Wavelet-Based Estimation of a Discriminant Function

Woojin Chang[1]

Seong-Hee Kim[2]

and

Brani Vidakovic[3]

Abstract. In this paper we consider wavelet-based binary linear classifiers. Both consistency results and implementational issues are addressed. We show that under mild assumptions on the design density wavelet discrimination rules are L_2-consistent. The proposed method is illustrated on synthetic data sets in which the "truth" is known and on an applied discrimination problem from the industrial field.

KEY WORDS: Discrimination, Wavelets, Regression.

[1]Woojin Chang is Ph.D student, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia, 30332-0265,

[2]Seong-Hee Kim is an Assistant Professor, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia, 30332-0265,

[3]Brani Vidakovic is an Associate Professor, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia, 30332-0265.

1 Introduction

Discrimination is one of main statistical procedures in the field of Pattern Recognition Theory. Based on historic (training) covariate measurements (univariate or multivariate) the decision maker is to classify a newly obtained observation. For instance, an observation may be classified as conforming or non-conforming, low or high, real or fake, black or white, etc, depending on the problem context. This unknown nature of the observation will be called a *class,* and in this paper we consider problems possessing only two possible exclusive classes, "0" and "1." Formally, the classifier is a function that maps the d-dimensional space of covariates to the set $\{0, 1\}$.

In this paper we are concerned with discriminator functions represented by wavelet decompositions. Our proposal builds on the existing theory of Fourier-based classifiers studied by Greblicki and Pawlak (1981), Hermite polynomials-based classifiers by Greblicki (1981), and generalized linear discriminators by Devorye, Gyorfi, and Lugosi (1991). Kohler (2001) argues that the use of standard wavelets in the general regression may produce suboptimal results if the distribution of the design is very non-uniform. It is likely true that the same holds for wavelet based discriminators. However, we have found that in practical and simulated situations when design distribution is clearly non-uniform, our discriminators work well.

The paper is organized as follows: Section 2 provides basic definitions and formulates the classification problem. In Section 3 we define wavelet based classifiers and state results concerning their L_2-consistency. Section 4 gives simulations and applications of the classifier from Section 3. Appendices contain proofs of the results from Section 3 as well as MATLAB program that

calculates the classifiers.

2 The Bayes Classification Problem

In this section we introduce the Bayes classification problem.

Let $(X, Y) \in R^d \times \{0, 1\}$ be a two-dimensional random variable. Let μ be probability measure of X and η regression of Y on X, i.e., for a Borel set $A \in R^d$

$$\mu(A) = P(X \in A)$$

and

$$\eta(x) = P(Y = 1|X = x) = E(Y|X = x).$$

It can be demonstrated that the pair (μ, η) uniquely determines the distribution of (X, Y).

Any function $g : R^d \to \{0, 1\}$ is a classifier. For a classifier g, the error (risk) function is the probability of error, i.e., $L(g) = P(g(X) \neq Y)$.

It can be demonstrated that Bayes classifier

$$g^*(x) = \mathbf{1}(\eta(x) > 1/2)$$

minimizes L, i.e., for any classifier g,

$$P(g^*(X) \neq Y) \leq P(g(X) \neq Y).$$

We will denote this minimal error with L^* and call it Bayes error.

The attribute Bayes comes from the fact that classification is made according to the posterior probability,

$$\eta(X) = P(Y = 1|X).$$

We also assume that a density of X, f, exists, If f_0 and f_1 are class-conditional densities, i.e., densities of X when $Y = 0$ and $Y = 1$ respectively, and p and $1-p$ class probabilities, $P(Y = 1)$ and $P(Y = 0)$, then the function

$$\alpha(x) = pf_1(x) - (1 - p)f_0(x)$$

has the representation $(2\eta(x) - 1)f(x)$, and the classifier g^* can be written as

$$g^*(x) = \mathbf{1}(\alpha(x) > 0). \tag{1}$$

Let $D_n = ((X_1, Y_1) \ldots, (X_n, Y_n))$ be a training set and X be a new observation. We estimate label Y by the decision $g_n(X) = g_n(X, D_n)$. The error probability is

$$L_n = P(Y \neq g_n(X)|D_n). \tag{2}$$

The expected error probability, $EL_n = P(Y \neq g_n(X))$ is completely determined by the distribution of (X, Y) and the classifier g_n. The classifying rule g_n is said to be consistent if $\lim_{n\to\infty} L_n = L^*$.

