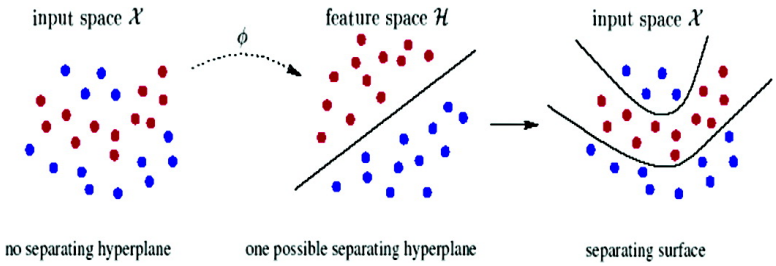
JOURNAL OF CHEMICAL INFORMATION AND MODELING

Article

One- to Four-Dimensional Kernels for Virtual Screening and the Prediction of Physical, Chemical, and Biological Properties

Chlo-Agathe Azencott, Alexandre Ksikes, S. Joshua Swamidass, Jonathan H. Chen, Liva Ralaivola, and Pierre Baldi

J. Chem. Inf. Model., **2007**, 47 (3), 965-974• DOI: 10.1021/ci600397p • Publication Date (Web): 06 March 2007



More About This Article

Additional resources and features associated with this article are available within the HTML version:

- Supporting Information
- Links to the 3 articles that cite this article, as of the time of this article download
- Access to high resolution figures
- Links to articles and content related to this article
- Copyright permission to reproduce figures and/or text from this article

View the Full Text HTML

One- to Four-Dimensional Kernels for Virtual Screening and the Prediction of Physical, Chemical, and Biological Properties

Chloé-Agathe Azencott,[†] Alexandre Ksikes,[‡] S. Joshua Swamidass,[†] Jonathan H. Chen,[†] Liva Ralaivola,[§] and Pierre Baldi*[,†,||]

School of Information and Computer Sciences, University of California—Irvine, Irvine, California 92697-3435

Received September 13, 2006

Many chemoinformatics applications, including high-throughput virtual screening, benefit from being able to rapidly predict the physical, chemical, and biological properties of small molecules to screen large repositories and identify suitable candidates. When training sets are available, machine learning methods provide an effective alternative to ab initio methods for these predictions. Here, we leverage rich molecular representations including 1D SMILES strings, 2D graphs of bonds, and 3D coordinates to derive efficient machine learning kernels to address regression problems. We further expand the library of available spectral kernels for small molecules developed for classification problems to include 2.5D surface and 3D kernels using Delaunay tetrahedrization and other techniques from computational geometry, 3D pharmacophore kernels, and 3.5D or 4D kernels capable of taking into account multiple molecular configurations, such as conformers. The kernels are comprehensively tested using cross-validation and redundancy-reduction methods on regression problems using several available data sets to predict boiling points, melting points, aqueous solubility, octanol/water partition coefficients, and biological activity with state-of-the art results. When sufficient training data are available, 2D spectral kernels in general tend to yield the best and most robust results, better than state-of-the art. On data sets containing thousands of molecules, the kernels achieve a squared correlation coefficient of 0.91 for aqueous solubility prediction and 0.94 for octanol/water partition coefficient prediction. Averaging over conformations improves the performance of kernels based on the three-dimensional structure of molecules, especially on challenging data sets. Kernel predictors for aqueous solubility (kSOL), LogP (kLOGP), and melting point (kMELT) are available over the Web through: http://cdb.ics.uci.edu.

1. INTRODUCTION

Many chemoinformatics applications, including high-throughput virtual screening, benefit from being able to rapidly predict the physical, chemical, and biological properties of small molecules to screen large repositories and identify suitable candidates.[1−3] Ab initio methods, such as quantum mechanical methods, have made great progress but can still not be applied systematically due to complexity and computational cost issues.[4] When annotated training data are available, machine learning methods that try to extract relevant information more or less automatically from the data provide a suitable alternative. Here, we develop machine learning kernel methods to address problems of predictive regression, where the goal is to predict numerical values associated with a molecule, such as its degree of solubility or melting temperature.

There have been previous applications of kernel methods, in particular in the form of support vector machines (SVMs), to predictive problems in chemistry.[5−7] Most of the previous work, however, focuses on binary classification problems (e.g., toxic/nontoxic) rather than regression problems, where the goal is to predict a numerical value associated with a particular property of a molecule (e.g., melting point). With the exception of the NCI data sets used in Swamidass et al.,[6] most of the previous applications are based on very small data sets containing at most a few hundred examples, and often much less. Such data sets are not always publicly available, and their small size casts some doubts on their suitability for large-scale machine learning methods. Moreover, most previous applications of SVMs to quantitative structure−activity relationships (QSAR) rely on the application of generic kernels [e.g., radial basis functions (RBFs) or Gaussian kernels] to more or less hand-picked, and problem-specific, vectors of molecular descriptors. In contrast to previous work, here we focus on regression problems, on public data sets with thousands of compounds, and on the development of kernel methods based on both generic and specific similarity measures (e.g., Tanimoto) applied to large, combinatorial, feature vectors that can be constructed automatically to efficiently represent molecules.

The methods leverage several rich molecular representations including 1D SMILES strings, 2D graphs of bonds, and 3D coordinates to expand the library of available kernels to include: surface kernels using Delaunay tetrahedrization and other techniques from computational geometry, pharmacophore kernels, and kernels capable of taking into account multiple molecular configurations, such as conform-

* Corresponding author e-mail: pfbaldi@ics.uci.edu.
[†] Institute for Genomics and Bioinformatics, UCI.
[‡] Current address: Department of Computer Science, University of Cambridge, Cambridge, U. K.
[§] Current address: Department of Computer Science, University of Provence/Aix-Marseille, Marseille, France.
[||] Department of Biological Chemistry, UCI.

10.1021/ci600397p CCC: $37.00
Published on Web 03/06/2007

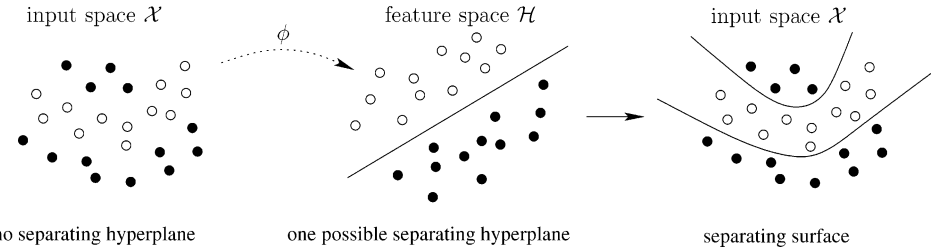


Figure 1. The kernel approach. Black dots have negative labels (−1) and white dots positive labels (+1). Left: Original complex nonlinearly separable problem in the input space $\mathcal{X}$. Middle: The mapping ϕ transforms the problem into a linearly separable problem in the feature space $\mathcal{H}$. Right: The hyperplane in feature space defines a complex nonlinear decision function in input space.

ers. The kernels are tested using cross-validation and redundancy-reduction methods on regression problems using several available data sets to predict, for instance, boiling points, melting points, aqueous solubility, octanol/water partition coefficients, and biological activity.

2. METHODS

2.1. Kernel Methods. Before we describe our library of kernels, we briefly review the basic principles behind kernel methods and SVMs. Further details can be found in the abundant literature.[8,10] For simplicity, let us consider a binary classification problem, but similar ideas apply to regression, as well as multiway classification problems. In a binary classification problem, the training set is of the form $\mathcal{S} = \{(x_1,y_1), ..., (x_l,y_l)\}$, $(x_i,y_i) \in \mathcal{X} \times \mathbb{R}$, $i = 1, ..., l$, where $y_i = \pm 1$ and $\mathcal{X}$ is an inner-product space (e.g., $\mathbb{R}^n$), with inner product denoted by $\langle .,. \rangle$. Learning is then the task of building a decision function $f : \mathcal{X} \rightarrow \mathbb{R}$ with the associated classification rule given by $y = 1$ if $f(x) > 0$ and $y = -1$ if $f(x) < 0$. Intuitively, the function f ought to achieve an optimal tradeoff between minimizing functional complexity and maximizing generalization performance.

