Heuristic Classifier Performance Bounds in High Dimensional Settings

Dr. Paul M. Baggenstoss *
Naval Undersea Warfare Center
Newport RI, 02841
401-841-7505 x 38240 (TEL)
401-841-7453 (FAX)
p.m.baggenstoss@ieee.org (EMAIL)
EDICS 6.1.6 / 3.5 / 6.17

March 12, 2002

Abstract

This paper is concerned with probability density estimation in high-dimensional settings. Simplified geometric arguments and supporting examples point to a performance bound which limits algorithm performance to that of either (1) nearest-neighbor or (2) single-kernel PDF estimators. A method of monitoring PDF estimation performance as well as recommendations for neural net and classification algorithm practitioners is provided.

1 Introduction

The subject of this paper is non-parametric probability density function (PDF) estimation in high-dimensional settings. As such, it is relevant to signal processing, estimation, and classification.

We refer herein to the data dimension as P, the largest number of random variables to be considered at once, i.e. in a joint distribution. When the form of the PDF to estimate is entirely unknown and must be estimated from training data, the PDF value itself at quantized "grid cell" locations effectively become parameters to estimate. The number of parameters effectively rises exponentially with P. The rapid increase in complexity of systems has been termed the *curse of dimensionality* by Richard Bellman [1]. Indeed, it has been shown that given that the PDF meets certain smoothness assumptions, the amount of training data required for nonparametric estimators rises exponentially in P [2]. It is conjectured by some researchers that the underlying structure of the data in most problems has a dimension rarely larger than about 4 or 5 [3]. Thus, some kind of transformation or projection onto a lower dimensional manifold is recommended. The most obvious method of dimension reduction is simply to discard the least important features, a process called *feature selection* a well-studied problem in itself [4], [5]. The dilemma is that the true information content of a feature cannot be measured without a good joint PDF estimate of *all* the features. On the other hand, a good PDF estimate cannot be obtained at a high dimension. As features are added, increasing P, it is possible that algorithm performance may actually get worse in spite of information content of the added feature. Likewise, eliminating an information-bearing feature may actually improve performance. Dimension reduction is a subject of ongoing research [6], [7], [3], [8]. That dimensionality is an overriding problem may at first contradict the fact that PDF estimators have been employed successfully at very high dimension. In this paper, we argue that in such applications, true PDF estimation is not happening. What is really happening is explained below in the context of one of two possible *primitive* PDF estimators.

Projection of the PDF estimate onto one or two dimensions is an effective method of monitoring PDF accuracy. If the low-dimensional projections are flawed, so must be the PDF estimate on $\mathcal{R}^P$. But accurate projections do not guarantee good PDF estimates. It is easy to be "fooled" by the apparent size of the kernels. An example that illustrates this phenomenon is the following [3]. Let the marginal distribution of each data dimension be distributed uniformly within the

*This work supported by Office of Naval Research

interval $[-1,\ 1]$. The data is therefore contained in the hypercube $[-1,\ 1]^P$. Imagine a radius-1 hypersphere inscribed inside the hypercube, touching the center of each "face". As P grows, the fraction of data that falls inside the hypersphere falls exponentially to zero. This is counter-intuitive because the inscribed hypersphere is very large when projected onto any axis. We suggest an alternate approach that is not as easily fooled.

The object of this paper is not to offer a solution to the dimensionality problem, but to offer an explanation for its existence and argue that it is not solvable when attacked as a high-dimensional problem. Solutions are offered in other papers [9], [10].

We begin the paper with a geometric argument to expose the nature of the curse of dimensionality. This argument is concisely represented by heuristic bound on performance. This provides a setting for the remainder of the paper. Next, we suggest methods of assessing PDF accuracy that do not rely on classifier performance. We end the paper with our concluding remarks and recommendations.

2 Heuristic Performance Bounds

The impetus for this work was that in high dimensional problems, it was found from experience that:

1. Complex classifiers rarely work better than simple classifiers (Fisher's linear discriminant, quadratic Bayesian, or Nearest Neighbor classifiers).

2. Simple classifiers tend to improve as the feature set dimension increases.

3. Complex classifiers improve with increasing dimension initially, but then performance stops improving, or drops as dimension increases. As dimension continues to increase, performance sometimes improves.

