An Adaptive Radial Basis Function Network Model for Statistical Detection

Mu Zhu

University of Waterloo

Acknowledgment

- Wanhua Su.
- Hugh A. Chipman.

Agenda

1. The detection problem.
2. Average precision.
3. Drug discovery and high throughput screening.
4. Methodology.
5. Radial basis function networks.
6. Support vector machines.
7. Results.
8. A statistical explanation.
9. Some ongoing work.

The Detection Problem

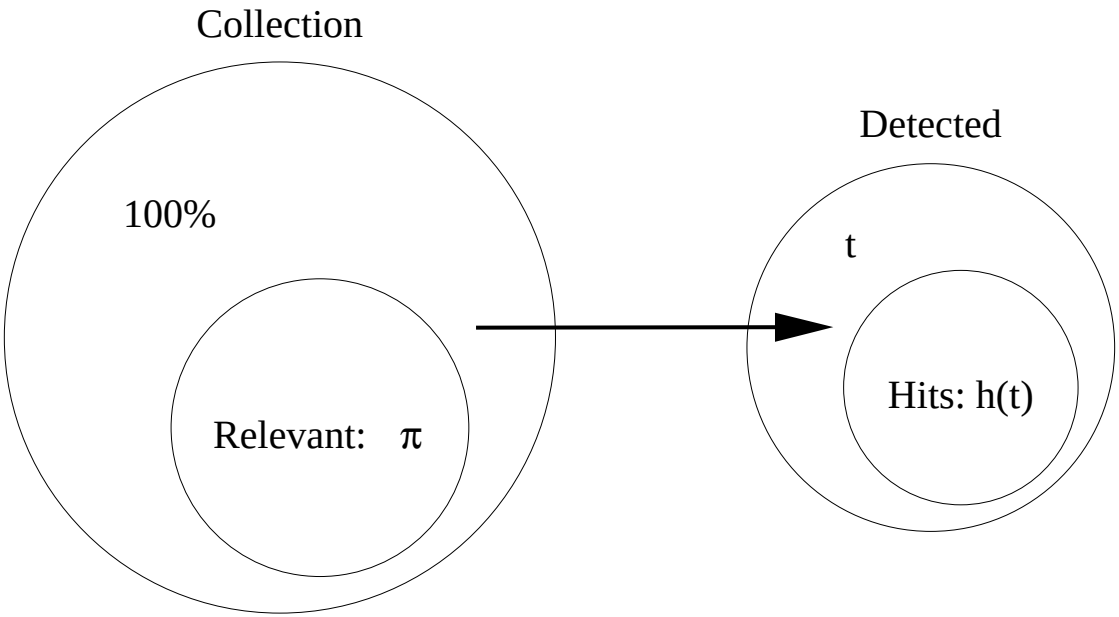


Figure 1: Illustration of a typical detection operation. A small fraction π of the entire collection $\mathcal{C}$ is of interest (relevant). An algorithm detects a fraction t from $\mathcal{C}$, out of which $h(t)$ is relevant.

The Typical Paradigm

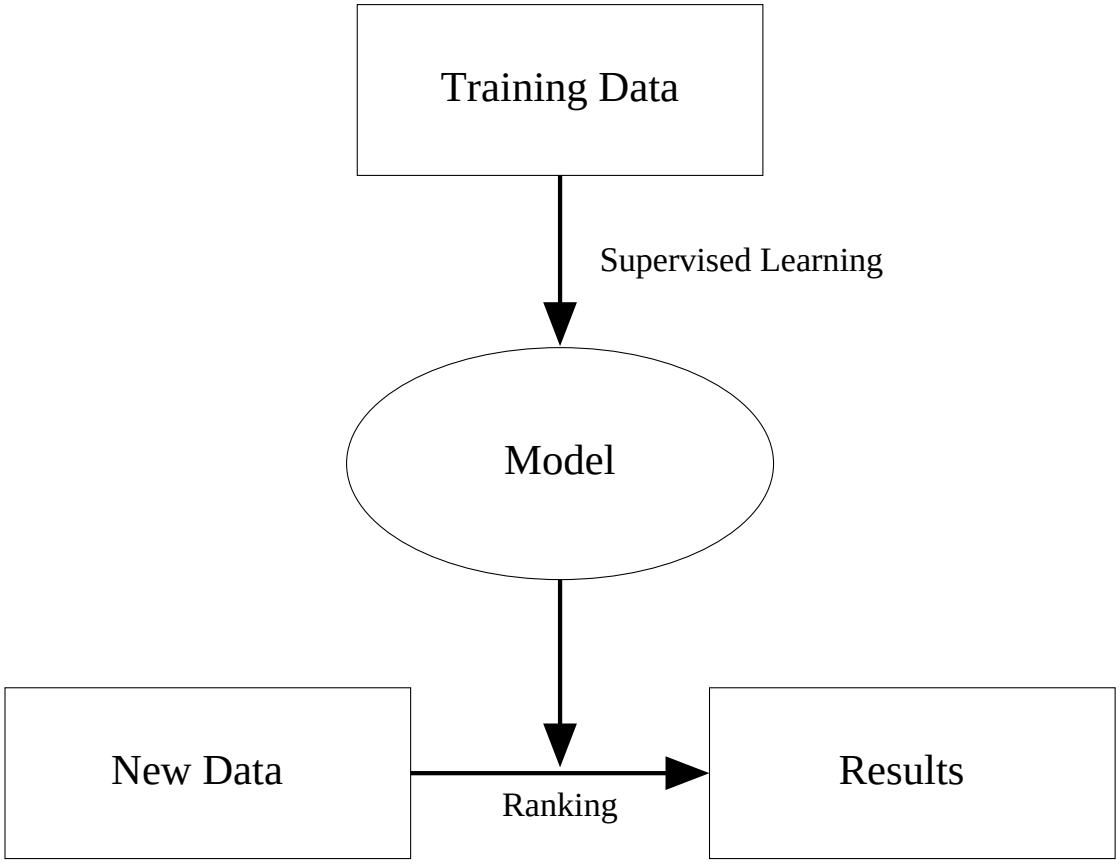


Figure 2: Illustration of the typical modelling and prediction process.

