AN ADAPTIVE METHOD FOR STATISTICAL DETECTION WITH APPLICATIONS TO DRUG DISCOVERY

Mu Zhu, Hugh A. Chipman and Wanhua Su
Department of Statistics and Actuarial Science,
University of Waterloo,
Waterloo, Ontario N2L 3G1, Canada.

Key Words: Classification; Hit curve; Kernel method; Nearest neighbor.

Abstract:

Researchers have tried to tackle various statistical detection problems using state-of-the-art classification techniques but are often disappointed at the results. The reason is two-fold. First of all, as classification problems, these statistical detection problems are heavily unbalanced: the class of interest is rare in the training data; an overwhelming majority of the training data belong to what can be called a background class. A primary example is drug discovery, where most of the chemical compounds in the data set are inactive whereas the goal is to detect a small number of active compounds. Secondly, the goal of statistical detection is fundamentally different from that of classification, making misclassification rate the wrong criterion to focus on. In this article, we develop an adaptive method for statistical detection and demonstrate that it can be an effective tool for drug discovery.

1. Statistical Detection

A typical statistical detection problem is as follows: We have data $\{y_i, \mathbf{x_i}\}_{i=1}^n$, where $\mathbf{x_i} \in \mathbb{R}^d$ is a vector of predictors and $y_i \in \{0, 1\}$, a class label. The setup is similar to a standard two-class classification problem, but the objective is quite different. In particular, class 1 is of primary interest, i.e., we want to detect this class against a general background class, class 0, but class 1 is extremely rare in the training data. Here are a few examples:

- *Drug Discovery.* Here $\mathbf{x}_i$ is a vector of descriptors for a chemical compound and y_i indicates whether the compound is considered an active drug agent for a certain disease. Most compounds are inactive; we are interested in catching the active ones.
- *Terrorist Screening.* Here $\mathbf{x}_i$ is a vector of descriptors for an airline passenger and y_i indicates whether the passenger could be a terrorist. Most passengers are not terrorists; we are interested in catching the terrorist.
- *Fraud Detection (Bolton and Hand 2002).* Here $\mathbf{x}_i$ is a vector of descriptors for a credit card transaction and y_i indicates whether the transaction is fraudulent. Most transactions are not fraudulent; we are interested in catching the fraud.

Because the class of interest is so extremely rare, traditional classification methods are *not* the most effective; new methods are needed that cater specifically to this type of problems.

1.1 Objective

It is very important to understand that the primary objective for the statistical detection problem is not automatic classification; hence misclassification rate is not the right criterion for consideration. This is because in these applications, further examinations, often very expensive, are almost always necessary. For example, in the drug discovery application, you never start to mass-produce a drug without conducting further lab tests; in the terrorist screening application, one never labels someone a terrorist without conducting further investigation; and in the fraud detection application, one never terminates a credit card account without confirming the fraud. Therefore, we are most interested in producing an effective *ranking* of all the candidates so that the costly further investigations are least likely to be carried out in vain. Since traditional classification methods often aim to reduce misclassification rate, this explains, to some degree, why they are not the most effective methods for statistical detection.

1.2 The Hit Curve

For the statistical detection problem, it is often informative to examine the so-called hit curve, which is a function $h(n)$ that gives the number of hits, i.e., items actually belonging to class 1, among the first n detected items. Figure 1 shows some typical hit

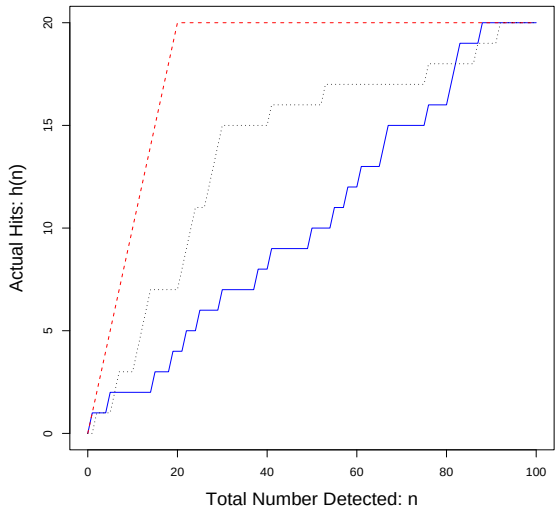

Figure 1: Illustration of several typical hit curves. The red and dashed curve is an ideal curve. The blue and solid curve is that of random selection. The black and dotted curve is that of a typical statistical detection method.

curves. In this illustration, there are 100 candidates; only 20 of them belong to the class of interest. The red and dashed curve is an ideal curve; every item detected is an actual hit until all potential hits are exhausted. The blue and solid curve is that of random selection. The black and dotted curve is that of a typical statistical detection method. Hit curves are also known as "gain charts" in some data mining applications.