The arguments to be presented attempt to explain this behavior using simple geometric arguments. By no means do we attempt to find quantitative results that can be verified experimentally. However, because the arguments are basic, it should be clear to the readers whether or not the assumptions are applicable to their problems.

1. This paper assumes that performance of algorithms is dependent only on PDF estimation. Clearly classification performance depends only on accurate PDF estimation in the boundary regions between signal classes. Yet, it can be argued that if we limit ourselves to speaking about these localized regions only, the same arguments hold.

2. We make arguments based on kernel-based PDF estimation. It can be said that our arguments do not apply to other methods. However consider that:

   (a) the minimum probability of error is achieved in theory by the probabilistic Bayesian classifier which requires the PDF [11],

   (b) kernel-based PDF estimators converge to the true PDF given smoothness of the PDF, enough data and enough kernels [12],

   (c) Most Neural Networks actually are PDF estimators [13].

To develop a language for the arguments to be presented, we consider four broad classes of PDF estimators:

1. **Variable Basis Function (VBF)** The PDF is approximated as a sum of positive basis functions. It is assumed that the approximation algorithm maximizes the approximation fit to the training data by determining the best set of basis function locations and individual shapes and sizes.

2. **Uniform Basis Function (UBF)** The PDF is approximated as a sum of positive basis functions with the constraint that all basis functions are identical except for location. This includes PNN and homoscedastic Gaussian mixtures and applies in a broad sense to to the case when all basis functions have the same *volume*, for example strophoscedastic mixtures [14]. It is assumed that the approximation algorithm maximizes the approximation fit to the training data by determining the best set of basis function locations and overall basis function size.

3. **Single Basis Function (SBF)** This is the special case of the first two methods where there is only one basis function. It is assumed that the approximation algorithm maximizes the approximation fit to the training data by determining the best single basis function (location, shape and size). Examples are the quadratic-Bayesian classifier or Fisher's linear discriminant.

4. **K-Nearest Neighbor (KNN)** The probability density is estimated from the volume of a ball (or other basis function) that contains K samples from the training set.

In this paper we use the terms *kernel* and *basis function* interchangeably.

Consider the problem of estimating an arbitrary PDF from N data samples of a random variable $\mathbf{x}$ using a UBF PDF estimate. The PDF is then used in an algorithm whose performance may be quantified. Consider $\mathbf{x}$ to be infinite-dimensional (such data may be formed by adding random non-informative data to each sample of a finite-dimensional data set). Let $\mathbf{x}^P$ be the first P dimensions of $\mathbf{x}$, with PDF $p_P(\mathbf{x}^P)$. Let $I(p_P)$ define the expected performance of the algorithm when $p_P(\mathbf{x}^P)$ is used. Clearly,

$$I(p_P) \geq I(p_Q); \quad P > Q.$$

Let $\hat{p}_P(\mathbf{x}^P)$ be an estimate of $p_P(\mathbf{x}^P)$ derived from N data samples. Assume that

$$I(\hat{p}_P) \leq I(p_P).$$

Let the probability mass of $p_P(\mathbf{x}^P)$ be confined (mostly) to the P-dimensional hypercube defined by $0 \leq x_i \leq D, \quad i = 1, \ldots, P$ and let d_0 be the smallest dimension of local variations in $p_P(\mathbf{x}^P)$. The determination of d_0 is illustrated in Figure 1. In the figure, d_0 is determined separately for each dimension by the smallest variation or "peak" in a sectional slice at a fixed value of the other dimension(s). It is the largest kernel width in that dimension that a UBF estimator of the PDF could have and still provide an accurate approximation to the true PDF. For simplicity, we assume it is the same for each dimension, i.e. $d_{0,i} = d_0$. Let $Q(N, P, M, d)$ be the expected algorithm performance when the PDF

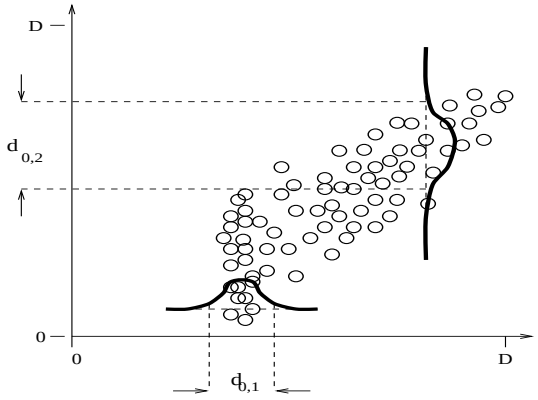

Figure 1: Determination of d_0 from data samples.

