Counting Clusters Using *R*-NN Curves

Rajarshi Guha,*,† Debojyoti Dutta,‡ David J. Wild,† and Ting Chen‡

School of Informatics, Indiana University, Bloomington, Indiana 47406, and Department of Computational Biology, University of Southern California, Los Angeles, California 90089

Received November 28, 2006

Clustering is a common task in the field of cheminformatics. A key parameter that needs to be set for nonhierarchical clustering methods, such as *k*-means, is the number of clusters, *k*. Traditionally, the value of *k* is obtained by performing the clustering with different values of *k* and selecting that value that leads to the optimal clustering. In this study, we describe an approach to selecting *k*, a priori, based on the *R*-NN curve algorithm described by Guha et al. (*J. Chem. Inf. Model.*, **2006**, *46*, 1713−722), which uses a nearest-neighbor technique to characterize the spatial location of compounds in arbitrary descriptor spaces. The algorithm generates a set of curves for the data set which are then analyzed to estimate the natural number of clusters. We then performed *k*-means clustering with the predicted value of *k* as well as with similar values to check that the correct number of clusters was obtained. In addition, we compared the predicted value to the number indicated by the average silhouette width as a cluster quality measure. We tested the algorithm on simulated data as well as on two chemical data sets. Our results indicate that the *R*-NN curve algorithm is able to determine the natural number of clusters and is in general agreement the average silhouette width in identifying the optimal number of clusters

1. INTRODUCTION

One of the common tasks in cheminformatics is the clustering of chemical data sets.[1] The fundamental goal of a clustering is to divide a set of molecules into groups such that the molecules within a group are more similar to each other than to molecules outside the group. A variety of clustering methods is available[2] and can be divided into two groups: hierarchical (such as Ward's algorithm[3]) and partitional (such as the *k*-means algorithm[4]). The former type of clustering either divides a data set into successively smaller clusters or builds up clusters starting from individual compounds. In the case of partitional clustering, there is no such hierarchical relationship between clusters, which are simply disjoint groups of compounds.

Clustering has been used in a wide variety of application areas ranging from compound acquisition,[5] conformational analysis,[6−8] visualization,[9] docking,[10−13] and database searching.[13−15] In applications of an exploratory nature, one is usually interested in observing how the data are clustered, and specifying the number of clusters is not necessarily important. On the other hand, certain situations may warrant the specification of a certain number of clusters to be generated.

The concept of the *number of clusters* is fundamentally different between hierarchical and partitional clustering algorithms. In the case of a hierarchical partitioning, one can generate varying numbers of clusters depending on what level the tree is cut. Thus, one does not necessarily need to specify the number of clusters beforehand. On the other hand, partitional clustering algorithms require that the number of clusters, *k*, be specified before the clustering can be performed. This presents us with a problem: how does one decide on the number of clusters before performing the clustering? The simplest approach is to perform the clustering and then obtain a measure of the quality of the clustering. The optimal number of clusters is determined by this measure. It is clear that this is a trial-and-error process. For small data sets, this is not a significant problem. However, for larger data sets, repeated clustering can be time-consuming.

An alternative approach to this problem is to visualize the data such that one can manually identify the number of clusters. This can be problematic due to both the size of the data set as well as the possibly high-dimensional nature of the data set. One alternative is to use a multidimensional scaling algorithm[16−18] to view the data set in two or three dimensions. One could also use a principal components analysis, though in this case it is possible that the structure of the data set is not obvious by simply viewing the first two or three principal components.

Clearly, it is useful to be able to estimate the number of clusters in a data set of arbitrary dimensions a priori. In this paper, we present an approach to identifying the number of clusters based on a nearest neighbor approach. We focus on its application to partitional clustering (more specifically, the *k*-means algorithm) and do not consider its application to hierarchical clustering algorithms. We test the algorithm on manual data as well as two different chemical data sets. The estimated numbers of clusters in all cases is confirmed by visual inspection of the data set (or the scaled data when the dimensionality is greater than three).

* Corresponding author. e-mail: rguha@indiana.edu.
† Indiana University.
‡ University of Southern California.

10.1021/ci600541f CCC: $37.00
Published on Web 06/30/2007

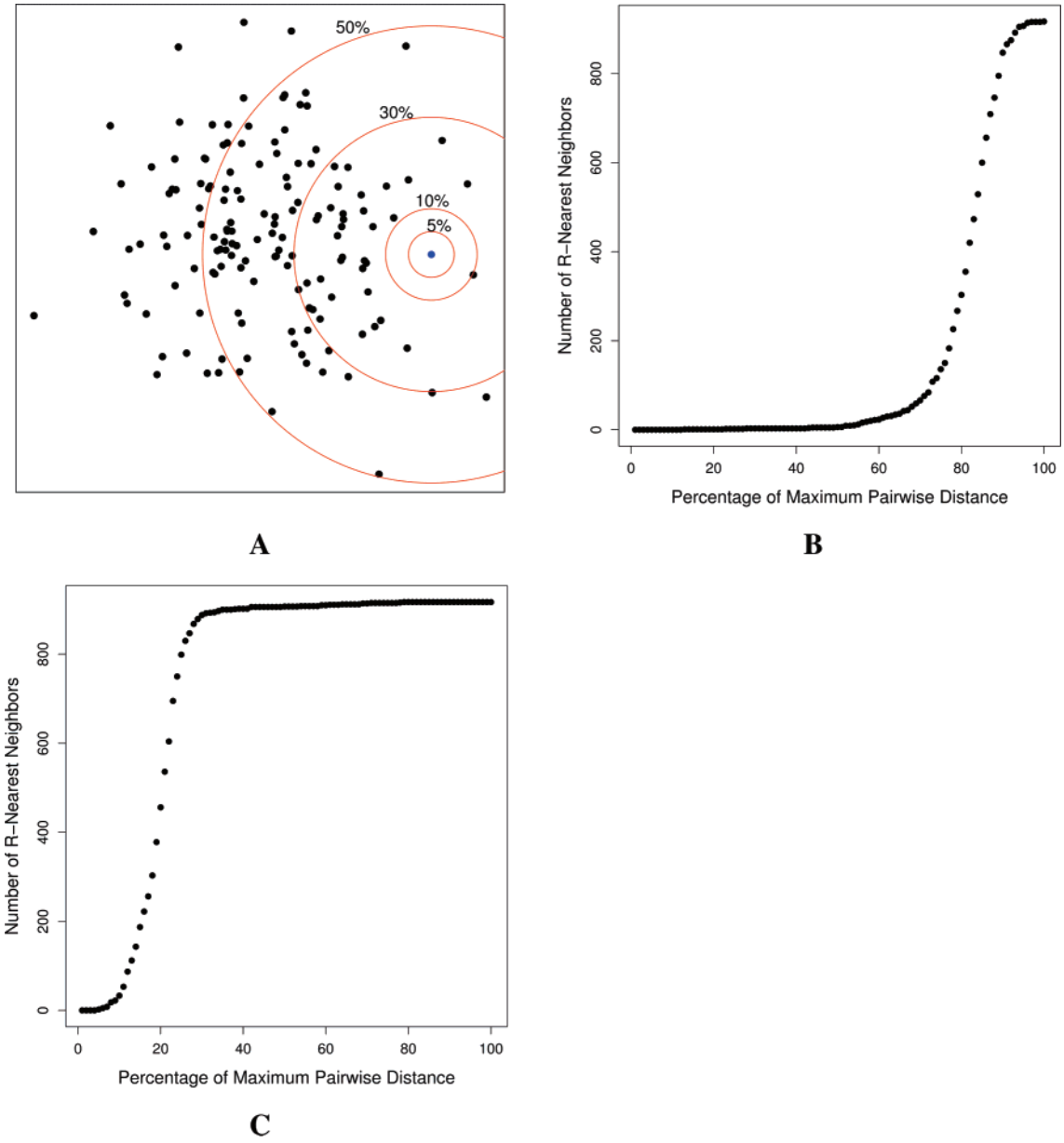

Figure 1. A schematic description of the calculation of R-NN curves (A). The percentages indicate the radius as a percentage of the maximum pairwise distance in the data set. Plots B and C are examples of the R-NN curve for a molecule in a sparse and dense region of the chemical space, respectively.