The Hit Curve

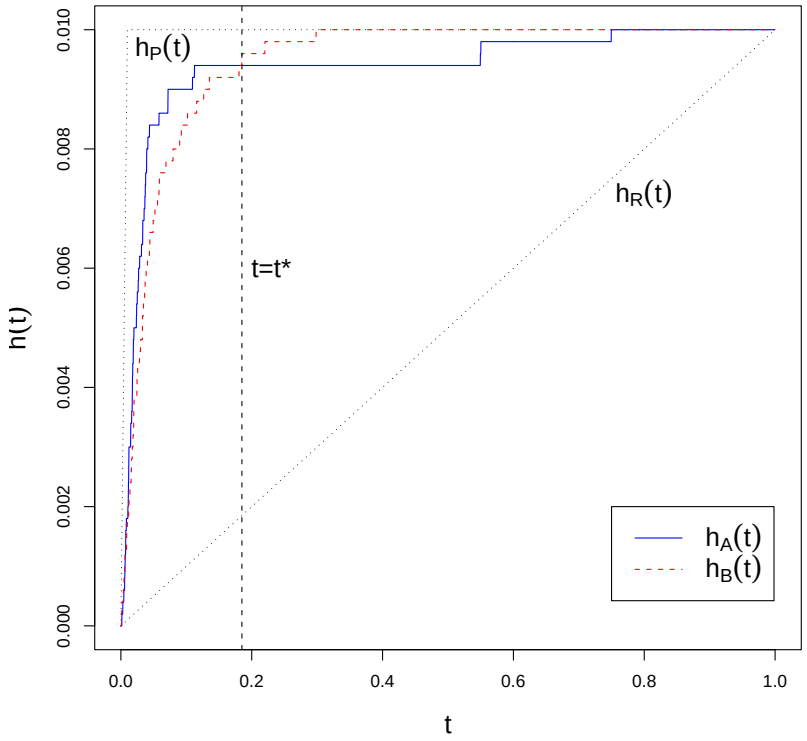


Figure 3: Illustration of some hit curves. Note that $h_A(t)$ and $h_B(t)$ cross each other at $t = t^*$; $h_P(t)$ is an ideal curve produced by a perfect algorithm; $h_R(t)$ corresponds to the case of random detection.

The Average Precision

Let $h(t)$ be the hit curve; let

$$r(t) = \frac{h(t)}{\pi} \quad \text{and} \quad p(t) = \frac{h(t)}{t}.$$

Then,

$$\text{Average Precision} = \int p(t)dr(t). \tag{1}$$

In practice, $h(t)$ takes values only at a finite number of points $t_i = i/n$, $i = 1, 2, ..., n$. Hence, the integral (1) is replaced with a finite sum

$$\int p(t)dr(t) = \sum_{i=1}^{n} p(i)\Delta r(i) \tag{2}$$

where $\Delta r(i) = r(i) - r(i-1)$.

A Simple Example

	Algorithm A			Algorithm B		
Item (i)	Hit	$p(i)$	$\Delta r(i)$	Hit	$p(i)$	$\Delta r(i)$
1	1	1/1	1/3	1	1/1	1/3
2	1	2/2	1/3	0	1/2	0
3	0	2/3	0	0	1/3	0
4	1	3/4	1/3	1	2/4	1/3
5	0	3/5	0	1	3/5	1/3

$$\text{AP(A)} = \sum_{i=1}^{5} p(i)\Delta r(i) = \left(\frac{1}{1} + \frac{2}{2} + \frac{3}{4}\right) \times \frac{1}{3} \approx 0.92.$$

$$\text{AP(B)} = \sum_{i=1}^{5} p(i)\Delta r(i) = \left(\frac{1}{1} + \frac{2}{4} + \frac{3}{5}\right) \times \frac{1}{3} = 0.70.$$

Drug Discovery Data

Original data from National Cancer Institute (NCI) with label added by GlaxoSmithKlein, Inc.

1. $n = 29,812$ chemical compounds, of which only 608 are active against the HIV virus.
2. $d = 6$ chemometric descriptors of the molecular structure, known as BCUT numbers.
3. Using stratified sampling, randomly split of the data to produce a training set and a test set (each with $n = 14,906$ and 304 active compounds).
4. Tuning parameters selected using 5-fold cross-validation on the training set, and compare performance on the test set.

High Throughput Screening (HTS)

