Integrated Sensing and Processing Decision Trees

Carey E. Priebe, *Member*, *IEEE Computer Society*, David J. Marchette, and Dennis M. Healy Jr.

Abstract—We introduce a methodology for adaptive sequential sensing and processing in a classification setting. Our objective for sensor optimization is the back-end performance metric—in this case, misclassification rate. Our methodology, which we dub *Integrated Sensing and Processing Decision Trees* (ISPDT), optimizes adaptive sequential sensing for scenarios in which sensor and/or throughput constraints dictate that only a small subset of all measurable attributes can be measured at any one time. Our decision trees optimize misclassification rate by invoking a local dimensionality reduction-based partitioning metric in the early stages, focusing on classification only in the leaves of the tree. We present the ISPDT methodology and illustrative theoretical, simulation, and experimental results.

Index Terms—Classification, clustering, adaptive sensing, sequential sensing, local dimensionality reduction.

1 Introduction

Statistical pattern recognition techniques often play a key role in extraction and exploitation of useful information from the inherently high-dimensional data collected by many of today's sensing systems. Dealing effectively with this data can impose stringent challenges for the designer of statistical pattern recognition algorithms when issues of throughput and statistical reliability of the overall sensor/exploitation system are a concern. This paper examines methodologies for addressing the challenges of exploiting high-dimensional data in sensor systems and are particularly suited to those (increasingly common) systems incorporating controllable sensor front ends whose particular measurement functions can be tuned.

For an illustration of the type of high-dimensional data exploitation, we are concerned with considering a land-use classification application based on measurements from a hyperspectral camera, such as that described in [16]. Such a sensor may provide the equivalent of hundreds of mega-pixel images of a scene, each image corresponding to the appearance of that scene in light from a narrow band of wavelengths. Taken together, these images present a finely resolved spectrum for each pixel in the scene and may provide information on the material composition at the spatial positions within the field of regard. All told, spatial and spectral information are presented in a data volume totaling on the order of a billion voxels per scene. Of course, for real scenes, these billions of degrees of freedom exhibit correlations; nevertheless, the raw data presented to the statistical pattern recognition algorithms comprise points in an overwhelmingly high-dimensional space and typically presents a challenge in the subsequent operations involved in moving, storing, and computing with this data.

In this example, and many others, we face a challenging requirement for effective and affordable exploitation of high-dimensional sensor data through the application of pattern classification techniques. In fact, the curse of dimensionality (see, for example, [2], [19], [17], [11]) implies that it can be counterproductive to attempt to perform statistical pattern recognition in the high-dimensional space of every measurement that can be made. Certainly one must be concerned with the possible effects of the curse of dimensionality in evaluating the performance and reliability of statistical pattern recognition tasks applied to the hyperspectral data example discussed above.

A potentially significant development is the proliferation of modern sensor systems that can be controlled to take measurements in a specified low-dimensional projection of the full space of the sensor's possible measurable degrees of freedom. In some situations, this is indeed motivated in part by one or more of the issues we have sketched: It may not be possible to effectively and affordably sense, process, transmit, or reliably exploit all of the dimensions that could be simultaneously sensed in principle. A straightforward example of this strategy is seen in various types of functional Magnetic Resonance Imaging (MRI). In standard Magnetic Resonance Imaging, images of an object are built after acquiring a sequence of many measurements, each one obtaining the values of the object's spatial Fourier transform along a particular line in Fourier space ([12], [9]). Each line takes a certain amount of time to acquire, and significant imaging time may be required as many lines must be obtained to build a high-resolution image. In various forms of functional imaging (involving some motion of the object), one often needs to restrict to a subset of all Fourier space measurements, generally corresponding to a low spatial resolution projection of the highest possible resolution image. Although higher resolution would be useful in many cases, acquisition of the full resolution data would

- *C.E. Priebe is with the Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD 21218-2682. E-mail: cep@jhu.edu.*
- *D.J. Marchette is with the Naval Surface Warfare Center, Code B10, Dahlgren Rd., Dahlgren, VA 22448-5100. E-mail: marchettedj@nswc.navy.mil.*
- *D.M. Healey Jr. is with the Department of Mathematics, University of Maryland, College Park, MD 20742-4015. E-mail: dhealy@math.umd.edu.*

Manuscript received 8 Apr. 2003; revised 4 Sept. 2003; accepted 15 Jan. 2004. Recommended for acceptance by A. Yuille.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0020-0403.

result in blurred images, as the object may have moved significantly over the time required to take all the required lines. In this case, the agility of the sensor is employed to make a simple trade of spatial resolution for temporal resolution. More sophisticated approaches have been proposed which apply a priori knowledge to reduce the dimension of the acquired data in different ways.

This type of control is currently being considered in a variety of sensor systems and applications in hopes of obtaining a better trade off between performance requirements and cost constraints. For instance, the hyperspectral sensor mentioned previously provides very finely resolved spectral information that seems very useful in some applications. However, this performance is obtained at a cost. Spectral resolution results in high-dimensional raw data sets presenting significant computational and communication challenges, particularly for time-critical applications. Furthermore, the narrow spectral range of the hyperspectral bands means that one must collect light for some time before obtaining enough photons in each given band to produce an image with reasonable signal-to-noise ratio. A further and very important difficulty in high-dimension spectral sensing is the potential for poor reliability of pattern classification algorithms applied to that data. Certainly, one must be concerned with the possible effects of the curse of dimensionality in statistical pattern recognition tasks applied to the full hyperspectral data discussed above.

In contrast to the hyperspectral camera, the multispectral sensor is similar in concept but resolves the light into fewer spectral bands. It offers coarser spectral resolution but could wind up providing better time resolution in video mode, lower dimensional data, and less overall data burden than a hyperspectral sensor. In light of this trade off, several research groups are currently developing "tunable" multispectral sensors which can resolve light into bands with adjustable bandwidth and other parameters. These tunable bands provide data with may be modeled as (adjustable) projections of the full hyperspectral information. The ability to obtain such projections on demand could potentially offer some of the benefits of both hyperspectral and multispectral sensors in one system, potentially offering some of the capability of hyperspectral imagery while still meeting constraints on data size.

To take advantage of this in applications involving statistical pattern recognition, we consider the use the tunable multispectral sensor for the direct measurement of informative features in what amounts to a reduced dimensionality subspace of the full hyperspectral measurement space. In order to realize the potential of a sensing/pattern recognition system of this type, we are motivated to consider the design of control strategies explicitly devised for enhanced performance in the statistical pattern classification. Such a methodology effectively integrates adaptive sensor technology with the pattern recognition task for which the sensor is employed, enabling users to take full advantage of sensor agility to enhance system-level performance. A simple special case of this notion has been introduced in [16] and illustrated for the tunable multispectral sensor example. This work represents a step in the larger context of a program of end-to-end cooptimization of sensor, processor, and exploitation subsystems, embodied in DARPA's "Integrated Sensing and Processing" (ISP) initiative.