2. METHODOLOGY

The approach to predicting the number of natural clusters in a data set is based on the *R*-NN curve algorithm described by Guha et al.[19] Before describing the algorithm to determine the number of clusters, we provide a brief overview of the *R*-NN curve algorithm.

The algorithm is based on the observation that, when the radius around a query point is increased, the number of neighbors that lie within the radius will also increase. This is schematically shown in Figure 1A, where the query point is colored blue. In general, the values of the radius are taken as percentages of the maximum pairwise distance (which is calculated exactly or obtained by sampling in the case of large data sets). It follows that, when the radius is equal to the maximum pairwise distance in the data set, the whole data set will be considered neighbors of the query point. When the nearest-neighbor count is plotted versus the radius, a sigmoidal plot is generated. The characteristic feature of this plot is that the length of the lower tail characterizes the query point's location in the space being considered. Thus, for a point in a dense region of the descriptor space, there will be an appreciable number of points even for small radii. On the other hand, for a query point located in a sparse region of the descriptor space, there will be no or very few neighbors for small to intermediate radii. Only when the hypersphere reaches the bulk of the data set will the nearest neighbor count start increasing. The result of this behavior is that a sigmoidal curve with a short lower tail indicates that the query point is located in a dense region of the descriptor space, and a long lower tail indicates that it is located in a sparse region of the descriptor space. Examples of these curves for points located in a sparse and dense region of a descriptor space are shown in Figure 1B and C, respectively.

Now, when the data are clustered, it is observed that the sigmoidal curve is characterized by *steps*. This can be understood by the fact that, when a point is located in, say,

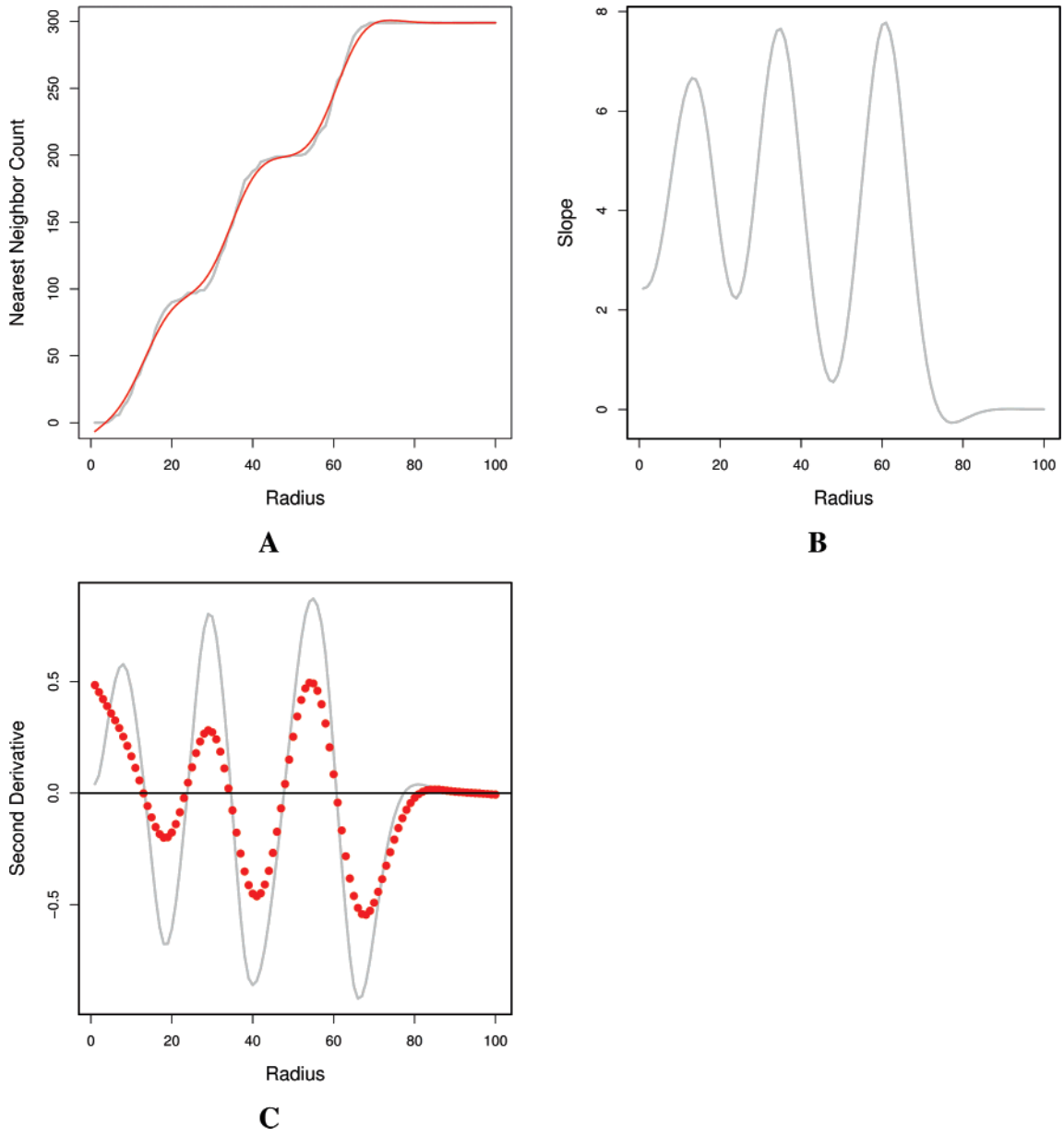

Figure 2. Plots of an R-NN curve and its subsequent transformations to obtain the number of steps in the original curve.

the bulk of one cluster, and as we increase the radius, the number of nearest neighbors increases. At one point, the radius will encompass the entire cluster, and subsequent increases will not add any new neighbors. Thus, the nearest neighbor count will be constant. However, at a certain value of the radius, it will encounter the another cluster. From this point onward, the nearest neighbor count will again increase with an increase in the radius. Clearly, for two clusters, the number of steps in the curve will be one; for three clusters, there will be two steps, and so on. Thus, by identifying the number of steps in the sigmoidal curves, one should be able to estimate the number of clusters present in the data sets, for a given descriptor space.

2.1. Counting Clusters. A number of approaches were considered to count the number of steps in a sigmoidal curve. One possible approach involves matching the generated curve against a set of canonical curves with a known number of steps. The curve-matching problem has been addressed, and a number of metrics such as the Frechét distance[20] and the Hausdorff distance[21] have been investigated. However, this does not always work well for a variety of curves and can be computationally intensive.

A simple approach is to consider the fact that the slope of the sigmoidal curve will exhibit maxima (corresponding to the linear portions of the curve) and minima (corresponding to the plateaus or steps in the curve). Thus, by obtaining the first derivative of the *R*-NN curve, we could then apply a peak picking routine to the result, the number of peaks being equal to the number of clusters. Our initial attempt resulted in a curve with a large number of peaks. This was partly due to the discontinuous nature of the original *R*-NN curves, since it was evaluated at 100 values of the radius. We then considered a smoothed version of the *R*-NN curve, and that is shown in Figure 2A. Though the resultant curve is much smoother, the first derivative still contained some smaller, but broad, maxima. Visually, it would be easy to ignore such peaks, but our automated peak picking routine would consider them in addition to the real (sharp) peaks. Thus, our next step was to smooth the first derivative (Figure 2B) and take its slope. That is, we end up with the second

Chart 1

Algorithm 1 The R-NN curve cluster counting algorithm

```
N_root,max ← −1
for molecule in dataset do
  C ← Evaluate R-NN curve
  C_S ← smooth (C)
  C''_S ← smooth (d²C_S/dR²)
  N_root ← Number of roots of C''_S
  if N_root > N_root,max then
    N_root,max ← N_root
  end if
end for
if N_root,max mod 2 = 0 then
  N_cluster ← N_root,max/2
else
  N_cluster ← (N_root,max + 1)/2
end if
```

derivative of the original *R*-NN curve (Figure 2C). Given the second derivative, we then fit a spline and then evaluate the number of roots of the curve, N_{root}, which can be used to evaluate the number of clusters as

$$N_{\text{cluster}} = \begin{cases} \dfrac{N_{\text{root}}}{2} & \text{if } N_{\text{root}} \text{ is even} \\ \dfrac{N_{\text{root}}+1}{2} & \text{if } N_{\text{root}} \text{ is odd} \end{cases} \quad (1)$$

The above procedure only considers an *R*-NN curve for a single molecule. It is apparent that not all the *R*-NN curves for a data set will exhibit the steps characteristic of a clustering. An example would be the *R*-NN curve for a point located between two clusters. Thus, to reliably identify the number of clusters, we must consider multiple *R*-NN curves. Our current implementation evaluates N_{root} for all the *R*-NN curves in the data set and then uses the maximum value of N_{root} to evaluate N_{cluster} in eq 1. The procedure is summarized in Algorithm 1 (see Chart 1).