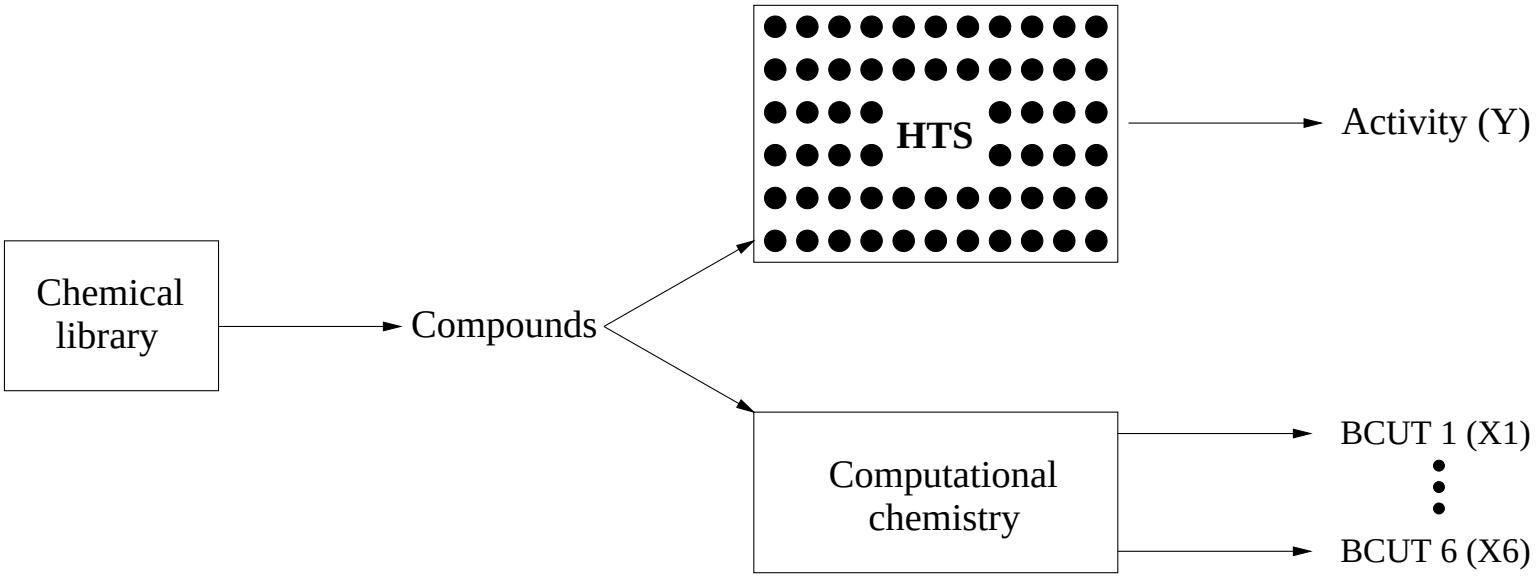


Figure 4: Illustration of the high throughput screening process. Based on Welch (2002).

Origin and Background of the Main Idea

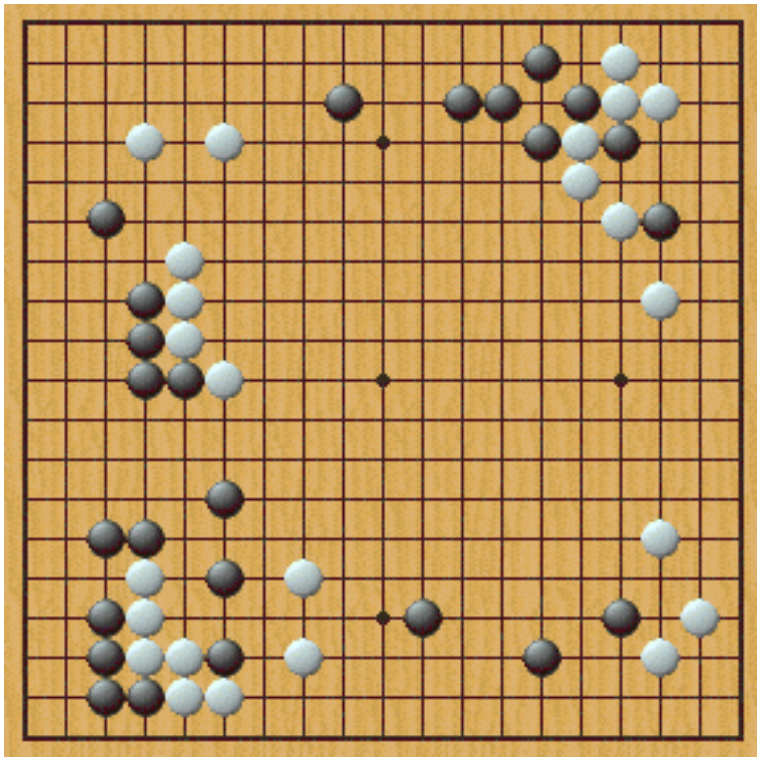


Figure 5: The ancient Chinese game of Go is a game in which each player tries to claim as many territories as possible on the board. Image taken from http://go.arad.ro/Introducere.html.

Key Ingredients

Definition 1. Let $\mathbf{x} \in \mathbb{R}^d$ be a training observation belonging to class 1; its *radius of influence* is defined as $\mathbf{r} = (r_1, r_2, ..., r_d)^T$ where

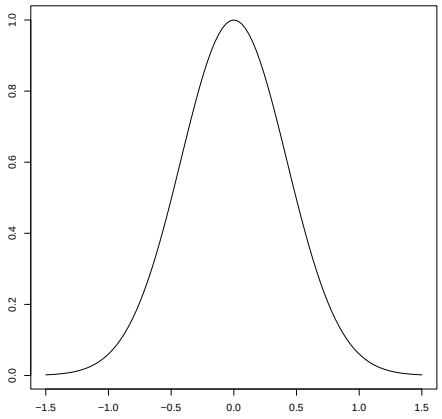
$$r_j = \frac{1}{K} \sum_{\mathbf{w} \in N(\mathbf{x}, K)} |x_j - w_j|$$

is the average distance in the j-th dimension between $\mathbf{x}$ and its K nearest class-0 neighbors. That is, every $\mathbf{w} \in N(\mathbf{x}, K)$ belongs to the background class.

Definition 2. $f(u)$ is called a *quasi kernel function* if $f(0) = 1$ and there exists a constant $c > 0$ such that $cf(u)$ is a regular kernel function, i.e., $\int cf(u)du = 1$.

