The Chain-Rule Processor: Optimal Classification Through Signal Processing

Dr. Paul M. Baggenstoss *†
Naval Undersea Warfare Center
Newport RI, 02841, USA
p.m.baggenstoss@ieee.org
http://www.npt.nuwc.navy.mil/csf/index.html

Abstract

The chain-rule processor is a method of constructing an optimal Bayes classifier from a bank of processors. Each processor is a feature extractor designed to separate the given class from a class-dependent reference hypothesis, thereby avoiding the curse of dimensionality. This work builds upon prior work in optimal classifier design using class-specific features. The chain-rule processor is an improvement that recursively applies the PDF projection theorem.

1 Introduction

The so-called M-ary classification problem is that of assigning a multidimensional sample of data $\mathbf{x} \in \mathcal{R}^N$ to one of M classes. The statistical hypothesis that class j is true is denoted by H_j, $1 \leq j \leq M$. The statistical characterization of $\mathbf{x}$ under each of the M hypotheses is described completely by the probability density functions (PDFs), written $p(\mathbf{x}|H_j),\ 1 \leq j \leq M$. Unfortunately, the PDFs are often unknown in many problems and must be approximated from training data.

The high dimension of the raw data usually precludes estimating the PDFs without knowing the parametric form beforehand. Because the classical theory does not offer any solutions to this problem, many practitioners are forced to extract a set of information-bearing features of lower dimension from the raw data, then abandon the raw data altogether by recasting the problem as though the features *were* the raw data. We contend that in problems of high complexity, this drastic step unnecessary. In problems of high complexity, where there are many statistical hypotheses, the dimension of the feature space that has enough information to optimally separate all the classes grows with the number of classes. But with a fixed amount of training data, the PDF of the features can be difficult to estimate above a very small dimension (usually about 5). Therefore, classifier performance can be severely limited either because of insufficient information in a low-dimensional feature set, or by unreliable PDF estimation at high dimensions. The root of the problem is that we seek a *common* feature space where all classes are separable. Any feature that has information pertaining to just one or two classes must be thrown into the common feature set regardless of whether it is informative about any other classes. The added feature dimensions are often completely irrelevant to some of the classes, yet these dimensions must be estimated under all hypotheses - and this can lead to very poor PDF estimation. Fortunately, there is a theoretical solution to this dilemma, which we now present.

*This was work supported by the Office of Naval Research.
†The author would like to acknowledge Dr. Roy Streit for his valuable advice.

2 The PDF Projection Theorem

In this section, we describe the theorem that makes the class-specific method possible. In Section 3, we use the theorem to construct a classifier.

2.1 Summary of the Theorem

For those readers who would like to skip most of this section, we summarize the main result as follows. The PDF projection has nothing to do with linear or nonlinear projection from high-dimensions to low dimensions. Instead, it has to do with estimating the PDF of a select set of features - only the relevant features necessary to characterize a given class - then *projecting* the PDF of the *low dimensional* features back to the *high dimensional* input raw data space. That is, constructing a function of the raw data from the feature PDF that not only is a PDF so it integrates to 1 on the raw data space, but is also a good approximation to the desired class PDF. This completely eliminates the need to construct decision boundaries in a feature space and makes

possible direct implementation of the optimal Bayes classifier on the raw data space. Let H_1 be any data class hypothesis. Let $\mathbf{z}_1 = T_1(\mathbf{x})$ be a set of features that describes class H_1 - such as parameter estimates of the relevant parameters that govern the data for that class. Then, the projection operation is written

$$\hat{p}_x(\mathbf{x}|H_1) = Q(\mathbf{x}, T_1, H_0)\, \hat{p}_z(\mathbf{z}_1|H_1) \tag{1}$$

where $\hat{p}_z(\mathbf{z}_1|H_1)$ is the estimated low-dimensional feature PDF of $\mathbf{z}_1$ and $\hat{p}_x(\mathbf{x}|H_1)$ is the projected PDF defined on the raw input data space - such as raw time-series of raw image pixel data - not features. The projection or "Q" function

$$Q(\mathbf{x}, T_1, H_0) = \frac{p(\mathbf{x}|H_0)}{p(\mathbf{z}_1|H_0)} \tag{2}$$

is the ratio of the PDF at the input and output of the feature transformation under some reference hypothesis H_0. It is determined exactly from the feature transformation and does not depend on training data, thus does not suffer from dimensionality issues. The method requires that the feature transformations are known and needs access to the raw data in order to compute equation (2). Thus, it cannot be viewed as another method of constructing decision functions on a given feature space.