is approximated using a UBF mixture with M mixture components of diameter d, and obtained from N data samples. The *Heuristic Performance Bounds* state that

$$\begin{aligned} Q(N, P, M, d) &\leq I(p_P)\ A(d) \\ Q(N, P, M, d) &\leq I(p_P)\ V(M, d, P). \end{aligned} \tag{1}$$

We now describe each term.

- We have described $I(p_P)$, the *ideal* expected performance, above.
- $A(d)$ is the *Potential Accuracy* for M infinite. This term represents the loss due only to the diameter of the mixture components, d exceeding d_0 (i.e., for M infinite). $A(d)$ is such that

$$\begin{aligned} \lim_{d \to d_0} A(d) &= 1 \\ A(d_1) &\leq A(d_2), \quad d_2 < d_1 \\ \lim_{d \to D} A(d) &= \alpha > 0 \end{aligned}$$

The last statement says that performance does not go to zero, even for very large kernel diameters. On the contrary, performance approaches that of a SBF PDF estimator. An example of a function meeting these requirements is

$$A(d) = \left[\alpha + (1-\alpha)\frac{d_0/d}{1+d_0/d}\right]. \tag{2}$$

- $V\,(M,d,P)$ represents the loss due only to volume coverage. Clearly the volume occupied by the approximation is no greater than Md^P. Let volume occupied by $p_{\mathcal{P}}(\mathbf{x}^P)$ be denoted $V_0 = (\gamma D)^P$, where $0 < \gamma \leq 1$. The point at which the volume of the approximation matches the volume occupied by $p_{\mathcal{P}}(\mathbf{x}^P)$ is when $Md^P = V_0$. Define this point to be $d^* \triangleq \left(\frac{V_0}{M}\right)^{\frac{1}{P}} = \gamma D M^{-\frac{1}{P}}$. Thus,

$$\begin{aligned}
\lim_{d\to D} V(M,d,P) &= 1 \\
V(M,d_1,P) &\leq V(M,d_2,P), \quad d_2 > d_1 \\
\lim_{d\to 0} V(M,d,P) &= \beta > 0
\end{aligned}$$

The last statement says that performance does not go to zero, even for very small diameters. On the contrary, performance approaches that of a KNN classifier. We have assumed that M is essentially fixed by the amount of available data, N. The amount of data needed to estimate the parameters of each mixture component is approximately linear in P (e.g. [15]). This weak dependence may be ignored. It should be expected that $V\,(M,d,P)$ drops *very rapidly* to β as d falls below d^*. An example of a function meeting these requirements is

$$V\,(M,d,P) = \left[\beta + (1-\beta)\frac{Md^P}{Md^P + (\gamma D)^P}\right].$$

We have mentioned already the work of Stone [2] which derives the fact that accurate PDF estimation requires N to increase exponentially in P. This corroborates the above analysis if we suppose that the amount of training data is roughly proportional to M, say $N = kM$. In order to remain below the critical point of volume collapse, we need $N \geq k\left(\frac{\gamma D}{d}\right)^P$. Some researchers have derived similar expressions based on special cases [6],[3], [16].

At this point, it is possible to make a very important observation: $A(d)$ falls as d *increases* toward D, but $V(M,d,P)$ falls *rapidly* as d *decreases* below d^*. But, $d^* \triangleq \left(\frac{V_0}{M}\right)^{\frac{1}{P}} = \gamma D\left(\frac{1}{M}\right)^{\frac{1}{P}}$ approaches γD as P rises. This is illustrated in Figure 2. So, it may be concluded that as P increases, performance becomes limited to either α or β, depending on the

