

An Evaluation of Some SVM Heuristics for Predicting Activity on pK_i Assays

Robert Burbidge

June 29, 2001

1 Introduction

An *agonist* is a signalling molecule which binds to a receptor inducing a conformational change which produces a response. An *antagonist* is a drug which attenuates the effect of an agonist. Antagonists may be competitive, non-competitive or uncompetitive. A *competitive* antagonist binds to a region of the receptor which overlaps the region bound by an agonist. The agonist and antagonist compete for the same binding site and cannot simultaneously occupy the receptor. The potency of a competitive antagonist is quantified by the equilibrium dissociation constant, K_B , as determined in a functional assay. This is the concentration of antagonist which would occupy 50% of the receptors at equilibrium. This may be determined by Schild analysis — a concentration-response curve is plotted for the agonist, in the presence of varying quantities of the antagonist. Theoretically, this value should be the same as the K_I value determined in a radioligand competition binding assay. In this assay the agonist is radio-labelled and is screened at only one concentration, usually below its equilibrium dissociation constant for the receptor, K_D . The specific level of binding of the agonist is then determined in the presence of a range of concentrations of a competing non-radioactive compound. The data for each competing ligand are usually fitted to a hyperbolic equation from which the $IC50$ can be determined. The $IC50$ is the concentration of antagonist required to reduce the specific binding of the radioligand by 50%. This is then converted to a K_I value by the Cheng-Prusoff equation

$$K_I = \frac{IC50}{1 + \frac{[L]}{K_D}}$$

where $[L]$ is the concentration of free radioligand used and K_D is its equilibrium dissociation constant for the receptor. Whereas the $IC50$ value for a compound may vary between experiments the K_I is an absolute value. Typically, the negative logarithm of this quantity, pK_I is reported as the measure of activity against the receptor. Since a ligand is screened over a limited range of concentrations it may not be possible to specify an exact pK_I , instead it may be reported as less than or greater than those limits implied by the concentration range.

Actual pK_I	SQ pK_I
<5.5	1
5.5-6.0	2
⋮	⋮
8.5-9.0	8
>9.0	9

Table 1: Rescaling of pK_I values to a semi-quantitative ranking.

2 Problem Description

The compounds analysed in this report were screened against 11 targets (receptors), of various classes, in competitive binding assays and the corresponding pK_I values calculated. Due to the limited range of concentrations in screening the pK_I values were rescaled to a semi-quantitative ranking as shown in table 1. In the remainder pK_I will refer to the semi-quantitative values. This resulted in activity values for 1416 compounds on the 11 screens (previously had been available only activity values for 581 compounds on 10 screens). Due to a data processing error one of the feature sets was not available for one of the compounds (SB-362439), so this compound was removed to leave 1415 compounds. To provide an idea of the diversity of the compounds, they were clustered in Daylight fingerprint space at 0.7 Tanimoto similarity. This gave 395 cluster centroids and 1167 analogues. The compounds were also clustered at 0.90, 0.95 and 0.99 Tanimoto similarity. Removing the original 395 centroids yielded sets of analogues of sizes 617, 786 and 917, respectively (all contained SB-362439).

The problem is to predict the pK_I values based on some set of descriptors. For the purposes of classification, a compound is treated as active if its actual pK_I is greater than 6, i.e., if the semi-quantitative pK_I is greater than 2. Two feature sets were available. The first is a set of commonly used physico-chemical descriptors including ‘pick-of-six’ descriptors previously found predictive at GSK, and one-dimensional descriptors such as molecular weight, CLOGP, number of hydrogen bond donors and acceptors, etc. The second is a set of generic 166-dimensional binary structural keys, proprietary to GSK.