The classification is easier problem than the regression – if η_n is a L_2-consistent estimator of η, then the classifier based on η_n is consistent, moreover, $EL_n - L^*$ converges to 0 faster than the L_2-norm of the difference $(\eta_n - \eta)$. We found that wavelet-based classifiers are comparable to regression classifiers when the later are feasible.

For more details and results about general Bayes classification problem we direct the reader to an excellent monograph by Devroye, Györfi, and Lugosi (1996).

3 Wavelet Based Classifier

The wavelet based classifier is preceded in the literature by the Fourier series classifier. All such classifiers can be put in the form: Classify $X = x$ to be in class 0 if: $\sum_{j=1}^{k} a_{n,j}\psi_j(x) \leq 0$. Functions ψ_j are fixed and represent the basis for the series estimate, $a_{n,j}$ are coefficients depending on the training sample of size n, and k, the number of basis functions, usually regulates smoothness.

The literature on Fourier series classifiers is rich. Work by Van Ryzin (1966), Greblicki and his team (Greblicki, 1981; Greblicki and Rutkovski, 1981; Greblicki and Pawlak, 1982; 1983) explore various theoretical concepts of consistency and rates of convergence of the classifiers.

Let the scaling function $\phi(x)$ generates orthonormal MRA and let the multiresolution subspace V_J be spanned by the functions $\{\phi_{J,k}(x) = 2^{J/2}\phi(2^J x - k),\ k \in Z\}$. Let $\psi(x)$ be a wavelet function corresponding to ϕ and $\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k), j, k \in Z$. Since $\alpha(x)$ is in L_2, it allows wavelet representation

$$\alpha(x) = \sum_k c_{J,k}\phi_{J,k}(x) + \sum_{j \geq J}\sum_k d_{j,k}\psi_{j,k}(x).$$

A *raw* wavelet-based linear classifier, $\hat{g}_J$, is defined as

$$\hat{g}_J(x) = \mathbf{1}(\hat{\alpha}_J(x) > 0), \tag{3}$$

where $\hat{\alpha}_J(x)$ is an estimator of the projection of α on V_J, i.e., an estimator of $\alpha_J(x) = \sum_k c_{J,k}\phi_{J,k}(x)$.

The coefficients $c_{J,k} = \int (2\eta(x)-1)f(x)\phi_{J,k}(x)\,dx = E[(2\eta(X)-1)\phi_{J,k}(X)]$ can be, by moment matching, estimated by $\hat{c}^n_{J,k} = \frac{1}{n}\sum_{i=1}^{n}(2Y_i - 1)\phi_{J,k}(X_i)$. Thus, one can take $\hat{\alpha}_{n,J}(x) = \sum_k \hat{c}^n_{J,k}\phi_{J,k}(x)$, and the estimator from (3) can

be rewritten as

$$\hat{g}_{n,J}(x) = \mathbf{1}(\hat{\alpha}_{n,J}(x) > 0), \tag{4}$$

If the wavelet basis is interpolating, or close to interpolating, then the coefficients $\{\hat{c}^n_{J,k},\ k \in Z\}$ can be thought as values of function α at sampled at equally spaced points. Let $\hat{L}_n(J) = P(Y \neq \hat{g}_{n,J}(X)|D_n)$ be the error probability of $\hat{g}_{n,J}$.

The estimator $\hat{g}_{n,J}(x)$ is consistent. The following result holds.