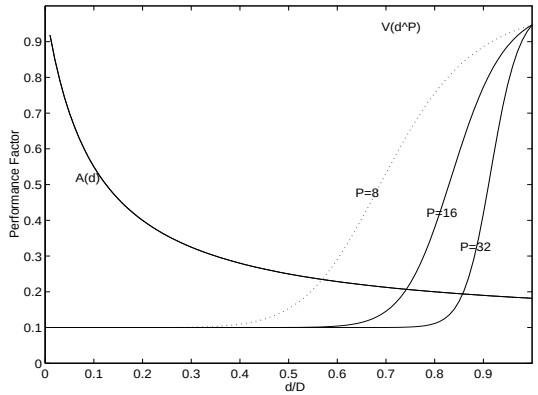

Figure 2: Ideal performance can never be achieved for high P. Either Potential Accuracy or Volume Coverage is small. For this plot, $M = 16,\ \beta = \alpha = 0.1,\ d_0 = .1, \gamma = 1$

value of d. We call the condition when $d < d^*$ and performance approaches β the *collapsed kernel* (CK) effect. We call the condition when performance approaches α the *expanded kernel* (EK) effect. We call this the *fundamental tradeoff*, the

tradeoff associated with *oversmoothing* vs. *undersmoothing*. While many practitioners attempt to find a compromise, we argue that at high dimensions, one can never escape suffering the brunt of one or the other.

It is possible to relate CK and EK effects to "primitive" classifiers such as SBF and KNN. The performance of an SBF classifier is α because the SBF kernel is matched to the volume of the data PDF. The performance of a KNN classifier is β. A UBF or VBF classifier with very narrow basis functions ($d < d^*$) and a kernel located at each training sample is equivalent to a KNN. It can be argued that UBF and VBF classifiers with more than one training sample per kernel still have a similar behavior because of the narrowness of the kernels. From this standpoint, it is clear why some complex Neural Networks work no better that simple SBF or KNN classifiers.

In classification problems, the values of α and β depend on the cluster-shapes and relative locations of clusters in the high-dimensional space.

1. If the classes are well-separated and unimodal, α and β may both be high and either an SBF or Nearest-Neighbor classifier will work well.
2. If they are unimodal but not well separated, α will be high, but β will be low and SBF classifier is recommended.
3. If their shapes are complex, yet are well separated, β will be high, but α will be low and a Nearest-Neighbor classifier is recommended.
4. In practice, the above two conditions may exist simultaneously in different parts of the data space because real-world data is not homogeneous. VBF classifiers may also exhibit a little of both conditions.

3 Supporting Example

Virtually all PDF estimation methods (including neural nets [13]) are consistent in the sense that there is a certain tendency to converge to the true PDF estimate given enough training data and/or low enough dimension. There are numerous methods of PDF estimation (for comparative studies and overviews see [4], [3], [17]). In this example, we utilize a multivariate PDF estimation approach based on a heteroscedastic Gaussian mixture (GM) approximation. A widely accepted technique for estimating the parameters of the GM model is the EM algorithm [11],[18]. The EM algorithm suffers from numerical problems when there is insufficient data leading some researchers to avoid it [17] or constrain the covariances of the kernels to be identical [19], or of uniform size with variable rotation [14]. Adding to the covariance estimates based on a Bayesian prior density argument is the preferred method of dealing with the problem [20], [21]. This involves simply adding a diagonal matrix, representing an independent measurement noise prior, to the kernel covariances at each iteration. We have obtained excellent results with this method.

When the PDF is estimated by optimizing the likelihood function, such as an EM algorithm, the total likelihood of the training data is maximized. Therefore, kernels are caught in opposing forces. Kernels must become smaller to increase their likelihood value, but larger to encompass more data. As P increases vith N and M fixed, a given number of training samples occupies an exponentially decreasing fraction of the data volume. Kernels tend to become associated with disjoint "subsets" of training data and tightly enclose the data subsets in order to maximize the likelihood. In the limit, the data "subsets" occupy subspaces of zero volume. However, depending on how the algorithm is initialized, these subspaces have arbitrary orientations and therefore appear "wide" when projected onto a given 2-dimensions. As a visual example, consider a flat disk-shaped kernel in a 3-dimensional volume enclosing 3 widely separated training data points in a plane. When projected onto 2 dimensions, unless the two basis functions upon which it is projected are aligned just right, it will appear wide. Thus, it makes sense that unless kernel width is constrained, the algorithm will tend toward CK effect, yet the projection of the PDF approximation onto 2 dimensions will widen. The use of Bayesian priors, as mentioned above, prevents the kernels from collapsing to zero volume, but the fractional volume still decreases exponentially.

