Gram-Schmidt Kernels Applied to Structure Activity Analysis for Drug Design

Huma Lodhi
Department of Computing
Imperial College, University of London
180 Queen's Gate, London SW7 2BZ, UK
hml@doc.ic.ac.uk

Yike Guo
Department of Computing
Imperial College, University of London
180 Queen's Gate, London SW7 2BZ, UK
yg@doc.ic.ac.uk

ABSTRACT

We introduce a novel approach for structure activity relationship analysis based on the use of a special kernel. The kernel efficiently performs Gram-Schmidt orthogonalisation in a kernel defined feature space.

We show that support vector machines in conjunction with Gram-Schmidt kernel, a recent method to extract features, can be adopted successfully to predict the inhibition of dihydrofolate reductase by pyrimidines. We show that the feature space generated by Gram-Schmidt kernel does capture enough information, to the extent that it can outperform state-of-the-art systems. We also show that support vector machines in conjunction with Gram-Schmidt kernels can be applied successfully to predict the inhibition of dihydrofolate reductase by triazines. A preliminary experimental comparison of performance of the kernel with standard kernels and decision trees is made showing encouraging results.

Keywords

drug discovery, structure activity relationship analysis, Gram-Schmidt kernel, support vector machines.

1. INTRODUCTION

Millions of people are suffering form fatal diseases such as cancer, AIDS, and many other bacterial and viral illnesses. The key issue now is how to design life-saving and cost-effective drugs so that the diseases can be cured and prevented. It would also enable the provision of medicines in developing countries where approximately 80% of the world population lives.

Structure activity relationship (SAR) analysis is a key drug discovery task. It is based on the assumption that chemical structure and activity of compounds are correlated. SAR is the task of predicting the activity of new compounds by observing the structure of the compound. The activity of a compound can be biological activity, chemical reactivity and toxicity. A new compound is assigned an activity by conducting qualitative (q) or quantitative (Q) structure activity relationships analysis. In other words QSAR analysis is a regression problem whereas qSAR analysis is a classification task.

In this paper we introduce a novel methodology based on the use of a special kernel [4] to predict the qualitative biological activity of a new compound. SAR analysis plays a crucial role in the design and development of drugs. It is used to select a subset of important molecules from a huge sea of molecules, hence forming a small library of useful molecules. An important factor in SAR analysis is the prediction of new compounds with low probability of error, as the false prediction can be costly and result in loss of useful information.

Kernel methods (KM) are class of learning algorithms that give state-of-the-art performance. The support vector machine (SVM) [3; 17] is a well known example. The building block of these methods is an entity known as the kernel. The non-dependence of theses methods on the dimensionality of the feature space and the flexibility of using any kernel make them a good choice for different predictive modelling especially for SAR analysis. KM maps the vectorial data $d_1, \ldots, d_n \in D$ into some higher dimensional feature space and trains a linear classifier in this higher dimensional space. The kernel trick provides an efficient way to construct such a classifier by providing an efficient method of computing the inner product between mapped instances in the feature space. One does not need to represent the input instances explicitly in the feature space. The kernel function computes the inner product by implicitly mapping the instances to the feature space.

A number of learning methods have been applied to SAR analysis including neural networks [6] and decision trees [8]. There also exists special methods such as Partial Least Squares (PLS) [7]. The application of kernel methods in SAR analysis has been pioneered by Burbidge et al. [2] and successively explored by others [5]. Note that Burbidge et al. formulated a classification problem whereas Demiriz et al. formulated a regression problem. Learning techniques have been applied to compute new features to enhance the prediction of activity of a compound. New features have been computed by applying inductive logic programming in [16].

In this paper we introduce a radically different approach to extract features for SAR analysis. We introduce Gram-Schmidt kernels [4] (GSK) to compute features for structure activity prediction. This method incorporates more information into a kernel matrix. In other words the computed features are more informative and useful. We applied this approach to drug design with qSAR. We show that the computed features improve the generalisation performance of an SVM. We applied the method to predict the inhibition of dihydrofolate reductase by pyrimidines. We compare our methodology with SVM-RBF, artificial neural networks,

radial basis function networks and C5.0 (results extracted from [2]) demonstrating that the approach delivers state-of-the-art performance and can outperform all of the above mentioned techniques. Furthermore, we conducted another set of experiments to predict the inhibition of dihydrofolate reductase by triazines. The preliminary experimental comparison of the performance of the kernel with the standard kernels and decision trees shows encouraging results. This shows the effectiveness of our methodology in SAR analysis.

