Proceedings of International Joint Conference on Neural Networks, San Jose, California, USA, July 31 – August 5, 2011

Application of Cover's Theorem to the Evaluation of the Performance of CI Observers

Frank Samuelson and David G. Brown, *Member, IEEE*

Abstract— **For any N points arbitrarily located in a d-dimensional space, Thomas Cover popularized and augmented a theorem that gives an expression for the number of the 2^N possible two-class dichotomies of those points that are separable by a hyperplane. Since separation of two-class dichotomies in d dimensions is a common problem addressed by computational intelligence (CI) decision functions or "observers," Cover's theorem provides a benchmark against which CI observer performance can be measured. We demonstrate that the performance of a simple perceptron approaches the ideal performance and how a single layer MLP and an SVM fare in comparison. We show how Cover's theorem can be used to develop a procedure for CI parameter optimization and to serve as a descriptor of CI complexity. Both simulated and micro-array genomic data are used.**

I. Introduction

A 1965 paper by Thomas Cover, entitled "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition," presents and elaborates upon a remarkable result, which he refers to as the "function-counting theorem" [1]. Restating that theorem slightly (in his formulation the hyperplanes are required to pass through the origin), the theorem becomes as follows: There are C(N,d) linearly separable dichotomies of N points in general position in Euclidean d-space, where

$$C(N,d) = 2\sum_{k=0}^{d}\binom{N-1}{k}. \qquad (1)$$

The brackets denote the binomial coefficient for N-1 things taken k at a time, and "general position" means that the points do not "line up," e.g., no more than two points lie on the same line in a two-dimensional space, no more than three points lie in the same plane in a three-dimensional space, etc. (Our C(N,d) corresponds to C(n,p) in a related exposition by Ripley [2].)

It is important to remember that this is not a statistical relationship: it is not that "on average" there are C(N,d) linearly separable dichotomies of N points in d-space, but rather that for each and every set of N points, regardless of the distribution of the points, there are precisely C(N,d) linearly separable dichotomies. In this paper we will refer to the above function-counting theorem as Cover's theorem and will assume that any referenced set of N points meets the requirement of being in general position.

CI observers are frequently designed for separating N exemplars in d-dimensional feature space into two classes. For example, given d=100 gene expression results for each of N=50 individuals, m labeled as "with disease" or class 1, and N-m labeled as "normal," or class 2, can we design a classifier to successfully separate those with a particular disease from those unaffected by it? Similarly, provided with a set of d = 30 measurements of metabolic information for each of 200 patients, m classified as helped by a particular drug therapy and N-m classified as unaffected, can we discriminate those who are candidates for use of that drug from those who are not?

Cover's theorem can be invoked directly or used to develop a procedure to examine CI observer performance in several ways, including the following:

1. It can be used to indicate a problem with data sufficiency. As we shall see shortly, in the above problem with d=100 and N=50, a simple linear discriminant can be found to perfectly separate any arbitrary labeling of the N points. There are too few patients to adequately constrain a solution.
2. It can be used as a test problem with a known solution or solution rate to test a candidate algorithm, e.g., for algorithm selection or parameter optimization.
3. It can be used to quantify the complexity of the CI algorithm. Whereas the linear decision surface, e.g., perceptron operates in a d-dimensional feature space, a more complicated CI will be seen to be equivalent to operation in a d'-dimensional space, where d' is usually greater than d, and hence allows for increased ease of separation of the two populations.

If we pick N points at random in d-space and assign their labels according to which side they fall with respect to a given hyperplane, then we know that by construction the points are separable by that hyperplane. A candidate CI observer would be expected to accomplish the desired separation; however, in practice even such apparently trivial problems are surprisingly difficult, as we shall see below for commonly available software packages.

For N points in d dimensions, C(N,d) is the number of the 2^N possible dichotomies or assignments of labels to the N points that are separable by a hyperplane. Fig. 1 illustrates this concept for N=4 points in d=2 dimensions. For assignment (a), points 1, 2, and 4 are labeled "+," and point 3 is labeled "o." For this arrangement, a separating hyperplane (here line) is easily discovered. On the other

Manuscript received January 28, 2011.

F. Samuelson is with the Food and Drug Administration, Center for Devices and Radiological Health, 10903 New Hampshire Ave., Bldg. 62, Rm. 3105, Silver Spring, MD 20993-0002 USA (phone: 301-796-2544; fax: 301-796-9925; e-mail: frank.samuelson@ fda.hhs.gov).