In a simple linearly separable classification problem, the function f corresponds to the decision hyperplane $f(x) = \langle w,x \rangle + b = 0$. Note that the parameters w and b are defined only up to a multiplicative constant; thus, additional constraints on their size can be introduced. Additional constraints are necessary to define the "optimal" hyperplane, typically in the form of maximal margin constraints maximizing the closest distance between the training points and the hyperplane. Under these assumptions (see references for details), the representer theorem[11,12] states that solving for the optimal hyperplane leads to a convex quadratic optimization problem such that the solution vector w is a linear combination of a subset of the training vectors, the support vectors, such that $w = \Sigma_{i=1}^{n} \alpha_i x_i$, for some $\alpha_i \in \mathbb{R}$, $i = 1, ..., n$. Thus, f can thus be rewritten as

$$f(x) = \sum_{i=1}^{n} \alpha_i \langle x_i,x \rangle + b \qquad (1)$$

As a side note, it is even possible to write the coefficients α_i in the stronger form $\alpha_i = \beta_i y_i$ with $\beta_i \geq 0$. If the problem is not exactly linearly separable, then there is a standard convex generalization of this approach using slack variables to allow for some of the classification constraints to be violated. In all cases, the fundamental point is that the optimal hyperplane can be expressed in terms of dot products in the original space.

However, for complex problems, the set of all possible linear functions (defined by $w \in \mathcal{X}$ and $b \in \mathbb{R}$) might not be rich enough to provide appropriate predictions (Figure 1). Kernel methods generalize the previous approach to the case where the input points are far from being linearly separable. The basic idea is to use a mapping ϕ to embed the original points in a new (Hilbert) space $\mathcal{H}$, called the feature space, equipped with a dot product, where the points $\phi(x)$ are exactly or approximately linearly separable so that the convex optimization methods described above can still be applied. The prediction function (eq 1) now has the form

$$f(x) = \sum_{i=1}^{n} \alpha_i \langle \phi(x_i),\phi(x) \rangle + b \qquad (2)$$

Thus, all we need to know are the dot products of the form $\langle \phi(x_i),\phi(x_j) \rangle$. The key here is to replace the dot product in $\mathcal{H}$ by a kernel function K such that $K(x,x') = \langle \phi(x),\phi(x') \rangle$, using the definition of positive definite kernels, Gram matrices, and Mercer's theorem.

Definition 1 (Positive Definite Kernel, Gram Matrix). Let $\mathcal{X}$ be a nonempty space. Let $K \in \mathbb{R}^{\mathcal{X} \times \mathcal{X}}$ be a continuous and symmetric function. K is a positive definite kernel if, for all $n \in \mathbb{N}$, for all $x_1, ..., x_n \in \mathbb{R}$, the $n \times n$ square matrix $K = [(K(x_i, x_j))_{1 \leq i,j \leq n}]$ is positive semidefinite, that is, all its eigenvalues are non-negative.

For a given set $\mathcal{S} = x_1, ..., x_n$, K is called the Gram matrix of the kernel with respect to $\mathcal{S}$. Positive definite kernels are also referred to as Mercer kernels.

Theorem 1 (Mercer's Theorem). For any positive definite kernel function $k \in \mathbb{R}^{\mathcal{X} \times \mathcal{X}}$, there exists a mapping $\phi \in \mathcal{H}^{\mathcal{X}}$ into a feature space $\mathcal{H}$ equipped with the inner product $\langle .,. \rangle_{\mathcal{H}}$ such that

$$\forall x, x' \in \mathcal{X} k(x, x') = \langle \phi(x),\phi(x') \rangle_{\mathcal{H}}$$

Replacing the dot product in eq 2 by a Mercer kernel K leads to the corresponding prediction function f for any input point x:

$$f(x) = \sum_{i=1}^{n} \alpha_i K(x_i,x) + b \qquad (3)$$

When the theory of self-reproducing Hilbert kernels is used, it is possible to construct the function ϕ from the kernel K, but this is not necessary here.

In short, kernel methods allow one to map complex nonlinear regression or classification problems to a new feature space where convex optimization methods can be used to solve the problem. Intuitively, a kernel defines a

similarity measure between two data points, that is, two molecules, in the original space. Given any two molecules A and B, a kernel has the form $K(A,B) = \langle \phi(A), \phi(B) \rangle$, where ϕ is the embedding into the feature space. The Gram matrix of pairwise similarities between any set of available molecules must be semidefinite positive (Mercer's condition) and defines the local geometry of the embedding feature space where classification or regression functions are implemented. Thus, the application of kernel methods relies on two steps: (1) the definition of the kernels and (2) the solution of a convex optimization problem to determine the optimal linear decision manifold associated with the corresponding Gram matrix of the training data. Computing the optimal manifold in feature space $\mathcal{H}$ can be done with off-the-shelf software; thus, the main focus here is on the construction of good kernel functions, that is, good similarity measures between molecules.

Most of the kernels for discrete objects in the literature[14−16] are convolution kernels and, more specifically, spectral kernels. Spectral kernels are derived by (1) building feature vectors recording the presence or absence, or the number of occurrences, of particular substructures (subsequences, subgraphs, etc.) in the given structure and (2) defining a similarity measure between these feature vectors. Unlike the original structures that are variable in size, it is important to note that the feature vectors have a fixed size; they are actually an extension of traditional chemical fingerprints.[17,18] Because small molecules have multiple representations, multiple kernels can be derived by using each representation. We first describe these representations and how they lead to spectral feature vectors. We then describe the similarity measures that are used to compare these feature vectors.

2.2. 1D Kernels Based on SMILES Strings. Small molecules can be represented in a unique way as SMILES strings.[19,20] Although SMILES strings require selecting a somewhat arbitrary order of the atoms of a molecule, they are widely used and are particularly useful in database organization and searches, since each molecule can be associated with a unique SMILES string. We can build a spectral representation of a string by counting or indexing the number of all possible substrings of length l, or length up to l, occurring in the string. Extensions can allow for word mismatches and insertions.[14]

2.3. 2D Kernels Based on Bond Graphs. Small molecules are most familiarly represented as labeled graphs of bonds. Labels on the nodes represent atom types (e.g., C, N, and O); labels on the edges represent bond types (e.g., single and double). For small molecules, these graphs are fairly small, in terms of both the number of nodes and the number of edges. Valence rules constrain the average degree to be typically less than three. In a spectral approach, several kinds of substructures can be considered, such as labeled paths or labeled trees. Here, we use all labeled paths of length d, or up to d, starting from each node in the graph (Figure 2). Paths are allowed to self-intersect and traverse the same node twice, to capture ring structures, but are not allowed to traverse the same edge twice, to avoid "totters".[25]

2.4. 2.5D Surface and 3D Kernels Based on Delaunay Tetrahedrizations. In many biological and other applications, it is the *surface* of a molecule that matters the most, rather than its interior, since it is the surface with its charges that mediates the interactions of the molecule with other

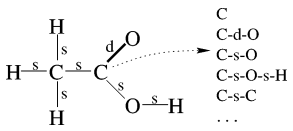


Figure 2. A molecule represented as a labeled graph. The labels on the nodes correspond to atom symbols, and those on the edges describe the type of covalent bond between atoms (e.g., "s" for single bond, "d" for double bond). Also shown are examples of labeled paths of lengths 0, 1, and 2 resulting from a depth-first search exploration of the graph, starting from one of the carbon atoms.

molecules. By "2.5D," we denote a new class of kernels derived from an explicit representation of the surface of the molecule. Starting from the 3D structure, we use techniques from computational geometry to derive essentially a triangulated graph that approximates the surface of the molecule. We then use the same spectral approach as in the 2D case to count labeled paths of length d, or up to d, in this new graph. For computational reasons, a smaller d value is typically used for the surface graph than for the bond graph, because the surface graph has higher connectivity, with an average degree of six, corresponding to the hexagonal tiling of a plane. For some applications, labels on the surface graph can include additional information, for instance, about electric charges.