Figures 3 through 5 illustrate this effect. The upper portion of each plot is a scatter diagram of the data samples available in the training set. The lower portion is a contour plot of the individual kernels at a fixed level. The Gaussian Mixture is marginalized (effectively integrated over all the remaining dimensions). The figures represent a sequence of increasing P. The number of kernels is 5 in all cases. The apparent kernel widths gradually widen.

4 Monitoring PDF Accuracy

We have argued that PDF approximation is the weak link in high-dimensional algorithm performance. But how is it possible to monitor PDF approximation accuracy without knowing the true PDF? In this section, we present some ideas.

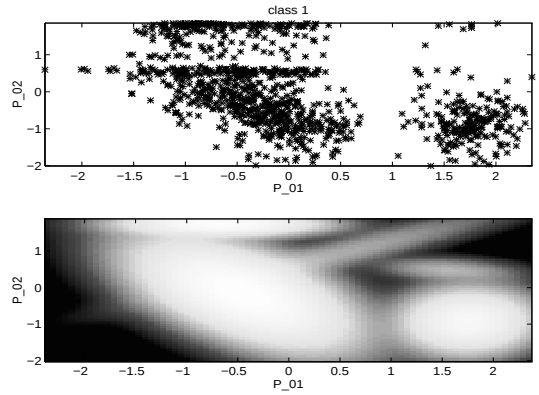

Figure 3: Gaussian Mixture approximation estimated at $P = 2$.

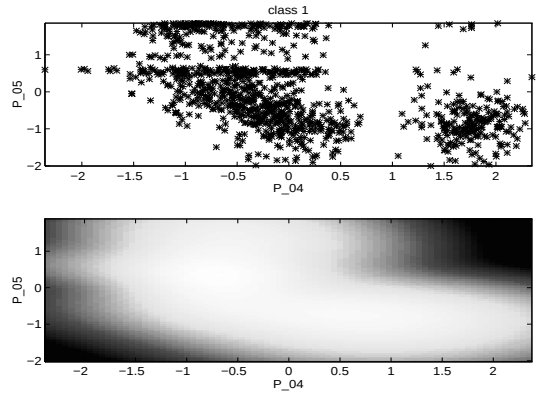

Figure 4: Gaussian Mixture approximation estimated at $P = 14$ and projected onto two dimensions.

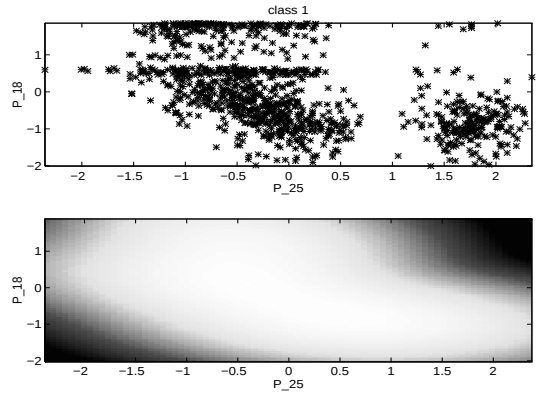

Figure 5: Gaussian Mixture approximation estimated at $P = 44$ and projected onto two dimensions.

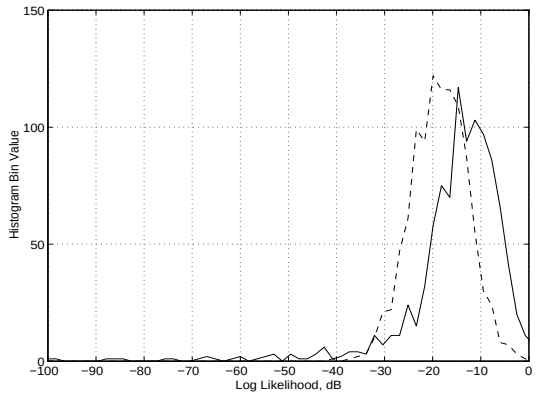

Figure 6: Histograms of the Log Likelihood for Testing data and Synthetic data, $P = 44$. Solid: testing, dashed: synthetic.