D. G. Brown is with the Food and Drug Administration, Center for Devices and Radiological Health, 10903 New Hampshire Ave., Bldg. 62, Rm. 4116, Silver Spring, MD 20993-0002 USA (e-mail: david.brown @fda.hhs.gov).

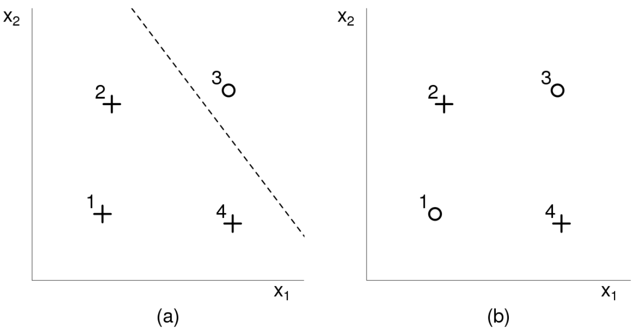

Fig.1. Example for d=2, N=4 for which (a) no separating hyperplane is possible and (b) a separating hyperplane exists.

hand, for assignment (b), points 1 and 3 are labeled “o,” and points 2 and 4 “+.” Clearly no hyperplane can separate the space such that all of the points labeled “o” lie on one side of the line and those labeled “+” lie on the other side. Aside from the complementary arrangement for which the “o” and “+” class labels are reversed, the other 14 of the $2^4 = 16$ possible arrangements are separable, so we confirm that C(4,2) = 2(1+3+3) = 14. In terms of the fraction of separable dichotomies, f(N,d), this is f(N,d)=14/16 = .875.

TABLE I
C(N,d) AND f(N,d) FOR SMALL VALUES OF d

d	N	C(N,d)	f(N,d)
1	1	2	1.0
	2	4	1.0
	3	6	0.75
	4	8	0.5
	N	2N	$N/2^{N-1}$
2	1	2	1.0
	2	4	1.0
	3	8	1.0
	4	14	0.875
	5	22	0.6875
	6	32	0.5
	N	N^2-N+2	$(N^2\text{-}N+2)/2^N$
3	1	2	1.0
	2	4	1.0
	3	8	1.0
	4	16	1.0
	5	30	0.9375
	6	52	0.8125
	7	84	0.65625
	8	128	0.5
	N	$(N^3\text{-}3N^2+8N)/3$	$(N^3\text{-}3N^2+8N)/(3x2^N)$

Table I gives results for equation 1 for C(N,d) and f(N,d) for one, two, and three dimensions. For a given value of d, f(N,d) =1 for small N and then monotonically decreases. It is also evident that N=2(d+1) corresponds to the midpoint, i.e., f(2(d+1),d)=0.5. This provides a means for generating a polynomial expression in N which evaluates to 2^N for sufficiently small N (N<(d+2)).

Note that f(N,d) can be considered to be the complement of a cumulative probability distribution (of N for fixed d), call it $p_d(N)$. By taking the difference f(N,d) - f(N+1,d), we can solve for $p_d(N)$:

$$p_d(N) = \frac{1}{2^N}\binom{N-1}{d}, \tag{2}$$

with the requisite property of summing to 1, with mean and variance 2(d+1), and recognizable immediately as belonging to the class of the negative binomial distributions.

Fig. 2 shows f(N,d) for several representative values of N (similar to Fig. 3.4 of Duda and Hart [3]). The figure illustrates the fraction of dichotomies that may be successfully separated by a hyperplane as a function of N and d. Note that for N≤d+1, a hyperplane always exists that will separate the two classes, and for N=2(d+1) exactly half of them are amenable to separation. Further, as N increases, f as a function of N/2(d+1) comes to approximate a step function: for values of N less than 2(d+1) all possible assignments of class labels are almost certainly separable, and for values greater than 2(d+1) they are almost certainly incapable of separation. Note that the above argument for nearly step-function behavior of the function f should be tempered slightly, since the absolute width of $p_d(N)$ is growing with N; it is the width normalized to N/2(d+1) that is decreasing.