1.3 Performance Measures

An obvious measure that can be used to compare the performance of different methods is the number of hits $h(N)$ among the first N detected items, where N is often determined by one's budget for further investigation (Bolton and Hand 2002).

However, considering $h(N)$ alone can sometimes be misleading. Consider the situation illustrated in Figure 2. The red and dashed curve $h_r(n)$ and the blue and solid curve $h_b(n)$ are hit curves of two hypothetical detection methods. If we set $N = 55$, then $h_r(N) > h_b(N)$, indicating that the method corresponding to the red curve is better, but this is not necessarily true since the method corresponding to the blue curve is more efficient in the sense that it makes much more detections early on.

Therefore, another possible performance measure worth considering is

$$H(N) = \sum_{n=1}^{N} h(n).$$

This can be thought of as the area under the hit curve $h(n)$ up to a certain point N and generally favors methods that make more detections early on. We are also investigating some other performance measures, but for this article, we shall only concentrate on the two mentioned above, namely $h(N)$ and $H(N)$.

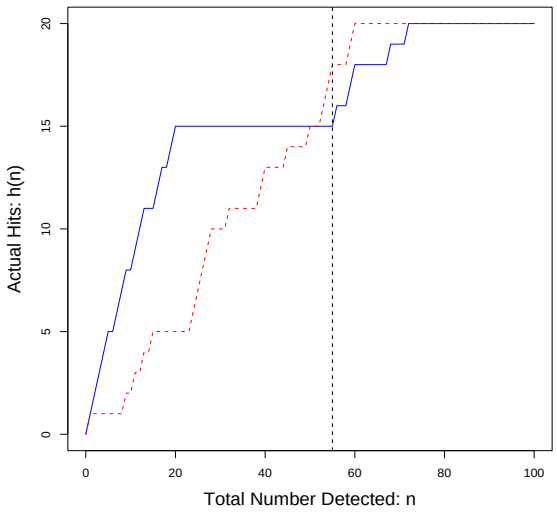

Figure 2: Hit curves for 2 different detection methods. Here, $h_r(55) > h_b(55)$ whereas $H_r(55) < H_b(55)$.

2. Drug Discovery Data

In this article, we focus on drug discovery as our main application and example. The original data are from the National Cancer Institute (NCI) with class labels added by GlaxoSmithKlein, Inc. There are 29,812 chemical compounds, of which only 608 are active against the HIV virus. Each compound is described by $d = 6$ chemometric descriptors known as BCUT numbers. For details of what these BCUT numbers are, refer to Lam *et al.* (2002).

A brief examination of the data (Figure 3) reveals that the class of interests are scattered everywhere. In fact, among traditional classification methods, it is known (Wang *et al.* 2002) that local methods such as nearest-neighbors (Cover and Hart 1967) or classification trees (Breiman *et al.* 1984) tend to work a lot

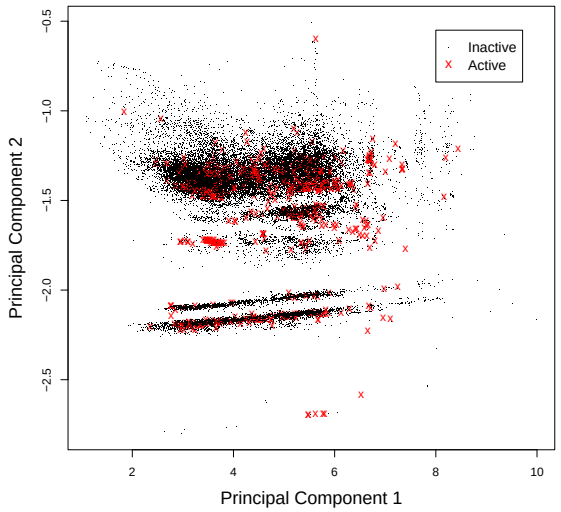

Figure 3: NCI data. Projecting the data onto the first two principal components clearly indicates the local nature of the problem.

better than global methods such as logistic regression. The method we shall present below can be regarded as an adaptive variation of nearest-neighbors.