Of course, the tunable multispectral system we have considered represents only one of many existing and planned sensor systems that could be used to take reduced dimensionality measurements in an agile and controllable way. As a next step toward addressing ISP, we present here an extensive generalization of the previous work of Priebe et al. [16], resulting in widely applicable methodologies for the effective control of such sensors in a variety of applications in which the exploitation subsystem is concerned with supervised statistical pattern recognition (classification).

Let us begin with a high-level description of some of the considerations for useful exploitation of such tunable sensor systems. Effective application of a flexible reduced dimensionality sensor systems starts with identification of a particular realizable projection of the overall measurable space that contains information useful for the pattern recognition task at hand. This step could be realized in a fairly standard way using training data and also may be informed by prior knowledge of phenomenology and application details. At this point, one may then set up and deploy the sensor, thus obtaining information about the world in the form of observations in this measurement space. When the sensor observes a particular situation or scene of interest, we obtain a point in the measurement space whose location provides the information used by the classifier or other pattern recognition algorithms. It is certainly consistent with common experience that the resulting information may be insufficient for unambiguous classification. For example, a particular observation may lie in a region of feature space known to be more liable to misclassification than other regions. In this case, it is possible that the particular disposition of the observation in the initial measurement space may suggest a new projection of the measurable space whose acquisition by the sensor would likely provide new information contributing to an improvement in the pattern recognition performance over that obtainable from the first measurement alone.

This line of inquiry is pushed beyond the realm of thought experiment by taking advantage of the recent engineering advances in adaptive sensor technology. As mentioned before, these are beginning to provide highly agile sensor systems that can be rapidly reconfigured on demand to provide various different looks at a single scene. These we consider to be modeled as low-dimensional projections of the full space of all possible measurement degrees of freedom. In this setting, we seek mathematical methodology for adaptive, sequential selection of the measurement projections to be acquired, based on task-specific metrics and in the context of previous measurements. In a sense, we would like the sensing system to play a good game of "20 questions."

Our approach leads to a new type of decision tree for guiding an agile sensor through the process of acquiring various "looks" at a scene (that is, measurements in various low-dimensional spaces) on the way toward classification of the scene. Some intuition for this concept may be found in the following heuristic: If it is hard to classify the data measured with particular sensor features, the problem may be broken down into smaller pieces by clustering (irrespective of class). Each cluster corresponds to a set of additional sensor measurements providing additional information for classifying data points in that cluster with respect to a particular classifier.

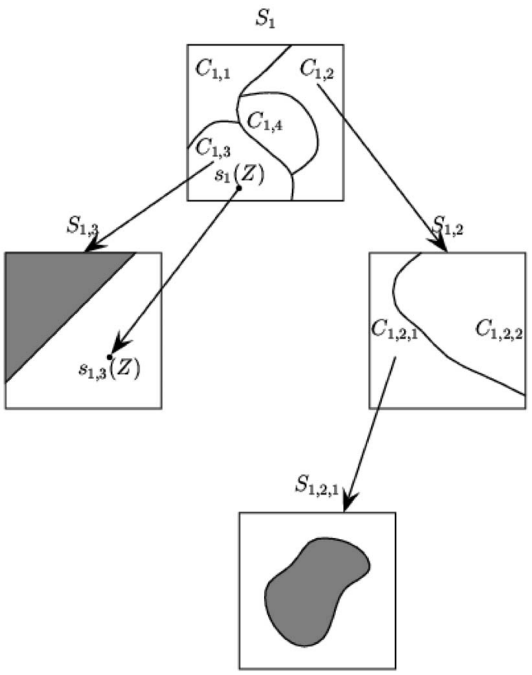


Fig. 1. Adaptive sequential sensing and processing via the ISP Decision Tree: On the top, we have the initial sensor space $S_1 = s_1(\Xi)$ partitioned into $S_1 = \cup_{j=1}^{k_1=4} C_{1,j}$. For observation Z (to be classified), we see that $s_1(Z) \in C_{1,3}$. In the middle left, we have the subsequent sensor space $S_{1,3}$ and we see that a linear classifier is utilized in this space to classify $s_{1,3}(Z)$. In the middle right, we have a partitioning of $S_{1,2} = \cup_{j=1}^{k_1=2} C_{1,2,j}$ and we see that a nonlinear classifier is utilized in $S_{1,2,1}$ (bottom). Shading in leaves indicates classification decision regions. (Spaces $S_{1,1}$, $S_{1,4}$, and $S_{1,2,2}$ are not shown.)

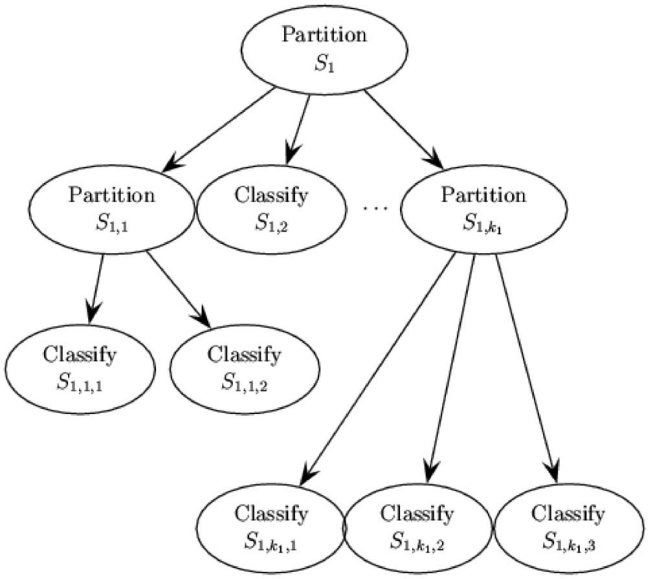


Fig. 2. Adaptive sequential sensing and processing via the ISP Decision Tree: Identify sensor settings and associated sensor space. If the performance of a classifier is adequate, stop; otherwise, partition the sensor space and repeat. At each stage, a (potentially) new set of features is chosen for the task at hand.

The decision tree is constructed (trained) so that its nodes or vertices define sensor "looks," or control settings, s_α, which we identify with particular maps from the field of regard of the sensor into associated low-dimensional feature or measurement spaces S_α. At the root of the tree, the initial sensor setting s_1 is trained for the purpose of getting an informative first look at the scene. We do not necessarily require classification on the basis of this single look, so s_1 is not chosen solely to optimize classification. Instead, this first look is intended to provide information that can guide subsequent looks at the scene in order to improve the ultimate classification. In practice, s_1 is chosen to produce features which allow good partitioning (under some predefined metric) of its associated initial sensor measurement space, S_1 into the disjoint union $\cup C_{1,j}$. See Figs. 1 and 2.