2. KERNEL METHODS

This section reviews the main ideas behind Support Vector Machines (SVMs) (a well known example of the kernel methods) and kernel functions. SVMs are a class of algorithms that combine the principles of statistical learning theory with optimisation techniques and the idea of a kernel mapping. They were proposed in 1992 [1]. An SVM is provided with set S of n training instances of the form

$$\{(\mathbf{d}_1, c_1), \ldots, (\mathbf{d}_n, c_n)\}.$$

Here $\mathbf{d}_i$ are the instances, which live in the instance space $\mathbf{D}$ and $c_i = c(\mathbf{d}_i)$ are the class labels, categories or targets. For binary classification $c_i \in \{-1, +1\}$, otherwise $c_i \in \{1, 2, \ldots, k\}$. The learning process of these methods consists of the following stages:

- Map the input data into some higher dimensional space through a non-linear mapping ϕ. The mapped space is known as the feature space F and the mapping is given by

  $$\phi : \mathbf{D} \rightarrow F.$$

  The mapping ϕ may not be known explicitly but be accessed via the kernel function described below.

- Construct a linear classifier f in the feature space as given by

  $$f(d) = \langle \mathbf{w}, \phi(\mathbf{d}) \rangle + b.$$

  Here $\mathbf{w}$ is the weight vector learned during the training phase. This weight vector is a linear combination of training instances. In other words

  $$\mathbf{w} = \sum_{i=1}^{n} \alpha_i c_i \phi(\mathbf{d}_i).$$

  Substituting the value of $\mathbf{w}$

  $$f(\mathbf{d}) = \sum_{i=1}^{n} \alpha_i c_i \langle \phi(\mathbf{d}_i), \phi(\mathbf{d}) \rangle + b.$$

  Hence the classifier is constructed only using the inner products between the mapped instances.

In other words, SVMs are based on the idea of constructing maximal margin hyperplane in the feature space. This unique hyperplane separates the data into two sets with maximum margin. Figure 1 shows a maximal margin hyperplane. The non-dependence of the solution on the dimensionality of the space where the separation takes place makes it possible to work in very high dimensional spaces without overfitting.

We now briefly describe a kernel function. A function that calculates the inner product between mapped instances in a

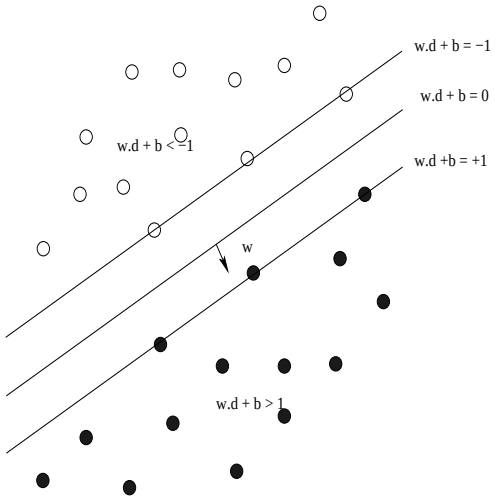

Figure 1: A maximal margin hyperplane

feature space is a kernel function, that is for any mapping $\phi : D \rightarrow F$, $K(d_i, d_j) = \langle \phi(d_i), \phi(d_j) \rangle$ is a kernel function. Note that the kernel computes this inner product by implicitly mapping the instances to the feature space. The mapping ϕ transforms an n dimensional instance into an N dimensional feature vector.

$$\phi(d) = (\phi_1(d), \ldots, \phi_N(d)) = (\phi_i(d)) \quad \text{for } i = 1, \ldots, N$$

The explicit extraction of features in a feature space generally has very high computational cost but a kernel function provides a way to handle this problem. The mathematical foundation of such a function was established during the first decade of twentieth century [14]. A kernel function is a symmetric function,

$$K(d_i, d_j) = K(d_j, d_i), \quad \text{for } i, j = 1, \ldots, n.$$

The $n \times n$ matrix with entries of the form $K_{ij} = K(d_i, d_j)$ is known as the kernel matrix. A kernel matrix is a symmetric, positive definite matrix. It is interesting to note that this matrix is the main source of information for KMs and these methods use only this information to learn a classifier. There are ways of combining simple kernels to obtain more complex ones.

