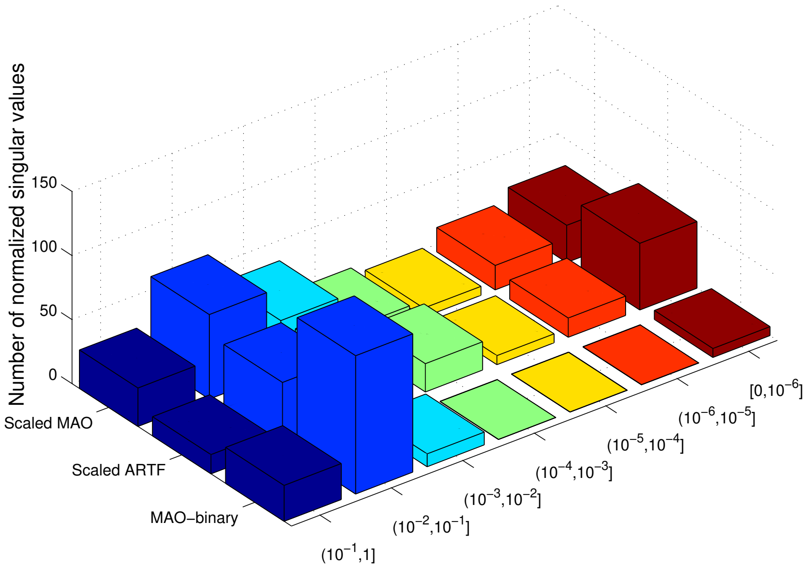


Figure 1: The distribution of the normalized singular values $\{\hat{\sigma}_i\}$

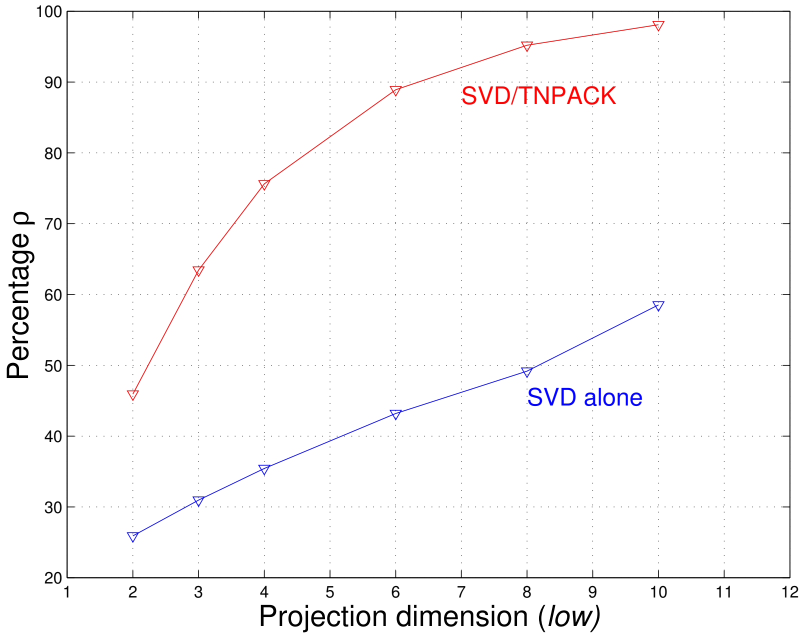


Figure 2: The percentage ρ defined in (9) increases with the number of dimensions of the projection space ($\eta = 0.1$ for ARTF)