2.2. Measuring Cluster Quality. After performing the clustering, we must then determine the quality of the clustering. One approach is to visualize the clustering. However, this is only possible when the clustering is performed in a 2D or 3D space. A more general approach is to use a measure that indicates the quality of the clustering. Many such measures are available such as the Goodman−Kruskal index,[22] Huberts Γ statistic,[23] the silhouette width,[2] the Dunn index,[24] and the Davies−Bouldin index.[25] Many of the traditional cluster quality measures are conceptually similar in that they try to ascertain whether an object is better placed in a specific cluster as opposed to some other cluster. In general, this question is answered by looking at some sort of distance (or similarity) between the object in question and the members of each cluster being considered. It should be noted that this study does not attempt to compare the utility of different measures of cluster quality. Rather, we desire to use a given measure to provide an external confirmation of the number of clusters predicted by the *R*-NN curve algorithm. Given this observation, we considered two of the many available cluster quality measures. More specifically, we investigated the use of the silhouette width and the Dunn index. Since we observed very similar results for both measures, we only report and discuss the results obtained using the silhouette width.

The silhouette width is a method that characterizes a clustering by providing a measure of the confidence of cluster assignments and has been used in a wide variety of studies.[26−29] The silhouette width is defined for each member, *i*, of a cluster, *j*, as

$$s(i) = \frac{b(i) - a(i)}{\max[a(i),b(i)]} \quad (2)$$

where $a(i)$ is the average distance between *i* and all the other members of the cluster *j* and $b(i)$ is the minimum of the average distance between *i* and the members of the other clusters. The above definition implies that $-1 \le s(i) \le +1$. A value close to +1.0 indicates that object *i* has been placed in the correct cluster, such that the average distance of the object *i* to other members of the cluster is small compared to the average distance to members of the closest cluster. A value of −1.0 indicates that the object *i* has been placed in the wrong cluster, so that the average distance to the members of the cluster is larger than the average distance to the members of the closest cluster. Finally, a value of 0 would indicate that the cluster membership of the object is unclear—the average distance to members of the two nearby clusters are essentially the same. The larger the value of the silhouette width for an object, the surer we can be that it has been placed in the correct cluster. If many objects have silhouette widths close to 0, one might infer that the data simply cannot be clustered distinctly.

Given the silhouette width of a single cluster member, we can then define the silhouette width for the entire clustering, termed the average silhouette width (ASW), as

$$\text{ASW} = \frac{1}{k}\sum_{j=1}^{k}\left[\frac{1}{N_j}\sum_{i=1}^{N_j} s(i)\right] \quad (3)$$

where N_j is the number of objects in cluster *j* and *k* is the number of clusters. The ASW is a dimensionless measure that characterizes the extent of cluster structure found in the data set. A general rule of thumb suggests that values of the ASW between 0.25 and 0.50 are indicative of cluster structure, though additional analysis may be required and values greater than 0.5 are indicative of reasonable to strong cluster structures.[30]

3. DATA SETS

We considered three data sets for this study. The first data set was in fact a collection of simulated 2D data sets that were generated using a Thomas cluster process.[31] We considered a number of such data sets with the number of clusters ranging from two to four. The distribution of points is plotted in Figure 3. These data sets were considered since the visualization of the clusters was obvious and thus would allow us to easily verify whether the *R*-NN curve algorithm was indeed identifying the number of clusters correctly. Similarly, these data sets also allowed us to judge the quality of the *k*-means clusterings when performed with a variety of *k* values.

Though the use of simulated data is useful for verification purposes, we are naturally more interested in the algorithm's ability to count clusters that may occur in chemical data sets. To this end, the second data set was created by combining two sets of structures. We considered a set of 277 dihydrofolate reductase (DHFR) inhibitors taken from the 756

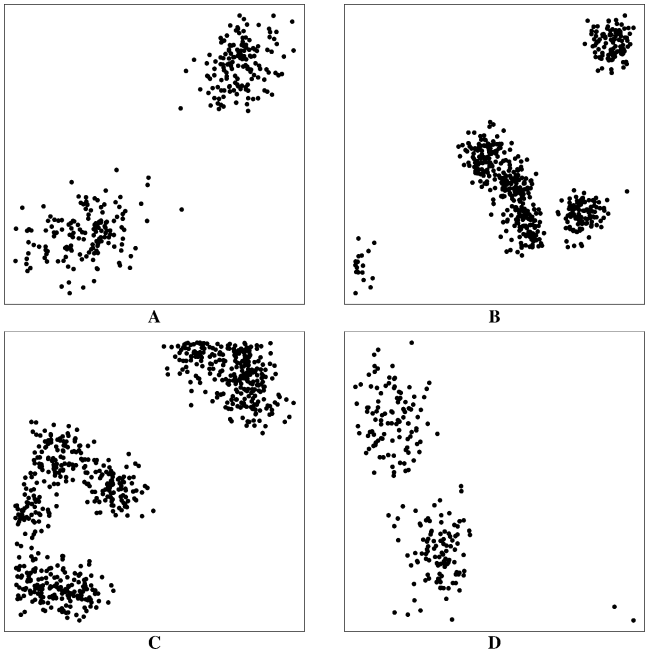

Figure 3. Distribution of simulated 2D data generated using a Thomas point process.[31]

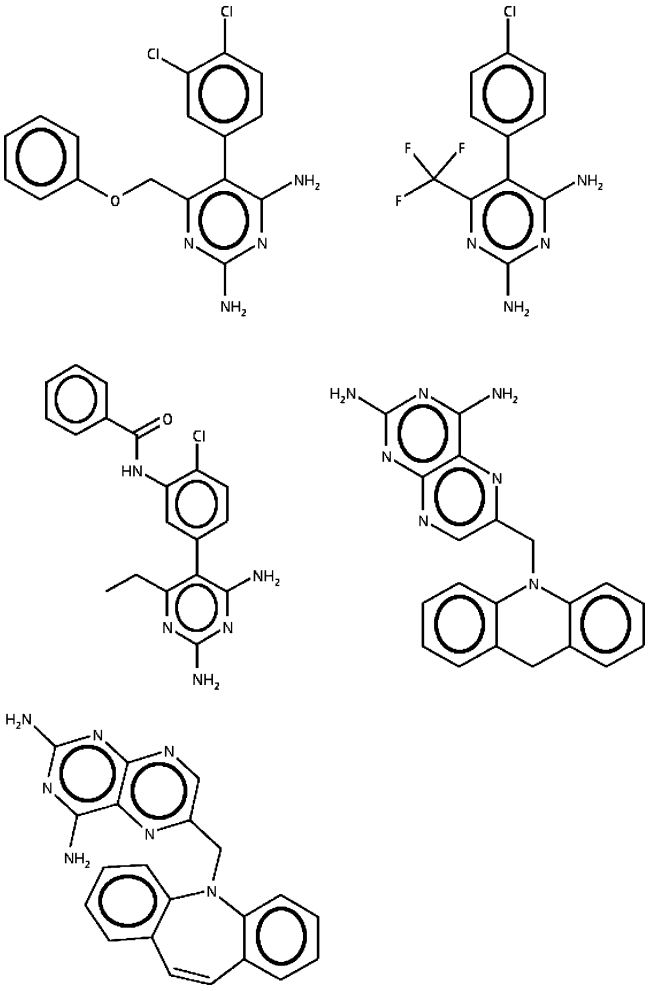

Figure 4. Some representative DHFR inhibitors.