Theorem 1 *Assume that the density for X, f, is compactly supported and belongs to L_∞. Let $J = J(n)$ be the multiresolution level depending on sample size n, in the sample $(X_1, Y_1), \ldots, (X_n, Y_n)$.*

Let K be the number of coefficients $\hat{c}^n_{J,k}$ in $\hat{\alpha}_{n,J}(x)$ from (4). If

$$J \to \infty \quad \textit{and} \quad \frac{K}{n} \to 0 \quad \textit{as } n \to \infty$$

then the wavelet-based classifier in (4) is consistent, i.e.,

$$\lim_{n\to\infty} \mathbb{E}\hat{L}_n(J) = L^* \ .$$

Remark. If the $\alpha(x)$ is compactly supported, K is finite. If the $\alpha(x)$ is rescaled to $[0, 1]$ then $K = 2^J$.

The proof of Theorem 1 is given in the Appendix.

The consistent linear estimator $\hat{g}_{n,J}$ gains in performance if regularized. Regularization is achieved by wavelet shrinkage. For given levels J and J_0

such that $J_0 < J$, starting with $\hat{c}^n_{J,k}$, one can obtain scaling and wavelet coefficients $\hat{c}^n_{J_0,k}$, $\hat{d}^n_{j,k}$ for $J_0 \leq j < J$, by utilizing fast Mallat's cascade algorithm. Thus, the original estimator $\hat{\alpha}_{n,J}$ depending on $\hat{c}^n_{J,k}$ can be represented as

$$\hat{\alpha}_{n,J}(x) = \sum_k \hat{c}^n_{J_0,k}\phi_{J_0,k}(x) + \sum_{J_0 \leq j < J} \sum_k \hat{d}_{j,k}\psi_{j,k}(x) \tag{5}$$

To regularize $\hat{\alpha}_{n,J}$ we apply wavelet shrinkage to "detail" coefficients, $d_{j,k}$. The shrunk coefficients we denote by $d^*_{j,k}$, where $J_0 \leq j < J$. In our analysis we used soft shrinkage policy $d^*_{j,k} = (|\hat{d}_{j,k}| - \lambda)_+$, with universal threshold $\lambda = \sqrt{2 \log K}\ \hat{\sigma}$, where $\hat{\sigma}$ is an estimator of standard deviation of wavelet coefficients at finest scale. Other shrinkage policies can be implemented as well.

¿From $\hat{c}_{J_0,k}$'s and $d^*_{j,k}$'s, utilizing the inverse wavelet transformation, we obtain the sampled values of regularized estimator of α_J. This estimator is given by

$$\tilde{\alpha}_{n,J,\lambda}(x) = \sum_k \hat{c}^n_{J_0,k}\phi_{J_0,k}(x) + \sum_{J_0 \leq j < J} \sum_k d^*_{j,k}\psi_{j,k}(x) \tag{6}$$

Thus, for the training sample of size n, multiresolution level J, and threshold λ, the proposed regularized discriminator is

$$\tilde{g}_{n,J,\lambda} = \mathbf{1}(\tilde{\alpha}_{n,J,\lambda} > 0), \tag{7}$$

where λ is the threshold level.

The regularized estimator is consistent as well, i.e., the following theorem holds:

Theorem 2 *Let J and K be as in Theorem 1 and let J_0 be multiresolution level such that $J_0 < J$. The regularized wavelet-based classifier $\tilde{g}_{n,J,\lambda}$ in (7) is consistent if*

$$J_0 \to \infty \quad \textit{and} \quad \frac{K}{n} \to 0 \quad \textit{as } n \to \infty.$$

The proof of Theorem 2 is given in the appendix.

4 Implementations

To apply the proposed nonlinear classifiers and select optimal multiresolution levels and threshold, we introduce the empirical errors. Empirical errors of classifiers $\hat{g}_{n,J}$ and $\tilde{g}_{n,J,\lambda}$, based on training data set of size n, and evaluated at data $\{(X_j, Y_j) : j = 1, \ldots, m\}$ are

$$\hat{L}_n(J, m) = \frac{1}{m} \sum_{j=1}^{m} \mathbf{1}(\hat{g}_{n,J}(X_j) \neq Y_j), \tag{8}$$

and

$$\tilde{L}_n(J, m, \lambda) = \frac{1}{m} \sum_{j=1}^{m} \mathbf{1}(\tilde{g}_{n,J,\lambda}(X_j) \neq Y_j), \tag{9}$$

respectively.