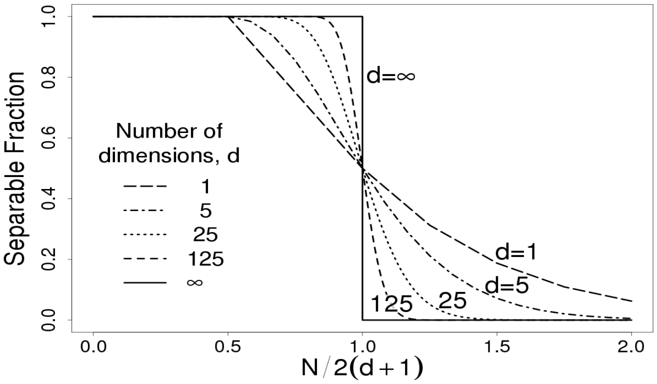

Fig.2. f(N,d) for d=1,5,25,125, and the limit of large d. The abscissa x is scaled so that the values of f(N,d)=0.5 lie superposed at x=1for all d.

In terms of our earlier question concerning data sufficiency, unless the number of cases (N) is greater than twice the number of features (d) being used to separate the cases into two classes (N>2(d+1)), the problem is not amenable to a meaningful solution: a simple hyperplane can be found to separate each and every random arrangement of the classes over the cases.

II. Methodology

A. Simulation Protocol

Our object is to obtain data similar to that of Fig. 2 for actual trained computational observers in order to compare them to the theoretical performance of a linear observer. For this paper we use three observers as described below. One is a perceptron, which is a linear discriminant, trained using the classical perceptron training rule [4]. The second is a publicly available implementation of a multi-level perception (MLP) with a single level, i.e., a simple linear perceptron, but with least squares, back-propagation learning. For the third CI algorithm we wanted to use a nonlinear decision surface and chose a support vector machine (SVM) as described below. For this paper the performance of a trained CI observer is rated by the fraction of times that it can perfectly separate sets of N points from two different classes in d dimensional space. This performance can also be measured in terms of the area under the corresponding receiver operating characteristic (ROC) curve as explained below.

In obtaining performance data for a randomly selected set of N points in d dimensions, there are two potential approaches. First, we could exhaustively examine the 2^N possible label assignments, or second, we could statistically sample the possible labeling schemes. The first approach is theoretically satisfying; however, as N increases, it rapidly becomes impractical. Therefore, we used the second approach.

We desired to verify that any observed defect between theoretically expected and actual performance for our algorithms did exist and was not some artifact introduced by our simulation procedure. To do this, we sliced a hyperplane through our randomly selected points and labeled those on one side as null-valued and on the other side as positive. Therefore we knew in advance that the points were separable and could see whether or not the CI algorithms still failed to discern that fact.

B. *CI Observers*

The classical perceptron was implemented from section 4.5.1 of Ref. 5. The other two computational observers used in this work are obtained directly from the program suite on the Comprehensive R Archive Network at http://cran.r-project.org. The MLP is a feed forward back propagation, MLP neural network called AMORE [6]. It was used in its simplest configuration as a single layer perceptron with back propagation learning. The standard suggested parameter settings were used unless otherwise specified. These are momentum = 0.5, learning rate = 0.075 (as optimized below, instead of the suggested value of 0.01), the error criterion was least mean squares, the hidden layer function was “tansig,” and the minimization method was adaptive gradient descent with momentum. These two CI observers both produce hyperplane decision surfaces, but they differ in the training rules that they employ.

The SVM was provided in a package (also on the Comprehensive R Archive Network) called e1071, and using LIBSVM [7,8]. The default settings for that algorithm were used unless otherwise specified, and included cost = 1, gamma (γ)= 1/d, tolerance = .001, epsilon = 0.1, and kernel = radial basis function.

III. Results

A. *Optimization*

Our first experiment employed the Cover’s Theorem (CT) methodology to optimize the learning rate for our MLP observer. Fig. 3 shows the results of our parameter optimization for “learning rate” for the AMORE perceptron in terms of both the success rate for perfect separation of the data and the area (AUC) under the ROC curve. Each point is the result of averaging over hundreds of experiments for which a known plane successfully separates the data, with N ranging from 3 to 27 and d from 3 to 9. The small vertical line within each data point is the plus or minus one standard deviation error bar for that point.