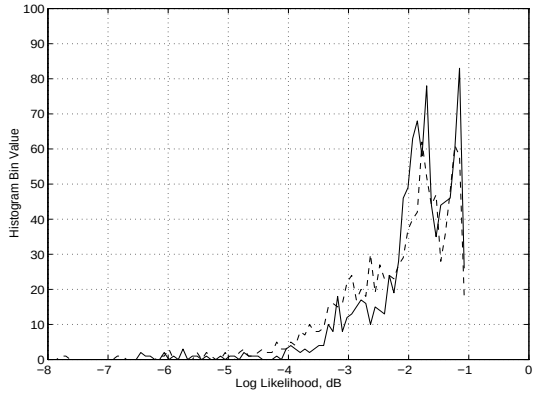

Figure 7: Histograms of the Log Likelihood for Testing data and Synthetic data, $P = 2$. Solid: testing, dashed: synthetic.

In one and two dimensions, it is possible to create plots such as Figures 3 to visually inspect the PDF approximation comparing with actual data histograms. But in higher dimensions, it is no longer feasible. It is necessary to project the multidimensional data and PDF approximation down to one or two dimensions to visualize it. But still, such methods depend on the choice of projection.

An idea we like is the following: Let there be 3 data sets, training data $\mathbf{x}^{Tr}$, testing data $\mathbf{x}^{Te}$, and synthetic data $\mathbf{x}^{Sy}$. After obtaining a PDF estimate $\hat{p}(\mathbf{x})$ from $\mathbf{x}^{Tr}$, generate a population of synthetic data $\mathbf{x}^{Sy}$ based on $\hat{p}(\mathbf{x})$. Then, compare histograms of $\log \hat{p}(\mathbf{x}^{Te})$, and $\log \hat{p}(\mathbf{x}^{Sy})$. Comparing these histograms can locate telltale problems. If $\hat{p}(\mathbf{x})$ is a good approximation to $p(\mathbf{x})$, the two histograms will match. However, if CK effect occurs, $\log \hat{p}(\mathbf{x}^{T})$ will exhibit a very wide spread of values including some very large negative samples. If the testing data falls inside the narrow kernels, the likelihood will be too high, if it falls outside, it will be much too low. If EK effect were to occur, the spread of values would be narrow. Furthermore, EK effect would be quite visible in the 2-D projections.

Consider Figure 6, created for a PDF estimate of dimension 44 (see Figure 5). This shows the testing data has both higher and lower values of likelihood than the synthetic data, indicating CK effect.

The experiment was repeated for the case of Figure 3 ($P = 2$) and the result is plotted in Figure 7. The histograms are much better matched indicating a better PDF estimate. The method compresses an enormous amount of information into a single plot and may provide the basis of non-subjective measures of fit such as total log-likelihood of testing data. This method is not foolproof since the matching of the log-likelihood distributions is not a guarantee of a correct PDF estimate. Combined with the 2-D projection method, however, it is a powerful test.

5 Conclusions and Recommendations

Starting from the premise that dimensionality plays a central role in algorithm performance, a approximate "conceptual" equation of dimensionality has been presented. The behavior of classifiers at higher dimension has been explained in the context of this relationship. Specifically, classifiers show a tendency to perform similarly to one of two "primitive" classifiers: a nearest neighbor classifier or a single-kernel classifier, or somewhere between the two.

We highly recommend doing three things whenever a new PDF estimator is evaluated: (1) Plotting marginalized PDF intensity and data scatter diagrams in 2 dimensions or data histograms together with marginalized PDF curves, (2) plotting KNN and SBF performance on the same graph as any new algorithm performance, (3) plotting histograms of log-likelihood for the two data populations: testing data and data synthesized from the trained PDF model. This will (a) locate PDF approximation errors, (b) put performance in the perspective of "primitive" methods (c) and determine whether CK or EK effects are happening.

6 Acknowledgements

The author would like to acknowledge the contributions of Stephen Greineder for help in developing equation (1) and other ideas. The ideas in this paper had their origin with the author's previous work at Raytheon in active classification [22] [23]. The problems associated with classification of short duration events in which there was an abundance of good features but a dearth of training data provided motivation for more comprehensive treatment. The ideas expressed in this paper took shape in a series of discussions including primarily the author, Stephen Greineder, and Tod Luginbuhl (thanks to Tod for helping sound out ideas and providing a wealth of information in mathematical statistics and current literature).