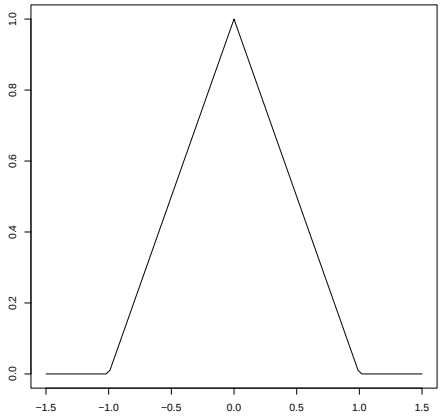
Some Quasi Kernels

Gaussian



$$f(u) = \exp\left(-\frac{u^2}{2}\right)$$

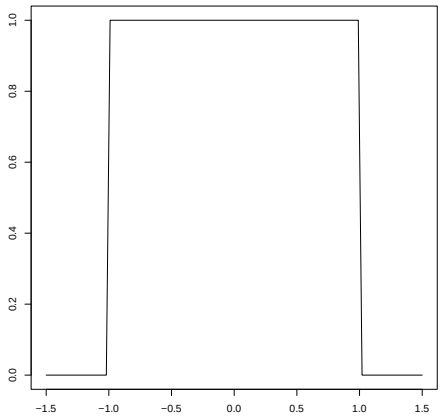
Triangular



$$f(u) = 1 - |u|$$

$$|u| \leq 1$$

Uniform



$$f(u) = 1$$

$$|u| \leq 1$$

Main Methodology

1. Given a new observation $\mathbf{z}$, each class-1 observation in the training data, $\mathbf{x}$, will cast a vote on $\mathbf{z}$ based on its radius of influence, $\mathbf{r}$:

$$v(\mathbf{z}; \mathbf{x}, \mathbf{r}) = \prod_{j=1}^{d} f\left(\frac{z_j - x_j}{\alpha r_j}\right)$$

   where $f(u)$ is a quasi kernel function and α, an extra global tuning parameter (to be explained later). Default setting: $\alpha = 1$.

2. The new observation will be ranked according to the average vote it receives:

$$F(\mathbf{z}) = \frac{\sum_{i=1}^{n} v(\mathbf{z}; \mathbf{x_i}, \mathbf{r_i}) I(y_i = 1)}{\sum_{i=1}^{n} I(y_i = 1)}.$$

3. Since only observations in the important but rare class are eligible to cast a vote, there is considerable computational saving (e.g., over K-NN).

Tuning Parameters

K: mild effects, insensitive; effect on the radius of influence not identical in every direction.

α: stronger effects; stretches or dampens the radius of influence identically in every direction.

Kernel Calibration

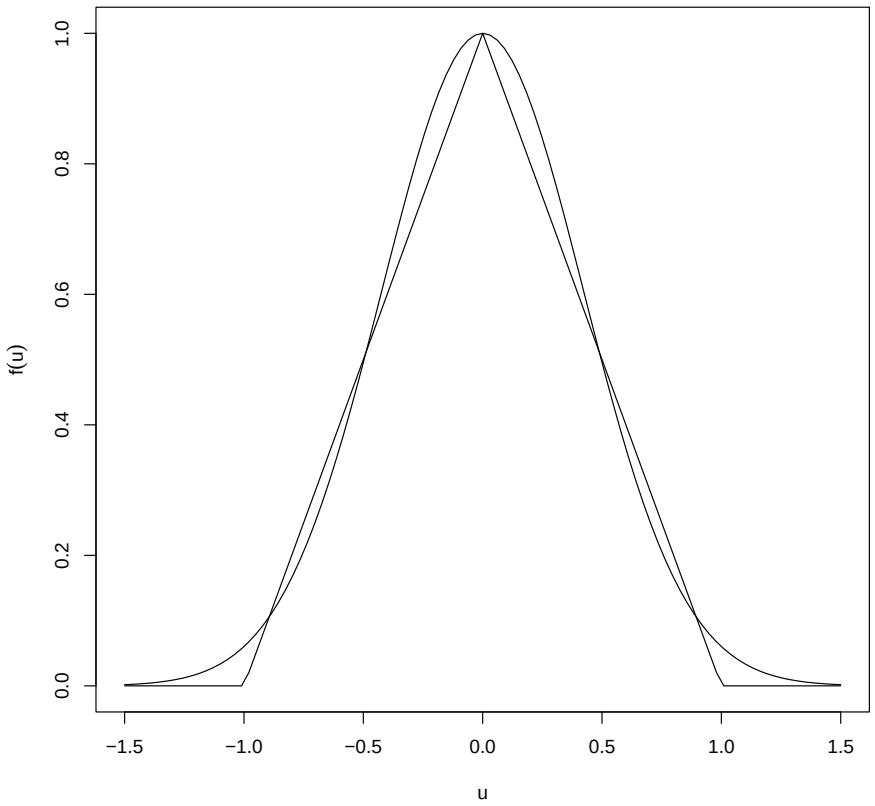


Figure 6: Calibrated quasi-kernels.

Effective radius of influence is different for the triangular and the Gaussian kernels. To make the comparisons easier, we calibrate as follows: Let

$$\begin{aligned} f(u) &= \exp\left(-\frac{u^2}{2\sigma^2}\right) \\ g(u) &= 1-|u|, \end{aligned}$$

set σ^2 to

$$\text{argmin} \int_{-1}^{1} \left(f(u)-g(u)\right)^2 du.$$

Optimal choice is $\sigma^2 \approx 0.178$.