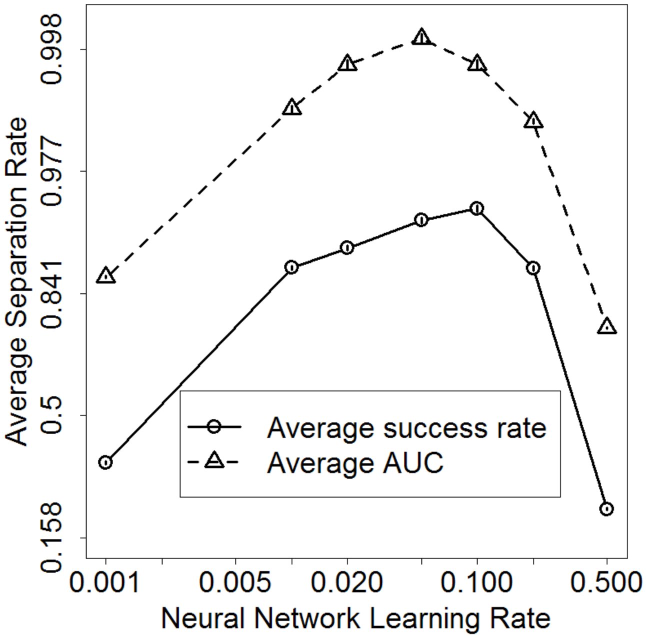

Fig. 3. Average separation rate and AUC as a function of neural network learning rate. The ordinate is scaled by the inverse of the error function and the abscissa by the logarithm.

The AUC values were taken directly from empirical ROC curves. Empirical ROC curves are plots of the fraction of true assignment of one class label to points that are actually of that class (true positive fraction (TPF)) versus the fraction of false assignment of the second class label to points of the first class (false positive fraction (FPF)). For medical data class 1 is traditionally a condition to be diagnosed and class 2 is disease free or normal. The ROC curve is thus a plot of sensitivity (TPF) versus the complement of specificity (FPF = 1 – true-negative-fraction (TNF)) for that medical condition. Fig. 4 demonstrates the construction of an ROC curve as a decision threshold “T” is swept across the output values “t_i” of a CI observer for each of the N points. Initially the output yields an assignment to class 2 no matter what the output value (TPF = 0, FNF = 0). As the threshold is raised, at $T=t_t$, points with t_i greater than t_t are labeled as class 1 and with t_i less than t_t are labeled as class 2, until finally as T is further increased, all points are labeled as belonging to class 1 (TPF = 1, FPF = 1). Given the ROC curve, AUC is the average sensitivity over all values of specificity. As seen in Fig. 3, we obtain a similar optimization for the learning rate for the present data using either the success rate or the AUC.

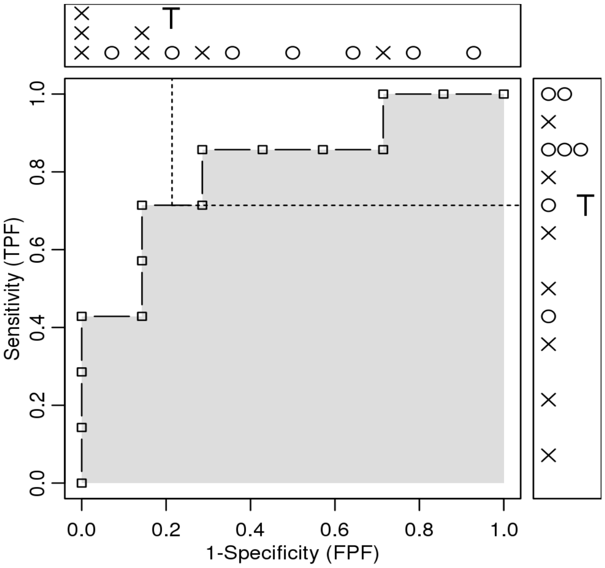

Fig. 4. ROC curve generated from N=14 data points with $n_1=n_2=7$ of each class, demonstrating construction of the curve from the CI observer output: "x" indicates a true class 2 (normal), and "o" is a true class 1 (disease present). The ROC curve is created as the value of the decision threshold T is swept across the output values of the CI, moving up the ordinate $1/n_1$ as each instance of x is encountered and over the abscissa $1/n_2$ for each instance of o.