To build the surface graph, we compute the Delaunay tetrahedrization of the atoms in the molecule and then use the α-shape algorithm to prune the tetrahedrization (Figure 3). Specifically, we first compute the Delaunay tetrahedrization of the molecule's atoms, where each atom is represented as a point in space, using the Computational Geometry Algorithms Library (http://www.cgal.org) with exact arithmetic. In addition to the vertices, the tetrahedrization is described also by its edges, faces, and tetrahedra. In a Delaunay tetrahedrization, the circumscribing sphere of each tetrahedron does not contain any other vertex of the tetrahedrization in its interior. The α-shape algorithm relies on this property to appropriately prune pieces of the tetrahedrization and generate the final shape.

The α-shape algorithm is used to remove low-density regions and derive an α-shape. An α-shape[21] is a subset of the tetrahedrization defined by retaining the same set of vertices but excluding some of the edges, faces, and tetrahedra present in the original tetrahedrization, according to a parameter α. Intuitively, an α-shape is obtained by carving out the tetrahedrization using a sphere-shaped scoop of radius $\sqrt{\alpha}$. If the circumscribing sphere of an edge, face, or tetrahedron is larger than a sphere of radius $\sqrt{\alpha}$, then it is removed from the tetrahedrization. The family of shapes obtained for different values of α corresponds to a family of solvent-accessible surfaces generated by using spherical probes of different size parametrized by α. If $\alpha = \infty$, the α-shape is the original tetrahedrization; if $\alpha = 0$, it is exactly restricted to the initial set of vertices. In the simulations, we use a value of $\sqrt{\alpha}$ equal to 3.8 Å, corresponding approximately to the radius of a water molecule plus the radius of a carbon atom (or a methyl group). Other values of $\sqrt{\alpha}$, in the range of 3.8−7.8 Å, lead to robust results. For a given α, the α-shape algorithm leaves three classes of edges: interior, regular, and singular. As the name implies, interior edges are buried below the surface of the α shape. Regular

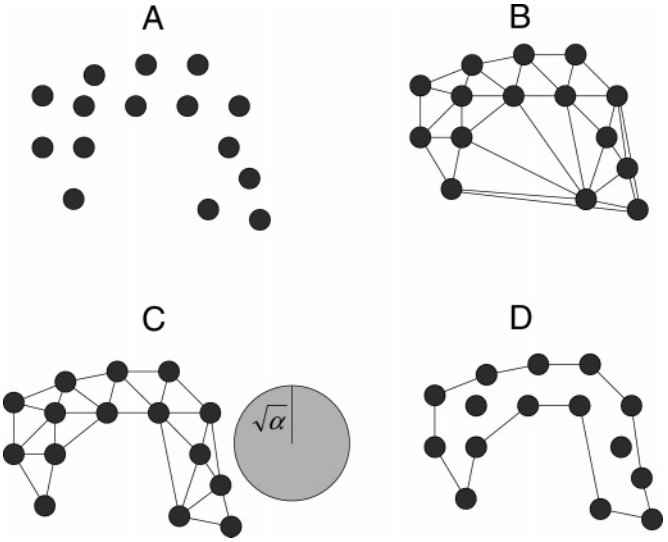


Figure 3. Illustration of the α-shape algorithm in two dimensions. α-shapes formalize our intuitive notion of shape for a set of points (A). We compute the α-shape by first computing the Delaunay triangulation (in 2D) or tetrahedrization (in 3D) of the points (B). Next, we remove all the geometric elements of the triangulation (or tetrahedrization) for which the radius of the corresponding circumscribing sphere is bigger than a parameter α. This procedure carves out regions with a low density of points from the convex hull, leaving us with the α complex (C). The surface graph, is defined by the edges at the boundary of the α-shape. Formally, this surface graph is composed of the "regular" and "singular" edges (D).

edges define the surface of the α-shape. Singular edges extend into space from the surface of the α shape but are not adjacent to any retained tetrahedron. We define a surface graph as a graph whose nodes are the atoms and whose edges are the singular and regular edges of the α-shape computed at a particular α value and conformation. This construction can be easily extended into a 3D kernel by including interior edges as well, or even all the edges of the original tetrahedrization.

In summary, given the original set of vertices associated with the atoms of the molecule in a given configuration, the Delaunay tetrahedrization constructs a set of edges on the corresponding nodes. These nodes and edges form a graph, and from this graph, various subgraphs can be derived by pruning some of the edges. The α-shape algorithm in particular allows one to identify edges associated with the surface of the molecule. In any case, for any subgraph, we can apply spectral techniques to derive a corresponding kernel. If only surface edges are retained, we call it a 2.5D Delaunay kernel; if in addition a significant set of interior edges is retained, we call it a 3D Delaunay kernel.

We use four approaches to type atoms on the basis of their local chemical environment. First, we simply use the element symbol. Second, we use our own Python implementation of XSCORE,[22] a program used to predict binding affinities between proteins and small molecules. It uses a general typing system which labels atoms as polar, hydrophobic, hydrogen-bond donating, hydrogen-bond accepting, and both hydrogen-bond donating and accepting. Third, we label atoms using the corresponding element symbol and hybridization state. For example, an sp^3 carbon atom is labeled *C*.3; an sp^2 nitrogen is labeled *N*.2, and so forth. Fourth, we use the OpenBabel implementation of Tripos' Sybyl labeling scheme, which has the most complex atom typing system, with the largest number of labels.

2.5. 3D Kernels Based on Atomic Coordinates and Pharmacophores. A simple 3D kernel can be derived by representing a molecule as a set of pairwise distances between labeled atoms or, more compactly, as a set of histograms of distances between atoms of certain types (e.g., C−C and C−N). We have developed variations of this kernel by considering triplets, or more generally *k*-tuples, of points. For instance, in the case of $k = 3$, we can use a pharmacophore representation, whereby a molecule is represented by the list of all of its triplets of atoms (or even groups of atoms), with the pairwise distances between the pairs of atoms in each triplet, and the corresponding labels, which, beyond atom type, can include information about size, polarity, electronegativity, and so forth. This approach is the same as the one recently described in Mahé et al.,[23,24] where the authors use a labeling scheme based on the Morgan indices[25,26] that increase the specificity of the labels by including topological information about adjacent atoms. If desirable, a more compact representation is derived by building histograms for each class of triplets (e.g., C−C−C, C−C−O) on the basis of the size of the smallest sphere that contains all three points (or the largest pairwise distance in the triplet). In all cases, we use the program CORINA[27,28] to derive the 3D coordinates needed for all kernels of dimension 2.5 and higher.