The next level of the tree is populated by daughter nodes, one for each disjoint region $C_{1,j}$ of the partition of initial sensor space. Each daughter node defines a particular sensor setting for a subsequent look $s_{1,j}$, to be obtained by applying the sensor tuned to that setting. The choice of a particular daughter node and its associated second look is determined by the outcome of the initial look: Sensor setting $s_{1,j}$ is used for the second look when the first look landed in the partition cell $C_{1,j}$.

The particular sensor settings $s_{1,j}$ (that is, the local dimensionality identification) and corresponding classifiers (or further partitionings) of their associated feature spaces are determined by invoking a criterion that insures they are most appropriate for initial observations that wind up in partition cell $C_{1,j}$. In the case of either partitioning or classification, a (potentially) different set of second look features is associated with each initial partition cell.

We pursue this idea in order to build up an *ISP Decision Tree* (ISPDT): a decision tree in which the leaves are classifiers and the nonleaf nodes are partitionings (clusterings). Modeling entities or scenes to be sensed as Ξ-valued random variables $X : \Omega \to \Xi$, each node of the tree determines the (potentially unique) surjective sensor setting map $s_\alpha : \Xi \to S\alpha$, onto its associated sensor space $S_\alpha := s_\alpha(\Xi)$.

The identification of these sensor settings, their associated feature spaces, and partition/classifier selections for those spaces are all based on a training data set $D := \{(X_1, Y_1), \cdots, (X_n, Y_n)\}$, where the X_i are Ξ-valued and the class labels Y_i are (say) $\{0, 1\}$-valued. For the partitioning stages, the class labels are ignored—the local dimensionality identification is to be addressed for the general, unlabeled case. Say, for instance, that S_1 is partitioned as $\cup_{j=1}^{k_1} C_{1,j}$ and that the $s_{1,j}$ have been identified. Then, in $S_{1,j}$ the training data set is given by $D_{1,j} := \{(s_{1,j}(X_i), Y_i) : s_1(X_i) \in C_{1,j}\}$. The set $D_{1,j}$ is then used to determine whether $S_{1,j}$ is appropriate for classification. That is, can a classifier $g_{1,j}$ be constructed so that, for unlabeled Ξ-valued observation Z with unobserved class label Y_Z, $P[g_{1,j}(s_1(Z), s_{1,j}(Z)) \neq Y_Z | s_1(Z) \in C_{1,j}]$ is acceptably small. If such a classifier cannot be constructed, then $S_{1,j}$ is further partitioned and the ISPDT grows. The general ISPDT methodology is illustrated in Figs. 1 and 2. This recipe is quite general, leaving the choice of partitioning (clustering) and classification algorithms to the practitioner.

This approach has some similarity to the hierarchical discriminant analysis of Swets and Weng [18]. A key difference is that the first step, defining the clusters or partition cells in which local dimensionality is to take place, may require substantially different features than the

subsequent classification. Other related work in the scientific and engineering literature includes [1], [8], [10], [13], [14].

In the next three sections of this paper, we illustrate the ISPDT concept and demonstrate its efficacy with theoretical, simulation, and experimental results. In particular, our final results clearly indicate the significant potential advantages of using ISPDT to control the operation of a proposed adaptive multichannel olfactory sensor. This result is based on real data collected with an existing version of this sensor and provides a nice example of the exercise of ISPDT in control of an adaptive sensor in order to obtain excellent exploitation performance.

2 Illustrative Theorem

Here, we present an existence proof—the existence of a model for which ISPDT yields zero classification error while any canonical d-dimensional Bayes classifier performs arbitrarily poorly—indicating that the ISPDT is indeed a useful addition to the statistical pattern recognition practitioner's toolbox. While the theorem is stated abstractly, the proof (by construction) is illustrative of the ISPDT "partition, then classify" recipe.

This theorem is in response to the following thought experiment. Consider a sensor which can sense an entity Z one feature at a time and assume that there are resources enough to sense exactly two features; at which point a decision must be made as to the value of the unobserved class label Y_Z associated with Z. For simplicity, we identify Z with a random vector whose components are the measurable features (canonical features). We observe one feature Z_i, and then we may subsequently choose a second feature Z_j to observe. We wish to develop a classifier g for which $L(g) = P[g(Z_i, Z_j) \neq Y_Z]$ is as small as possible.

For $K \geq 2$, let $\mathcal{P}_K$ represent the collection of all K-class classification models. That is, $P \in \mathcal{P}_K$ consists of K class-conditional distributions together with prior probabilities of class membership. Let $L_P^*(d)$ be the Bayes optimal probability of misclassification under P minimized over all d-dimensional canonical marginals—that is, minimized over all subsets of size d of the components of the observation vector Z. Let $L_P(g_{ispdt})$ denote the probability of misclassification under P for ISPDT (with appropriate choices of clusters and classifiers). The following theorem shows that there are situations in which the ISPDT methodology is beneficial. This is accomplished by identifying class-conditional distributions such that the Bayes classifier using any collection of d canonical features results in nonzero classification error while the ISPDT classifier, with the choice of second feature depending on the first observation, yields zero error.

Theorem 2.1. *For any $d \in \mathbb{Z}_+$, $\sup_{P \in \mathcal{P}_K} L_P^*(d) - L_P(g_{ispdt}) = (K-1)/K$.*

Proof. Since $L_P^*(d) \leq (K-1)/K$ for any K-class model and any d, it suffices to demonstrate that, for any $\epsilon > 0$ and any $d \in \mathbb{Z}_+$, there is a model $P = P(d, \epsilon)$ such that $L_P^*(d) - L_P(g_{ispdt}) > (K-1)/K - \epsilon$.

We proceed by construction, for $K = 2$. The case of general K proceeds analogously. Given $\epsilon > 0$ and $d \in \mathbb{Z}_+$, we specify a two-class model P for which $L_P(g_{ispdt}) = 0$ and $L_P^*(d) > 1/2 - \epsilon$.

Let the class label Y be a $\{0,1\}$-valued random variable and let the feature vector $X = [X_0, X_1, X_2, \cdots, X_q]'$ be $\{1, 2, \cdots, 2q\} \times \{0,1\}^q$-valued. Consider class 0 observations $X|_{Y=0} \sim F_0$ with F_0 defined as follows: Let $X_0|_{Y=0} \sim \text{DiscreteUniform}(\{1, 2, \cdots, 2q\})$. For each $i = 1, \cdots, q$, let $X_i|_{X_0=j, Y=0} \sim \text{Bernoulli}(0)$ for $j = 2i-1$, $X_i|_{X_0=j,Y=0} \sim \text{Bernoulli}(1)$ for $j = 2i$, and $X_i|_{X_0=j,Y=0} \sim \text{Bernoulli}(1/2)$ otherwise. Finally, consider (Z, Y_Z) with (unobserved) class label $Y_Z \sim \text{Bernoulli}(\frac{1}{2})$ and $Z|_{Y_Z=0} \sim F_0$, $Z|_{Y_Z=1} \sim F_1$. As constructed, $L^* := P[g_{Bayes}(Z) \neq Y_Z] = 0$.