3. An Adaptive Detection Method

The basic idea behind our method is quite straightforward: due to the underlying objective of the statistical detection problem, every observation in the training data belonging to the important but rare class of interest ought to be taken extremely seriously; in other words, we should deliberately overfit the class of interest. We first introduce two basic ingredients.

Radius of Influence Let $\mathbf{x} \in \mathbb{R}^d$ be a training observation belonging to class 1; let $N(\mathbf{x}, K)$ be its K nearest class-0 neighbors, i.e., every $\mathbf{w} \in N(\mathbf{x}, K)$ belongs to the background class. The *radius of influence* of $\mathbf{x}$ 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. In other words, a training observation belonging to class 1 will have a large radius of influence if nearby class-0 observations are far away. The motivating idea is that we want to extend our neighborhood along each axis until the number of class-0 observations becomes too large.

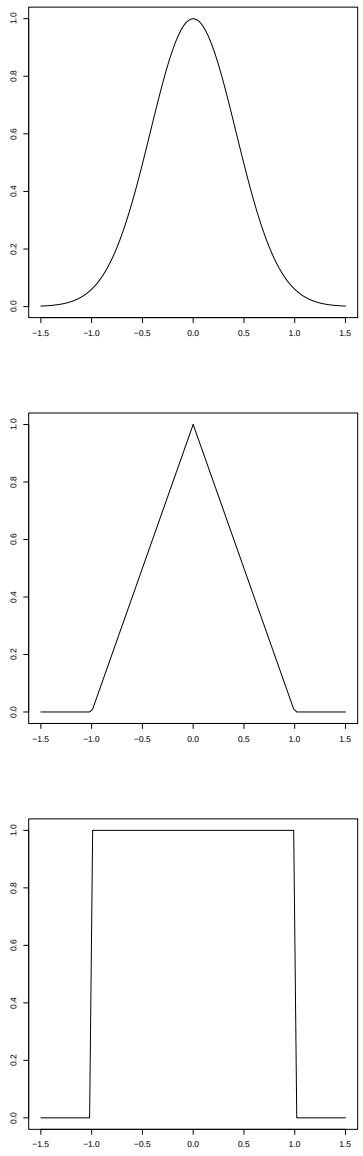

Figure 4: Some quasi kernel functions. Top: Gaussian, $f(u) = \exp\left(-\frac{u^2}{2}\right)$. Middle: Triangular, $f(u) = 1 - |u|, |u| \leq 1$. Bottom: Uniform, $f(u) = 1, |u| \leq 1$.

Quasi Kernel Function $f(u)$ 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 common quasi kernel

functions are shown in Figure 4.

3.1 Prediction

Here is how the method works: 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 below (Section 3.3); clearly, setting $\alpha = 1$ is the same as not introducing this extra tuning parameter at all. A new observation will be ranked according to the average vote it receives from class-1 training cases:

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

The reason why quasi kernel functions rather than regular kernel functions are used is now clear: If $\mathbf{x}$ has a large radius of influence, it ought to cast a stronger vote on $\mathbf{z}$. Now consider the special case where the distance between $\mathbf{x}$ and $\mathbf{z}$ is zero. With the quasi kernel, $\mathbf{x}$ will cast exactly one vote on $\mathbf{z}$; whereas with the regular kernel, $\mathbf{x}$ will cast a much stronger vote on $\mathbf{z}$ if its radius of influence is small.

There are two important advantages in this approach: 1) the radius of influence is different in each dimension; and 2) since only observations in the rare (but important) class are eligible to cast a vote, there is considerable computational saving, e.g., over nearest-neighbors.

3.2 Choice of Quasi Kernel

Our experience so far has indicated that if the uniform kernel is used, a lot of observations can be tied in terms of the average vote they receive; this does *not* produce an effective ranking. Significant improvements can, therefore, be obtained by choosing $f(u)$ to be Gaussian or triangular. Whether there exist significant differences between the Gaussian and the triangular kernels is a more difficult question. To make the comparison easier, we calibrate the Gaussian kernel as follows: Let

$$f(u) = \exp\left(-\frac{u^2}{2\sigma^2}\right) \quad \text{and} \quad g(u) = 1 - |u|;$$