2.2 Statement of Theorem

It is well known how to write the PDF of $\mathbf{x}$ from the PDF of $\mathbf{z}$ when the transformation is 1:1. This is the change of variables theorem from basic probability. Let $\mathbf{z} = T(\mathbf{x})$, where $T(\mathbf{x})$ is an invertable and differentiable multidimensional transformation. Then,

$$p_x(\mathbf{x}) = |\mathbf{J}(\mathbf{x})|\ p_z(T(\mathbf{x})), \tag{3}$$

where $|\mathbf{J}(\mathbf{x})|$ is the determinant of the Jacobian matrix of the transformation

$$\mathbf{J}_{ij} = \frac{\partial z_i}{\partial x_j}.$$

What we seek is a generalization of (3) which is valid for many-to-1 transformations. Define

$$\mathcal{F}(T, f_z) = \{f_x(\mathbf{x}) : \mathbf{z} = T(\mathbf{x}) \text{ and } \mathbf{z} \sim f_z(\mathbf{z})\},$$

that is, $\mathcal{F}(T, f_z)$ is the set of PDFs $f_x(\mathbf{x})$ such that if $\mathbf{z} = T(\mathbf{x})$, then $\mathbf{z}$ has PDF $f_z(\mathbf{z})$. If $T(\ )$ is many-to-one, $\mathcal{F}(T, f_z)$ will contain an infinite number of members. Therefore, it is impossible to uniquely determine $f_x(\mathbf{x})$ from $T(\ )$ and $f_z(\mathbf{z})$. We can, however, find a particular solution if we constraint $f_x(\mathbf{x})$. The applicable constraint is that

$$\frac{f_x(\mathbf{x})}{p_x(\mathbf{x}|H_0)} = \frac{f_z(\mathbf{z})}{p_z(\mathbf{z}|H_0)}, \tag{4}$$

or that the likelihood ratio (with respect to H_0) is the same in both the raw data and feature domains for some predetermined reference hypothesis H_0. We will soon show that this constraint produces desirable properties. The particular form of $f_x(\mathbf{x})$ is uniquely defined by the constraint itself, namely

$$f_x(\mathbf{x}) = \frac{p_x(\mathbf{x}|H_0)}{p_z(\mathbf{z}|H_0)}\ f_z(\mathbf{z}); \quad \text{at} \quad \mathbf{z} = T(\mathbf{x}). \tag{5}$$

Theorem 1 proves that (5) is, indeed, a PDF.

Theorem 1 (PDF Projection Theorem). *Let H_0 be some fixed reference hypothesis with known PDF $p_x(\mathbf{x}|H_0)$. Let $\mathcal{X}$ be the region of support of $p_x(\mathbf{x}|H_0)$. In other words $\mathcal{X}$ is the set of all points $\mathbf{x}$ where $p_x(\mathbf{x}|H_0) > 0$. Let $\mathbf{z} = T(\mathbf{x})$ be a continuous many-to-one transformation (the continuity requirement may be overly restrictive). Let $\mathcal{Z}$ be the image of $\mathcal{X}$ under the transformation $T(\mathbf{x})$. Let $p_z(\mathbf{z}|H_0)$ be the PDF of $\mathbf{z}$ when $\mathbf{x}$ is drawn from $p_x(\mathbf{x}|H_0)$. It follows that $p_z(\mathbf{z}|H_0) > 0$ for all $\mathbf{z} \in \mathcal{Z}$. Now, let $f_z(\mathbf{z})$ be a any other PDF with the same region of support $\mathcal{Z}$. Then the function (5) is a PDF on $\mathcal{X}$, thus*

$$\int_{\mathbf{x}\in\mathcal{X}} f_x(\mathbf{x})\ \mathrm{d}\mathbf{x} = 1.$$

Furthermore, $f_x(\mathbf{x})$ is a member of $\mathcal{F}(T, f_z)$.
Proof: *These assertions are proved in a prior publication [1],[2].*