Radial Basis Function Networks

- A radial basis function (RBF) network has the form:

$$f(\mathbf{x}) = \sum_{m=1}^{M} \beta_m K(\mathbf{x}; \boldsymbol{\mu}_m, \mathbf{r}_m),$$

  where $K(\mathbf{x}; \boldsymbol{\mu}, \mathbf{r})$ is a kernel function with center $\boldsymbol{\mu}$ and radius (or bandwidth) vector $\mathbf{r} = (r_1, r_2, ..., r_d)^T$.

- For example, a common choice of the kernel is

$$K(\mathbf{x}; \boldsymbol{\mu}, \mathbf{r}) = \prod_{j=1}^{d} \phi\left(x_j; \mu_j, r_j\right)$$

  where $\phi(x; \mu, r)$ is the density function for $\mathrm{N}(\mu, r^2)$.

Separating Hyperplanes

- Given $\mathbf{x}_i \in \mathbb{R}^d$, a hyperplane in $\mathbb{R}^d$ is characterized by

$$f(\mathbf{x}) = \boldsymbol{\beta}^T \mathbf{x} + \beta_0 = 0.$$

- Given $y_i \in \{-1, +1\}$ (two classes), a hyperplane is a separating hyperplane if there exists $c > 0$ such that

$$y_i(\boldsymbol{\beta}^T \mathbf{x}_i + \beta_0) \geq c \quad \forall i.$$

- A hyperplane can be reparameterized by scaling, e.g.,

$$\boldsymbol{\beta}^T \mathbf{x} + \beta_0 = 0 \quad \text{is the same as} \quad s(\boldsymbol{\beta}^T \mathbf{x} + \beta_0) = 0.$$

- A separating hyperplane satisfying

$$y_i(\boldsymbol{\beta}^T \mathbf{x}_i + \beta_0) \geq 1 \quad \forall i$$

(i.e., scaled so that $c = 1$) is sometimes called a canonical separating hyperplane (Cristianini and Shawe-Taylor 2000).

Separating Hyperplanes and Margins

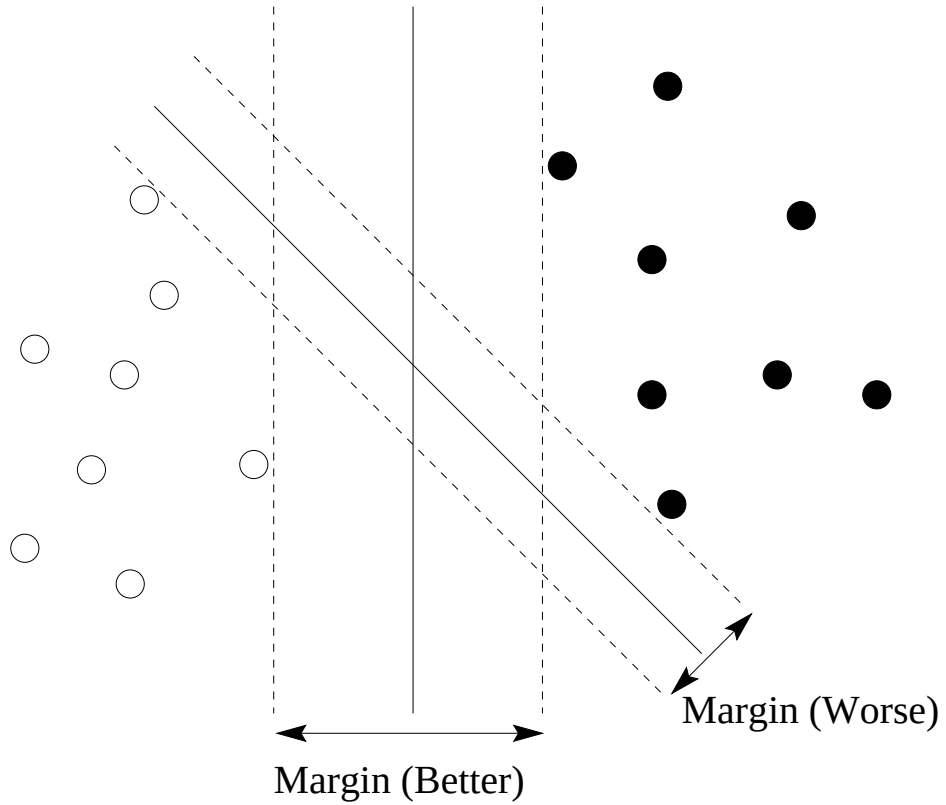


Figure 7: Two separating hyperplanes, one with a larger margin than the other.

The Support Vector Machine

- It can be calculated that a canonical separating hyperplanes has margin equal to $\frac{1}{\|\boldsymbol{\beta}\|}$.
- The support vector machine (SVM) finds a "best" (maximal margin) canonical separating hyperplane to separate the two classes (labelled $+1$ and -1) by solving

$$\min \quad \frac{1}{2}\|\boldsymbol{\beta}\|^2 + \gamma \sum_{i=1}^{n} \xi_i$$

$$\text{s.t.} \quad \xi_i \geq 0 \quad \text{and} \quad y_i(\boldsymbol{\beta}^T \mathbf{x}_i + \beta_0) \geq 1 - \xi_i \quad \forall i.$$

SVM: Characterizing the Solution

- The solution for $\boldsymbol{\beta}$ is characterized by

$$\hat{\boldsymbol{\beta}} = \sum_{i \in SV} \hat{\alpha}_i y_i \mathbf{x}_i,$$

  where $\hat{\alpha}_i \geq 0$ $(i = 1, 2, ..., n)$ are solutions to the dual optimization problem and SV, the set of "support vectors" with $\hat{\alpha}_i > 0$ strictly positive.