2.6. Beyond 3D Kernels: Conformers and Isomers. Molecules often exist in multiple configurations. Movable bonds, such as rotatable bonds, give rise to conformers, and stereocenters give rise to symmetries and isomers. While the SMILES and graph of bonds are unchanged, these alternative configurations impact the 3D structure of the molecule and the representations derived from it. One simple way to accommodate a class of configurations is to sample the class and represent a molecule as a family of molecules, each with its own set of 3D coordinates. This approach casts our problem in the framework of multi-instance problems.[29] We name the kernels derived from these approaches as 3.5D when applied to a family of surfaces (2.5D) and 4D when applied to a family of 3D representations. In all cases, in order to derive a kernel, one has at least two choices: (1) one can derive a profile vector for each molecule by, for

instance, averaging the vectors associated with the different configurations and then apply a similarity measure to the profile vectors or (2) use the average kernel value among all possible pairs, as described below. For each molecule of a data set, we generate a set of up to 15 conformations of minimal steric energy using CORINA, corresponding to up to 225 pairs of representations. The 3.5D kernel between two molecules is calculated as the average of the 2.5D kernel computed over all corresponding pairs. Similarly, the 4D kernel is computed as the average of the 3D kernel computed over all corresponding pairs. In the next section, we show that this averaging approach preserves the Mercer kernel properties. Alternative approaches to simple averaging are described, for instance, in Ray and Page[30] or Cheung and Kwok.[31]

2.7. Similarity Measures. So far, we have seen how we can associate a feature vector, or a family of vectors in the 3.5D and 4D cases, to each molecule. These vectors or fingerprints can be further processed to reduce their dimensionality. This is typically the case for the 2D binary vector representation based on the presence or absence of subgraphs in the graph of bonds, which is routinely compressed to a shorter binary fingerprint, typically of length 512 or 1024, using a modulo operator.[20,32] With or without this postprocessing step, to complete the definition of the kernels, we need to define similarity measures between such vectors. A standard approach that yields well-known Mercer kernels includes taking dot products or Euclidean distances, possibly composed with another suitable function, such as a Gaussian exponential.[8] In the case of binary fingerprints, we also use the Tanimoto similarity measure[20] between two binary fingerprints defined by the ratio of the number of common bits set to one to the total number of bits set to one in the two fingerprints $K(\vec{A}, \vec{B}) = (\vec{A} \cap \vec{B})/(\vec{A} \cup \vec{B})$. For nonbinary fingerprints based on actual counts, we have developed the MinMax measure[6,33] given by

$$K(\vec{A}, \vec{B}) = \frac{\sum_i \min(A_i,B_i)}{\sum_i \max(A_i,B_i)} \quad (4)$$

where $\vec{A} = (A_i)$. This reduces to the Tanimoto measure in the case of binary fingerprints. It has previously been shown[6] that both the Tanimoto and MinMax measures lead to kernels that satisfy the Mercer's condition.

For the 3D kernels, similarity between histograms of pairs of atoms or pharmacophores represented by a single radius can again be measured using the Euclidean distance v (i.e., the sum of squared differences between histogram bins), or a Gaussian kernel of the form $\exp[-v^2/\lambda^2]$. In the case of pharmacophores represented by the dimensions and labels of a triangle of three atoms, kernel similarity between pharmacophores can be derived by considering both the labels and the length of the sides of the triangle. For instance, we can assume that the distance is infinite if the atoms involved are not identical (e.g., CCO versus CCN). If the atoms are identical, then we can compute the Euclidean distances between the lengths of the various edges. When the triangle has some symmetry because two or three of the atom labels are identical (e.g., CCO), we can average over all possible ways of matching the two triangles or just use the best match.

Finally, in the 3.5D and 4D cases, if molecule A is represented by configurations A_1, ..., A_r and molecule B is represented by configurations B_1, ..., B_s, we can define the kernel

$$K'(A,B) = \sum_{ij} K(A_i,B_j)/(rs) \quad (5)$$

where K is any kernel defined on the individual configurations, or associated vectors. To prove that K' is indeed a Mercer kernel, it is sufficient to notice that, if $K(A_i,B_j) = \langle \phi(A_i), \phi(B_j) \rangle$, then $K'(A,B)$ can be written as $K'(A,B) = \langle \sum_i \phi(A_i)/r, \sum_j \phi(B_j)/s \rangle$. Thus, K' is a dot product with respect to the embedding that transforms a molecule A into $\phi(A) = \sum_i \phi(A_i)/r$. Here, all molecules must have the same r, or each molecule must come with its fixed value of r, to ensure that the kernel is symmetric and well-defined. Variations on these ideas, for instance by introducing weights on particular substructures or combining different kernels, have been explored (results not shown) but in general do not seem to lead to any significant improvements.

2.8. Kernels Parametrization. The following kernel parameters are investigated in detail in the simulations:

1D Kernels. These kernels are tested using Euclidean distance and all substrings of length up to l, with l varying from 2 to ∞. The case $l = \infty$ is actually easy to implement and used throughout in the results—given two molecules A and B, to compute the dot product of the feature vectors, one needs only to look at the actual substrings contained in the SMILES strings of the two molecules, up to the length of the shorter of the two strings.

2D Kernels. These kernels are tested using both Tanimoto and MinMax similarities with paths of length up to d, with d varying from 2 to 10.

2.5D Kernels. These kernels are tested using Tanimoto, MinMax, and Euclidean distance similarities together with the four labeling schemes described above: Element, XSCORE, Element-Hybridization (denoted by EH), and Sybyl. Paths of length 3 or 4 in the surface graph are investigated.

3D Delaunay Kernels. These kernels are tested using Tanimoto, MinMax, and Euclidian similarities, together with the four labeling schemes described above: Element, XSCORE, Element-Hybridization (denoted by EH), and Sybyl. Paths of length 3 or 4 in the surface graph augmented with interior edges are investigated.

3D Histogram Kernels. These kernels are tested using Euclidean distance with two, three, or four atoms per tuple (pairs, triplets, or quadruplets), with bins of size 0.05, 0.1, or 0.5 Å.

3D Pharmacophores Kernels. These kernels are tested with the default parameters in Mahé et al.[23] (edges kernel: RBF kernel of bandwidth 1.0; atoms kernel: binary kernel based on Morgan indices) using the code provided by these authors.[24]

3.5D Kernels. The 3.5D version of all previous 2.5D kernels are tested by averaging kernel values across multiple (up to 15) conformations of each molecule.

4D Kernels. The 4D version of all the previous 3D kernels are tested by averaging the kernel values across multiple conformations of each molecule.

2.9. Training and Optimization. For a given kernel, we use the e-SVR implementation of SVM in LibSVM[34] to find the optimal manifold. This implementation uses two parameters: the error/margin tradeoff C and the insensitivity ϵ. We use two nested cross-validation procedures to estimate the performance and optimize these parameters. In the results reported here, we use leave-one-out cross-validation for small data sets and 10-fold cross-validation for larger data sets. The choice of optimal C and ϵ parameters is performed using an *exhaustive* grid search over typical values. This simple approach avoids overfitting and is easy to parallelize since C and ϵ are independent of each other.

2.10. Redundancy Reduction. Redundancy reduction in prediction is common practice in areas such as bioinformatics but has not been applied systematically to data sets in chemoinformatics. To study and avoid the biases introduced by redundant data, we use the algorithm in Hobohm et al.[35] to derive redundancy-reduced data sets, by iteratively thinning clusters of molecules with high similarity, until no molecules in the training sets have a similarity greater than some preset threshold (see Results, section 4.7).

3. DATA

The library of kernels is tested over many data sets. Results are reported for seven regression data sets. Three of them are small data sets comprising less than 300 molecules (Benzodiazepines, Alkanes, Melting Points/Bergström); the other four are larger data sets comprising between 1000 and 4500 molecules [Aqueous Solubility (Huuskonen), Aqueous Solubility (Delaney), Octanol/Water Partition (XLOGP), and Melting Points/Karthikeyan]. Additional results on four small data sets (BZR, COX2, DHFR, and ER) used in Mahé et al.,[23] and on the Mutag and PTC data sets presented in previous work,[6] are also briefly discussed.