For example given a kernel K and a set of n vectors the polynomial construction is given by

$$K_{poly}(d_i, d_j) = (K(d_i, d_j) + c)^p$$

where p is a positive integer and c is a nonnegative constant. Clearly, we incur a small computational cost, to define a new feature space. The feature space corresponding to a degree p polynomial kernel includes all products of at most p input features. Hence polynomial kernels create images of the examples in feature spaces having huge numbers of dimensions.

Furthermore, Gaussian RBF kernels define feature space with infinite number of dimension and it is given by

$$K_{gauss}(d_i, d_j) = \exp\left(\frac{-\|d_i - d_j\|^2}{2\sigma^2}\right)$$

A Gaussian kernel allows an algorithm to learn a linear classifier in an infinite dimensional feature space.

Require: A kernel k, training set $S = \{(d_1, c_1), \ldots, (d_n, c_n)\}$ and number T
for $i = 1$ to n **do**
$\text{norm2}[i] = k(\mathbf{d}_i, \mathbf{d}_i)$;
end for
for $j = 1$ to T **do**
$i_j = \text{argmax}_i(\text{norm2}[i])$;
$\text{index}[j] = i_j$;
$\text{size}[j] = \sqrt{\text{norm2}[i_j]}$;
for $i = 1$ to n **do**
$$\text{feat}[i,j] = \frac{\left(k(\mathbf{d}_i, \mathbf{d}_{i_j}) - \sum_{t=1}^{j-1} \text{feat}[i,t] * \text{feat}[i_j,t]\right)}{\text{size}[j]};$$
$\text{norm2}[i] = \text{norm2}[i] - \text{feat}(i,j) * \text{feat}(i,j)$;
end for
end for
return $\text{feat}[i,j]$ as the j-th feature of input i;
To classify a new example $\mathbf{d}$:
for $j = 1$ to T **do**
$$\text{newfeat}[j] = \frac{\left(k(\mathbf{d}, \mathbf{d}_{i_j}) - \sum_{t=1}^{j-1} \text{newfeat}[t] * \text{feat}[i_j,t]\right)}{\text{size}[j]};$$
end for
return $\text{newfeat}[j]$ as the j-th feature of the example $\mathbf{d}$;

Figure 2: The GSK algorithm.

3. GRAM-SCHMIDT KERNELS FOR STRUCTURE ACTIVITY PREDICTION

In this section we describe our adaptation of Gram-Schmidt kernels (GSK) [4] to predict qualitative biological activity for drug design. GSK is a recently proposed method for extracting features in the kernel defined feature space. It is based on the idea of building a more informative kernel matrix as compared to the standard kernels. In this way, GSK defines a more informative feature space and the examples that live in a physiochemical space are mapped into a highly informative space. This technique makes the incorporation of information into a kernel matrix feasible and attractive. According to this technique a set of features can be computed in time proportional to $O(Tn^2)$, where T is the number of features required to create an effective feature space. This technique significantly speeds up the computation of feature space.

GSK is based on Gram-Schmidt decomposition that builds the projection as the span of a subset of (the projections of) a set of k training instances. These are selected by performing a Gram-Schmidt orthogonalisation of the training vectors in the feature space. Hence, once a vector is selected the remaining training points are transformed to become orthogonal to it. The next vector selected is the one with the largest residual norm. The whole transformation is performed in the feature space using the kernel mapping to represent the vectors obtained. GSK is an iterative procedure that greedily selects a training instance at each iteration and extracts features. At each iteration the criterion for selecting an instance is the maximum norm. Figure 2 gives complete pseudo-code for extracting the features in the kernel defined feature space. This procedure has been termed GSK algorithm.

The GSK algorithm takes a set $S = \{(\mathbf{d}_1, c_1), \ldots, (\mathbf{d}_n, c_n)\}$ of n training instances. As input an underlying kernel function and number T are also fed to the algorithm. The number T specifies the number of dimension of the new informative feature space.