- This means the resulting hyperplane can be written as

$$\hat{f}(\mathbf{x}) = \hat{\boldsymbol{\beta}}^T \mathbf{x} + \hat{\beta}_0 = \sum_{i \in SV} \hat{\alpha}_i y_i \mathbf{x}_i^T \mathbf{x} + \hat{\beta}_0 = 0.$$

SVMs and RBF Networks

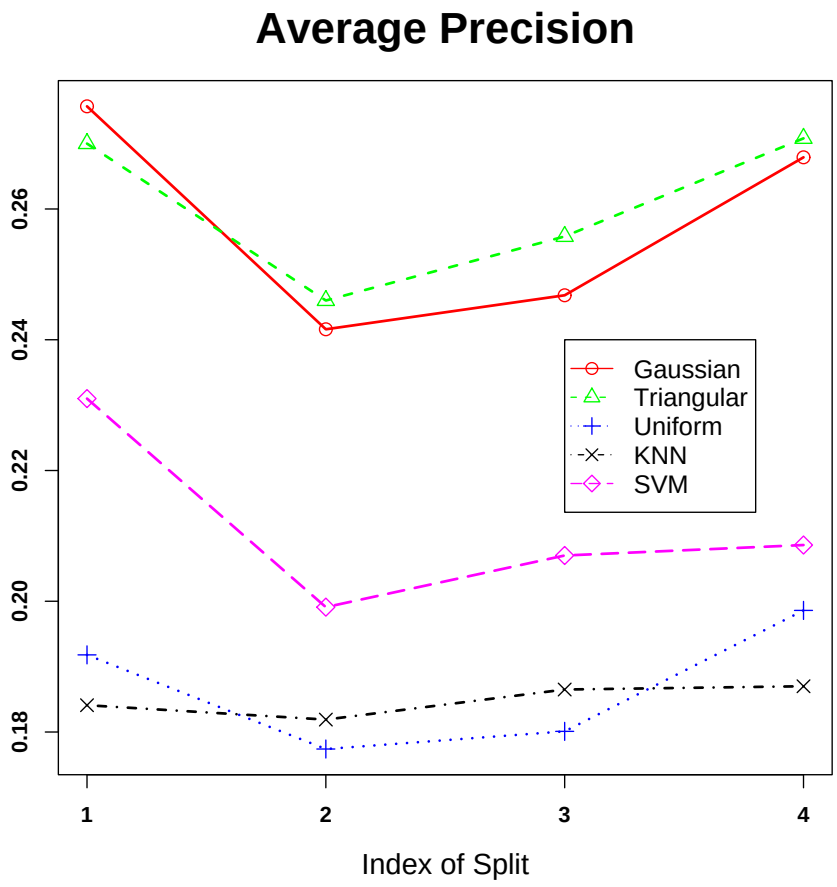
- Can replace the inner product $\mathbf{x}_i^T\mathbf{x}$ with a kernel function $K(\mathbf{x};\mathbf{x}_i)$ to get a nonlinear decision boundary:

$$\hat{f}(\mathbf{x}) = \sum_{i\in SV} \hat{\alpha}_i y_i K(\mathbf{x};\mathbf{x}_i) + \hat{\beta}_0 = 0.$$

  The boundary is linear in the space of $h(\mathbf{x})$ where $h(\cdot)$ is such that $K(\mathbf{u};\mathbf{v}) = \langle h(\mathbf{u}), h(\mathbf{v})\rangle$ is the inner product in the space of $h(\mathbf{x})$.

- Hence SVM can be viewed as an automatic way of constructing an RBF network (Schölkopf *et al.* 1997).

Performance Results: Drug Discovery Data



The original data set is randomly split by stratified sampling for four times to produce 4 different training and test sets. Each time, models are built on the training set with tuning parameters selected by 5-fold cross-validation and tested on the test set.

Performance Results: ANOVA Set-up

Do a simple ANOVA comparison by constructing four orthogonal contrasts:

$$\begin{aligned} C_1 &= \frac{\mu_T + \mu_G}{2} - \frac{\mu_U + \mu_K + \mu_S}{3}, \\ C_2 &= \mu_S - \frac{\mu_K + \mu_U}{2}, \\ C_3 &= \mu_U - \mu_K, \\ C_4 &= \mu_G - \mu_T, \end{aligned}$$

where $\mu_K, \mu_S, \mu_U, \mu_T$ and μ_G are the average result of K-NN, SVM, and our RBF method using the uniform kernel, the triangular kernel and the Gaussian kernel, respectively.

Performance Results: ANOVA Summary

Source	SS ($\times 10^{-4}$)	df	MS ($\times 10^{-4}$)	F_0	P-Value
Methods					
C_1	203.737	1	203.737	380.050	< 0.0001
C_2	17.854	1	17.854	33.304	< 0.0001
C_3	0.036	1	0.036	0.068	0.7987
C_4	0.140	1	0.140	0.262	0.6180
$+)$	221.768	4	55.442	103.42	< 0.0001
Splits	15.318	3	5.106	9.525	0.0017
Error	6.433	12	0.536		
Total	243.519	19			