3 Performance Analysis

To evaluate the performance of the various classification algorithms the data were partitioned randomly into training and test sets of sizes 395 and 1020 respectively. The proportion of active compounds in each of the training sets, π_{+1}^s is shown in table 3. The risk functional for classification is

$$\begin{aligned}
 R(\theta) &= \lambda(+1, -1)\pi_{-1}\Pr\{f(\mathbf{x};\theta) = +1|y = -1\} \\
 &+ \lambda(-1, +1)\pi_{+1}\Pr\{f(\mathbf{x};\theta) = -1|y = +1\},
 \end{aligned}
 \tag{1}$$

Assay	π_{+1}^s	Assay	π_{+1}^s
1	0.28	7	0.49
2	0.38	8	0.56
3	0.70	9	0.52
4	0.38	10	0.35
5	0.56	11	0.42
6	0.58		

Table 2: Proportion of active compounds (positives) in each of the training sets.

where π_k is the prior probability of class $k \in \{-1, +1\}$ and $\lambda(l, k)$ is the loss incurred for making prediction l when the true class is k . As many algorithms are adversely affected by unbalanced data it was decided to minimize the following risk functional

$$R(\theta) = \Pr\{f(\mathbf{x}; \theta) = +1 | y = -1\} + \Pr\{f(\mathbf{x}; \theta) = -1 | y = +1\}, \quad (2)$$

which amounts to setting the cost ratio, $\lambda(-1, +1)/\lambda(+1, -1)$, equal to π_{-1}/π_{+1} . In practice, the training set proportions π_k^s are used instead. Minimizing this risk functional is equivalent to minimizing

$$\frac{FP}{N} + \frac{FN}{P}$$

instead of

$$\frac{FP + FN}{N + P}$$

where, FP , FN are the number of false positives and false negatives, respectively, and N , P are the number of negatives and positives in the training set. The two approaches are equivalent for balanced data.

4 Support Vector Classification

For each of the 11 screens an SVM was trained using the physico-chemical descriptors. Since the data are known to be non-linear a Gaussian RBF kernel $K(\mathbf{x}, \mathbf{z}) = \exp(-\|\mathbf{x} - \mathbf{z}\|^2/2\sigma^2)$ was used. The RBF width σ was set using Jaakkola’s heuristic, viz., the median separation of negative points to their nearest positive neighbour. This quantity is on the order of the separation between the two classes. For overlapping classes, however, this quantity is likely to be too small and result in a model with many support vectors that overfits the training data. The parameters affecting convergence were as follows for all SVM methods. The working set size for the decomposition routine was 10, the maximum number of iterations was set at 5000 to ensure that all experiments could be completed in a reasonable amount of time. The parameter C was varied in

$\{1, 10\}$ and chosen as that which minimized the expected risk, as estimated by a generalized version of Joachim’s $\alpha\xi$ estimator of the leave-one-out error.

The errors and corresponding risks on the training and test sets are shown in table 4, together with the area under the ROC curve (AUROC) on the test data. The ROC curve is generated by varying the threshold b in the SVM solution, $f(\mathbf{x}; \mathbf{w}, b) = \text{sgn}(\mathbf{w}^T \mathbf{x} + b)$.

pK_I	Error		Risk		AUROC
	Train	Test	Train	Test	
1	0.06	0.25	0.10	0.56	0.78
2	0.03	0.28	0.06	0.57	0.76
3	0.03	0.30	0.05	0.73	0.73
4	0.05	0.28	0.09	0.57	0.78
5	0.02	0.31	0.04	0.63	0.72
6	0.05	0.37	0.11	0.76	0.69
7	0.04	0.32	0.08	0.64	0.75
8	0.02	0.27	0.04	0.54	0.79
9	0.03	0.33	0.05	0.66	0.74
10	0.05	0.25	0.09	0.50	0.81
11	0.09	0.24	0.19	0.50	0.82

Table 3: Error rates and risk for SVM.