2.3 Optimality Conditions

While it is interesting that $f_x(\mathbf{x})$ is a PDF, it is not yet clear that $f_x(\mathbf{x})$ is the best choice. We generally we would like to use $f_x(\mathbf{x})$ as an approximation to the PDF $p_x(\mathbf{x}|H_1)$. Define

$$\hat{p}_x(\mathbf{x}|H_1) \triangleq \frac{p_x(\mathbf{x}|H_0)}{p_z(\mathbf{z}|H_0)}\ \hat{p}_z(\mathbf{z}|H_1) \quad \text{at} \quad \mathbf{z} = T(\mathbf{x}). \tag{6}$$

From Theorem 1, we see that (6) is a PDF. Furthermore, if $T(\mathbf{x})$ is a sufficient statistic for H_1 vs H_0, then as $\hat{p}_z(\mathbf{z}|H_1) \to p_z(\mathbf{z}|H_1)$, we have

$$\hat{p}_x(\mathbf{x}|H_1) \to p_x(\mathbf{x}|H_1).$$

This situation produces an optimal classifier if it can be achieved for all classes. If the reader can excuse the misuse of the terms optimal and sufficient, if sufficiency is *approximate*, it produces a classifier that is *approximately* optimal. This is because the PDF projection operator produces a valid PDF regardless of sufficiency.

Theorem 1 allows maximum likelihood (ML) methods to be used in the raw data space to optimize the accuracy of

the approximation. Let $\hat{p}_z(\mathbf{z}|H_1)$ be parameterized by the parameter $\boldsymbol{\theta}$. Then, the maximization

$$\max_{\boldsymbol{\theta},T,H_0} \left\{ \frac{p_x(\mathbf{x}|H_0)}{p_z(\mathbf{z}|H_0)}\, \hat{p}_z(\mathbf{z}|H_1;\boldsymbol{\theta}) \right\}$$

is a valid ML approach and can be used for model selection (with appropriate data cross-validation).

2.4 Reference Hypotheses and Calculation of "Q" function

One of the issues in constructing the "Q" function (2) is that it must be possible to calculate both the numerator and denominator for any value of $\mathbf{x}$ or $\mathbf{z}_1$. Problems arise especially in the PDF tails. For many feature transformations, where the numerator and denominator PDFs may be derived analytically, tail behavior is not a problem. For still more feature transformations, the PDFs may be accurately analyzed in the tails using the saddlepoint approximation [3] or analyzed using asymptotic CR bound analysis. Subject to certain requirements, the reference hypothesis H_0 can vary on-the-fly to match the input data sample as well as possible. This helps produce simpler and better-conditioned expressions for the numerator and denominator of the Q-function and allows reduction of feature dimension. A special case of variable reference hypothesis is when the features are maximum likelihood (ML) estimates. The Q-function can be derived from the Cramer-Rao lower bound. For more information on these issues, the reader is referred to the website [6].

2.5 The Chain Rule

In many cases, it is difficult to derive the ratio $p(\mathbf{x}|H_0)/p(\mathbf{z}_1|H_0)$ for an entire processing chain. On the other hand, it may be quite easy to do it for one stage of processing at a time. This is especially true because up-front processing is usually simpler - FFT, wavelets, etc - and yields well to analysis. Another advantage of breaking the processing into stages is to take advantage of redundancy in the up-stream processing. The chain rule is just the recursive application of the PDF projection theorem. Consider a processing chain:

$$\mathbf{x} \overset{T_1(\mathbf{x})}{\rightarrow} \mathbf{y} \overset{T_2(\mathbf{y})}{\rightarrow} \mathbf{w} \overset{T_3(\mathbf{w})}{\rightarrow} \mathbf{z} \tag{7}$$

By applying (6) to the first stage, we obtain

$$p_x(\mathbf{x}|H_1) \;=\; \frac{p_x(\mathbf{x}|H_0)}{p_y(\mathbf{y}|H_0)}\, p_y(\mathbf{y}|H_1). \tag{8}$$

If this process is continued recursively to expand the last term each time, we obtain

$$\begin{aligned} p_x(\mathbf{x}|H_1) \;=\; & \frac{p_x(\mathbf{x}|H_0)}{p_y(\mathbf{y}|H_0)}\, \frac{p_y(\mathbf{y}|H_0')}{p_w(\mathbf{w}|H_0')} \\ & \cdot \frac{p_w(\mathbf{w}|H_0'')}{p_z(\mathbf{z}|H_0'')}\, p_z(\mathbf{z}|H_1) \end{aligned} \tag{9}$$

where H_0, H_0', H_0'' are reference hypotheses suited to each stage in the processing chain.