inhibitors studied by Sutherland et al.[32] and a set of 277 compounds from the Design Institute for Physical Property Data (DIPPR) Project 801 database that had been previously modeled by Goll and Jurs.[33] The reason for choosing these

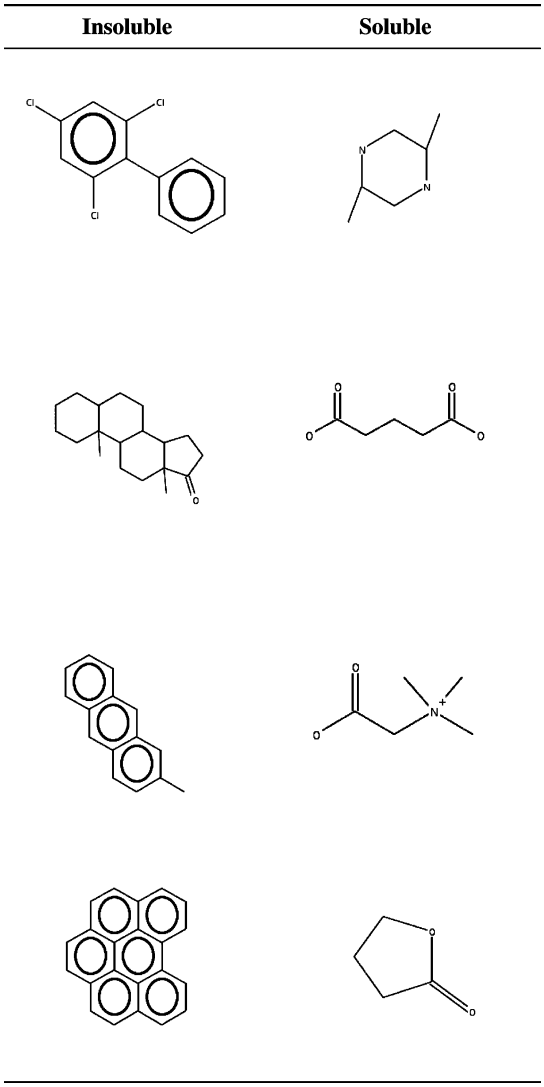

Figure 5. Some representative structures from the aqueous solubility data set.

data sets was that there was in general significant structural differences between the two groups. The DHFR inhibitors were based around a 5-(4-chlorophenyl)-6-ethyl-2,4-pyrimidinediamine or 2,4-diaminopteridine scaffold, and some representative structures are shown in Figure 4. The DIPPR set consisted mainly of substituted hydrocarbons and did not have any specific common scaffold, though it is possible to broadly divide the data set into aliphatic and aromatic compounds. The differences in the two sets are characterized by their average Tanimoto similarity of 0.38 and 0.14 (based on 1052 bit BCI fingerprints[34]). The structural differences in the two sets of molecules thus allowed us to derive measures of the clustering quality in a relatively easy fashion. We then evaluated a set of 147 molecular descriptors using the Molconn-Z[35] software package. For future reference, we term this data set the *mixed data set*. The initial descriptor pool was reduced by randomly removing descriptors that had a Pearson correlation greater than 0.6 with other descriptors as well as removing descriptors that exhibited zero variance. This resulted in 23 descriptors.

The final data set we considered was derived from the aqueous solubility data set studied by Huuskonen.[36] The original data set consisted of 1236 compounds along with the logarithm of their measured aqueous solubility (log *S*,

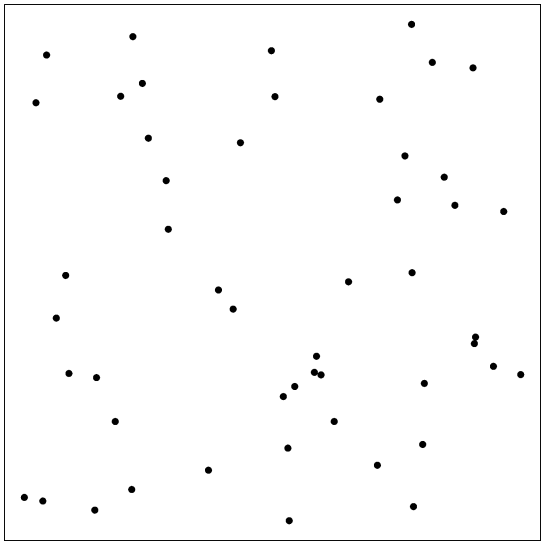

Figure 6. A set of 2D points derived from a uniform random distribution. The points exhibit no discernible clustering and thus serve as a control data set for the R-NN curve algorithm

where *S* is the solubility measured in moles per liter). Though the data set consisted of structurally diverse compounds, we decided to consider a subset that represented the most soluble and the most insoluble compounds. Thus, we selected 94 compounds whose log *S* was less than −6.0 and 84 compounds whose log *S* was greater than 0. Representative structures from these two groups are shown in Figure 5. We then evaluated a set of 147 descriptors using MOE,[37] which was then reduced to a 47-descriptor pool by randomly removing descriptors that had a Pearson correlation greater than 0.6 with other descriptors as well as removing descriptors that exhibited zero variance.

4. RESULTS

Before describing the results of our tests on the three data sets, we performed an experiment that would serve as a control. We generated a set of 2D data points from a uniform random distribution which should not exhibit any clustering. Visual inspection of the plot in Figure 6 indicates that this is so. Thus, we expect that the *R*-NN algorithm should also predict that there are no clusters present; that is, the data set characterizes a boundary condition of the algorithm. Upon applying the algorithm to the data set, we observed that it predicted that no clusters were present.

4.1. Simulated Data. The plots of the simulated 2D data sets are shown in Figure 3. It is clear that visual inspection can be used to clearly identify the number of clusters present. However, the number of clusters in data sets **C** and **D** are a little subjective. In the case of **C**, the left-hand cluster of points appears to be *joined* by a bridge, but visually one would expect that the left-hand cluster is really composed of two individual clusters. For the case of **D**, it is apparent that there are three clusters. However, one of them is composed of only two points and is thus near-singleton. However, since it is significantly far from the other two clusters, we consider it as a unique cluster. With the exception of **A**, these data sets provide insight into the behavior of the *R*-NN and *k*-means algorithms and the utility of the silhouette width as a measure of cluster quality.

Table 1. A Summary of the Quality of the *k*-means Clusterings for the Simulated Data Sets[a]

simulated data set	*k*	average silhouette width	ASW/cluster
A	**2**	0.81	0.83, 0.77
	3	0.56	0.81, 0.44, 0.49
B	2	0.66	0.88, 0.61
	3	0.54	0.87, 0.42, 0.49
	4	0.55	0.86, 0.69, 0.48, 0.30
C	2	0.69	0.81, 0.61
	3	0.65	0.77, 0.67, 0.51
	4	0.53	0.67, 0.53, 0.51, 0.43
D	2	0.65	0.65, 0.66
	3	0.45	0.61, 0.35, 0.23
	4	0.47	0.36, 0.34, 0.33, 0.21

[a] The quality is measure using the average silhouette width. Bold values of *k* indicate the number of clusters predicted by the *R*-NN algorithm.

Table 1 summarizes the quality of clustering for each of the simulated data sets using various values of *k*. The values of *k* that are in bold indicate the number of clusters that were predicted by the *R*-NN algorithm. For the case of **A**, we see that the predicted number of clusters is two. When *k*-means clustering was performed using $k = 2$ and $k = 3$, we see that the former led to a significantly higher value of the average silhouette width, indicating a better quality of clustering. However, this is not surprising as the plot indicates two well-separated clusters. We have also included the silhouette values for each of the clusters for a given *k*. Thus, for data set **A**, we see that, for $k = 2$, each of the individual clusters exhibits a high degree of clustering, whereas when $k = 3$, one of the clusters has a high degree of structure but the remaining two can be said to exhibit a low degree of clustering.