Hit Curves: Drug Discovery Data

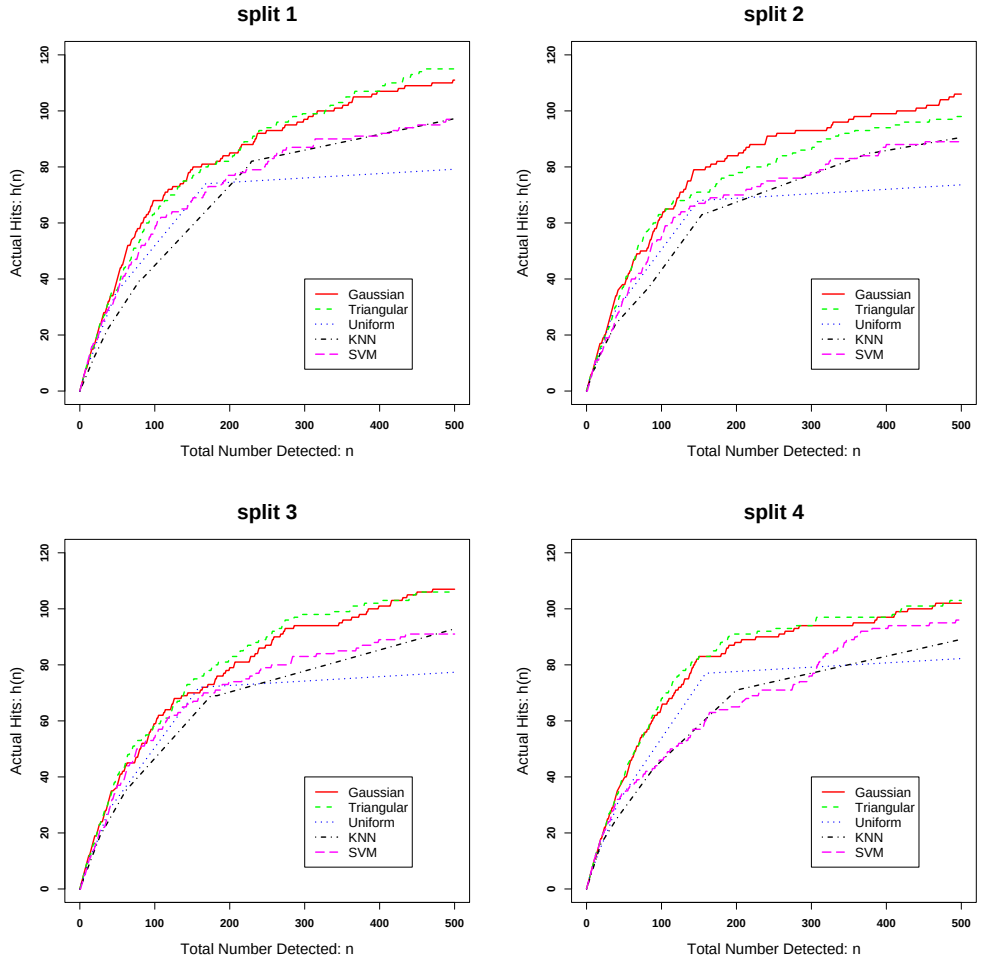


Figure 8: Only the initial part of the curves (up to $n = 500$) are shown.

The Number of SVs Used by SVM

	Number of Inactive SVs	Number of Active SVs
Split 1	12475	300
Split 2	12394	300
Split 3	12433	299
Split 4	3091	301
Total Possible	14602	304

A Statistical Explanation

- The "best" score function should be the posterior probability:

$$f(\mathbf{x}) \equiv P(y = 1|\mathbf{x}) = \frac{\pi_1 p_1(\mathbf{x})}{\pi_1 p_1(\mathbf{x}) + \pi_0 p_0(\mathbf{x})}. \tag{3}$$

- In order to rank items from a new data set $\{\mathbf{x}_i; i = 1, 2, ..., N\}$, it is clear that a very accurate estimate of $f(\mathbf{x}_i)$ is not crucial as long as $f(\mathbf{x}_i)$ ranks these observations in the correct order. That is, any monotonic transformation of f will do.
- Moreover, for detection problems it can often be assumed that the density for the background class, $p_0(\mathbf{x})$, is relatively flat when compared with $p_1(\mathbf{x})$.

Statistical Explanation (cont'd)

- If p_0 is a very flat, i.e., close to being a constant everywhere, it is clear from (3) that we can arbitrarily put any positive number in place of p_0 without affecting the ordering of $f(\mathbf{x}_i)$.
- This means we no longer need to estimate p_0; the potential saving here is significant since the background class 0 is actually the majority class.
- In reality, p_0 is not a constant and its surface will have some small ripples.
- What is the effect of these ripples on the function f?