The algorithm starts by measuring the norm of each instance. At iteration j the training instance with maximum norm is chosen and we denote its index by i_j. A new jth feature is extracted for the ith training instance

$$\text{feat}[i,j] = \frac{\left(k(\mathbf{d}_i, \mathbf{d}_{i_j}) - \sum_{t=1}^{j-1} \text{feat}[i,t] * \text{feat}[i_j,t]\right)}{\text{size}[j]},$$

where $\text{size}[j]$ is the residual norm of the chosen example. This process is repeated in the feature space T times, where T is the chosen dimensionality of the feature space. Finally the instances are transformed into a new T dimensional space. The feature for a new example is extracted by projecting it into the space obtained by the Gram-Schmidt orthogonalisation of k training vectors. It is given by

$$newfeat[j] = \frac{\left(k(\mathbf{d}, \mathbf{d}_{i_j}) - \sum_{t=1}^{j-1} \text{newfeat}[t] * \text{feat}[i_j,t]\right)}{\text{size}[j]}$$

for $j = 1 \ldots T$.

3.1 Choosing Error/Margin Parameter and Underlying Kernel Function

We employed GSK in conjunction with an SVM in qSAR data analysis. The goal of a classifier is to predict the inhibition of dihydrofolate reductase by pyrimidines with low probaility of error (task1) and to predict the inhibition of dihydrofolate reductase by triazines with low probability of error (task2). We first describe the selection of parametes for task1.

We can view the learning process of an SVM with GSK as comprising two stages. In the first stage highly informative features are extracted in a kernel induced feature space. In the second stage a soft margin hyperplane classifier is trained. In order to train a linear SVM classifier we used SVM^{light} [11]. The generalisation performance of an SVM can be controlled by the free parameter C. It is a trade off parameter between margin maximisation and error. For the underlying kernel function we employed a Gaussian RBF. The effectiveness of a structure activity prediction system based on GSK can be influenced by the free parameter σ. Note that for each new value of this parameter, we obtain a new kernel and in turn the resultant feature space contains new information. Highly informative features can be computed by using an optimal value of σ. We now have two tunable parameters σ (for underlying kernel function) and C (for SVM). In order to choose an optimal value of C a range of values of C were selected. The set is given by $\{1, 10, 100\}$. Similarly a range of values of σ were selected based on a heuristic method [10] . The set is given by $\{\sigma, 2\sigma, 3\sigma, 4\sigma\}$, where σ is the median Euclidean distance between a positive instance and its its nearest negative instance in the training set. We mapped the data into a feature space defined by GSK using each value of σ in the set. Once we have computed a feature space, we trained linear SVMs using different values of C. The optimal values of C and σ which minimise the upper bound on generalisation error [12] were chosen. Note that we prefer higher values of σ in a scenario where more than one values of σ give approximately same estimate. Table 1 illustrates the behaviour of

SVM in conjunction with GSK for different values of σ in 128 dimensional feature space ($T = 128$).

σ	2σ	3σ	4σ
77.52	85.74	84.65	80.5

Table 1: Generalisation (Accuracy) of SVM in conjunction with GSK for different values of σ for fold 1 of dataset 1.

We now describe the selection of parametes for task2. We use the value of C which has shown high performance for task1. We also observed the support vectors to assess our selection. For the underlying kernel function, we employed linear and Gaussian RBF. As described earlier the performance of an SVM with GSK can be influenced by the parameter σ. We now have only one tunable parameter σ. In order to choose an optimal value of σ a range of values of γ $\{\gamma, 2\gamma, 3\gamma\, 4\gamma\}$ were selected. We set $\gamma = 1/2\sigma^2 = .1$. For comparison we applied an SVM with standard kernels and C4.5 [15]. We now explain the selection of parameters for an SVM with standard kernels. As standard kernels we applied linear and Gaussian kernels. Note that we applied the same methodology to set the parameter C and σ as described (for an SVM with GSK). We trained SVM classifier using SVM^{light} [11]. We used C4.5 with default parameters by keeping pruning on.

4. DATASETS

We performed structure activity relationship analysis by conducting experiments on UCI datasets. We selected two collections described below.

4.1 Dataset 1

In order to study the performance of the proposed methodology we chose benchmark dataset described in [13; 2] in detail. In this problem the aim of a learning algorithm is to predict the inhibition of dihydrofolate reductase by pyrimidines with low probability of error. In order to obtain a solution for QSAR a regression problem is solved, as QSAR analysis are generally regression problems. This problem has been converted into a binary classification task by considering a greater activity relationship between pairs of compounds. In this way each instance in the dataset is assigned an integer label. The dataset contains 55 compounds that are divided into 5-fold cross-validation series. The examples in the dataset are in vectorial form. For each drug there are 3 positions of possible substitution and the number of attributes for each substitution position is 9. These attributes are namely, polarity, size, flexibility, hydrogen-bond donor, hydrogen-bond acceptor π donor, π acceptor, polarisability and σ effect. Each drug, now is described by 27 attributes. Two drugs for each examples in the dataset makes the dimensionality of the space 54.