Data set **B** is a slightly tougher test of the *R*-NN curve algorithm. Visually, there are four distinct clusters. However, the average silhouette width listed in Table 1 indicates that the best clustering is obtained when $k = 2$. This is clearly in opposition to what we observe in the plot. If we consider the individual silhouette values for each cluster for a given *k*, we see that the values for two of the clusters for $k = 4$ are quite similar to the values observed for $k = 2$. The extra two clusters do not exhibit a high degree of clustering. Thus, on the basis of the individual silhouette values, one might tend to accept $k = 4$ as the proper clustering, even though the average silhouette value is lower than when $k = 2$. It is also interesting to note that, when $k = 3$, only one of the clusters has a high silhouette value, whereas the other two have quite poor values. More interestingly, the *R*-NN algorithm predicts that there should be three clusters. This can be understood by considering the fact that, when clusters are radial in nature, the *R*-NN algorithm will not be able to differentiate between them. However, though the ASW is lower for $k = 3$ compared to $k = 2$, it is only marginally poorer compared to $k = 4$. That is, by mispredicting the number of clusters as three rather than four, the clustering quality is not significantly decreased.

Data set **C** presents an interesting problem. The points on the left-hand side of the plot could be considered as two separate clusters. However, the fact they are joined together indicates that it is also possible to consider them to be a single, albeit distorted, cluster. For this case, the *R*-NN

Table 2. A Summary of the Quality of Clustering of the DHFR+DIPPR Combined Data Set for the Three Descriptor Sets Considered[a]

descriptors	*k*	ASW	ASW/cluster
SsssN, SdssC, SsOH, SHBd	2	0.71	0.77, 0.51
	3	0.67	0.91, 0.59, 0.51
	4	0.73	0.91, 0.78, 0.59, 0.55
SaasC, SdssC, Qv, SaaN, Xvc3, SHCsats	2	0.67	0.79, 0.51
	3	0.70	0.74, 0.69, 0.59
	4	0.61	0.74, 0.68, 0.42, 0.14
all 23 descriptors	2	0.19	0.27, 0.17
	3	0.35	0.47, 0.42, 0.34
	4	0.53	0.54, 0.46, 0.46, 0.26

[a] Note that the first two sets were selecetd randomly from the reduced pool of 24 descriptors. Values of *k* in bold indicate the number of clusters predicted by the *R*-NN algorithm. SsssN, sum of E-state values for sp^3 nitrogens;[44] SdssC, sum of E-state values for sp^2 carbons;[44] SsOH, sum of E-state values for oxygen in hydroxyl groups;[44] SHBd, sum of E-state values for strong hydrogen bond donors;[44] SaasC, sum of E-state values for aromatic carbons;[44] Qv, general polarity; SaaN, sum of E-state value for pyrrole nitrogens;[44] Xvc3, third-order valence cluster index.[45]

algorithm predicted three clusters. However, when we perform the clustering using *k*-means, the ASW indicates that the optimal clustering occurs when $k = 2$. If we look at the silhouette values for individual clusters for a given *k*, we see that, when $k = 2$, one of the clusters has a high value. However, when $k = 3$, the highest value of silhouette width is lower compared to when $k = 2$, but the next cluster has a higher silhouette width compared to the second cluster when $k = 2$. The overall average is pulled down by the value for the third cluster. It is clear that, by looking at individual silhouette widths, one is able to get a clearer picture of the situation. However, it is also true that, if one is to consider the individual silhouette widths, choosing an optimal *k* does become more subjective than simply using an average silhouette width. In this case, the *R*-NN algorithm is able to correctly identify the natural number of clusters in the data set.

Finally, for data set **D** it is visually clear that the data set consists of three clusters. The *R*-NN algorithm predicts this as the natural number of clusters. However, when we perform the *k*-means clustering using $k = 3$, it is significantly lower than for $k = 2$. Clearly, the ASW metric considers the clustering whereby the two points in the lower right-hand corner are merged with one of the other clusters as a better clustering, and this is indicated by the individual silhouette widths.

4.2. Mixed Data Set. The first step in the clustering of this data set was the choice of chemical space. As noted above, we evaluated a set of 147 Molconn-Z topological descriptors. This was processed to remove correlated and zero-variance descriptors, resulting in a reduced pool of 23 descriptors. Without prior knowledge as to the suitability of a specific subset of these descriptors, we selected two random subsets and also considered the entire 23-descriptor space for the purposes of clustering. For all the scenarios, we applied the *R*-NN curve algorithm to predict the number of clusters and then performed a *k*-means clustering as described previously. The results of the clusterings are summarized in Table 2.

It can be seen that, for two of the three descriptor sets chosen, the *R*-NN algorithm predicts four clusters. This is not too surprising since the DHFR inhibitors are, broadly, based on two scaffolds, both of which are structurally dissimilar to the DIPPR data set, which itself consists of branched aliphatic and aromatic compounds. As a result, when combined with the DIPPR data set, one could expect that there would be four clusters. However, the fact that the six-descriptor subset leads to a prediction of three clusters is not too surprising since the spatial distribution of points from one descriptor space does not necessarily carry over to different descriptor spaces.[38]

When we consider the average silhouette width for the various clusterings, we see that the values are relatively consistent with the predictions made by the *R*-NN curve technique for all the descriptor sets considered. If we also consider the silhouette widths for the individual clusters for a given *k*, we see that the values are quite consistent as well. Thus for the four-descriptor case, we see that, when $k = 4$, the two best clusters have a higher silhouette width than the maximum silhouette widths observed for $k = 2$ or $k = 3$. On the other hand, this is not the case for the six-descriptor case. For this case, we can see that, when $k = 4$, one of the clusters is of very poor quality, with a silhouette width of 0.14. As a result, this lowers the average silhouette width for the clustering. For the $k = 2$ case, we see that the difference in silhouette widths is relative large, whereas for the $k = 3$ case, the two best clusters have silhouette widths which are somewhat close to each other. In addition, the lowest silhouette width for $k = 3$ is still higher than the lowest for $k = 2$.

We next attempted to visualize the distribution of the structures in the given chemical spaces using a principal components analysis. Plots of the first two principal components for the three descriptor sets are displayed in Figure 7. In each of the plots, the point are colored by their class membership—black circles for DHFR inhibitors and blue squares for the molecules from the DIPPR collection. In the case of the four-descriptor subset, there is a relatively clean separation between the two classes along the horizontal. An immediate feature of the plot is the banding in the vertical direction. However, it is also possible to consider the two small groups toward the bottom right as belonging to a single cluster. These two principal components explain a total of 97% of the total variance and thus can be expected to be a faithful representation of the structure of the four-dimensional data.

The situation is a little clearer for the six-descriptor case. The first two principal components explain 94% of the total variance and can thus be expected to be a good representation of the overall structure of the six-dimensional data set. As before, we observe banding in the vertical direction, but the distinction between the bands is much clearer. For this case, the fact that there are three clusters is relatively clear. However, note the small group of points between the two right-most bands. One could consider this a separate cluster, but as was noted for the case of the simulated data sets, a *k*-means clustering will tend to merge this into one of the larger clusters. In addition, due to the design of the *R*-NN algorithm, such a cluster may be hidden from view by the large ones next to it.

Finally, for the all-descriptor case, it is evident that the clustering is not very distinct, and this is confirmed by the low values of the average silhouette width. Furthermore, the