To select the wavelet base, multiresolution levels J and J_0, and threshold level λ, we minimized corresponding empirical errors.

By simulational analysis we found that for various m, the choice of *Symmlet 8* (Daubechies least asymmetric 8-tap wavelet filter), $J = 6$ or 7, $J_0 = 3$, and λ universal threshold with the soft-shrinkage policy, produced consistently good results.

We discuss in detail two simulational studies in which the true classes are known, and a real-life example from the industrial practice.

4.1 Simulated Data Example: 0 - 1 Discrimination

In this simulation we want to discriminate between observations coming from two different normal populations.

The training set, $\{(X_i, Y_i), i = 1, \ldots, n\}$, ($n$ is even) is generated as follows. For the first half of data, $X_i,\ i = 1, \ldots, \frac{n}{2}$ are sampled from the standard normal distribution and $Y_i = 0,\ i = 1, \ldots, \frac{n}{2}$. For the second half, $X_i,\ i = \frac{n}{2} + 1, \ldots, n$ are sampled from normal distribution with mean 2 and variance 1, while $Y_i = 1,\ i = \frac{n}{2} + 1, \ldots, n$. In Figure 1(a) the raw training data are shown. A superposition of wavelet regularized discriminator and a standard discriminator based on the logistic regression are depicted in Figure 1(b).

The validation set $\{(X_j, Y_j), j = 1, \ldots, m\}$ is generated in the same way. We compare the empirical errors $\hat{L}_n(J, m)$ with $\tilde{L}_n(J, m, \lambda)$ and the error of the logistic regression classifier,

$$L_n^{\text{logit}}(m) = \frac{1}{m} \sum_{j=1}^{m} \mathbf{1}\Big(\mathbf{1}(f(X_j) > 0.5) \neq Y_j\Big),$$

where f is fitted logistic regression.

The results for various values of n and $m = 200$ are given in Table 1. In this simulation we set $J = 6$ for both $\hat{L}_n(J, m)$ and $\tilde{L}_n(J, m, \lambda)$. In $\tilde{L}_n(J, m, \lambda)$, *Symmelet 8* is used for wavelet transformation and the soft shrinkage rule with universal threshold is applied to wavelet coefficients.

As evident in Table 1, the raw classifier exhibits uniformly the largest error. The errors of regularized wavelet classifier and logistic regression classifier are comparable.

n	$\hat{L}_n(6, 200)$	$\tilde{L}_n(6, 200, \lambda)$	$L_n^{\text{logit}}(200)$
80	0.272	0.170	0.158
200	0.200	0.164	0.154
400	0.187	0.176	0.171
800	0.169	0.163	0.163
2000	0.160	0.157	0.158

Table 1: Empirical errors using n training data points, $J = 6$, and $m = 200$ validation data points

4.2 Simulated Data Example: 0 - 1 - 0 Discrimination

In the following simulated example the linear logistic regression classifier is not possible.

We generate the training data set, $\{(X_i, Y_i), i = 1, \ldots, n\}$, ($n$ is a multiple of 3) as follows. In the first third of data, $X_i,\ i = 1, \ldots, \frac{n}{3}$ is generated from normal distribution with mean -2 and variance 1, with $Y_i = 0,\ i = 1, \ldots, \frac{n}{3}$. In the second third of data, $X_i,\ i = \frac{n}{3}+1, \ldots, \frac{2n}{3}$ are standard normal random variables and $Y_i = 1,\ i = \frac{n}{3} + 1, \ldots, \frac{2n}{3}$. Finally, in the last third of data, $X_i,\ i = \frac{2n}{3} + 1, \ldots, n$ are generated from normal distribution with mean 2 and variance 1, and $Y_i = 0,\ i = \frac{2n}{3} + 1, \ldots, n$.