There is a special embedded relationship between these hypotheses. Let $\mathcal{H}_y$ be the *region of sufficiency* (ROS) for $\mathbf{y}$, the set of all hypotheses for which $\mathbf{y}$ is a sufficient statistic [1]. Let $\mathcal{H}_w$ and $\mathcal{H}_z$ be similarly defined for $\mathbf{w}$ and $\mathbf{z}$. Because information is lost at each stage in the chain, the region of sufficiency shrinks each time. Thus, we have $\mathcal{H}_z \in \mathcal{H}_w \in \mathcal{H}_y$. For optimality of the classifier, we also require $H_1 \in \mathcal{H}_z, \quad H_0'' \in \mathcal{H}_z, \quad H_0' \in \mathcal{H}_w, \quad H_0 \in \mathcal{H}_y$. The factorization (9) together with the embedding of the hypotheses we call the chain-rule processor (CRP).

3 Building a Classifier

3.1 Extending to M Classes

We now consider the M-ary classification problem. If we adapt equations (1), (2) for multiple classes as follows. Let

$$\hat{p}(\mathbf{x}|H_j) = Q(\mathbf{x}, T_j, H_{0,j})\; p(\mathbf{z}_j|H_j), \tag{10}$$

where $\mathbf{z}_j = T_j(\mathbf{x})$ is a class-specific feature set for class j and $p(\mathbf{z}_j|H_j)$ is its PDF under class H_j. Notice that the "Q"-function uses a class-dependent reference hypothesis - there is no need to make it common. As we have stated, $\hat{p}(\mathbf{x}|H_j)$ so defined is *always* a PDF (it integrates to 1 on $\mathbf{x}$) and it is a member of of $\mathcal{F}(T_j, f_z)$. The result is the class-specific Neyman Pearson classifier

$$j^* = \arg\max_j Q_j(\mathbf{x}, T_j, H_{0,j})\; p(\mathbf{z}_j|H_j). \tag{11}$$

Notice that the Q-function is a function of $\mathbf{x}$ that depends only on the transformation $T_j(\mathbf{x})$ and the reference hypothesis $H_{0,j}$, so it may be determined *a priori* without training. $Q_j(\mathbf{x}, T_j, H_{0,j})$ is the correct way to compensate the likelihoods in a class-specific classifier. The ability of the Q-function to provide a "peak" at the "correct" feature set gives the classifier a measure of classification performance without needing to train. In some applications, the PDF of the features is not even needed - the Q-function does all the work.

[1] We mean that for a binary test between any pair of hypotheses in the set, the feature is a sufficient statistic.

3.2 Class-Specific Modules

The derivation of Q-functions for a particular feature transformation is something that needs to be done only once. Once complete, the Q-function calculation can be bundled together with the feature calculation to arrive at a software package called a class-specific "module". Capitalizing on the chain-rule, the designer of a class-specific classifier can draw from a library of modules which can be connected together into "chains" to form each arm of a classifier. All probabilities are represented in the log-domain, so the log-Q functions are added together. At the end of the chain, the aggregate log-Q function is added to the log-PDF of the features. Q-functions have been derived for many features useful in speech analysis, time-series analysis, and general classification. As time goes on, more feature sets become available.

4 Example

The class-specific method is not an algorithm like a neural network that can be applied to an existing set of features and compared with dozens of existing algorithms. Comparison with traditional feature-based classifiers on open databases is problematic for several reasons. First, it is a method that requires defining its own features and requires access to the raw data. Second, it is highly dependent on the choices that are made in design including features and reference hypotheses. The closest thing to a "fair" comparison would be to develop class-specific features, then collect all the features from all the classes into one set and offer this to the conventional classifier. Such an experiment is available [4] and shows that the conventional classifier requires two orders of magnitude (more than 100 times) more training data to reach the same maximum performance level.