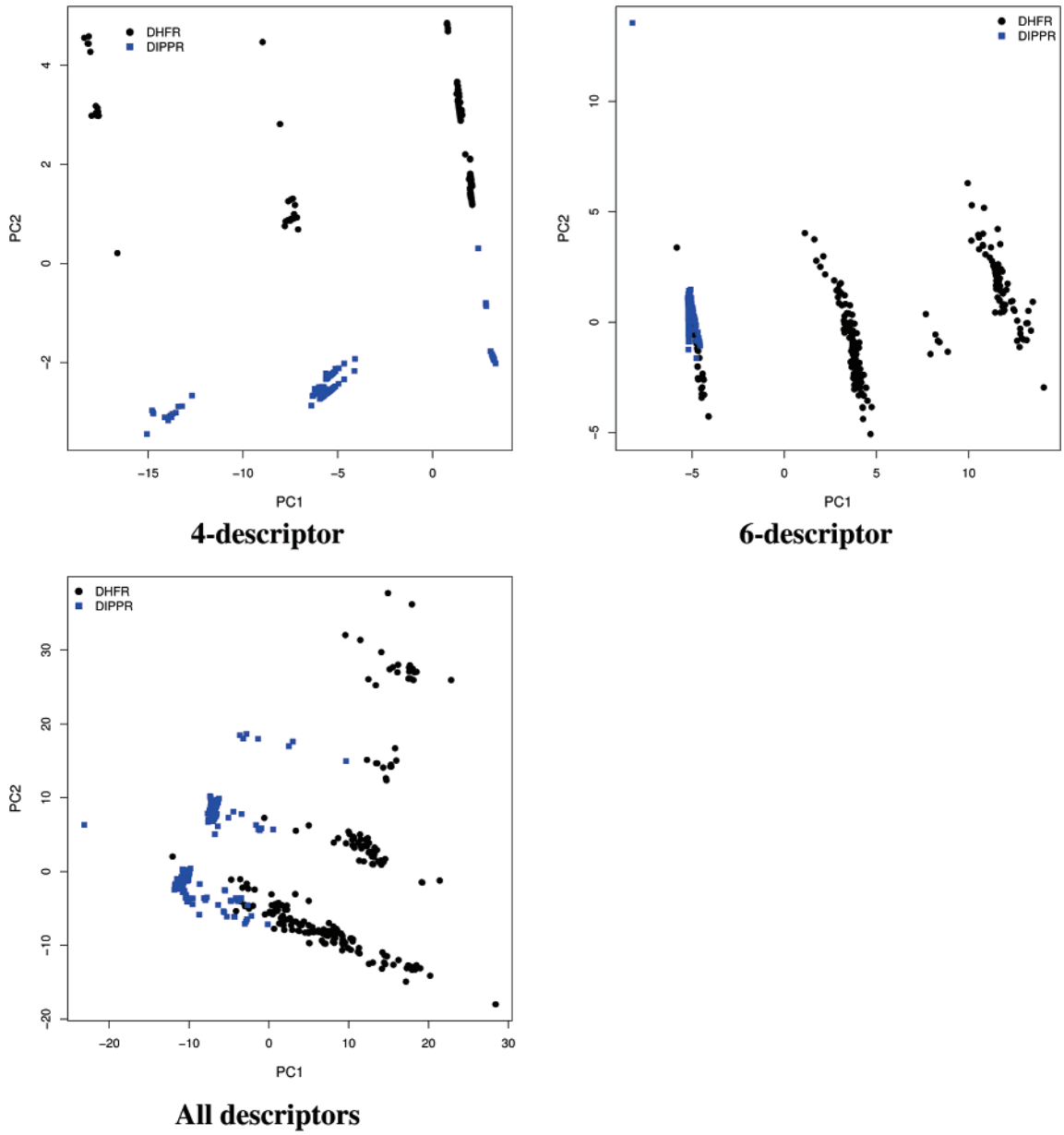

Figure 7. Plots of the first two principal components for the three descriptor sets selected for the DHFR+DIPPR data set. Blue squares correspond to points from the DIPPR data set, and black circles correspond to points from the DHFR data set.

first two principal components only explain 66% of the total variance. Clearly, one cannot fully explain the structure of the data set using just the first two principal components. As before, if one considers the banding, then three or four clusters are discernible. However, given the ASW and the principal components plot for the all-descriptor case, a definitive answer is not forthcoming.

4.3. Aqueous Solubility Data Set. As with the mixed data set, we had no prior knowledge as to a good subset of descriptors to perform clustering. However, since the data set was generated on the basis of a property value (log *S*), we decided to build a classification model. Thus, molecules whose log $S < -6.0$ were assigned to the "insoluble" class, and those with log $S > 0$ were assigned to the "soluble" class. To develop the classification model, we considered a random forest[39] due to its ability to perform automatic feature selection. The model was built using the whole 47-descriptor pool, and the out-of-bag estimate of the error was 0% over a number of runs. This is not surprising since the molecules were selected from the original data set[36] such that they were clearly separated in terms of their log *S* values. The model was then analyzed to determine the four most important descriptors[39] (using the mean decrease in accuracy as the importance measure). We then performed the *R*-NN analysis and subsequent *k*-means clustering using these four descriptors. We also considered the whole 47-descriptor pool to investigate the behavior in absence of feature selection. The results are summarized in Table 3. In both cases, the value of *k* predicted by the *R*-NN method agrees with the value of *k* determined using the average silhouette width. In the four-descriptor case, we see that for $k = 2$ one of the clusters has a silhouette width of 0.78, indicating a high degree of clustering. This is confirmed if we view the plot of the first two principal components (which account for 95% of the total variance) for the four-descriptor data set in Figure 8A. The soluble compounds form a relatively tight cluster, whereas the insoluble compounds form a relatively more scattered one. It is evident that $k = 2$ is the natural number

Table 3. A Summary of the Quality of Clustering of the Aqueous Solubility Data Set for the Two Descriptor Sets Considered[a]

descriptors	*k*	ASW	ASW/cluster
PEOE-VSA, pmiY, ASA-H, log $P_{o/w}$	**2**	0.67	0.78, 0.57
	3	0.31	0.52, 0.43, 0.33
	4	0.27	0.43, 0.42, 0.33, 0.33
all 47 descriptors	**2**	0.31	0.43, 0.18
	3	0.22	0.36, 0.32, 0.08
	4	0.26	0.46, 0.15, 0.06, −0.02

[a] Values of *k* in bold indicate the number of clusters predicted by the *R*-NN algorithm. PEOE-VSA, sum of approximate van der Waals surface area for atoms with partial charge in the range [0:05; 0:10); pmiY, *y* component of the principal moment of interia; ASA-H, water-accessible surface area of all hydrophobic atoms; log $P_{o=w}$, log of the octanol/water partition coefficient.

of clusters. For $k = 3$ and $k = 4$, we see that the average silhouette widths are significantly lower than for $k = 2$.

A similar situation is observed when we consider the whole 47-descriptor pool. However, in this case, we see that, though *k* is correctly predicted as 2 by both the *R*-NN curve algorithm as well as the average silhouette width, the cluster quality is in general quite poor, compared to the four-descriptor case. The first two principal component for this case only explain 49% of the total variance and are plotted in Figure 8. It is clear that there is a large degree of scatter and that the clustering is not very distinct. This is not surprising since the four descriptors were chosen on the basis of their importance to the predictive ability of the random forest model. Thus, the four descriptors should characterize a good partitioning of the data set into soluble and insoluble classes. This observation is further strengthened when we consider the representative structures shown in Figure 5. The insoluble compounds are characterized by hydrophobic features, whereas the soluble compounds are characterized by polar features. These features are characterized well by the four descriptors which include log $P_{o/w}$ and the water-accessible surface area.

5. DISCUSSION

Though the ability of the *R*-NN curve algorithm to detect the natural number of clusters is clear for well-defined clusters, the validity of the predicted value of *k* can be doubtful when clusters are less crisp. Furthermore, the *R*-NN curve algorithm cannot handle data sets where the clusters may be distributed in a concentric fashion. In such cases, one or more clusters may be hidden from view and will end up being considered as a single cluster. One possible approach to alleviating this problem is to replace the hypersphere around a query point with an angular slice. By rotating the slice, we would then be able to take into account the density of neighbors in different directions. The disadvantage of this approach is that it would significantly increase the running time of the algorithm. Further investigation is required to decide whether the increased reliability is worth the increased time requirement.