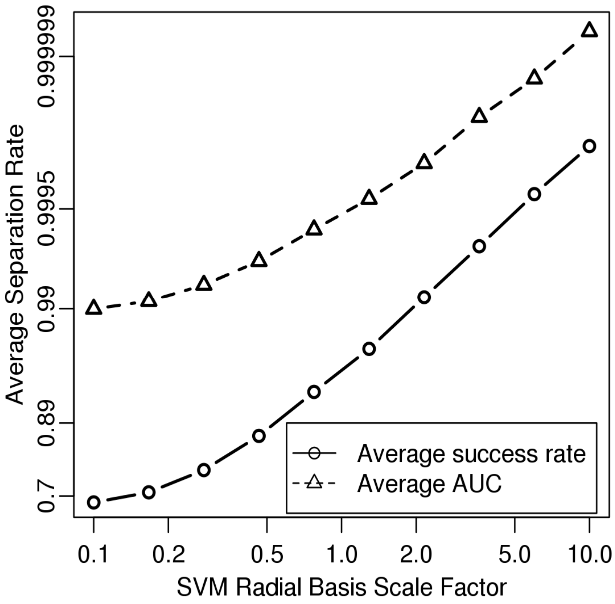

Fig. 5. Average separation rate and AUC (as transformed through the inverse Gaussian) as a function of SVM radial basis scale factor, s=dγ.

Fig. 5 is the corresponding curve for the optimization of the radial basis function "radius" parameter for the SVM. Here, however, there is no optimal value. As the parameter is increased, the fitting ability increases monotonically. Increasing this parameter increases the ability of the SVM to fit surfaces of ever higher complexity, as discussed further in Section C. Note that this result implies that our use of the term "optimization" is misnomer. Our CI will be able to better fit the training data, but that may be irrelevant to the underlying task at hand. True optimization would have to take into account the accompanying loss of generalizability of the algorithm for the particular data set under consideration.

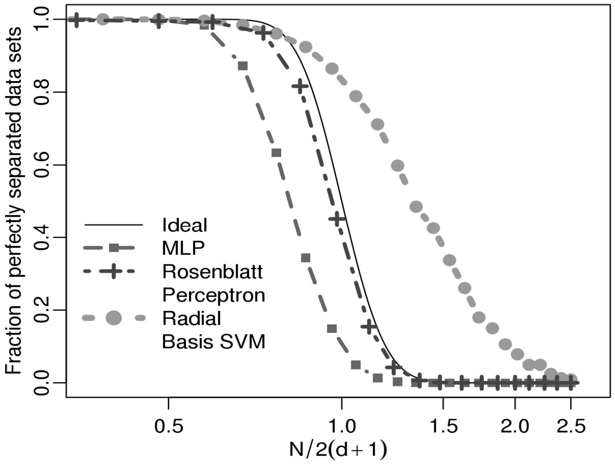

Fig. 6. f(N,d) curves for the classical perceptron, single layer MLP, and SVM compared to the ideal linear result.

B. *Performance characterization*

For any given parameterization, the performance of a proposed CI observer may be compared against that of the ideal linear observer as given by Cover's theorem. Fig. 6 gives the performance of all three of our CI algorithms in the same format as Fig. 2. The classical perceptron is seen to approach ideal performance for this task, the single layer MLP falls significantly short of ideal performance, and the (nonlinear) SVM exceeds what the ideal linear observer could accomplish. It may seem odd that the MLP curve is so far inferior to the ideal; however, it should be remembered that in addition to any defect incurred in the training process, it is well known that least squares measures of performance result in boundary hyperplanes that may not exactly separate even perfectly linearly separable populations. (See, e.g., Ref. 5, Fig. 4.13.).

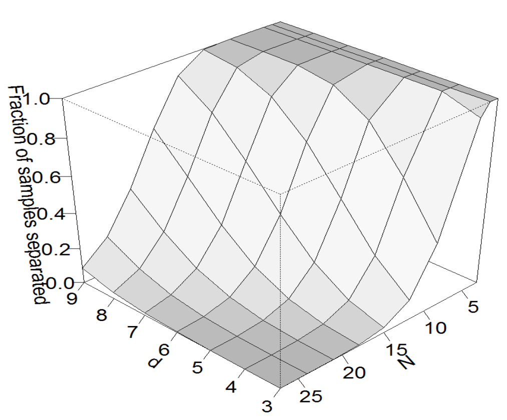

Fig. 7. 3-D perspective plot of f(N,d) as predicted by Cover's theorem.