Benzodiazepines QSAR. This data set[36] consists of 72 1,4-benzodiazepine-2-ones, together with their measured affinity toward the $GABA_A$ receptor (GABA = γ-aminobutyric acid). Although it is rather small, it is characterized by good molecular diversity, which is significant for QSAR analysis. The best performance on this data set is reported in Micheli et al.[37] with a correlation coefficient of 0.999 obtained by a cascade correlation neural network.

Alkanes Boiling Point. This data set[38] consists of the first 150 noncyclic alkanes (C_nH_{2n+2} with $n < 11$), together with their boiling points in degrees Celsius, covering the range $[-164, +174]$. The best performance over this data set is reported in Micheli et al.[39] with a squared correlation coefficient of 0.999 obtained as a mean over four runs of 10-fold cross-validation.

Melting Points (Bergström). This data set[40] consists of 277 druglike compounds together with their melting points. When using $^2/_3$ of these molecules for training, Bergström et al. report a squared correlation coefficient of 0.63 and a root-mean-square error (RMSE) of 44.6 °C over a test set composed of the remaining 92 compounds. These results are obtained by partial least-squares projection on latent structures.

Aqueous Solubility (Huuskonen). This data set[41] consists of 1026 molecules together with their aqueous solubility. Fröhlich et al.[42] report a squared correlation coefficient of 0.90 for an 8-fold cross-validation, using support vector machines with a RBF kernel.

Aqueous Solubility (Delaney). This data set[43] originally consisted of 2874 molecules together with their aqueous solubility. The so-called "small" subset consists of 1144 low-molecular-weight organic compounds. When focusing on this subset of small molecules, Delaney reports an average absolute error of 0.75 logM/L, obtained by linear regression.

XLOGP. XLOGP is a linear regression method developed by Wang et al.[44] for predicting the octanol/water partition coefficient (logP), an important index of lipophilicity which is a key determinant of the pharmacokinetic properties of a molecule. The current version of XLOGP (ftp2.ipc.pku.edu.cn) comes with annotated training and testing sets with 1853 and 138 molecules, respectively. XLOGP achieves a squared correlation coefficient of 0.947 and a root-mean-squared error of 0.349 on the training set, as well as a squared correlation coefficient of 0.944 and a root-mean-squared error of 0.348 on the testing set.

Melting Points (Karthikeyan). The second large melting point data set[45] contains 4173 molecules annotated with melting points and a wide range of additional properties. Using a large set of 2D descriptors and a fully connected neural network, Karthikeyan et al. reached a correlation coefficient of 0.65 and a root-mean-square error of 52.0 °C over an internal validation set of 1042 compounds, as well as a correlation coefficient of 0.66 and a root-mean-square error of 41.4 °C over an external validation set composed of Bergström's 277 molecules. [After we got the original results from the authors and reanalyzed the data, their "R^2" coefficient is a simple (nonsquared) correlation coefficient.]

4. RESULTS

These data sets are used to compare the various classes of kernels with different parameter settings. Consistently, with the relevant literature, prediction performance in regression is assessed using three metrics: squared correlation coefficient (r^2), RMSE, and average absolute error (AAE). For each data set, the optimal parameters for each kernel are presented in the corresponding subsection. Here, d denotes the length of the paths for 2D kernels, as well as for 2.5D and 3D Delaunay kernels. k denotes the number of atoms considered per tuple in the 3D contact histogram kernels.

4.1. Small Data Sets. On the benzodiazepines data set, the 2D kernel achieves a squared correlation coefficient of 0.69. However, running the experiment over the same training and testing split as in Micheli et al.,[37,39] we obtain a correlation coefficient of 0.98, which is not significantly different from the 0.999 value reported by these authors. This kernel is more likely to capture the global structure of the molecules and their chemical groups. On the prediction of the boiling point of alkanes, the 1D, 2D, and 3D histogram kernels achieve results comparable to those in Cherqaoui and Villemin[38] and Micheli et al.[39] This is consistent with their observation that most of the relevant information is captured by the knowledge of how many carbons are attached to each carbon. On the Bergström data set, using leave-one-out cross-

Table 1. Leave-One-Out Squared Correlation Coefficient Results for the Alkanes, Benzodiazepines (BZD), and Bergström Data Sets Using Different Kernels[a]

kernel/method	BZD	alkanes	melting (Bergström)
1D	0.49	*0.95*	**0.69**
2D	**0.69**	0.94	*0.36*
2.5D Delaunay	0.41	0.87	0.22
3D Delaunay	0.41	0.90	0.27
3D pharmacophores	0.33	0.77	0.13
3D histogram + Gaussian	*0.58*	**0.97**	0.30
3.5D Delaunay	0.28	0.79	0.26
4D Delaunay	0.35	0.79	0.30
4D histogram + Gaussian	0.55	**0.97**	0.30

[a] Best results are in bold, second best in italics.

validation, we obtain an improved squared correlation of 0.69, versus the 0.63 value reported by Bergström et al. Detailed results on the small data sets are reported in Table 1.

The optimal parameters for these sets are as follows.

Benzodiazepines. For 2D kernels, considering all paths of length up to d, with MinMax similarity, is optimal. For Delaunay kernels, considering all paths of length $d = 3$, with Element-Hybrdization labeling and MinMax similarity, is optimal. For histogram kernels (3D and 4D), triplets of atoms $k = 3$ with bins of size = 0.05 Å are optimal.

Alkanes. For 2D kernels, considering all paths of length up to $d = 2$, together with MinMax similarity, is optimal. For Delaunay kernels, considering all paths up to $d = 3$, with Element labeling and Tanimoto similarity, is optimal. For histogram kernels (3D and 4D), pairs of atoms $k = 2$ with binning size bin = 0.5 Å are optimal.

Bergström. For 2D kernels, considering all paths of length up to $d = 5$ with MinMax similarity is optimal. For Delaunay kernels, considering all paths of length up to $d = 2$, with Sybyl labeling and MinMax similarity, is optimal. For histogram kernels (3D and 4D), pairs of atoms $k = 2$ with binning size bin = 0.5 Å are optimal.

In our view, however, results derived on these very small data sets are at best indicative of real performance, especially considering that the feature vectors can have high dimensionality (e.g., 100 000). Thus, here, we focus primarily on the results derived on the larger data sets.

4.2. Aqueous Solubility (Huuskonen). Table 2 reports the 10-fold cross-validation performance of different kernels compared to the published results on the prediction of aqueous solubility on the Huuskonen data set. The best performance is achieved by a contact histogram kernel with pairs of atoms ($k = 2$), and bins of size = 0.5 Å, closely followed by a 2D kernel with path length $d = 2$ and MinMax similarity applied to the counts. The corresponding squared correlation coefficient is 0.91, compared to the 0.90 value reported by Fröhlich et al.[42]

4.3. Aqueous Solubility (Delaney). Table 3 reports the 10-fold cross-validation performances of different kernels compared to the published results on the prediction of aqueous solubility on the Delaney data set. The best performance is achieved by a 2D kernel with path length $d = 2$ and MinMax similarity applied to the counts. The resulting kSOL kernel solubility predictor achieves an average absolute error of 0.44, compared to the 0.75 value reported by Delaney,[43] corresponding to an improvement of about 2.5% in terms of relative absolute error.

Table 2. Prediction Performance for Aqueous Solubility Using 10-Fold Cross-Validation on the 1026 Compounds of the Huuskonen Data Set[a]

kernel/method	r^2	RMSE	AAE
1D	0.82	0.21	0.16
2D [$d = 2$, MinMax]	*0.90*	*0.15*	*0.11*
2.5D Delaunay [$d = 3$, EH, MinMax]	0.86	0.18	0.14
3D Delaunay [$d = 3$, EH, MinMax]	0.88	0.17	0.13
3D pharmacophores	0.84	0.20	0.14
3D histogram + Gaussian [$k = 2$, bin = 0.5]	**0.91**	**0.15**	**0.11**
3.5D Delaunay [$d = 3$, Sybyl, MinMax]	0.86	0.18	0.14
4D Delaunay [$d = 3$, Sybyl, MinMax]	0.86	0.18	0.14
4D histogram + Gaussian [$k = 2$, bin = 0.5]	**0.91**	**0.15**	**0.11**
published results[42]	0.90		

[a] Best results are in bold, second best in italics.