Examining the Ripple Effects

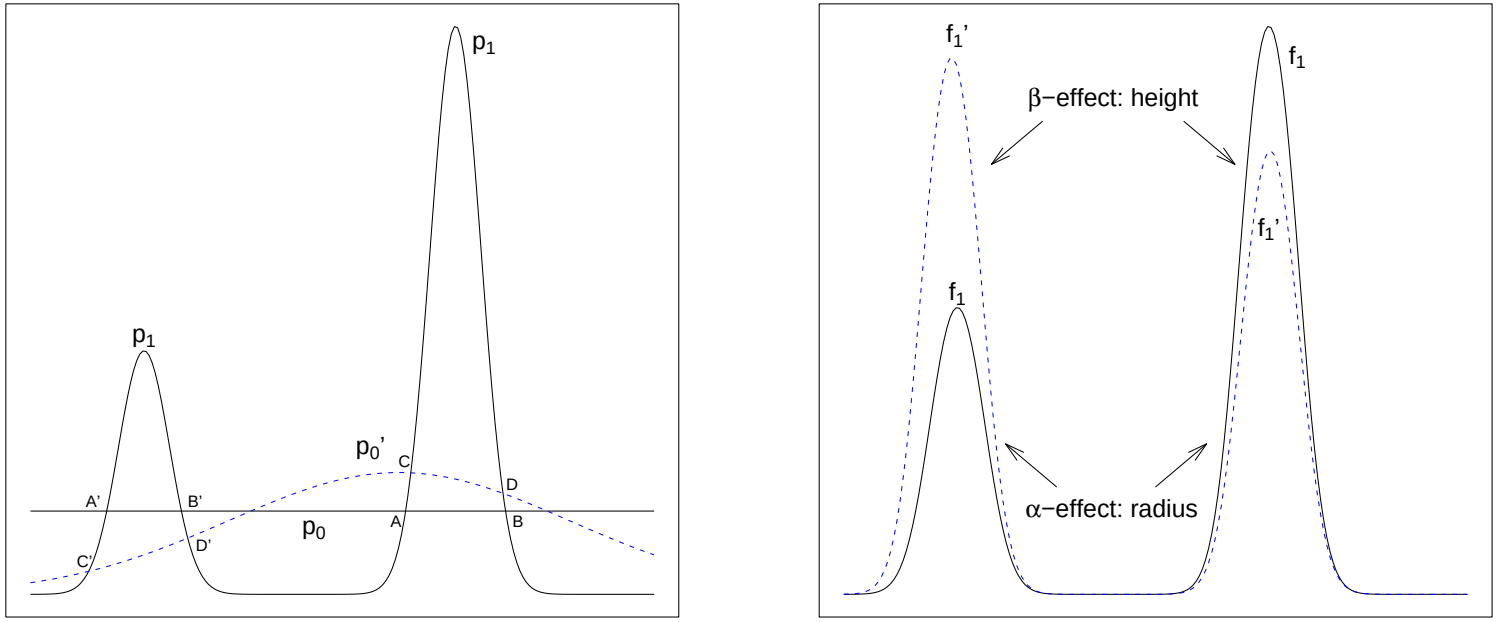


Figure 9: Illustration of the ripple effect. Left: Density functions. Right: The posterior probability.

In order to build a predictive model for
statistical detection problems,
it suffices to

- ☞ model the rare (but important) class alone and
- ☞ make local adjustments for the two ripple effects.

The Quasi Kernel Adjusts for the β-effect

- Take a proper kernel function belonging to a location-scale family:

$$\frac{1}{r} f\left(\frac{z-x}{r}\right).$$

  Can explicitly parameterize the two ripple effects as follows:

$$r^{\beta'} \frac{1}{\alpha r} f\left(\frac{z-x}{\alpha r}\right) \quad \propto \quad r^{\beta'-1} f\left(\frac{z-x}{\alpha r}\right) \quad \equiv \quad r^{\beta} f\left(\frac{z-x}{\alpha r}\right)$$

- Using quasi kernel functions, we have effectively decided that $\beta = 0$, which is equivalent to (implicitly) choosing $\beta' = 1$.
- If one regards an RBF network using proper kernel functions as a mixture model, then our RBF network using quasi kernel functions can be seen as scaling each mixture component by a factor proportional to r and hence adjusting for the β-effect.
- But is r the right scaling factor? Could it be r^2 or $\sqrt{r}$?

Evidence: r Is the Right Scaling Factor

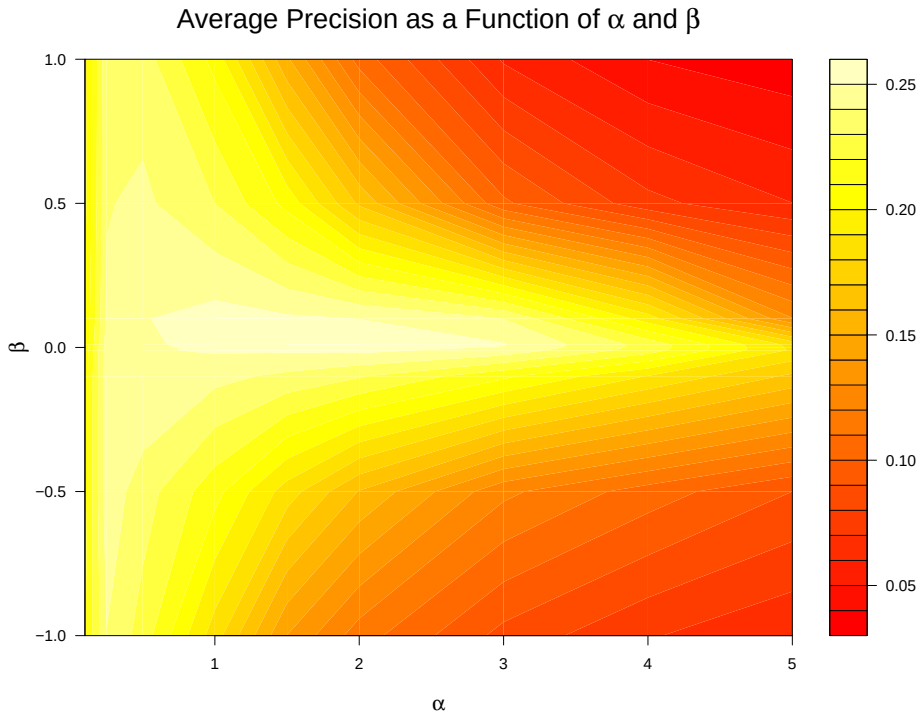


Figure 10: Choosing α and β (while fixing $K = 5$) using 5-fold cross-validation on the training data.

Some Ongoing Work

1. Want to produce empirical evidence for the statistical explanation on the drug discovery problem.
2. Want to turn the statistical explanation into more formal statements.
3. Want to modify the algorithm to implement more explicitly what the "theory" suggests.

References

Cristianini, N. and Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press.

Schölkopf, B., Sung, K. K., Burges, C. J. C., Girosi, F., Niyogi, P., Poggio, T., and Vapnik, V. (1997). Comparing support vector machines with gaussian kernels to radial basis function classifiers. *IEEE Transactions on Signal Processing*, **45**(11), 2758–2765.

Welch, W. (2002). *Computational Exploration of Data*. Course Notes, University of Waterloo.

Zhu, M. (2004). Recall, precision and average precision. Working Paper 2004-09, Department of Statistics and Actuarial Science, University of Waterloo.