For ISPDT, let the first feature observed be Z_0. Then, the second feature observed, Z_{Z_0}, depends on the value of Z_0. As constructed, $L(g_{ispdt}) = P[g_{Z_0}(Z_0, Z_{Z_0}) \neq Y_Z] = 0$. Notice that the appropriate clustering is trivial here; since the random variables are all integer valued, we consider each integer to constitute a cluster. Notice also that, as constructed, ISPDT requires only a linear classifier—but that classifier, g_{Z_0}, depends on the first feature observed.

For $\mathcal{I} \subset \{1, \cdots, q\}$, $L_{\mathcal{I}}^*$ (the Bayes optimal probability of misclassification using the associated $|\mathcal{I}|$-dimensional canonical marginal) is $1/2$. For $\mathcal{I} = \{0\} \cup \mathcal{I}'$ with $\mathcal{I}' \subset \{1, \cdots, q\}$, $|\mathcal{I}'| = q' \leq q$, we have $L_{\mathcal{I}}^* = 1/2 - q'/(2q)$. The desired result follows, with $d = q' + 1$ and choosing q such that $q'/(2q) < \epsilon$. □

Notice that the supremum is not achieved. That is, if $L_P^*(d) = (K-1)/K$ for $d \geq 2$, then $L_P(g_{ispdt}) = (K-1)/K$ as well.

This proof illustrates the basic idea of the ISPDT approach. First, one selects a subspace in which a useful partition of the data can be obtained—in our construction, the value observed for Z_0 indicates the best choice for the second feature. Then, a subsequent subspace is defined for each partition cell wherein classification can be performed with adequate precision—in our construction, a linear classifier depending on the value of Z_0 can perfectly classify the two-dimensional observation (Z_0, Z_{Z_0}). The features used for the partitioning and those used in each of the various subsequent classifiers may differ. This is the key to the ISPDT adaptive sequential sensing and processing.

In the sequel, we illustrate this idea with a simulation example and an experimental example.

3 Simulation Example

In this section, we illustrate the ISPDT via a simulation example. The sensor space is six-dimensional, but the sensor is restricted to measuring only two (canonical) variables at a time. This may be due to limitations of the sensor and/or throughput constraints. An ISPDT produces optimal performance.

We specify the joint distribution for this simulation example. We consider a two-class problem with equal priors, so that the class label $Y \sim \text{Bernoulli}(\frac{1}{2})$. The feature space for this example is six-dimensional. For $x \in \Re^2$ and $r \in \Re$, we let $B(x, r)$ denote the (Euclidean) ball of radius r centered at x and $U(B(x, r))$ denote the uniform distribution on $B(x, r)$. For $x \in \Re^2$ and positive definite real matrix Σ, we let $N(x, \Sigma)$ denote the (bivariate) normal (Gaussian) distribution. Let

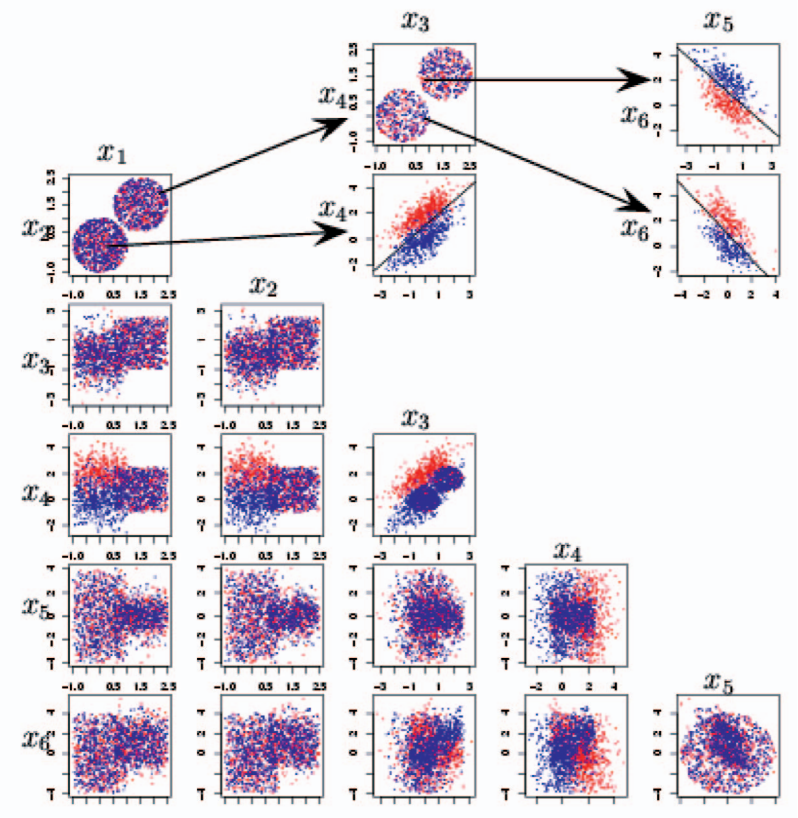


Fig. 3. Scatterplot matrices representing the data for the Simulation Example. (The ISP decision tree is depicted across the top; see also Fig. 4.) In none of the two-dimensional projections of the scatterplot matrix does the data separate well by class, although it does cluster in $[X_1, X_2]'$. The data in $[X_3, X_4]'$ associated with each of the two clusters in $[X_1, X_2]'$ is depicted in the top middle. Note that, conditional on $[X_1, X_2]' \in B([0,0]', 1)$, linear classification is optimal in $[X_3, X_4]'$; however, an additional clustering step is necessary when $[X_1, X_2]' \in B([1.5, 1.5]', 1)$. The data in $[X_5, X_6]'$ associated with each of the two clusters in $[X_3, X_4]'|_{[X_1,X_2]' \in B([1.5,1.5]',1)}$ is depicted in the top right. Again, for these data linear classification is optimal. Note that without performing the additional clustering in $[X_3, X_4]'$ the classes would be completely overlapped.

$$\Sigma_1 = \begin{bmatrix} 1 & \frac{3}{4} \\ \frac{3}{4} & 1 \end{bmatrix}$$

and

$$\Sigma_2 = \begin{bmatrix} 1 & -\frac{3}{4} \\ -\frac{3}{4} & 1 \end{bmatrix}.$$

The first set of two features, $[X_1, X_2]'$, are distributed according to a mixture of two uniforms, independent of the class label Y:

$$[X_1, X_2]' \sim \frac{1}{2} U\big(B([0,0]', 1)\big) + \frac{1}{2} U\big(B([1.5, 1.5]', 1)\big).$$