4.2 Dataset 2

We conducted second set of experiments on a subset of a benchmark dataset. This dataset is described in detail in [9]. The dataset is used to predict the inhibition of dihydrofolate reductase by triazines. In order to reproduce cancer cells triazines inhibit dihydrofolate enzymes. The dataset comprises of 186 compounds that are organised into six folds. As described in the preceeding section the dataset has been converted into a binary classification dataset by considering a greater activity relationship between pairs of compounds. There are two drugs for each instance. The instances are labelled on the basis of greater activity relationship between two drugs. Each drug is characterised by 6 position of possible substitution. The number of physiochemical attributes tht can be substituted are 10. These attributes are namely polarity, polarisability, hydrogen-bond donar, hydrogen-bond acceptor, π donar, π acceptor, size, flexibility, σ effect and braching. In this way there are 60 attributes for each drug. Tow drugs for each instance make the total number of attributes 120. In other words instances live in a physiochemical space where dimensionality of space is 120. In order to analyse the performace of GSK in qSAR analysis we took a subet of first fold (computational reasons). The selected subset comprises 1300 examples. We selected randomly 80% of the data for training the classifier and 10% for evaluation. We repeated this process 10 times. Note that for this set of experiments we set T to 256.

5. EXPERIMENTAL RESULTS AND DISCUSSION

Algorithm	T					
	16	32	64	128	200	256
SVM-GSK	76.5	82.0	86.2	87.6	88.1	**88.8**

Table 2: The performance (accuracy) of SVM with GSK. The results are averaged over five cross-validation folds. The results show the influence of computed features on generalisation performance of SVM

In this section we present our experimental results that show that our adaptation of SVM-GSK is effective for qSAR analysis. The GSK algorithm requires a set of examples, an underlying kernel function and the number T that specifies the dimension of the feature space. The algorithm starts by calculating the norm of each example giving a norm vector. Of all the examples, one with maximum norm is chosen. Once, an example is selected, the algorithm focuses on extracting features relative to this example. Given that the features are extracted, the next action is to update the norm vector. This process is repeated for T number of times. At the end of feature extraction process, that is done in the feature space a set of training documents and a norm vector is obtained. By exploiting the information obtained in the training phase, the algorithm extract features for new examples.

We employed GSK in conjunction with an SVM on the datasets described in the preceeding section.

We conduted a set of experiments to predict the inhibition of dihydrofolate reductase by pyrimidines. We investigated the influence of computed features on the generalisation performance of an SVM. Furthermore we observed the affect of variability of dimensionality of space on performance. The results are shown in Table 2. The results are averaged over 5-cold cross validation folds. To accomplish our goals we started with a small dimensional feature space. We increased the dimensionality of the feature space in intervals by extracting more features. Our results demonstrates that our adaptation of GSK is very effective. The results also demonstrate the generalisation performance of an SVM

classifier varies with respect to the dimensions of the feature space. The higher values of accuracy demonstrate that the computed features are effective and informative. Table 3 shows the published results on the same dataset. These results are extracted from [2]. The reported results are averaged over five cross-validation folds. These results demonstrate that our methodology outperforms all the methods listed in Table 3. Our results show the effectiveness of an SVM learner when it manipulates the information encoded in a kernel matrix that contains more information than a standard kernel matrix. Note that the free parameter T can be tuned by adopting the strategy explained in section 3.1.

Algorithm	Accuracy
SVM-RBF	87.33
1-NN	83.62
NN (manual)	86.97
RBF	78.76
C5.0	81.30

Table 3: Reported accuracies extracted from

Furthermore, we performed preliminary experiments to predict the inhibition of dihydrofolate reductase by triazines. We conduted experiments on subset of dataset as described in 4.2. We studied the affect of computed features on the generalisation performance of an SVM. The results are shown in Table 4. The results are averaged over 10 random splits of the data. The average accuracy is given and note that this table also show the the standard deviations. Our results show the technique GSK computes infromative features. We compared the performance of GSK with other techniques. The results show that an SVM classifier in conjunction with GSK is slighly better than the generalistion performance of an SVM with standard kernel. It is worth noting that GSK outperforms C4.5. We would like to further extend this work