we choose σ^2 as

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

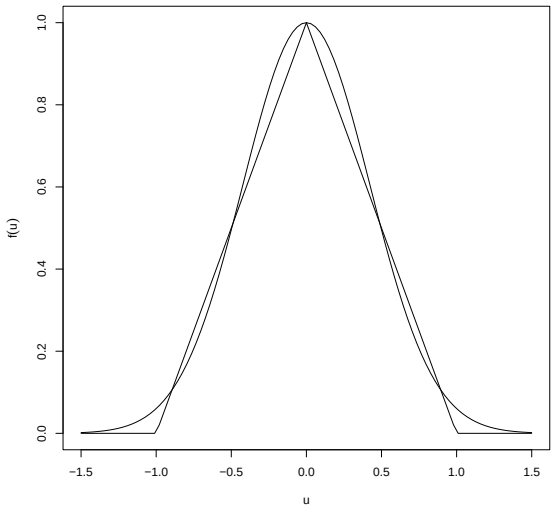

Figure 5: Calibrated quasi kernels.

By discretizing the integral, it is easy to solve, e.g., using the Gauss-Newton algorithm, that the optimal solution is $\sigma^2 \approx 0.178$; see Figure 5. Below, whenever the Gaussian kernel is mentioned, it shall be understood that we are referring to the calibrated Gaussian kernel.

3.3 Tuning Parameters

Other than the choice of the quasi kernel, there are two tuning parameters in our method, K and α, which we briefly discuss in this section. In practice, we choose both K and α by cross-validation. This is a standard operation; hence details are omitted. Instead, we shall try to explain intuitively the effect of these tuning parameters as well as why the extra global tuning parameter α is introduced.

- K: It is clear that, generally, increasing K would increase the radius of influence. However, our experiences with the drug discovery data so far have shown that the effect of K is quite mild. Figure 3 shows that most compounds are very close to each other. Therefore, a small increase in K generally does not produce a significant increase in the radius of influence. Such insensitive nature makes the parameter K quite difficult to tune. This is why the extra global parameter α is introduced.

- α: The parameter α stretches or dampens the radius of influence. For the drug discovery data,

because the compounds are close, stretching the radius of influence by a factor of 2 can sometimes be equivalent to increasing K by a few thousands. Therefore, the effect of α on the radius of influence is much stronger than K.

Of course, there is a fundamental difference between the effects of α and of K on the radius of influence. In particular, the effect of K is not identical in every direction; whereas the parameter α stretches or dampens the radius of influence identically in every direction. This is why the parameter K must be retained; otherwise the radius of influence would become identical in every direction, destroying a nice adaptive feature of our procedure. Of course, one could introduce a separate parameter α_j for each dimension j, but this would make tuning very difficult, especially if the number of dimensions d is large.

4. Some Results

We now report some results on the drug discovery data described in Section 2. We only compare our method with K-nearest-neighbors (K-NN) because an earlier study (Wang *et al.* 2002) has already concluded, using the same data, that K-NN is one of the best methods amongst a number of techniques.

Altogether, four attempts are made to evaluate the performance of different methods. In each attempt, we randomly split the data to produce a training set and a test set, each with 14,906 compounds, of which 304 are active compounds. All tuning parameters are selected using 5-fold cross-validation on the training set. After training, each method is allowed to select 500 compounds from the test set.

The performance of each method is measured and compared by $h(500)$, the number of hits when 500 compounds are selected, and $H(500) = \sum_{n=1}^{500} h(n)$, the area under the hit curve up to $n = 500$ (Figure 6), as well as by comparing the entire hit curve $h(n)$ up to $n = 500$ (Figure 7). Our results render the following conclusions:

1. Applying our method with the uniform kernel, we achieve results that are generally comparable and sometimes superior to K-NN. Recall that our method has a significant computational advantage over K-NN since only a small number of active compounds from the training set are eligible to cast a vote. This means even with comparable performances in terms of prediction, our method still appears to be more attractive.

2. When the Gaussian or the triangular kernel is used, however, our method consistently offers a significant improvement over K-NN.

3. There is no significant difference whether the Gaussian kernel or the triangular kernel is used. This is expected since we pre-calibrated the Gaussian kernel to "match" the triangular kernel.