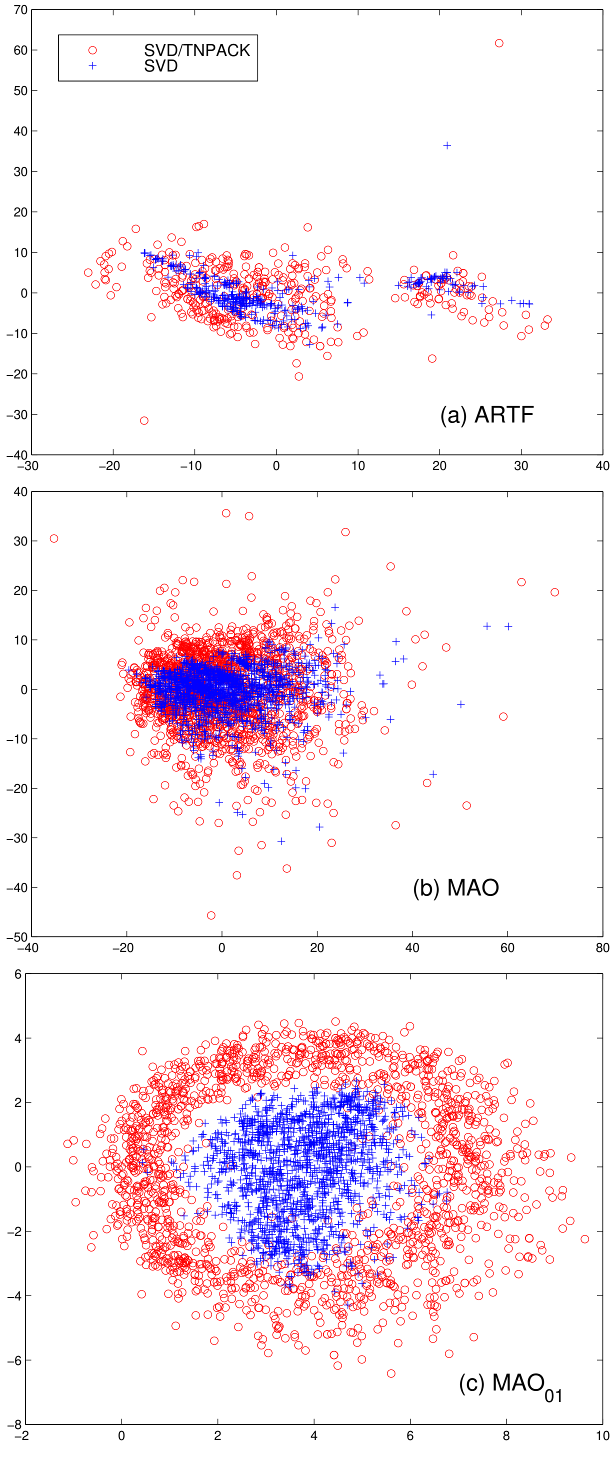


Figure 3: The 2D SVD and SVD/TNPACK mappings for ARTF, MAO, and MAO_{01}

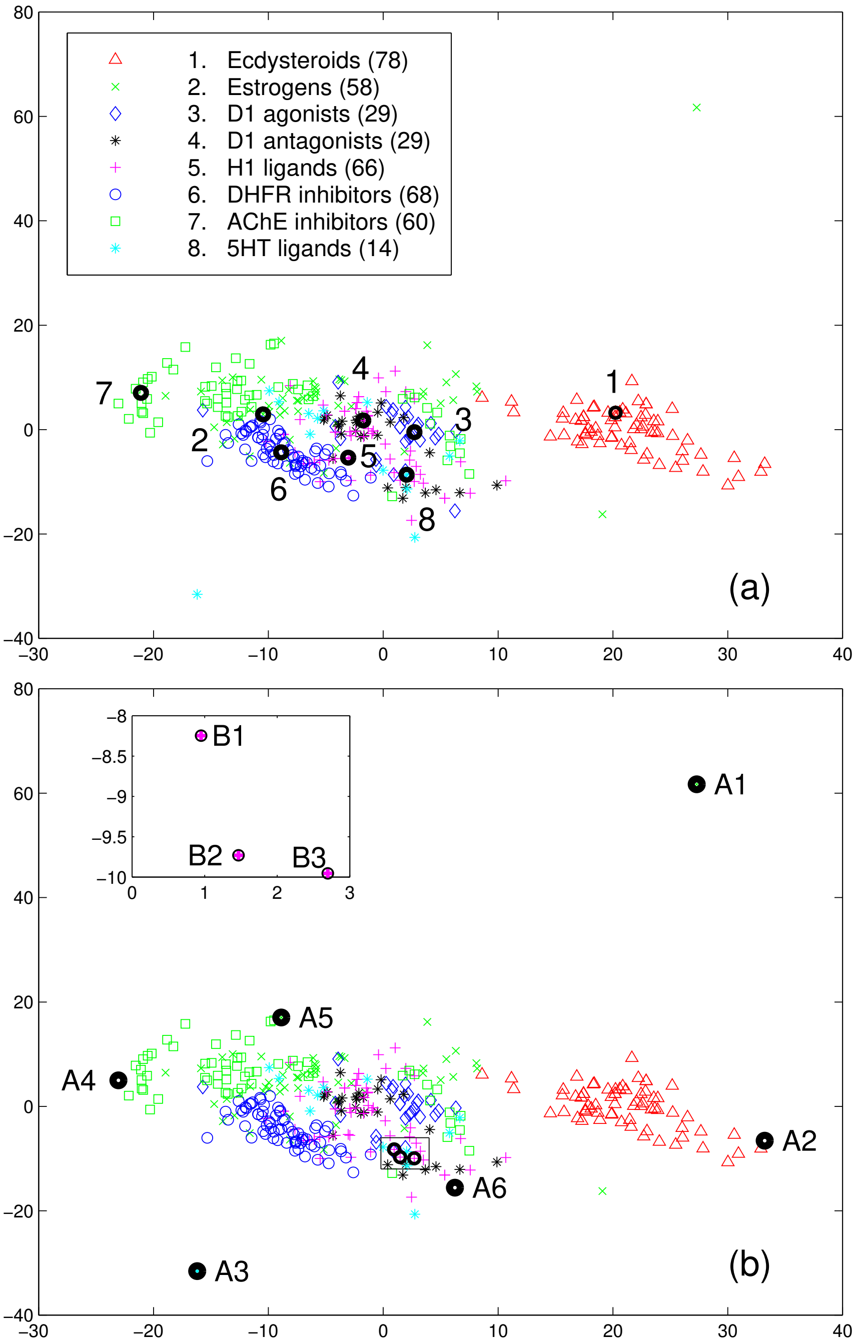


Figure 4: The 2D SVD/TNPACK mapping of the eight pharmacological classes of ARTF: (a) with eight chemical representatives marked by black circles, and (b) with a diversity sample (A1–A2) and a similarity sample (B1–B3). See Figures 5 and 6 for their chemical structures

Figure 5: Chemical structure representatives for the eight classes of ARTF (see Figure 4a)

Figure 6: Chemical structures for the diversity and similarity applications of the 2D SVD/TNPACK projection for ARTF (see Figure 4b)