The distribution of the second set of two features, $[X_3, X_4]'$, depends on the mixture component into which the first two features fall and on the class label Y:

$$\begin{aligned} [X_3, X_4]'|_{[X_1,X_2]' \in B([0,0]',1), Y=0} &\sim N([0,2]', \Sigma_1) \\ [X_3, X_4]'|_{[X_1,X_2]' \in B([0,0]',1), Y=1} &\sim N([0,0]', \Sigma_1) \\ [X_3, X_4]'|_{[X_1,X_2]' \in B([1.5,1.5]',1)} &\sim \frac{1}{2} U\big(B([0,0]', 1)\big) \\ &\quad + \frac{1}{2} U\Big(B([\sqrt{2.5}, \sqrt{2.5}]', 1)\Big). \end{aligned}$$

The third set of two features, $[X_5, X_6]'$, is normally distributed with mean vector and covariance matrix depending on X_1, X_2, X_3, X_4, and on Y:

$$\begin{aligned} [X_5, X_6]'|_{[X_1,X_2]' \in B([0,0]',1)} &\sim U\big(B([0,0]', 4)\big) \\ [X_5, X_6]'|_{\substack{[X_1,X_2]' \in B([1.5,1.5]',1) \\ [X_3,X_4]' \in B([0,0]',1)}, Y=0} &\sim N([0,0]', \Sigma_2) \\ [X_5, X_6]'|_{\substack{[X_1,X_2]' \in B([1.5,1.5]',1) \\ [X_3,X_4]' \in B([0,0]',1)}, Y=1} &\sim N([0,2]', \Sigma_2) \\ [X_5, X_6]'|_{\substack{[X_1,X_2]' \in B([1.5,1.5]',1) \\ [X_3,X_4]' \in B([\sqrt{2.5},\sqrt{2.5}]',1)}, Y=0} &\sim N([0,2]', \Sigma_2) \\ [X_5, X_6]'|_{\substack{[X_1,X_2]' \in B([1.5,1.5]',1) \\ [X_3,X_4]' \in B([\sqrt{2.5},\sqrt{2.5}]',1)}, Y=1} &\sim N([0,0]', \Sigma_2). \end{aligned}$$

Example data (1,000 observations from each class) are depicted in Fig. 3. The pairs plot (scatterplot matrix) is shown in the lower left triangle. The two clusters in $[X_1, X_2]'$ give rise to two different distributions in $[X_3, X_4]'$. In one case, the classes separate well, while in the other a further clustering gives rise to two different distributions in $[X_5, X_6]'$.

The ISP decision tree for this example is depicted across the top of Fig. 3 and again in Fig. 4. The tree is constructed as follows: For each pair of canonical variables (i, j), let $\mathcal{X}_{i,j}$ and

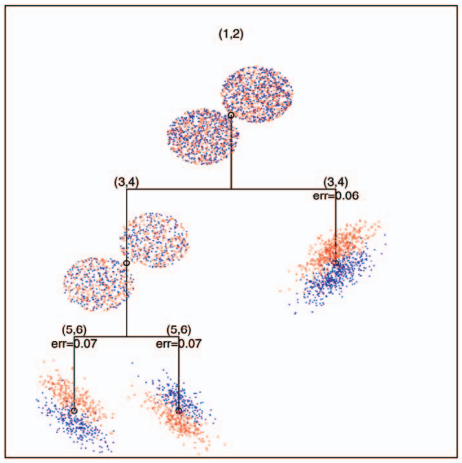


Fig. 4. The ISPDT decision tree for the Simulation Example. (This tree is also depicted across the top of Fig. 3.) For each node, the two features used are indicated numerically. The classifier is linear, the clusterer is 2-means. The features that produce "best clustering," for this tree, are the features which yield the largest distance between clusters (the minimum Euclidean distance between points in separate clusters). Classifier error ("err") is resubstitution error using the linear classifier.

$\mathcal{Y}_{i,j}$ represent the 2-means clustering of the data. Let $\rho_{i,j} := \min_{x \in \mathcal{X}_{i,j}, y \in \mathcal{Y}_{i,j}} d(x,y)$ be the minimum Euclidean distance between points in separate clusters and let $\widehat{L}_{i,j}$ represent the resubstitution estimate of the probability of misclassification for the linear classifier based on the two clusters. At each stage, we check $\min_{i,j} \widehat{L}_{i,j}$ against a threshold (here, we use 0.1). If there is a classifier which performs satisfactorily ($\widehat{L}_{i,j} < 0.1$), we stop and the construction of the tree is complete; otherwise, we choose the pair of canonical variables $(i^*, j^*) := \arg\min_{i,j} \rho_{i,j}$ which provides the most distinct clustering, split the data according to these clusters, and repeat the process for each newly generated branch of the tree.

A Monte Carlo experiment consisting of 100 replications is reported. For each replication, 1,000 training observations are drawn from each class. The ISPDT results in using 2-means clustering on the first two dimensions and then again on the appropriate subset of the third and fourth variates. Normals are then fit to the appropriate subsets so that a linear classifier

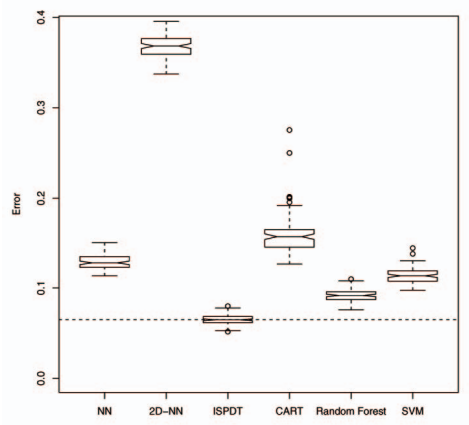


Fig. 5. Boxplots depicting probability of misclassification results for the Simulation Example. The dashed horizontal line is Bayes optimal. The ISPDT procedure yields (asymptotic) Bayes optimality.

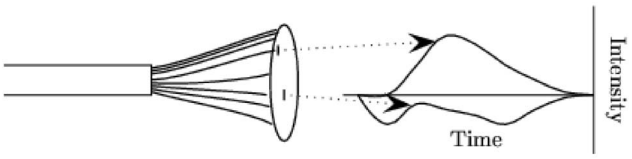


Fig. 6. The Tufts artificial nose consists of optical fibers (shown here spread apart) doped with a solvatochromic dye. Reaction of the polymer matrix with an analyte produces photons that are sampled at two wavelengths to produce a response for each fiber. These photons are captured by a CCD device, resulting in a time series of light intensity above (or below) the background intensity. The responses of two fibers, sampled at a single wavelength, are illustrated as curves in the figure.

results in the leaves. This asymptotically optimal (by construction) ISPDT procedure yields $P[g(Z) \neq Y_Z] \approx 0.0654$; the resultant tree is depicted, via arrows, across the top of Fig. 3. For comparison: A nearest neighbor classifier in the full six-dimensional space yields $P[g(Z) \neq Y_Z] \approx 0.1290$, the best (canonical) two dimensional nearest neighbor classifier yields $P[g(Z) \neq Y_Z] \approx 0.3687$, the CART methodology ([3]) yields $P[g(Z) \neq Y_Z] \approx 0.1588$, a support vector machine (SVM) ([20]) with radial kernels yields $P[g(Z) \neq Y_Z] \approx 0.1143$, a random forests approach ([4]) yields $P[g(Z) \neq Y_Z] \approx 0.0918$, and Bayes optimal performance for this example is $L^* \approx 0.0653$. Standard errors are small relative to the differences in means reported here, see Fig. 5.