5 Adaptive Scaling

The performance of the learned classifier is highly sensitive to the width of the RBF kernel, σ . Jaakkola’s heuristic provides a reasonable rule-of-thumb for the scale of the margin band. Since the solution depends only on support vectors it seems reasonable that a better solution will be obtained if Jaakkola’s heuristic is applied only to support vectors. The support vectors are not known until training is complete. Instead, one can apply Jaakkola’s heuristic to the current set of support vectors every h_a iterations. The kernel widths are thus adaptively scaled during training. This heuristic is termed LAIKA (Locally Adaptive Iterative Kernel Approximation). For these experiments $h_a = 100$, in practice the final solution is not very sensitive to h_a . A fairly large value is used to speed up computation as the kernel matrix and gradient information must be updated when the kernel parameter is updated. For these data, which are expected to lead to dense models, this heuristic is not likely to make much difference. The error rates and risk for LAIKA are shown in table 5

pK_I	Error		Risk		AUROC
	Train	Test	Train	Test	
1	0.04	0.25	0.05	0.62	0.73
2	0.03	0.28	0.05	0.57	0.74
3	0.03	0.30	0.05	0.73	0.71
4	0.04	0.27	0.08	0.57	0.76
5	0.02	0.31	0.04	0.62	0.70
6	0.03	0.36	0.06	0.76	0.67
7	0.03	0.31	0.06	0.63	0.74
8	0.02	0.27	0.04	0.54	0.75
9	0.03	0.33	0.06	0.65	0.72
10	0.03	0.22	0.06	0.47	0.81
11	0.09	0.24	0.19	0.49	0.80

Table 4: Error rates and risk for LAIKA.

6 Automated Rejection

In the support vector machine solution all training points \mathbf{x}_i with $\alpha_i > 0$ appear in the solution. This includes *bounded* support vectors, for which $\alpha_i = C, 0 < \xi_i < 1$, which lie within the margin band, and training errors, for which $\alpha_i = C, \xi_i > 1$. Intuitively, it is not desirable for training errors to appear in the solution, especially not with maximal weight. One way to remove these points is simply to retrain on the subset of training data that were correctly classified. An online approximation to this is to remove points during training that have been misclassified for the last h_e iterations. These points are likely to become training errors. This heuristic is termed STAR (Sparsity Through Automated Rejection). In these experiments $h_e = 100$, this heuristic is sensitive to the choice of h_e . If chosen too small then too many points are rejected and the model overfits the data. If too large then the heuristic behaves very similarly to the SVM, as few points are rejected. The error rates and risk are shown in table 6.

7 Combined Approach

8 Conclusions and Further Work

References

B.D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, 1996.

pK_I	Error		Risk		AUROC
	Train	Test	Train	Test	
1	0.07	0.28	0.00	0.61	0.78
2	0.03	0.28	0.00	0.58	0.76
3	0.05	0.32	0.00	0.76	0.68
4	0.00	0.38	0.00	1.00	0.79
5	0.02	0.31	0.00	0.62	0.73
6	0.00	0.43	0.00	1.00	0.68
7	0.00	0.46	0.00	0.94	0.74
8	0.02	0.27	0.00	0.54	0.78
9	0.00	0.51	0.00	1.00	0.72
10	0.00	0.35	0.00	0.93	0.81
11	0.00	0.31	0.00	0.74	0.82

Table 5: Error rates and risk for STAR.

pK_I	Error		Risk		AUROC
	Train	Test	Train	Test	
1	0.00	0.28	0.00	1.00	0.77
2	0.03	0.28	0.00	0.58	0.76
3	0.00	0.32	0.00	1.01	0.67
4	0.00	0.38	0.00	1.00	0.59
5	0.00	0.42	0.00	0.86	0.73
6	0.00	0.43	0.00	1.00	0.55
7	0.00	0.49	0.00	0.99	0.57
8	0.03	0.28	0.00	0.57	0.77
9	0.00	0.48	0.00	0.98	0.58
10	0.00	0.36	0.00	1.00	0.63
11	0.00	0.60	0.00	1.00	0.61

Table 6: Error rates and risk for STAR+LAIKA.