Table 3. Prediction Performance for Aqueous Solubility Using 10-Fold Cross-Validation on the 1144 Compounds of the "Small" Data Set of Delaney[a]

kernel/method	r^2	RMSE	AAE
1D	0.87	0.59	0.56
2D [$d = 2$, MinMax]	**0.91**	**0.61**	**0.44**
2.5D Delaunay [$d = 3$, EH, MinMax]	0.88	0.72	0.52
3D Delaunay [$d = 3$, EH, MinMax]	0.88	0.72	0.51
3D pharmacophores	0.85	0.83	0.61
3D histogram + Gaussian [$k = 2$, bin = 0.5]	*0.91*	*0.63*	*0.45*
3.5D Delaunay [$d = 3$, Sybyl, MinMax]	0.87	0.75	0.55
4D Delaunay [$d = 3$, EH, MinMax]	0.87	0.75	0.53
4D histogram + Gaussian [$k = 2$, bin = 0.5]	*0.91*	*0.64*	*0.47*
published results[43]			0.75

[a] Best results are in bold, second best in italics.

Table 4. Prediction Performance for the logP Coefficient Using 10-Fold Cross-Validation on the 1991 Compounds of the XLOGP Data Set[a]

kernel/method	r^2	RMSE	AAE
1D	0.91	0.47	0.33
2D [$d = 5$, MinMax]	**0.94**	**0.38**	**0.25**
2.5D Delaunay [$d = 3$, Sybyl, MinMax]	0.91	0.45	0.30
3D Delaunay [$d = 3$, Sybyl, MinMax]	*0.92*	*0.43*	*0.29*
3D pharmacophores	0.87	0.54	0.38
3D histogram + Gaussian [$k = 2$, bin = 0.5]	*0.92*	*0.43*	*0.30*
3.5D Delaunay [$d = 3$, Sybyl, MinMax]	0.90	0.48	0.32
4D Delaunay [$d = 3$, Sybyl, MinMax]	0.90	0.49	0.32
4D histogram + Gaussian [$k = 2$, bin = 0.5]	*0.92*	*0.44*	*0.31*

[a] Best results are in bold, second best in italics.

4.4. XLOGP. The 10-fold cross-validation average performances of the different kernels on the prediction of the octanol−water partition coefficient logP are presented in Table 4. Again, the 2D kernel, with path length $d = 5$ and the MinMax similarity applied to the counts, performs the best with a squared correlation coefficient of 0.94, similar to what is reported by Wang et al.[44] In addition, we also use the same training set of 1853 compounds used to train XLOGP, and the same testing set containing 138 compounds. Using this particular split, the 2D kernel achieves a square correlation of 0.946 with a RMSE of 0.338, slightly above

Table 5. Prediction Performance for the Melting Point Using 10-Fold Cross-Validation on the 4173 Compounds in the Karthikeyan Data Set[a]

kernel/method	r^2	RMSE	AAE
1D	*0.52*	*44.88*	*34.30*
2D [$d = 10$, MinMax]	**0.56**	**42.71**	**32.58**
2.5D Delaunay [$d = 3$, Sybyl, MinMax]	0.49	46.07	35.37
3D Delaunay [$d = 3$, Sybyl, MinMax]	0.50	45.62	35.01
3D histogram + Gaussian [$k = 2$, bin = 0.1]	0.27	55.01	43.38
3.5D Delaunay [$d = 3$, EH, MinMax]	0.44	48.35	37.44
4D Delaunay [$d = 3$, EH, MinMax]	0.35	55.36	43.43
4D histogram + Gaussian [$k = 2$, bin = 0.1]	0.40	50.40	39.85
published results[45]	0.42	52.0	41.3

[a] Best results are in bold, second best in italics.

Table 6. Leave-One-Out Accuracy of the 3D and 4D Contact Histogram Kernel on the Mutag and PTC Data Sets

data set	3D contact histogram	4D contact histogram	published[6]
Mutag	86.7%	88.8%	89.1%
PTC FM	59.3%	60.5%	64.5%
PTC FR	67.2%	69.6%	66.9%
PTC MM	63.1%	64.0%	66.5%
PTC MR	62.5%	62.8%	65.7%

Table 7. Accuracy of the Pharmacophores Kernel on the Training and Testing Sets Studied by Mahé et al.[23]

data set	pharmacophores kernel[23]	best performance
BZR	78.5%	79.8% [2D with $d = 2$, MinMax]
COX2	69.8%	70.1% [1D]
DHFR	81.9%	83.0% [1D]
ER	79.8%	82.1% [2D with $d = 2$, MinMax]

the 0.944 correlation and 0.348 RMSE reported in the literature. Consistent results were further obtained on a third validation set containing 105 organic compounds gathered by Breindl et al.[46] (not shown). Thus, the 2D kernel yields an optimal octanol−water partition coefficient predictor called kLOGP.

4.5. Melting Points (Karthikeyan). Table 5 reports the 10-fold cross-validation results obtained on the data collected by Karthikeyan et al. The 2D kernels, with a path length of $d = 10$, achieve the best performance with an average absolute error of 32.6 °C, which is a significant improvement over the 39.8 °C reported by these authors. The corresponding predictor (kMELT) has a correlation coefficient of 0.8 and, since the average melting temperature in the data is about 166 °C, an absolute relative error of about 20%.

4.6. Additional Results on Classification Problems. In order to further evaluate the newly introduced kernels, we present additional results on various small classification data sets. Table 6 describes the performance of 4D histogram kernels on the Mutag and PTC data sets and shows how, by averaging over configurations, 4D kernels can perform better than 3D kernels. Table 7 describes the best performance obtained on the data sets studied by Mahé et al.[23] using pharmacophore kernels. 2D kernels with the MinMax similarity measure outperform pharmacophores kernels on two of these four small data sets: BZR with an accuracy of 79.8% versus 78.5% and ER with an accuracy of 82.1% versus 79.8%. On the two remaining data sets, 1D kernels do better than pharmacophore kernels: COX2 with an accuracy of 70.1% versus 69.8% and DHFR with an accuracy of 83.0% versus 81.9%.

Table 8. Effect of Redundancy Reduction on the Squared Correlation Coefficient for logP on the XLOGP Testing Set[a]

kernel/method	whole set	reduced set
1D	0.91	0.90
2D	0.94	0.95
3D histogram	0.92	0.91

[a] The predictor is trained either on the whole set of 1,853 compounds, or on a reduced set of 1,615 compounds, constructed so that no pair of examples has a MinMax similarity greater than 0.8.

4.7. Redundancy Reduction. To assess whether these results could have been biased by data redundancy, we perform data redundancy reduction on the Delaney, XLOGP, and Karthikeyan data sets. Using various similarity measures together with a threshold of 80%, we are able to remove about 15% of the instances from each of these data sets. Experiments run with these subsets lead to results comparable to those obtained with the corresponding full sets, as shown in Table 8. Thus, redundancy does not appear to be a major issue for these particular data sets.

5. DISCUSSION

By utilizing spectral feature vectors derived from different molecular representations (1D−4D), we have developed machine learning kernels for small molecules for the effective prediction of important properties. The quality and robustness of the predictors is obviously related to the quality and size of the available training sets. Using the larger training sets available to us, we have derived three predictors for solubility (kSOL), the octanol/water partition coefficient (kLOGP), and the melting point (kMELT), which are available over the Web at http://cdb.ics.uci.edu, together with all the data sets used here and any additional information. All three predictors achieve performance superior or comparable to the best existing predictors. Solubility and octanol/water partition coefficient prediction seem close to optimal, while there is still room for improving the prediction of melting points.