This simulation example illustrates the power of the basic idea of the ISPDT; by selecting local regions in which to perform different processing, the ISPDT can improve dramatically over global methods. Note that the decisions in the tree are not made on the optimal (for classification) splits. This allows for a smaller, more flexible tree. Smaller trees mean a smaller number of sensor settings and, hence, more efficient sensors.

4 Experimental Example

The Tufts "artificial nose" is a chemical sensor designed to be nonspecific and cross-reactive. That is, its response to a mixture of odorants is not a linear combination of the responses to the individual odorants nor is its response a simple function of chemical composition or concentration. There is currently no theory describing the response expected from the sensor under any particular scenario. This makes it an interesting sensor from a pattern recognition standpoint since it requires the construction of nonparametric classifiers.

The Tufts sensor consists of a bundle of 19 optical fibers. Each fiber is chemically doped with a solvatochromic dye (see [21]). This doping results in a sensor for which a change in fluorescence intensity is in response to interactions of the dye in each fiber with the chemical environment ([5]). An observation is obtained by passing an analyte (a single compound or a mixture) over the fiber bundle in a four second pulse or "sniff." The information of interest is the change over time in emission fluorescence intensity of the dye molecules for each of the 19 fiber-optic sensors (see Fig. 6).

The task at hand is the identification of an odorant observation. Specifically, we consider the detection of trichloroethylene (TCE) in complex backgrounds. (TCE, a

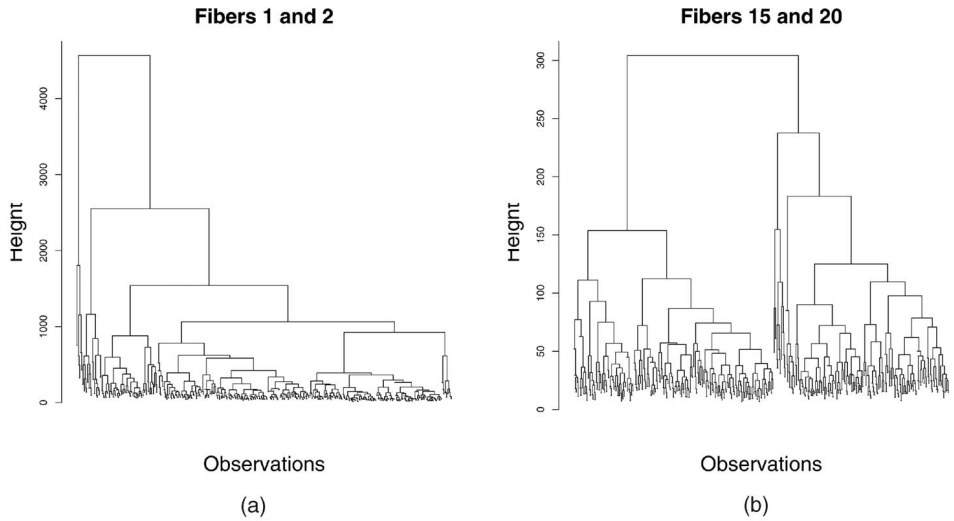


Fig. 7. Dendograms of the clustering of the training data for the nose, using (a) fibers 1 and 2 and (b) fibers 15 and 20.

carcinogenic industrial solvent, is of interest as the target due to its environmental importance as a ground water contaminant.)

The Tufts artificial nose, as with all real sensors, has a finite lifetime, driven mainly by the number of sniffs on a given fiber. Thus, one would like to reduce the number of times any given fiber is used. Instead of using all 19 fibers, an ideal sensor would only use a subset of the fibers at any one time, choosing a different subset depending on the current environment or the problem at hand. The ISPDT provides a methodology to do this, as will now be illustrated.

The data set we will consider here (see [15]) consists of recordings sensor responses to various analytes at various concentrations. Each observation is a measurement of the fluorescence intensity response at each of two wavelengths (620 nm and 680 nm) for each sensor in the 19-fiber bundle as a function of time. While the process is naturally described as functional with time ranging over a 20 second interval, the data as collected are discrete with the 20 seconds recorded at 60 equally spaced time steps for each response. Thus, each observation consists of 2,280 values: 19 fibers at 2 wavelengths sampled 60 times. The sensor responses are inherently aligned due to the "sniff" signifying the beginning of each observation. The response for each sensor for each observation is normalized by subtracting the background sensor fluorescence (the intensity prior to exposure to the analyte) from each response to obtain the change in fluorescence intensity for each fiber at each wavelength.

Construction of the database involves taking replicate observations for various analytes in various concentrations. In addition to TCE in air, eight diluting odorants are considered: BTEX (a mixture of benzene, toluene, ethylbenzene, and xylene), benzene, carbon tetrachloride, chlorobenzene, chloroform, kerosene, octane, and Coleman fuel. Dilution concentrations of 1:10, 1:7, 1:2, 1:1, and saturated vapor are considered. In addition, there are 40 observations of TCE alone, with no confusers. The database contains 352 observations from class 0, the TCE-absent class. These consist of 32 observations of pure air and 40 observations of each of the eight diluting odorants at various concentrations in air. There are likewise 760 class 1 (TCE-present) observations; 40 observations of pure TCE, 80 observations of TCE diluted to various concentrations in air, and 80 observations of TCE diluted to various concentrations in each of the eight diluting odorants in air are available. Thus, there are 1,112 observations in the training database. This database is well designed to allow for investigation of the ability of the sensor array to identify the presence of one target analyte (TCE) when its presence is obscured by a complex background; this is referred to as the "needle in the haystack" problem.

The time series for each (fiber, wavelength) pair on each observation is smoothed using smoothing splines with the smoothing parameter chosen using crossvalidation ([6]). This smoothing has been shown to improve the performance of classifiers in previous work ([15]).

The data set was randomly split into a training set and a test set of equal size (556 observations in each). The problem is to design a classifier (using the training set) that will detect the presence of TCE. The performance of the classifier is then evaluated using the test set.

For this experiment, we employ the nearest-neighbor classifier (as contrasted with the linear classifier used in the Simulation Example). This is for illustration purposes only. However, as will be seen, the performance of the ISP decision tree is quite good even with this simple classifier.

The partitioning algorithm we employ is standard agglomerative complete linkage hierarchical clustering ([7]) (as contrasted with the 2-means clusterer used in the Simulation Example). This produces a dendogram, such as is shown in Fig. 7.