To illustrate the design of chain-rule processors, we develop the Q-function for the autocorrelation function (ACF). Let

$$\mathbf{z} = [\hat{r}_0, \hat{r}_1 \dots \hat{r}_P]',$$

where P is the order of the desired autoregressive (AR) process and the circular autocorrelation samples are

$$\hat{r}_t = \frac{1}{N} \sum_{i=1}^{N} x_i \; x_{[i+t]},$$

where $[i+t]$ means $(i+t)$ modulo-N. Note that $\mathbf{z}$ is related by a 1:1 transformation to linear prediction (LPC) coefficients or reflection coefficients (RC) [5],[3], and is extremely useful for time-series analysis and spectral analysis. It is easily shown that an alternative way to compute $\hat{r}_t$ is as follows:

$$\hat{r}_t = \frac{1}{N^2} \sum_{k=0}^{N/2} \epsilon_k \; y_k \; \cos(2\pi k t/N),$$

where $\epsilon_k = 2$ for $k = 1, 2 \dots N/2-1$ and 1 otherwise, and where y_k are the DFT magnitude-square bins

$$y_k = \left| \sum_{n=1}^{N} x_n \exp(-i 2\pi n k/N) \right|^2 . \tag{12}$$

This can be written as the matrix equation

$$\mathbf{z} = \mathbf{C}' \, \mathbf{y}, \tag{13}$$

where the definitions of $\mathbf{y}$ and $\mathbf{C}$ are obvious. Thus, we can break the ACF calculation into two stages : DFT (12), followed by linear transformation (13).

4.1 Stage 1

In the first stage, we compute the DFT (12). For our reference hypothesis for this stage, we use H_0, the standard normal density (WGN hypothesis with unit variance). We have

$$p_x(\mathbf{x}|H_0) = (2\pi)^{-N/2} \exp\left\{ -\frac{1}{2} \sum_{i=1}^{N} x_i^2 \right\}. \tag{14}$$

Note that under H_0, $\mathbf{y}$ is a set of independent RVs. It is easily shown that $y_0, y_{N/2}$ obey the $\chi^2(1)$ density with mean N and variance $2N^2$. Also, $y_1 \cdots y_{N/2-1}$ obey the exponential density with mean N and variance N^2. Thus,

$$p(\mathbf{y}|H_0) = \prod_{i=0}^{N/2} p(y_i|H_0), \tag{15}$$

where

$$p(y_i|H_0) = \frac{(y_i/N)^{-1/2}}{N\sqrt{2\pi}} \quad \exp\left\{ -\frac{y_i}{2N} \right\} \quad i = 0, N/2, \tag{16}$$

and

$$p(y_i|H_0) = \frac{1}{N} \; \exp\left\{ -\frac{y_i}{N} \right\}, \quad 1 \le i \le N/2 - 1. \tag{17}$$

4.2 Stage 2

In stage 2, we compute (13) and use a variable reference hypothesis H_0. We let H_0 be the hypothesis that $\mathbf{y}$ obeys the AR spectrum corresponding to $\mathbf{z}$. Thus, we must use the Levinson algorithm to solve for the P-th order AR coefficients $\sigma_0^2, \mathbf{a}^z$. Let $A^z(k)$ be the DFT of $\mathbf{a}^z$ (padded to length N), then

$$P^z(k) = \frac{\sigma_0^2}{|A^z(k)|^2}$$

is the AR spectrum corresponding to $\mathbf{a}^z, \sigma_0^2$. We let

$$\mathbf{y}^z \triangleq [P^z(0) \dots P^z(N/2)]' \tag{18}$$

We need to evaluate $p_y(\mathbf{y}|H_0)$ and $p_z(\mathbf{z}|H_0)$. We assume that under H_0, $\{y_i\}$ are a set of independent exponential and χ^2 RVs with means corresponding to $\{y_i^z\}$, the elements of $\mathbf{y}^z$. Specifically,

$$p_y(y_i|H_0) = \frac{1}{y_i^z\sqrt{2\pi}}\,(y_i/y_i^z)^{-1/2}\,\exp\left\{-\frac{y_i}{2y_i^z}\right\}, \quad (19)$$

for $i = 0, N/2$, and and

$$p_y(y_i|H_0) = \frac{1}{y_i^z}\,\exp\left\{-\frac{y_i}{y_i^z}\right\}, \quad 1 \le i \le N/2-1. \quad (20)$$