Figs. 7, 8, and 9 are 3-D perspective plots of f(N,d) as a function of N and d. Fig. 7 shows the theoretical result of

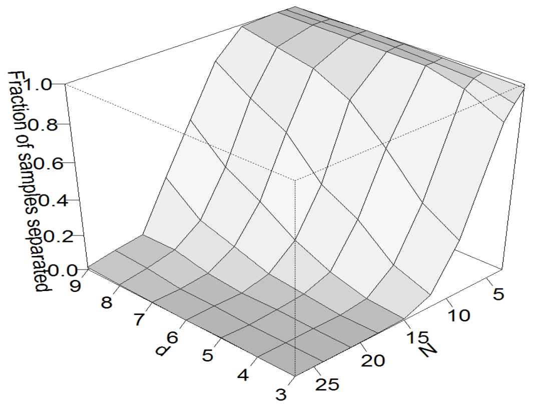

Fig. 8. 3-D perspective plot of f(N,d) for an AMORE perceptron.

Cover's theorem. It is similar to Fig. 2, except that curves for various valued of d are not superimposed but rather form a second axis. It also differs in that the N axis is no longer normalized to N/(2(d+1)), since there would have to be a different normalization for each value of d. The same general features are evident, with f=1 for N less than d+1, f=0.5 for N=2(d+1) and a monotonic decrease for increasing N for fixed d.

This serves as a reference for comparison with Fig. 8 for the AMORE MLP. There is an obvious decline in performance, even though the theoretical result for the perceptron as a linear decision surface is identical with Fig. 7. Thus, as previously seen in Fig. 6, separation using publicly available algorithms may be found to trail expectations, in some cases badly.

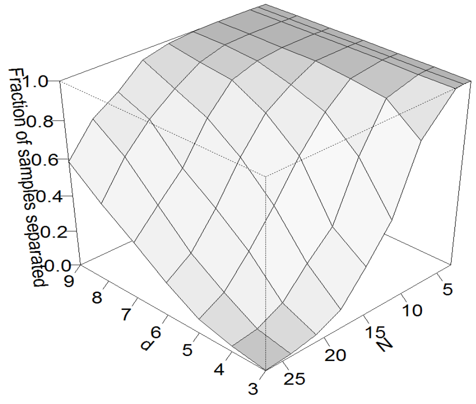

Fig. 9. 3-D perspective plot of f(N,d) for an SVM

Fig. 9 provides the same information for our SVM algorithm. Its performance exceeds that of the perceptron and that of Cover's theorem, which is to be expected in view of the extra complexity of the algorithm. The SVM uses a nonlinear surface rather than a simple hyperplane to segregate the two classes. Figs. 10 and 11 present the above information in a different format. They are 3D perspective plots of the data sets for which we have ensured that there is a separating hyperplane. For ideal performance, the surface would be a uniform plane at f=1, and therefore is uninformative and not shown. Figs. 10 and 11 are for the AMORE MLP and the SVM algorithm respectively. It is evident that both of them fall well short of the theoretical limit and that the SVM again markedly out performs the perceptron.

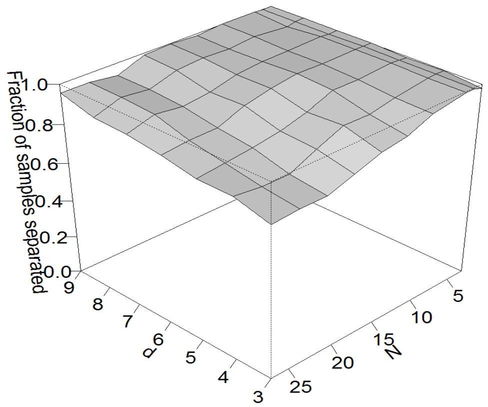

Fig. 10. 3-D perspective plot of f(N,d) for a perceptron for preseparated case.

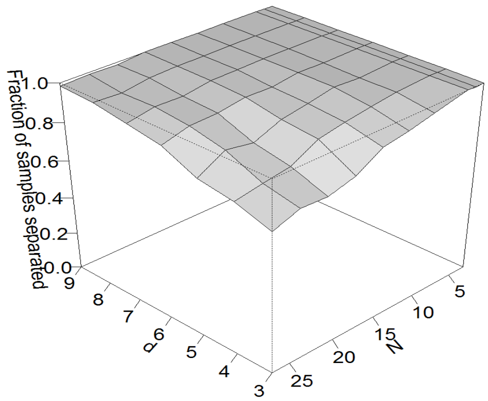

Fig. 11. 3-D perspective plot of f(N,d) for an SVM algorithm for the preseparated case.