Algorithm	Kernel	Accuracy	
		Mean	SD
SVM	*linear*	83.03	2.99
	gaussian	88.04	2.30
	GSK_{linear}	83.88	2.20
	$GSK_{gaussian}$	**88.15**	1.11
C4.5		80.66	2.42

Table 4: The generalisation performance (Accuracy) of SVM with GSK, standard kernels (linear kernel and Gaussian RBF kernel) and C4.5. The results are averaged over 10 runs of the techniques. We also report standard deviation.

by performing an extensive experimental analysis on other datasets and for harder problems such as QSAR with GSK. We believe that features extracted using GSK will be useful and will lead to a successful QSAR analysis.

6. CONCLUSIONS

The paper has introduced a novel methodology based on a special kernel for SAR analysis. The performance of the Gram-Schmidt kernel was empirically tested for structure activity prediction by applying it to a publicly available datasets. This kernel can be used with any kernel-based learning system, for example in clustering, regression, ranking, etc. In this paper we have focused on qSAR analysis, using a Support Vector Machine.
We have shown that the feature space generated by GSK is more informative than the feature spaces generated by standard kernels. The extracted features are highly effective and informative. A system based on GSK can outperform state-of-the-art systems. The experiments indicate that GSK can provide an effective alternative to the standard kernels used in previous SVM applications for SAR analysis.

Acknowledgements

First author would like to thank John Shawe-Taylor for useful and valuable discussions.

7. REFERENCES

[1] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In D. Haussler, editor, *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pages 144–152, Pittsburgh, PA, July 1992. ACM Press.

[2] R. Burbidge, M. Trotter, B. Buxton, and S. Holden. Drug design by machine learning: Support vector machines for pharamaceutical data analysis. *Computers and Chemistry*, 26(1):5–14, 2001.

[3] N. Cristianini and J. Shawe-Taylor. *An introduction to Support Vector Machines*. Cambridge University Press, Cambridge, UK, 2000.

[4] N. Cristianini, J. Shawe-Taylor, and H. Lodhi. Latent semantic kernels. *Journal of Intelligent Information Systems*, 18(2/3):127–152, 2002. Special Issue on Automated Text Categorization.

[5] A. Demiriz, K. P. Bennett, C. M. Breneman, and M. J. Embrechts. Support vector machine regression in chemometrics. *Computing Science and Statistics*, 2001.

[6] J. Devillers. *Neurak Networks and Drug Design*. Academic Press, 1999.

[7] I. E. Frank and J. H. Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35(2), 1993.

[8] D. M. Hawkins, S. S. Young, and A. Rusinko. Analysis of a large structure activity data set using recursive partitoning. *Quanttitative Structure Activity Relationships*, 16:296–302, 1997.

[9] J. D. Hirst and R. D. Kingand M. J. E. Sternberg. Quantitative structure-activity relationships by neural networks and inductive logic programming: 2. the inhibition of dihydrofolate reductase by triazines. *Computer Aided Molecular Design*, 8:421–432, 1994.

[10] T. Jaakkola, M. Diekhans, and D. Haussler. A discriminative framework for detecting remote protein homologies. *Journal of Computational Biology*, 7(1,2):95–114, 2000.

[11] T. Joachims. Making large–scale SVM learning practical. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods — Support Vector Learning*, pages 169–184, Cambridge, MA, 1999. MIT Press.

[12] T. Joachims. Estimating the generalisation performance of a SVM efficiently. In P. Langley, editor, *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 431–438. Morgan Kaufmann, 2000.

[13] R. D. King, S. Muggelton, R. A. Lewis, and M. J. E. Sternberg. Drug design by machine learning: the use of inducyive logic programming to model the structure-activity relationships of timethoprim analogues binding to dihydrofolate reducates. *Proc. Natl. Acad. Sci. USA*, 89:11322–11326, 1992.

[14] J. Mercer. Functions of positive and negative type and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society London (A)*, 209:415–446, 1909.

[15] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

[16] A. Srinivasan and R. D. Ling. Feature construction with inductive logic programming: A study of quantitative prediction of biological activity aided by structural attributes. *Data Mining and Knowledge Discovery*, 3(1):37–57, 1999.

[17] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer Verlag, New York, 1995.