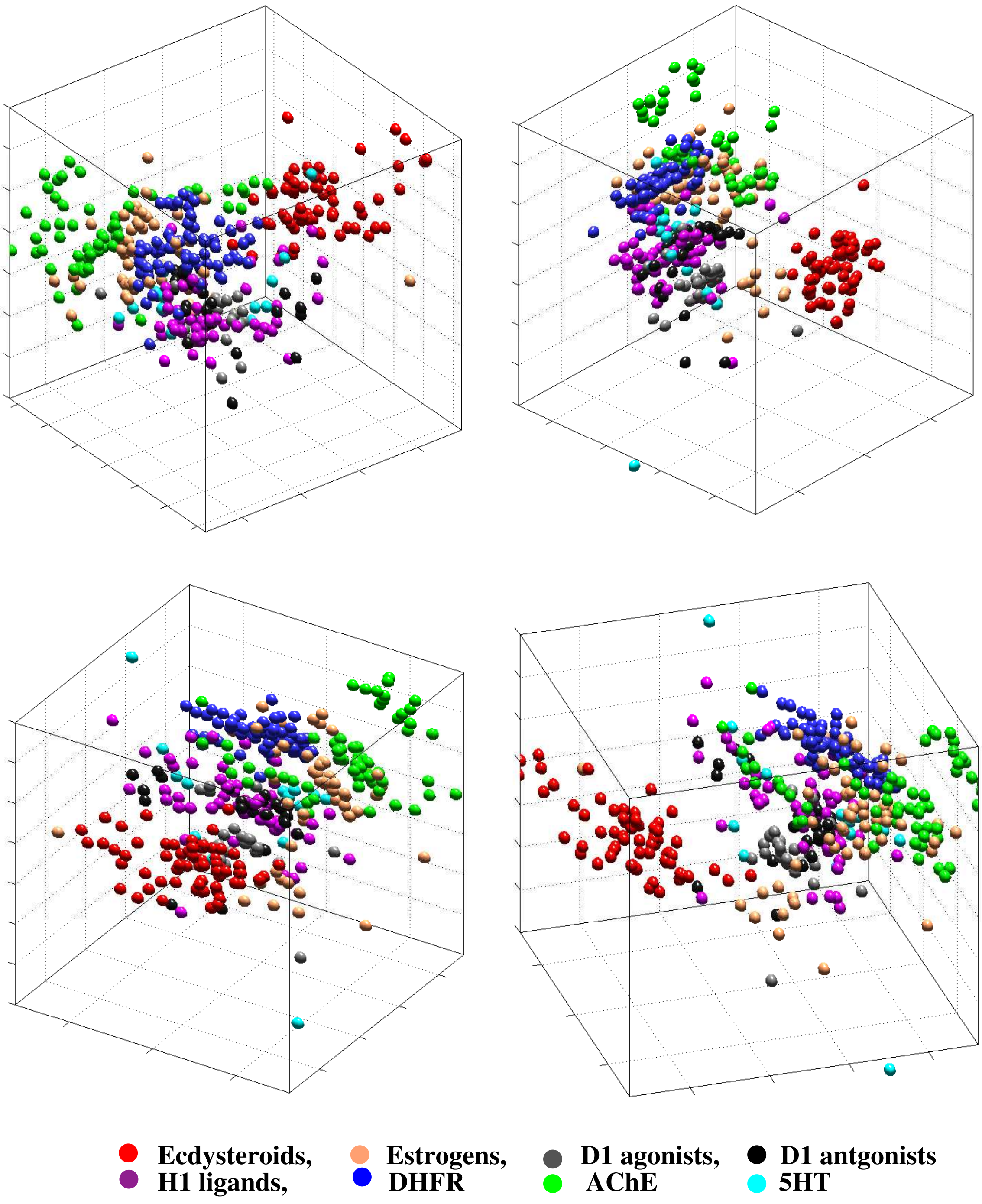
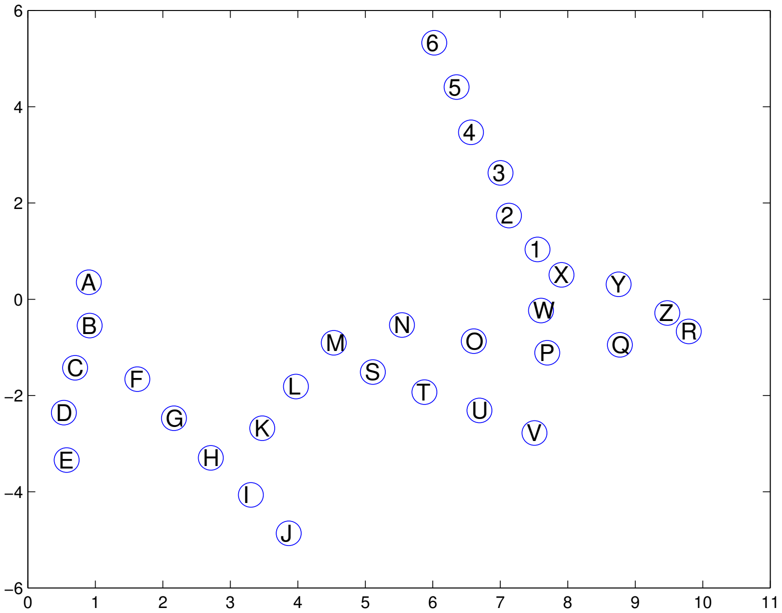


Figure 7: Four different views of the 3D SVD/TNPACK mapping for ARTF

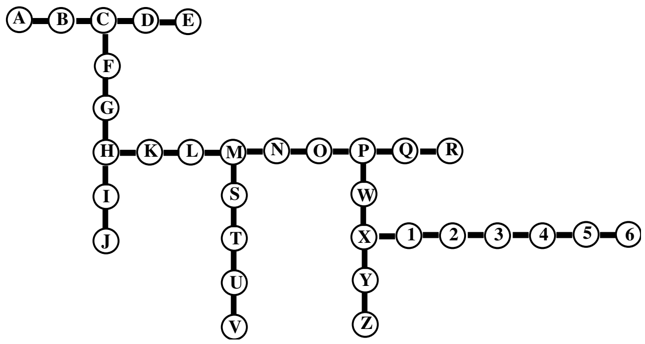
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	1	2	3	4	5	6
1	2	3	4	5	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
0	0	0	0	0	1	2	3	4	5	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
0	0	0	0	0	0	0	0	0	0	1	2	3	4	5	6	7	8	3	3	3	3	6	6	6	6	6	6	6	6	6	6
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	4	1	2	3	4	2	2	2	2	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	4	5	6

The dataset (32 5D−vectors) used by Kohonen



SVD/TNPACK map of the above dataset (ρ=70% **with** η=0.1)

```
  B C D E * Q R * Y Z
A * * * * P * * X *
  * F * N O * W * * 1
* G * M * * * * 2 *
  H K L * T U * 3 * *
* I * * * * * * 4 *
  * J * S * * V * 5 6
```



Kohonen map **Minimal spanning tree**

Figure 8: Comparison of SVD/TNPACK versus the Kohonen methods