To this point our results are strictly based on simulated data. In order to check our procedure against actual experimental data, we utilized a publicly available genomic micro-array data set [9]. For that set N=130 patients and d= 22283 gene expression probes. Fig. 12 is a variant of Fig. 6, plotting separation fraction against the quantity N/2(d+1). Now however, instead of varying N for a fixed value of d, we fix N=130 and vary d, and hence the values of N/2(d+1) are similar to those of Figure 6. For all N=130 patients, 20 to 200 different probes (d) were selected at random, which gave values of N/2(d+1) over the range of interest. The class of each patient was also randomly assigned. The data was input to the CI observers, and the fraction of all data sets that were perfectly separated were obtained by averaging over hundreds of experiments. The results of these experiments are shown in Fig. 12. Clearly the MLP perceptron performs significantly worse than the optimal theoretical performance for a hyperplane.

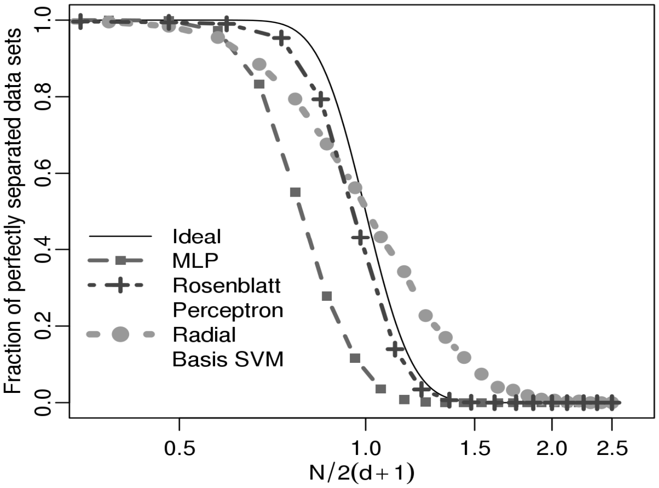

Fig. 12. f(N,d) for the classical perceptron, MLP, and SVM compared to ideal performance on a micro-array data set [9].

The performance of the CI observers on this set of real genomic data is very similar to that of the simulated data shown earlier. This demonstrates that this method of evaluation is general, robust, and does not depend on the distribution of the data.

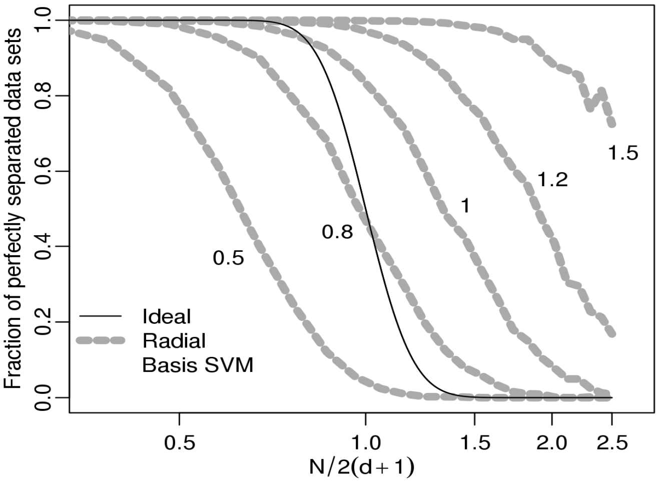

Fig. 13. f(N,d) curves for the SVM for different values of the radial basis function scale compared to the ideal linear result on a simulated data set.

C. *Complexity index*

More sophisticated CI algorithms produce decision surfaces that are more complicated than simple hyperplanes. For this reason they are able to outperform the ideal hyperplane in separating two-class dichotomies. Figs. 13 and 14 demonstrate this observation for our SVM observer for our simulated and real-world data sets respectively. The commonplace observation that the SVM operates by transforming the feature space into one of higher dimensionality is thus born out in practice [10].