We use *Symmlet 8* and soft shrinkage rule with $\lambda = \sqrt{2 \log K}\ \hat{\sigma}$ to construct 0-1-0 discriminator. The training set of X's and the corresponding wavelet-regularized discriminator are shown in Figures 2(a) and (b).

The evaluation set $\{(X_j, Y_j), j = 1, \ldots, m\}$ is generated in an analogous manner. In $\tilde{L}_n(J, m, \lambda)$, *Symmlet 8* is used for wavelet transformation and soft shrinkage rule with the universal threshold is applied to wavelet coeffi-

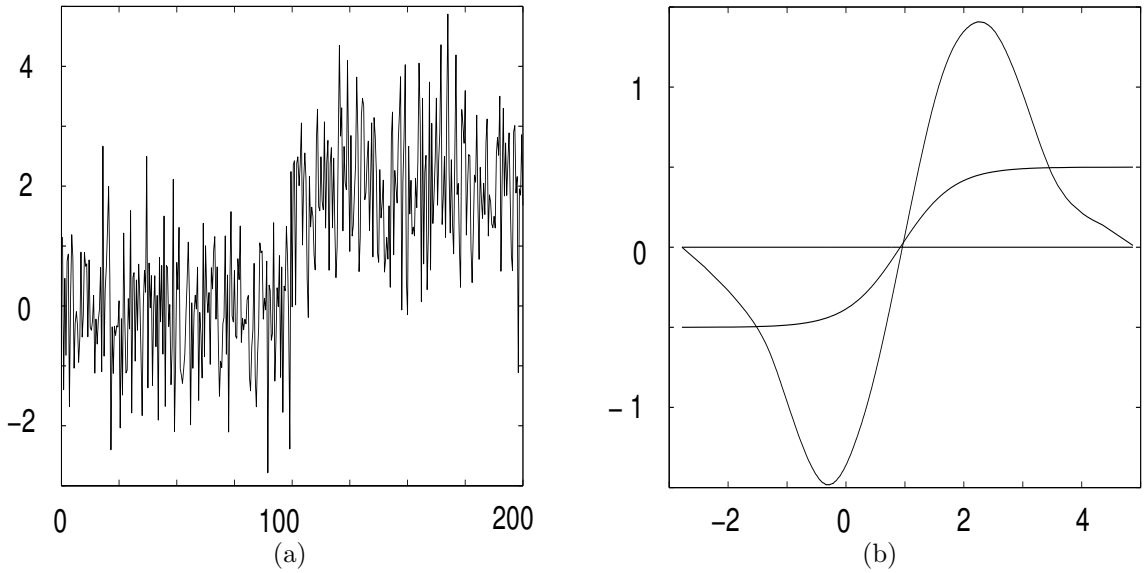

Figure 1: (a) Noisy training data (b) Discriminator functions

cients. The results for various values of n and $m = 300$ are presented in Table 2. We set $J = 7$ for both $\hat{L}_n(J, m)$ and $\tilde{L}_n(J, m, \lambda)$, and compare $\tilde{L}_n(J, m, \lambda)$ with $\hat{L}_n(J, m)$.

As evident from Table 2, regularized classifier uniformly dominates the linear classifier.

4.3 Application in Paper Producing Process

We consider an example from the book of Pandit and Wu (1993, pp. 496–497) which presents 100 data points of the observed basis weights in response to an input in the stock flow rate of a papermaking process. The values were taken at one-second intervals. The following brief description of the papermaking process is from the section 11.1.1 of Pandit and Wu (1993). A schematic

n	$\hat{L}_n(7, 300)$	$\tilde{L}_n(7, 300, \lambda)$
120	0.340	0.213
300	0.288	0.221
600	0.247	0.218
900	0.232	0.212
1200	0.214	0.202

Table 2: Average empirical errors using training data of size n, $J = 7$, and $m = 300$ evaluation data points.

diagram can be found there, too.