Another aspect of the current implementation of the *R*-NN curve algorithm is that it considers all the points in the data set. For large data sets, this can become time-consuming. One approach to avoid this is to sample the *R*-NN curves that are to be analyzed for cluster detection. The simplest solution is to randomly sample the *R*-NN curves. However, it is not entirely clear what percentage of the *R*-NN curves for the data set should be sampled. Our experiments indicate that a random sample of 60% of the data set is sufficient to be able to predict *k* correctly (compared to the prediction when the whole data set is used). In two of the three data sets, we were able to correctly predict *k* with 45% of the data set. However, a more logical approach is to only consider the *R*-NN curves for the data points that lie in the densest regions of the data set. This is because, in the presence of clustering, the steps in the *R*-NN curves for the compounds in the main body of a cluster will be more pronounced than if we consider the *R*-NN curves for outlying compounds (whose *R*-NN curves would be characterized by long lower tails, see Figure 1B). Thus, for a point lying in

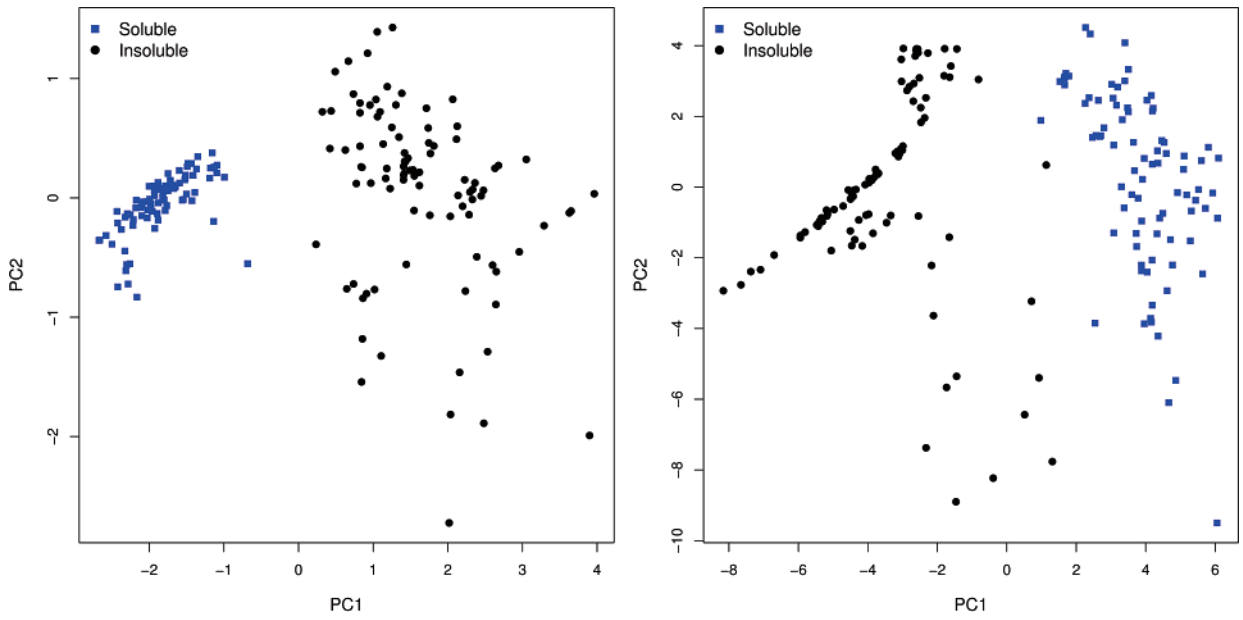

Figure 8. Principal components plots for the aqueous solubility data set. A is derived from the four most important descriptors from the random forest classi_cation model. B is derived from the whole 47-descriptor pool.

the middle of a cluster, as we increase the radius, we get a significant number of neighbors. As the radius increases beyond the cluster, the number of neighbors will become very small (or zero), and thus the *R*-NN curve will become close to flat. If there is another cluster nearby, then as the radius increases, the number of neighbors will start to increase again. This type of behavior will be less distinct if we start with a point that lies in a sparse region of the descriptor space.

The implementation of this approach is relatively simple since by selecting a suitable $R_{\max(S)}$ value as a cutoff we can choose the *R*-NN curves of the compounds in the dense regions of the descriptor space. Since the evaluation of $R_{\max(S)}$ involves a numeric differentiation of the *R*-NN curve, this approach does not take significantly longer than the more simplistic random sampling approach.

This study employed the *k*-means algorithm for the actual clustering. We also investigated the use of the Partitioning Around Medioids[2] algorithm. This method is an extension of the traditional *k*-means algorithm and is designed to be more robust. The results obtained using this algorithm were identical to what was obtained using *k*-means and hence are not included here. We also considered the use of a hierarchical clustering algorithm. However, the results obtained from such an algorithm are not directly comparable to our approach since no *k* has to be specified. Rather, the tree structure obtained by hierarchical clustering algorithms can be cut at a specific level, leading to individual clusters. A number of level selection algorithms are available and have been reviewed in ref 40. The goal of level selection in hierarchical clustering is to indicate where in the tree one can perform a cut, leading to *k* number of clusters such that they are optimal. In this context, optimality is generally a tradeoff between the number of clusters and the tightness of the individual clusters as characterized by inter- and intracluster variances. Though not directly comparable, we were interested in seeing whether a given level selection method would lead to the same number of clusters as predicted for the data set by the *R*-NN curve algorithm. For this purpose, we used the Kelley[41] level selection algorithm. In general, the number of clusters indicated by the level selection algorithm was much higher than indicated by the *R*-NN curve algorithm. Part of the reason is due to the difference in the underlying clustering algorithms. However, another reason for this difference is that the *R*-NN curve approach is limited by the resolution of the *R*-NN curves being analyzed. Small changes in the slope are not captured due to the relatively low resolution as well as due to the smoothing process. One could increase the resolution of the *R*-NN curves, but this would also lead to an increase in run time.

This aspect of the *R*-NN algorithm also leads to the observation that, even when there is no distinct clustering, the method may predict a certain number of clusters. Part of this reason is that small, localized variations in the neighbor density will lead to stepping in the *R*-NN curve. These steps may have nonzero slopes, but it is possible that the smoothing process will cause the algorithm to consider these artifacts as indicative of clustering. Whether such variations in local density can be considered as clusters is subjective. At the same time, this observation also allows us to provide a measure of confidence in the predicted number of clusters. That is, if the step in the sigmoidal *R*-NN curve has a slope of 0, we can be relatively sure that we are indeed characterizing a distinct cluster. As the slope increases away from zero, we would conclude that we are faced with an increasingly indistinct clustering. Though useful, this approach would be challenging to implement since it could be confused by artifacts in the *R*-NN curve arising due to low resolution.

Finally, we consider the computational complexity of the algorithm. Let f be the complexity to evaluate the roots in each step. Since there are n points, our current algorithm determines the *R*-NN curve for each point. Given a linear scan algorithm for near neighbors, the complexity of this step is $O(n)$. This is because we evaluate the *R*-NN curves for a fixed set of radii, independent of n. Thus, the overall time complexity for the cluster counting algorithm is $O(fn^2)$. Given the quadratic complexity, this approach is not very feasible for large data sets. To improve the time complexity one can use faster near-neighbor methods, such as KD-trees[42] or locality sensitive hashing,[19,43] to get a sublinear near-neighbor query time (nearest neighbors are detected in $o(n)$ time). Thus, the execution time can then be shown to be subquadratic in n. In addition, as noted above, rather than evaluating the *R*-NN curve for all the points in a data set, a sampling procedure (such as biased sampling based on the L_2 norm) could be employed.

6. CONCLUSIONS

We have presented an algorithm that determines the number, k, of natural clusters present in a data set of arbitrary dimensions. The algorithm is based on the notion of *R*-NN curves, which are a graphical representation of the spatial location of a compound in a chemical descriptor space. The characteristic feature of such *R*-NN curves is that, in the presence of clustering, the normally sigmoidal curves exhibit steps. By identifying the number of steps in these curves, taken over the whole data set, we are able to determine the number of clusters present. This approach provides an alternative to the trial-and-error approach of performing multiple clusterings with different values of k and then choosing that k which leads to the best cluster quality.

We measured the performance of our algorithm on one artificial and two chemical data sets of dimensionality ranging from 2 to 47. In general, the value of k predicted by the *R*-NN curve algorithm matched the number of clusters when the data sets were viewed visually. For the data sets with a dimensionality greater than two, multidimensional scaling was performed, and the results appear to confirm the predicted values of k. It was also interesting to note that the number of clusters predicted by use of the average silhouette width matched the number of clusters predicted by the *R*-NN curve algorithm in most of the cases. However, in cases where the average silhouette width led to the wrong number of clusters being suggested, the *R*-NN algorithm was able to identify the correct number of clusters.

ACKNOWLEDGMENT

This work was supported by NIH Grant No. NIH-NHGRI/P20 HG 003894-01.

REFERENCES AND NOTES

(1) Willett, P. *Similarity and Clustering Chemical Information Systems*; Research Studies Press Ltd.: Letchworth, U.K., 1987.