References

[1] R. E. Bellman, *Adaptive Control Processes*. Priceton, New Jersey, USA: Princeton Univ. Press, 1961.

[2] C. J. Stone, "Optimal rates of convergence for nonparametric estimators," *Annals of Statistics*, vol. 8, no. 6, pp. 1348–1360, 1980.

[3] D. W. Scott, *Multivariate Density Estimation*. Wiley, 1992.

[4] Duda and Hart, *Pattern Classification and Scene Analysis*. Wiley, 1973.

[5] S. J. Raudys and A. K. Jain, "Small sample size effects in statistical pattern recognition: Recommendations for practitioners," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, no. 3, pp. 252–264, 1991.

[6] N. Intrator, *Feature Extraction Using an Exploratory Projection Pursuit Neural Network*. PhD thesis, Brown University, 1991.

[7] S. Aeberhard, D. Coomans, and O. de Vel, "Comparative analysis of statistical pattern recognition methods in high dimensional settings," *Pattern Recognition*, vol. 27, no. 8, pp. 1065–1077, 1994.

[8] A. Finch, "A neural network for dimension reduction and application to image segmentation," in *Proceedings of the 1994 International Conference on Artificial Neural Networks (ICANN-94)*, pp. 252–264, 1994.

[9] P. M. Baggenstoss, "Structural learning for classification of high dimensional data," in *Proceedings of the 1997 International Conference on Intelligent Systems and Semiotics*, pp. 124–129, National Institute of Standards and Technology, 1997.

[10] P. M. Baggenstoss, "Class-specific features in classification.," *IEEE Trans Signal Processing*, pp. 3428–3432, December 1999.

[11] R. L. Streit, "A neural network for optimum Neyman-Pearson classification," in *Proc. International Joint Conference on Neural Networks*, (San Diego, California), pp. 685–690, June 1990.

[12] Anderson and Moore, *Optimal Filtering*. PH, 1979.

[13] D. W. Ruck, S. K. Rogers, M. Kabrisky, M. E. Oxley, and B. W. Suter, "The multilayer perceptron as an approximation to a Bayes optimal discriminant function," *IEEE Trans. on Neural Networks*, vol. 1, no. 4, pp. 296–298, 1990.

[14] R. L. Streit and T. E. Luginbuhl, "Estimation of Gaussian mixtures with rotationally invariant covariance matrices," *Communications in Statistics: Theory and Methods*, vol. 26, December 1997.

[15] R. L. Streit, "An upper bound on feature vector dimension as a function of design set size for two Gaussian populations," *Naval Undersea Warfare Center, New London Conn, Technical Memorandum TM 921048*, Mar 1992.

[16] D. Psaltis, R. R. Snapp, and S. S. Venkatesh, "On the finite sample performance of the nearest neighbor classifier," *IEEE Trans. on Information Theory*, vol. 40, no. 3, pp. 820–837, 1994.

[17] J.N.Hwang, S.R.Lay, and A. Lippman, "Nonparametric multivariate density estimation: A comparative study," *IEEE Trans SP*, vol. 42, pp. 2795–2810, October 1994.

[18] L. I. Perlovsky, "A model-based neural network for transient signal processing," *Neural Networks*, vol. 7, no. 3, pp. 565–572, 1994.

[19] R. L. Streit and T. E. Luginbuhl, "Maximum likelihood training of probabilistic neural networks," *IEEE Trans. on Neural Networks*, vol. 5, pp. 764–783, 1994.

[20] D. M. Titterington, A. F. M. Smith, and U. E. Makov, *Statistical Analysis Of Finite Mixture Distributions*. John Wiley & Sons, 1985.

[21] G. J. McLachlan, *Mixture Models*. Dekker, 1988.

[22] P. M. Baggenstoss, "Improved echo classification in shallow water," in *OCEANS 96 MTS/IEEE Proceedings*, pp. 1197–1203, IEEE, 1996.

[23] P. M. Baggenstoss, "Improved active classification incorporating spatial energy distribution of reverberation and environmental adaptation.," *U.S. Navy Journal of Underwater Acoustics (Accepted for publication)*, 1997.