Since we are using the nearest-neighbor classifier for this experiment, we must replace the resubstitution estimate of the probability of misclassification from the Simulation Example with the deleted (crossvalidated) estimate, employed in a greedy manner for up to five fibers. In addition, the ISPDT approach requires a method for selecting "good" clustering, including choosing the fibers on which to cluster. For this experiment, we use an approach based on the

TABLE 1
Cluster Statistics for the Clusters Chosen for the Training Data Using the Dendogram in Fig. 7b

Cluster Size	Classes	Dimensionality	Fibers
207	135 72	5	1 2 3 17 31
228	169 59	3	2 25 34
89	52 37	4	2 8 12 13
14	10 4	1	11
14	12 2	1	8
4	2 2	1	5

dendograms. An inspection of Fig. 7 suggests that fibers 15 and 20 produce more well-defined clusters than fibers 1 and 2; the cluster splits occur at (relatively) larger values of "height" for the second fiber pair than the first. Specifically, we choose the pair of fibers for which the clustering maximizes

$$\frac{1}{(n-1)\max\{h_i\}}\sum_{i=1}^{n-1} h_i,$$

where h_i is the height of dendogram for the ith split—the distance between clusters at which the split occurs. This corresponds to clusters that are more well-defined. We choose the fibers based on this criterion, without consideration of the class labels for the data.

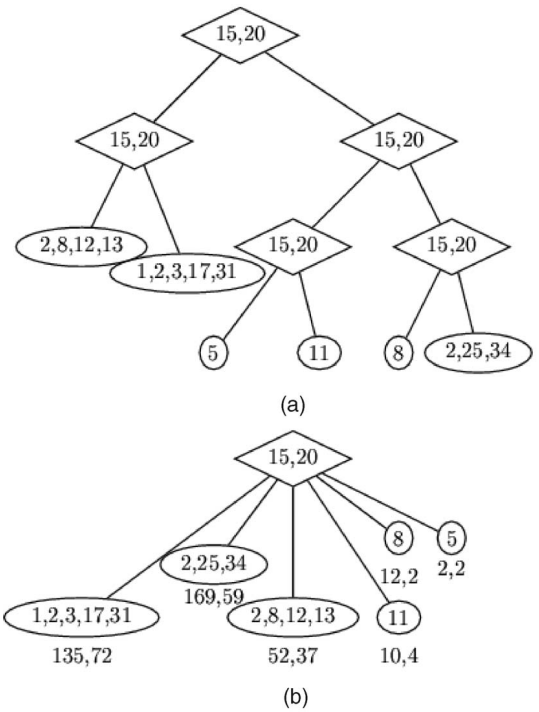


Fig. 8. The ISPDT for the nose data. The top figure represents the binary tree while the bottom represents the collapsed tree. The node labels indicate the fibers used for clustering or classification. The diamond shaped nodes indicates clustering, ovals indicate classification. The number of training observations from each class are indicated below the node in the bottom tree.

TABLE 2
Results of Several Classifiers on the Test Data for the Tufts Artificial Nose

Algorithm	Error
NN: All fibers	0.169
NN: Fibers 15&20	0.302
NN: Best 2 fibers	0.182
kNN: All fibers	0.133
CART	0.167
SVM	0.140
Random Forests	0.124
ISP Decision Tree	0.061

The experiment is as follows: First, we select a "good" clustering using two fibers (in this case, fibers 15 and 20 result in the best clustering). Then, for each cluster we select between one and five fibers, using a sequential search, which provide the best classification under a nearest-neighbor classifier. That is, we select the best single fiber, then the best pair of fibers conditional on one being the best single fiber, etc. We stop this procedure when adding the next fiber produces no improvement, or when we have selected five fibers. For this step, classifier performance is determined by cross-validation on the training observations in the cluster.

The number of clusters is chosen by inspection of the dendogram (Fig. 7b). For this experiment, we chose to use six clusters. (Robustness to this model selection choice is discussed briefly at the end of this section.) The cluster statistics are found in Table 1. For each cluster, the number of training observations is shown, along with the number of observations from each class, the number of fibers chosen by the above algorithm, and the fibers chosen. In this experiment, the clustering methodology, the number of fibers to use in the clustering, the number of clusters, and the maximum number of fibers per cluster were all chosen arbitrarily—with no attempt to choose optimally—for purposes of illustration. It is clear that improvements can be obtained by careful choice of these parameters.

Because we chose to cluster the tree into six clusters initially, the ISPDT tree is only of depth two (see Fig. 8). This is equivalent to a binary tree where each clustering node uses the same fibers (15 and 20). Experiments show that adding the flexibility of choosing different fibers at the clustering nodes does not improve the classification and, so, the shallower tree is preferred.

Results for this experiment are presented in Table 2. The ISP decision tree method is compared with several versions of nearest neighbor, each on a different set of fibers, and with CART, SVM, and random forests on the full data. The errors presented are those computed on the withheld testing set. First, the nearest neighbor result for the entire 38 fiber set is shown, followed by the result on the two fibers selected for the initial clustering, the best two fibers for nearest neighbor classification (chosen via cross-validation on the training set), the k-nearest neighbor classifier using all the fibers optimized over k ($k^* = 3$) and, finally,

the ISP decision tree method. As can be seen, the ISP decision tree yields a significant improvement over these competitors.

One might argue that the results in Table 2 are unfair. After all, the ISP decision tree is really using more than two fibers; in fact, it is using a total of 13 distinct fibers, although it never uses more than seven (the initial two plus at most five) for classifying any single observation. Thus, we should consider the performance of the nearest neighbor using the best 13 fibers for comparison. Since $\binom{38}{13}$ is large, we implemented a greedy algorithm similar to that used within each cluster to find the best j fibers for nearest-neighbor classification, $j = 1, \ldots, 38$, on the full training data set. This greedy-selected optimal fiber combination produces a nearest-neighbor misclassification rate of 0.129 (more than twice the ISPDT misclassification rate).

To explore the question of robustness to selection of the clusters, we investigate the error for different choices of the number of clusters. We varied the number of clusters in the procedure from 2 to 12, using the dendogram of Fig. 7b. Our choice of six clusters turned out to be optimal and all selections produced superior classification results to the full data performance (0.169). These results are (in order from $k = 2, \ldots, 12$): 0.162, 0.090, 0.097, 0.090, 0.061, 0.061, 0.094, 0.094, 0.094, 0.092, 0.092.

5 Discussion

This paper describes a mathematical methodology to implement an adaptive sequential sensing and processing system for classification applications. By selecting different regions for different sensing/processing, ISPDT implements both a local dimensionality identification and local classification similar to standard decision trees. Unlike standard trees, however, the partitioning steps in ISPDT allow branching decisions to be made based on criteria other than classification performance. In some situations, such as that described in the simulation example, a decision tree cannot make the "right" choice for branches on a single variable based on classification alone. Either combinations of variables must be considered or a methodology such as ISPDT must be implemented.