Because H_0 is "close" to $\mathbf{z}$, we approximate $p_z(\mathbf{z}|H_0)$ by the central limit theorem (CLT). Under H_0, the elements of $\mathbf{y}$ are independent with mean $\mathbf{y}^z$ and diagonal covariance $\mathbf{\Sigma}_y^z$,

$$\mathbf{\Sigma}_y^z(i,i) = \begin{cases} 2(y_i^z)^2, & i = 0, N/2 \\ (y_i^z)^2, & 1 \le i \le N/2-1. \end{cases}$$

We can then easily compute the mean and covariance of $\mathbf{z}$:

$$\mathbf{z}^z = \mathcal{E}(\mathbf{z}|H_0) = \mathbf{C}'\,\mathbf{y}^z,$$

and

$$\mathbf{\Sigma}_z^z = \mathbf{C}'\,\mathbf{\Sigma}_y^z\,\mathbf{C}.$$

$$\begin{aligned} p_z(\mathbf{z}|H_0) &= (2\pi)^{-\frac{(P+1)}{2}}|\det(\mathbf{\Sigma}_z^z)|^{1/2} \\ &\quad e^{-\frac{1}{2}\,(\mathbf{z}-\bar{\mathbf{z}}_{0z})'(\mathbf{\Sigma}_z^z)^{-1}(\mathbf{z}-\bar{\mathbf{z}}_{0z})} \qquad (21) \\ &\simeq (2\pi)^{-\frac{(P+1)}{2}}|\det(\mathbf{\Sigma}_z^z)|^{1/2} \end{aligned}$$

where in the last step, we make the approximation $\mathbf{z}^z \simeq \mathbf{z}$. This approximation becomes better as N becomes larger. Note also that the method just described is closely related to the ML approach. In fact, $\mathbf{\Sigma}_z^z$ can be related to the Fisher's information of the ACF estimates [5].

4.3 Validation

It is always important to validate the Q-function before implementation of the classifier. To this purpose, we have developed a method for end-to-end testing of a chain-rule processor [6]. However, because the current example has been analyzed through a different approach, it is possible to compare the results. Direct implementation of the Q-function without breaking into two stages is possible if we can compute the PDF $p(\mathbf{z}|H_0)$. This density has been published [3]. We compared the two methods in a simulation. We created data using an AR(4) model. AR coefficients were randomly created in each trial by choosing the reflection coefficients uniformly in the range [-1,1], then transforming into AR coefficients. Although there is insufficient space for a graph, the log-Q functions of the two methods track very closely across a wide range. The maximum difference did not exceed 0.5 in the log space although the values were spread across a range of more than 300, a factor of e^{300}. This wide range is due to the fact that samples are in the far tails of $p(\mathbf{z}|H_0)$.

5 Conclusions

The PDF projection theorem makes it possible to apply the classical theory of hypothesis testing in problems where PDFs must be approximated from training data. It allows the PDF approximation to be carried out in a class-specific low-dimensional feature space, while the likelihood comparisons can be made in the common raw data space. Doing this avoids the curse of dimensionality if the class-dependent features are each of low dimension. The theorem requires that the statistics of the input and output data of the transformation be known for a particular class-dependent reference hypothesis. The chain-rule is the recursive application of the PDF projection theorem. The resulting architecture, called the chain-rule processor, facilitates the analysis of complex chains of transformations. The ability to change the reference hypothesis "on the fly" further facilitates the analysis. The analysis of a chain which computes the autocorrelation function was demonstrated. More examples and documentation may be found on the website [6].

References

[1] P. M. Baggenstoss, "A theoretically optimum approach to classification using class-specific features.," *Proceedings of ICPR, Barcelona*, 2000.

[2] P. M. Baggenstoss, "A modified Baum-Welch algorithm for hidden Markov models with multiple observation spaces.," *IEEE Trans. Speech and Audio*, pp. 411–416, May 2001.

[3] S. M. Kay, A. H. Nuttall, and P. M. Baggenstoss, "Multidimensional probability density function approximation for detection, classification and model order selection," *IEEE Trans. Signal Processing*, Oct 2001.

[4] 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.

[5] S. Kay, *Modern Spectral Estimation: Theory and Applications*. Prentice Hall, 1988.

[6] P. Baggenstoss, "Class-specific web page," *http://www.npt.nuwc.navy.mil/csf/index.html*.