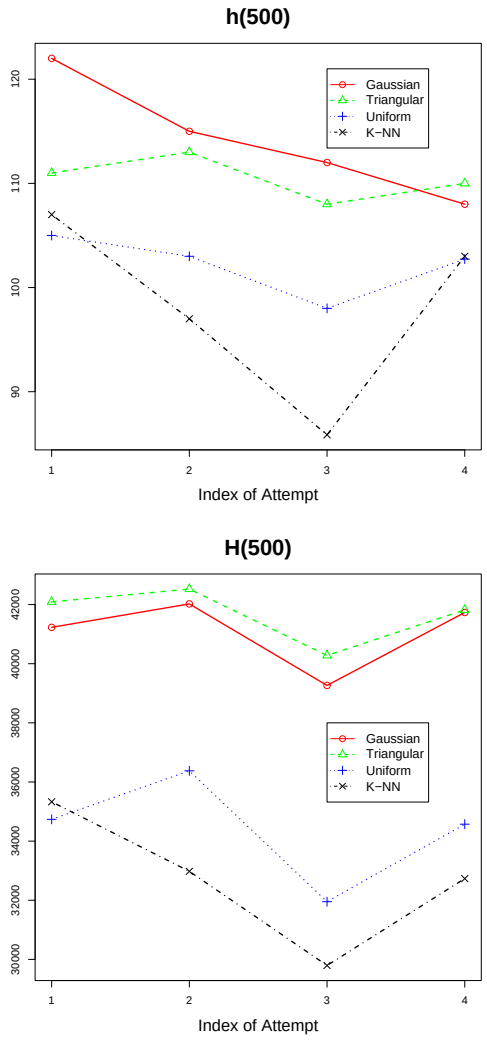

Figure 6: The number of hits, $h(N)$, as well as the area under the hit curve, $H(N)$, of different methods from four different attempts. N is taken to be 500.

5. Conclusion

We have proposed an adaptive procedure for statistical detection. On a particular drug discovery prob-

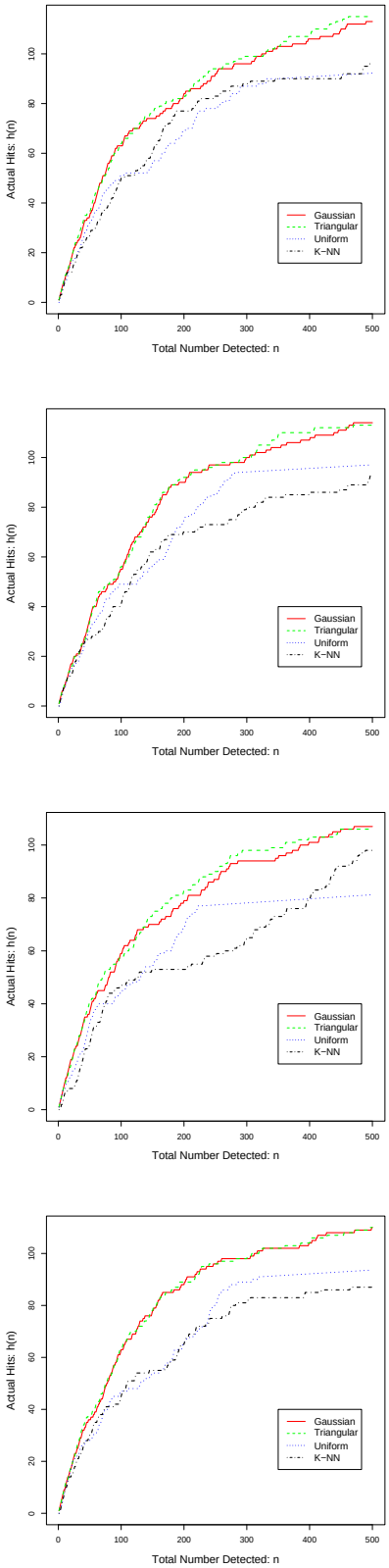

Figure 7: Hit curves of different methods from four different attempts.

lem, our procedure competes well with and outperforms existing methods. In this particular problem, local effects seem critical in prediction performance, making our method especially well-suited. Future work includes generalization to other data sets and application areas.

Acknowledgment

This work is partially supported by the Natural Science and Engineering Research Council of Canada as well as the Mathematics of Information Technology and Complex Systems network.

References

Bolton, R. J. and Hand, D. J. (2002). Statistical fraud detection: A review. *Statistical Science*, **17**(3), 235–255.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees*. Wadsworth.

Cover, T. M. and Hart, P. E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, **13**(1), 21–27.

Lam, R. L. H., Welch, W. J., and Young, S. S. (2002). Uniform coverage designs for molecule selection. *Technometrics*, **44**(2), 99–109.

Wang, Y., Chipman, H. A., and Welch, W. J. (2002). Mining nuggets of activity in high dimensional space from high throughput screening data. Research Report RR-02-01, Institute for Improvement in Quality and Productivity, University of Waterloo.