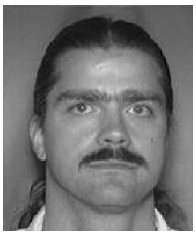
The ISPDT has been shown, via theoretical, simulation, and experimental examples, to be a powerful tool for statistical pattern recognition. It should be stressed that the ISPDT is not an algorithm, but rather a methodology. The issue of partition/classifier selection must be addressed in any real application. The improvement in performance demonstrated in the simulation and the experiment is substantial. It is hoped that further investigation of these ideas will lead to new and powerful classification systems for the next generation of adaptive sensors.

Acknowledgments

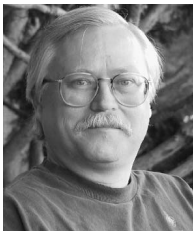
Support for this effort is provided in part by US Defense Advanced Research Projects Agency as administered by the US Air Force Office of Scientific Research under contract DOD F49620-01-1-0395 and the US Office of Naval Research grant N00014-01-1-0011. The artificial nose database used in this paper was provided by David Walt's laboratory at Tufts University. Additional domain expertise was provided by Peter Jurs' laboratory at Penn State University and by John Kauer's laboratory at Tufts University. The authors are grateful to anonymous reviewers for their valuable input.

References

[1] K. Abe and J. Jia, "Improvement of Decision Tree Generation by Using Instance-Based Learning and Clustering Method," *Proc. IEEE Conf. Systems, Man, and Cybernetics,* vol. 1, pp. 696-701, 1996.
[2] R.E. Bellman, *Adaptive Control Processes.* Princeton, N.J.: Princeton Univ. Press, 1961.
[3] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, *Classification and Regression Trees.* Boca Raton, Fla.: Chapman & Hall/CRC, 1998.
[4] L. Breiman, "Random Forests, Random Features," Technical Report 567, Dept, of Statistics, Univ. of California, Berkeley, 1999.
[5] T.A. Dickinson, J. White, J.S. Kauer, and D.R. Walt, "A Chemical-Detecting System Based on a Cross-Reactive Optical Sensor Array," *Nature,* vol. 382, pp. 697-700, 1996.
[6] P.J. Green and B.W. Silverman, *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach.* Chapman and Hall, 1994.
[7] J.A. Hartigan, *Clustering Algorithms.* New York: John Wiley & Sons, 1975.
[8] T. Hastie and R. Tibshirani, "Discriminant Adaptive Nearest Neighbor Classification," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 18, pp. 607-616, 1996.
[9] D.M. Healy Jr., D. Warner, and J.B. Weaver, "Applications of Adapted Wavelet Encoding in Functional Magnetic Resonance Imaging," *Time-Frequency Methods in the Engineering and Biological Sciences,* M. Akay, ed., New York: IEEE Press, 1997.
[10] T.K. Ho, "The Random Subspace Method for Constructing Decision Forests," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 8, pp. 832-844, Aug. 1998.
[11] A.K. Jain, R.P.W. Duin, and J. Mao, "Statistical Pattern Recognition: A Review," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 1, pp. 4-37, Jan. 2000.
[12] Z.P. Liang and P.C. Lauterbur, *Principles of Magnetic Resonance Imaging.* New York: IEEE Press, 2000.
[13] D.J. Marchette and W.L. Poston, "Local Dimensionality Reduction Using Normal Mixtures," *Computational Statistics,* vol. 14, pp. 469-489, 1999.
[14] W. Pedrycz and Z.A. Sosnowski, "Designing Decision Trees with the Use of Fuzzy Granulation," *IEEE Trans. Systems, Man, and Cybernetics, Part A: Systems and Humans,* vol. 30, no. 2, pp. 151-159, 2000.
[15] C.E. Priebe, "Olfactory Classification via Interpoint Distance Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, pp. 404-413, 2001.
[16] C.E. Priebe, D.J. Marchette, and D.M. Healy Jr., "Integrated Sensing and Processing for Statistical Pattern Recognition," *Modern Signal Processing.* D. Rockmore and D.M. Healy Jr., eds., Cambridge Univ. Press, 2004.
[17] D.W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization.* New York: John Wiley & Sons, 1992.
[18] D.L. Swets and J. Weng, "Hierarchical Discriminant Analysis for Image Retrieval," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, pp. 386-401, 1999.
[19] G.V. Trunk, "A Problem of Dimensionality: A Simple Example," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 1, no. 3, pp. 306-307, 1979.
[20] V.N. Vapnik, *The Nature of Statistical Learning Theory.* Springer, 1995.
[21] J. White, J.S. Kauer, T.A. Dickinson, and D.R. Walt, "Rapid Analyte Recognition in a Device Based on Optical Sensors and the Olfactory System," *Analytical Chemistry,* vol. 68, pp. 2191-2202, 1996.



Carey E. Priebe received the BS degree in mathematics from Purdue University in 1984, the MS degree in computer science from San Diego State University in 1988, and the PhD degree in information technology (computational statistics) from George Mason University in 1993. From 1985 to 1994, he worked as a mathematician and scientist in the US Navy research and development laboratory system. Since 1994, he has been a professor in the Department of Applied Mathematics and Statistics, Whiting School of Engineering, Johns Hopkins University, Baltimore, Maryland. His research interests are in computational statistics, kernel and mixture estimates, statistical pattern recognition, and statistical image analysis. At Johns Hopkins, he holds joint appointments in the Department of Computer Science and the Center for Imaging Science. He was elected fellow of the American Statistical Association in 2002. He is member of the IEEE Computer Society.



David J. Marchette received the BA degree in 1980 and the MA degree in mathematics in 1982, from the University of California at San Diego. He received the PhD degree in computational sciences and informatics in 1996 from George Mason University under the direction of Ed Wegman. From 1985-1994, he worked at the Naval Ocean Systems Center in San Diego doing research on pattern recognition and computational statistics. In 1994, he moved to the Naval Surface Warfare Center in Dahlgren, Virginia, where he does research in computational statistics and pattern recognition, primarily applied to image processing, automatic target recognition, and computer security.

Dennis M. Healy Jr. received the bachelors degree in physics and mathematics from the University of California at San Diego (UCSD) in 1980 and a doctorate in mathematics from UCSD in 1986. He is a professor of mathematics at the University of Maryland, a program manager for the Microsystems Technology Office at DARPA, and a program consultant for NIH/NIAAA. In a previous millennium, he served as program manager for DARPA's Applied and Computational Mathematics Program in the Defense Sciences Office and as associate processor at Dartmouth College with joint appointments in the Departments of Mathematics and Computer Science. His research concerns applied computational mathematics in real-world settings including medical imaging, optical fiber communications, design and control of integrated sensor systems, statistical pattern recognition, and fast nonabelian algorithms for data analysis.