In comparing different kernels associated with different representations, the results obtained on the larger data sets confirm the trends previously observed in classification problems.[6] Overall, in the current chemoinformatics environment, the 2D spectral kernels tend to yield the best results, sometimes closely followed by 3D contact histogram kernels. Within the 2D kernels, as the size of the data sets increases, we expect kernels based on MinMax similarity with count feature vectors rather than binary features, and deeper paths rather than shallow paths, to perform the best. We also observe, with the 2D and 2.5D kernels, that the labeling scheme including element hybridization, or the more complex Sybyl labeling scheme, performs better than the more simple schemes. In 2.5D, varying the probe radius $\sqrt{\alpha}$ from 3.8 to 7.8 Å does not noticeably improve the results. In any case, we note that cross-validation procedures remain essential for assessing performance and for selecting optimal kernel parameters since, even within one class of kernels, there is no one-size-fits-all solution that is valid across all data sets.

These results may seem at first surprising with respect to the 1D kernels since SMILES strings contain exactly the

same information as 2D graphs of bonds. However, this can be explained by noticing that the spectral 1D kernels used here are based on substrings of the SMILES string. The decomposition of SMILES strings into segments of contiguous letters does not take into account the branching (parentheses) structure of SMILES strings and is a weaker decomposition than the decomposition of the 2D graph of bonds into all paths up to a given length originating from all the nodes. It is possible to use a richer set of substructures for SMILES that would take into account branching; however, this would most likely end up resembling, and hence be redundant with, the 2D kernel approach.

The superiority of 2D kernels with respect to 2.5D surface, 3D pharmacophore, and 3.5D or 4D conformer kernels may also seem surprising since the 3D structure of a molecule contains more information than its 2D graph of bonds. This may be explained, however, by the simple observation that the 3D structures of the molecules, which are required to compute these kernels, are not present in the data sets, nor is any information about stereochemistry. Here, to compute these kernels, the coordinates of the atoms in these structures are predicted, and the orientations around stereocenters are assigned arbitrarily. These 3D structure predictions are likely to introduce errors that affect the performance of molecular property predictors. Further support for this hypothesis is provided by the comparative results of 3D and 4D kernels on simple and challenging data sets, as discussed below.

Within the class of 3D kernels, the various 3D binning parameters tested do not seem to yield significant differences in prediction quality, and more complex kernels, such as the 3D pharmacophore kernel, do not give better results than the simpler 3D kernels based on contact histograms. In our experience, these more complex kernels applied to predicted structures are still outperformed by the best 2D kernels, as shown in several additional experiments that were carried on the four data sets used by Mahé et al.[23] for classification problems (Table 7). One exception is the alkane boiling point data set where the 3D contact histogram kernels yield the best predictions. This exception is consistent with the above hypothesis for explaining the relative weakness of 3D kernels because the 3D structure of noncyclic alkanes is relatively simple and accurately predicted by CORINA. With further progress in the size and quality of data set annotation, particularly 3D structure annotation and prediction, we can expect 3D and 4D kernels to become more useful, for instance, as an extra filter in virtual screening experiments.[47]

One important new observation derived from the 4D kernels is that averaging over several predicted 3D conformations per molecule in general leads to noticeable improvements. This is particularly true when averaging is applied to less well-performing kernels, involving triplets or quadruplets of atoms rather than pairs, or to very challenging data sets containing a high proportion of chiral molecules. In our data sets, the proportion of chiral molecules varies from 10% for the benzodiazepines to about 60% for the melting point data sets. Further support for the correlation in performance between 3D structure prediction and 3D kernel prediction is provided by the observation that 3D contact histogram kernels perform their best on the aqueous solubility and octanol−water partition coefficient data sets, which contain a low (about 20%) percentage of chiral molecules. In contrast, on the melting points data sets which contain a high (about 60%) percentage of chiral molecules, these 3D kernels are strongly outperformed by the 2D kernels. In the latter case, averaging over several conformations with the 4D contact histogram kernels noticeably improves the predictive performance of the 3D kernels. In addition, 4D kernels perform very well for classification tasks on the Mutag and PTC data sets presented in previous work.[6] Thus, in short, averaging kernels appears to be a simple but promising approach to address conformational issues which are highly relevant for three-dimensional ligand-based virtual screening, and which have not been addressed in previous attempts at defining 3D kernels.

Finally, while for relatively large data sets 2D kernels remain the method of choice, due to their simplicity, computational efficiency, and prediction accuracy, a performance comparison of the kernels on smaller data sets yields more variable results. On small data sets, with less than a few hundred molecules, 2D kernels do not always yield the best performance, and all performance assessments conducted on these small data sets must be considered with some caution because of the statistical fluctuations induced by small samples. On these small data sets, long feature vectors are likely to lead to overfitting; thus, short feature vectors, where features are more or less "hand-picked" using expert knowledge or feature-selection techniques, may still present some advantages.

ACKNOWLEDGMENT

Work supported by an NIH Biomedical Informatics Training grant (LM-07443-01) and NSF grants EIA-0321390 and IIS-0513376 to P.B., by the UCI Medical Scientist Training Program, and by a Harvey Fellowship to S.J.S. We would like also to acknowledge the OpenBabel project and OpenEye Scientific Software for their free software academic license and Drs. Chamberlin, Nowick, Piomelli, and Weiss for their useful feedback.

REFERENCES AND NOTES

(1) Agrafiotis, D. K.; Lobanov, V. S.; Salemme, F. R. Combinatorial Informatics in the Post-Genomics Era. *Nat. Rev. Drug Discovery* **2002**, *1*, 337−346.
(2) Lipinski, C.; Hopkins, A. Navigating Chemical Space for Biology and Medicine. *Nature* **2004**, *432*, 855−861.
(3) Dobson, C. M. Chemical Space and Biology. *Nature* **2004**, *432*, 824−828.
(4) Leach, A. R. *Molecular Modeling. Principles and Applications*; Prentice Hall: London, U. K., 2001.
(5) Xue, C. X.; Zhang, R. S.; Liu, H. X.; Liu, M. C.; Hu, Z. D.; Fan, B. T. Support Vector Machines-Based Quantitative Structure−Property Relationship for the Prediction of Heat Capacity. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1267−1274.
(6) Swamidass, S. J.; Chen, J.; Bruand, J.; Phung, P.; Ralaivola, L.; Baldi, P. Kernels for Small Molecules and the Prediction of Mutagenicity, Toxicity, and Anti-Cancer Activity. *Bioinformatics* **2005**, *21*, i359−368; Proceedings of the 2005 ISMB Conference.
(7) Sun, J.; Chen, H. F.; Xia, H. R.; Yao, J. H.; Fan, B. T. Comparative Study of Factor Xa Inhibitors Using Molecular Docking/SVM/HQSAR/3D-QSAR Methods. *QSAR Comb. Sci.* **2006**, *25*, 25−45.
(8) Schölkopf, B.; Smola, A. J. *Learning with Kernels, Support Vector Machines, Regularization, Optimization and Beyond*; MIT University Press: Cambridge, MA, 2002.
(9) Boser, B.; Guyon, I.; Vapnik, V. A Training Algorithm for Optimal Margin Classifiers. In *Proc. of the 5th Workshop on Comp. Learning Theory*; ACM Press: New York, 1992; Vol. 5.
(10) Cortes, C.; Vapnik, V. Support Vector Networks. *Mach. Learn.* **1995**, *20*, 1−25.
(11) Kimeldorf, G.; Wahba, G. Some Results on Tchebycheffian Spline Functions. *J. Math. Anal. Appl.* **1971**, *33*, 82−95.