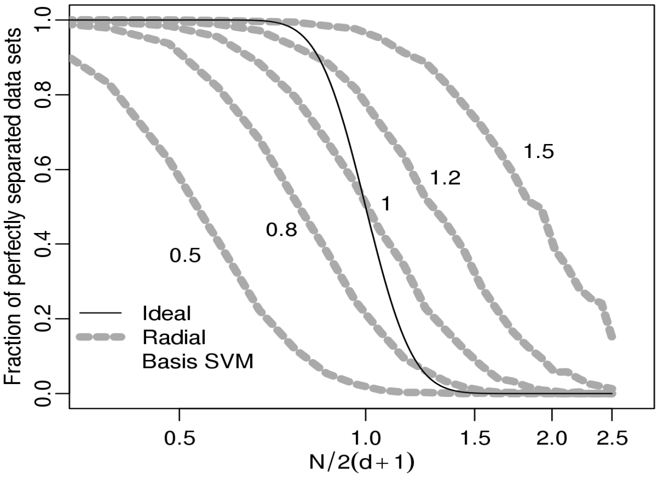

Fig. 14. f(N,d) for the SVM for different values of the radial basis function scale compared to ideal linear performance on a micro-array data set [9].

IV. Discussion

Our procedure, based on Cover's result and the random assignment of class over simulated data, provides a rigorous linear ideal benchmark against which to compare CI observer performance. This is in contrast with the typical practice of comparing a proposed CI observer to one of unknown absolute efficiency on an arbitrary data set as a standard of comparison. We have shown that this methodology provides a measure of data sufficiency, since it implies a requirement for the number of exemplars (patients) to be at least on the order of or greater than twice the number of features. We have also demonstrated that even the trivial problem of discovering linearly separable dichotomies can be difficult in practice for publicly available CI observers. The data we present show that for linear methods, even for only a small number of data points, theoretically separable data are frequently not directly separated using commonly available methods. Our method also demonstrates the degree of improvement in class separation achievable by more complex methods.

The most intriguing aspect of this study is the indication that our method may lead to a natural measure of the inherent complexity of CI algorithms. This should be expressed as an increase in the dimensionality of the underlying feature space, e.g., by calculating the N/2(d+1) abscissa at which f(N,d)=1/2 for a particular algorithm. The degree to which this is a stable and useful measure will be the subject of future research.

V. Conclusion

Cover's function counting theorem provides a useful metric for comparison of performance among different CI algorithms and for optimization of CI observer algorithm parameters. This is preferable to the present practice of comparing performance based on that of some other CI observer "floating" or not tied to an understanding of what the ideal performance should be. This paper presented basically qualitative measures of improvement, without a quantitative scale; however, we believe that these can be transformed into quantitative measures with a sound decision theoretic basis.

References

[1] T. M. Cover, "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition," *IEEE Trans. Elec. Comp.*, vol. EC-14, June 1965, pp. 326-334.

[2] B. D. Ripley, Pattern Recognition and Neural Networks. Cambridge: Cambridge University Press, 1996, pp. 119-120.

[3] R. O. Duda and P. E. Hart, Pattern Classification and Scene Analysis. New York: John Wiley & Sons, 1973, p. 70.

[4] F. Rosenblatt, Principles of Neurodynamics: Perception and theory of Brain Mechanisms. Washington, D.C.: Spartan Books, 1962.

[5] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning. New York: Springer, 2001, p. 107.

[6] A. V. Pernia Espinoza, J. B. Ordieres Mere., F. J. Martinez de Pison, A. Gonzalez Marcos, "TAOrobust Backpropagation Learning Algorithm," *Neural Networks*, vol. 18-2, 2005, pp. 191-204.

[7] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for Support Vector Machines," http://www.csie.ntu.edu.tw/~cjlin/libsvm and http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.ps.gz .

[8] R.-E. Fan, P.-H. Chen, and C.-J. Lin, "Working Set Selection Using the Second Order Information for Training SVM," http://www.csie.ntu.edu.tw/~cjlin/papers/quadworkset.pdf

[9] K. R. Hess, et al. "Pharmacogenomic predictor of sensitivity to preoperative chemotherapy with paclitaxel and fluorouracil, doxorubicin, and cyclophosphamide in breast cancer," *J. Clin. Oncol.* 24, 2006, pp. 4236-44, and http://bioinformatics.mdanderson.org/ pubdata.html.

[10] V. Vapnik, The Nature of Statistical Learning Theory. New York: Springer, 1995, Section 5.6.