(2) Kaufman, L.; Rousseeuw, P. *Finding Groups in Data: An Introduction to Cluster Analysis*, 2nd ed.; Wiley: New York, 1990.
(3) Ward, J. Hierarchical Grouping to Optimize an Objective Function. *J. Am. Stat. Assoc.* **1963**, *58*, 236−244.
(4) MacQueen, J. Some Methods for Classification and Analysis of Multivariate Observations. In *Proc. 5th Berkeley Symp. Math. Stat. Prob*.; University of California Press: Berkeley, CA, 1967; Vol. 5.
(5) Engels, M.; Gibbs, A.; Jaeger, E.; Verbinnen, D.; Lobanov, V.; Agrafiotis, D. A Cluster-Based Strategy for Assessing the Overlap between Large Chemical Libraries and Its Application to a Recent Acquisition. *J. Chem. Inf. Model.* **2006** [Online early access].
(6) Chema, D.; Goldblum, A. The "Nearest Single Neighbor" Method—Finding Families of Conformations within a Sample. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 208−217.
(7) Barnett-Norris, J.; Guarnieri, F.; Hurst, D.; Reggio, P. Exploration of Biologically Relevant Conformations of Anandamide, 2-Arachidonylglycerol, and Their Analogues Using Conformational Memories. *J. Med. Chem.* **1998**, *41*, 4861−4872.
(8) Feher, M.; Schmidt, M. Fuzzy Clustering as a Means of Selecting Representative Conformers and Molecular Alignments. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 810−818.
(9) Yamashita, F.; Itoh, T.; Hashida, M. Visualization of Large-Scale Aqueous Solubility Data Using a Novel Hierarchical Data Visualization Technique. *J. Chem. Inf. Model.* **2006**, *46*, 1054−1059.
(10) Bottegoni, G.; Cavalli, A.; Recanatini, M. A Comparative Study on the Application of Hierarchical-Agglomerative Clustering Approaches to Organize Outputs of Reiterated Docking Runs. *J. Chem. Inf. Model.* **2006**, *46*, 852−862.
(11) Murray, C.; Cato, S. Design of Libraries to Explore Receptor Sites. *J. Chem. Inf. Comput. Sci.* **1998**, *39*, 46−50.
(12) Cleves, A.; Jain, A. Robust Ligand-Based Modeling of the Biological Targets of Known Drugs. *J. Med. Chem.* **2006**, *49*, 2921−2938.
(13) Vidal, D.; Thormann, M.; Pons, M. A Novel Search Engine for Virtual Screening of Very Large Databases. *J. Chem. Inf. Model.* **2006**, *46*, 836−843.
(14) Li, W. A Fast Clustering Algorithm for Analyzing Highly Similar Compounds of Very Large Libraries. *J. Chem. Inf. Model.* **2006**, *46*, 1919−1923.
(15) Butina, D. Unsupervised Data Base Clustering Based on Daylight's Fingerprint and Tanimoto Similarity: A Fast and Automated Way To Cluster Small and Large Data Sets. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 747−750.
(16) Sammon, J. A Nonlinear Mapping for Data Structure Analysis. *IEEE Trans. Comput.* **1969**, *18*, 401.
(17) Kruskal, J. Multidimensional Scaling By Optimizing Goodness of Fit to a Nonmetric Hypothesis. *Psychometrika* **1964**, *29*, 1−27.
(18) Agrafiotis, D.; Lobanov, V. Nonlinear Mapping Networks. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 1356−1362.
(19) Guha, R.; Dutta, D.; Jurs, P.; Chen, T. R-NN Curves: An Intuitive Approach to Outlier Detection Using a Distance Based Method. *J. Chem. Inf. Model.* **2006**, *46*, 1713−1722.
(20) Fréchet, M. Sur Quelque Points du Calculi Fonionnel. *Rend. Circolo Math. Palermo* **1906**, *22*, 1−74.
(21) Rucklidge, W. In *Efficient Visual Recognition Using the Hausdorff Distance; Lecture Notes in Computer Science*; Springer: Berlin, Germany, 1996; Vol. 1173.
(22) Goodman, L.; Kruskal, W. Measures of Associations for Cross-Validation. *J. Am. Stat. Assoc.* **1954**, *49*, 732−764.
(23) Hubert, L.; Arabie, P. Comparing Partitions. *J. Classification* **1985**, *2*, 193−218.
(24) Dunn, J. Well Separated Clusters and Optimal Fuzzy Partitions. *J. Cybernetics* **1974**, *4*, 95−104.
(25) Davies, D.; Bouldin, D. A Cluster Separation Measure. *IEEE Trans. Pat. Recognit. Mach. Intell.* **1979**, *1*, 224−227.
(26) Domingues, F.; Rahnenfuhrer, J.; Lengauer, T. Automated Clustering of Ensembles of Alternative Models in Protein Structure Databases. *Protein Eng., Des. Sel.* **2004**, *17*, 537−543.
(27) Kapp, A.; Tibshirani, R. Are Clusters Found in One Dataset Present in Another? *Biostatistics* **2007**, *8*, 9−32.
(28) Liu, W.; Di, X.; Yang, G.; Matsukazi, H.; Huang, J.; Mei, R.; Ryder, T.; Webster, T.; Dong, S.; Liu, G.; Jones, K.; Kennedy, G.; Kulp, D. Algorithms for Large Scale Genotyping Microarrays. *Bioinformatics* **2003**, *19*, 2397−2403.
(29) Rao, S.; Rodriguez, A.; Benson, G. Evaluating Distance Functions for Clustering Tandem Repeats. *Genome Inf. Ser.* **2005**, *16*, 3−12.
(30) IDAMS Statistical Software. http://www.unesco.org/webworld/idams (accessed May 31, 2007).
(31) Thomas, M. A Generalization of Poissons Binomial Limit for Use in Ecology. *Biometrika* **1949**, *36*, 18−25.
(32) Sutherland, J.; O'Brien, L.; Weaver, D. Spline-Fitting with a Genetic Algorithm: A Method for Developing Classification Structure-Activity Relationships. *J. Chem. Comput. Sci.* **2003**, *43*, 1906−1915.
(33) Goll, E.; Jurs, P. Prediction of the Normal Boiling Points of Organic Compounds from Molecular Structures with a Computational Neural Network Model. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 974−983.
(34) Digital Chemistry. http://www.digitalchemistry.co.uk (accessed May 31, 2007).
(35) Molconn-Z. http://www.edusoft-1c.com/molconn (accessed May 31, 2007).
(36) Huuskonen, J. Estimation of Aqueous Solubility for a Diverse Set of Organic Compounds Based on Molecular Topology. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 773−777.
(37) Chemical Computing Group Inc. Molecular Operating Environment (MOE 2004.03). http://www.chemcomp.com/ (accessed May 31, 2007).
(38) Shanmugasundaram, V.; Maggiora, G.; Lajiness, M. Hit Directed Nearest-Neighbor Searching. *J. Med. Chem.* **2005**, *48*, 240−248.
(39) Breiman, L.; Friedman, J.; Olshen, R.; Stone, C. In *Classification and Regresion Trees*; CRC Press: Boca Raton, FL, 1984.
(40) Wild, D.; Blankley, C. Comparison of 2D Fingerprint Types and Hierarchy Level Selection Methods for Structural Grouping Using Ward's Clustering. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 155−162.
(41) Kelley, L.; Gardner, S.; Sutcliffe, M. An Automated Approach for Clustering an Ensemble of NMR-Derived Protein Structures into Conformationally Related Subfamilies. *Protein Eng.* **1996**, *9*, 1063−1065.
(42) Bentley, J. Multidimensional Binary Search Trees Used for Associative Searching. *Commun. ACM* **1975**, *18*, 509−517.
(43) Dutta, D.; Guha, R.; Jurs, P.; Chen, T. Scalable Partitioning and Exploration of Chemical Spaces Using Geometric Hashing. *J. Chem. Inf. Model.* **2006**, *46*, 321−333.
(44) Kier, L.; Hall, L. In *Molecular Structure Description: The Electrotopological State*; Academic Press: Burlington, MA, 1999.
(45) Kier, L.; Hall, L. In *Molecular Connectivity in Structure-Activity Analysis*; John Wiley and Sons: New York, 1986.

CI600541F