(12) Schölkopf, B.; Herbrich, R.; Smola, A. J. *A Generalized Representer Theorem*; Technical Report NC-TR-00-081 for NeuroCOLT: London, 2000.
(13) Aizerman, M.; Braverman, E.; Rozonoér, L. Theoretical Foundations of the Potential Function Method in Pattern Recognition Learning. *Autom. Remote Control (Engl. Transl.)* **1964**, *25*, 821−837.
(14) Leslie, C.; Eskin, E.; Cohen, A.; Weston, J.; Noble, W. Mismatch String Kernels for Discriminative Protein Classification. *Bioinformatics* **2004**, *20*, 467−476.
(15) Lodhi, H.; Saunders, C.; Shawe-Taylor, J.; Cristianini, N.; Watkins, C. Text Classification Using String Kernels. *J. Mach. Learn. Res.* **2000**, *2*, 419−444.
(16) Vishwanathan, S. V. N.; Smola, A. J. Fast Kernels for Strings and Tree Matching. In *Adv. in Neural Information Processing Systems*; MIT: Great Britain, 2003; Vol. 15.
(17) Flower, D. R. On the Properties of Bit String-Based Measures of Chemical Similarity. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 378−386.
(18) Raymond, J. W.; Willett, P. Effectiveness of Graph-Based and Fingerprint-Based Similarity Measures for Virtual Screening of 2D Chemical Structure Databases. *J. Comput.-Aided Mol. Des.* **2001**, *16*, 59−71.
(19) Weininger, D.; Weininger, A.; Weininger, J. L. SMILES. 2. Algorithm for Generation of Uniques SMILES Notation. *J. Chem. Inf. Comput. Sci.* **1989**, *29*, 97−101.
(20) James, C. A.; Weininger, D.; Delany, J. Daylight Theory Manual. Available at http://www.daylight.com/dayhtml/doc/theory/index.html (accessed Feb 2007).
(21) Edelsbrunner, H.; Mücke, E. P. Three-Dimensional Alpha Shapes. *ACM Trans. Graphics* **1994**, *13*, 43−72.
(22) Wang, R.; Lu, Y.; Wang, S. Comparative Evaluation of Eleven Scoring Functions for Molecular Docking. *J. Med. Chem.* **2003**, *46*, 2287−2303.
(23) Mahé, P.; Ralaivola, L.; Stoven, V.; Vert, J.-P. The Pharmacophore Kernel for Virtual Screening with Support Vector Machines. *J. Chem. Inf. Model.* **2006**, *46*, 2003−2014.
(24) Perret, J.-L.; Mahé, P. ChemCpp User Guide. chemcpp.sourceforge.net/doc/chemcpp_user-guide.pdf (accessed Feb. 2007).
(25) Mahé, P.; Ueda, N.; Akutsu, T.; Perret, J.-L.; Vert, J.-P. Graph Kernels for Molecular Structure−Activity Relationship Analysis with Support Vector Machines. *J. Chem. Inf. Model.* **2005**, *45*, 939−951.
(26) Morgan, H. L. The Generation of Unique Machine Description for Chemical Structures - A Technique Developed at Chemical Abstracts Service. *J. Chem. Doc.* **1965**, *5*, 107−113.
(27) Sadowski, J.; Gasteiger, J.; Klebe, G. Comparison of Automatic Three-Dimensional Model Builders Using 639 X-Ray Structures. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 1000−1008.
(28) Gasteiger, J.; Sadowski, J.; Schuur, J.; Selzer, P.; Steinhauer, L.; Steinhauer, V. Chemical Information in 3D-Space. *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 1030−1037.
(29) Dietterich, T.; Lathrop, R.; Lozano-Perez, T. Solving the Multiple Instance Problem with Axis-Parallel Rectangles. *Artif. Intell.* **1997**, *89*, 31−71.
(30) Ray, S.; Page, D. Multiple Instance Regression. In *Proc. 18th Int. Conf. on Mach. Learn*, Williamstown, MA, June 28−July 1, 2001; Morgan Kaufman Publishers; San Francisco, CA, 2001.
(31) Cheung, P.-M.; Kowk, J. T. A Regularization Framework for Multiple-Instance Learning. In *Proc. 23rd Int. Conf. on Mach. Learn*; Pittsburgh, PA, June 25−29, 2006; ACM Press: New York, 2006.
(32) Swamidass, S. J.; Baldi, P. Correcting Fingerprint Similarity Measures to Improve Chemical Retrieval. Submitted, **2006**.
(33) Ralaivola, L.; Swamidass, S. J.; Saigo, H.; Baldi, P. Graph Kernels for Chemical Informatics. *Neural Networks* **2005**, special issue on Neural Networks and Kernel Methods for Structured Domains. In press.
(34) Chang, C. C.; Lin, C. J. LIBSVM: A Library for Support Vector Machines.. Available at http://www.csie.ntu.edu.tw/cjlin/libsvm (accessed Feb 2007).
(35) Hobohm, U.; Scharf, M.; Schneider, R.; Sander, C. Selection of Representative Data Sets. *Protein Sci.* **1992**, *1*, 409−417.
(36) Hadjipavlou-Litina, D.; Hansch, C. Quantitative Structure−Activity Relationship of the Benzodiazepines. A Review and Reevaluation. *Chem. Rev.* **1994**, *94*, 1483−1505.
(37) Micheli, A.; Sperduti, A.; Starita, A.; Bianucci, A. M. Analysis of the Internal Representations Developed by Neural Networks for Structures Applied to Quantitative Structure−Activity Relationship Studies of Benzodiazepines. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 202−218.
(38) Cherqaoui, D.; Villemin, D. Use of Neural Network to Determine the Boiling Point of Alkanes. *J. Chem. Soc., Faraday Trans.* **1994**, *90*, 97−102.
(39) Micheli, A.; Sperduti, A.; Starita, A.; Biancucci, A. M. A Novel Approach to QSPR/QSAR Based on Neural Networks for Structures. In *Soft Computing Approaches in Chemistry*; Cartwright, H., Sztandera, L. M., Eds.; Springer-Verlag: Heidelberg, Germany, 2003.
(40) Bergström, C.; Norinder, U.; Luthman, K.; Artursson, P. Molecular Descriptors Influencing Melting Point and Their Role in Classification of Solid Drugs. *J. Chem. Inf. Model.* **2003**, *43*, 1177−1185.
(41) 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.
(42) Fröhlich, H.; Wegner, J. K.; Zell, A. Towards Optimal Descriptor Subset Selection with Support Vector Machines in Classification and Regression. *QSAR Comb. Sci.* **2004**, *23*, 313−318.
(43) Delaney, J. S. ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1000−1005.
(44) Wang, R.; Fu, Y.; Lai, L. A New Atom-Additive Method for Calculating Partition Coefficients. *J. Chem. Inf. Comput. Sci.* **1997**, *37*, 615−621.
(45) Karthikeyan, M.; Glen, R.; Bender, A. General Melting Point Prediction Based on a Diverse Compound Data Set and Artificial Neural Networks. *J. Chem. Inf. Model.* **2005**, *45*, 581−590.
(46) Breindl, A.; Beck, B.; Clark, T. Prediction of the n-Octanol/Water Partition Coefficient, logP, Using a Combination of Semiempirical MO-Calculations and a Neural Network. *J. Mol. Model.* **1997**, *3*, 142−155.
(47) Lin, T.; Melga, M.; Swamidass, S. J.; Purdon, J.; Tseng, T.; Gago, G.; Kurth, D.; Baldi, P.; Gramajo, H.; Tsai, S. Structure-Based Inhibitor Design of AccD5, an Essential Acyl-CoA Carboxylase Carboxyltransferase Domain of Mycobacterium Tuberculosis. *Proc. Natl. Acad. Sci. U.S.A.* **2006**, *103*, 3072−3077.

CI600397P