> The Fourdrinier papermaking process starts with a mixture of water and wood fibers (pulp) in the mixing box. The gate opening in the mixing box can be controlled to allow a greater or smaller flow of the thick stock (a mixture of water and fiber) entering the headbox. A turbulence is created in the headbox by means of suspended plates to improve the consistency of the pulp. The pulp then descends on a moving wire screen, as a jet from the headbox nozzles. Water is continuously drained from the wet sheet of paper so formed on the wire screen. The paper sheet then passes through press roles, driers, and calender roles to be finally wound.
>
> It is important to produce paper of as uniform a thickness as possible since irregularities on the surface such as ridges and valleys cause trouble in later operations such as winding, coating, printing, etc. This uniformity is measured by what is called a

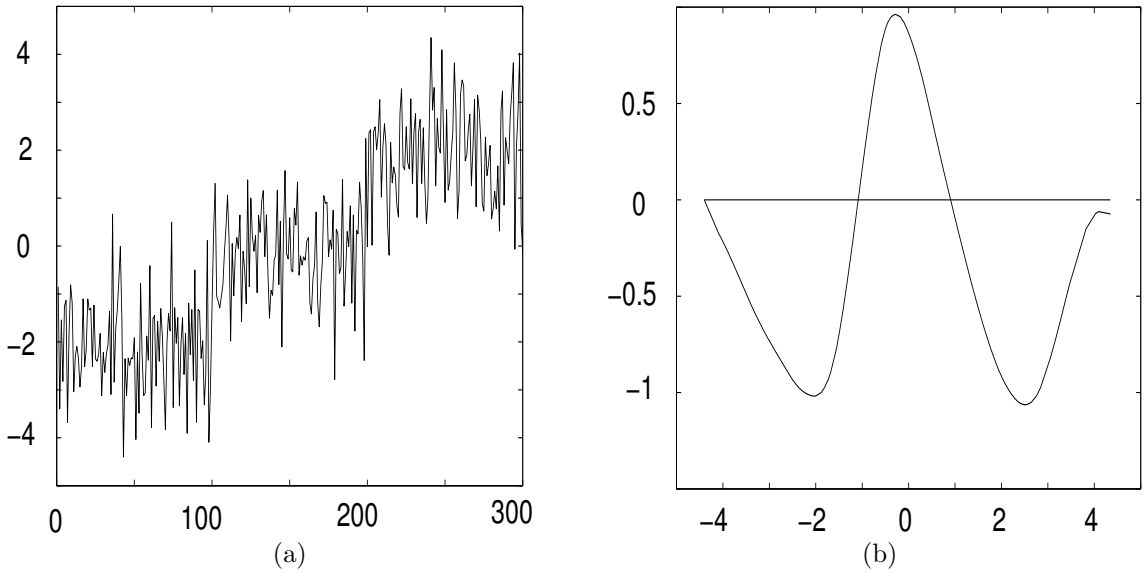

Figure 2: (a) Noisy training data (b) Discriminator function

basis weight, the weight of dry paper per unit area. It may be measured directly or by means of a beta-ray gauge that makes use of the beta-ray absorption properties of the paper material. The regulation of paper basis weight is one of the major goals of the paper control system.

The basis weight is affected by variables such as stock consistency, stock flow, headbox turbulence and slice opening, drier steam pressure, machine speed, etc. However, thick stock flow is often used as the main control input measured by the gate opening in the mixing box.

Based on the above description, we selected the stock flow as the only input in our problem. We are looking for a good predictor of the output basis

weight. Let B_t and S_t for $t = 1, 2, \ldots, 100$ denote the basis weight and the stock flow rate at time t, respectively. The output basis weight, B_t depends not only on its past values but also on the stock flow. However, as stock flow must go through several steps such as refining, pressing, drying etc. *to be paper products,* the stock flow at time t cannot directly affect the basis weight at the same time. However, we assumed that S_{t-1} affects B_t. Further analysis found that $0.7B_{t-1} + 0.25S_{t-1}$ is a good predictor of B_t.

Now we define $\{(X_t, Y_t), t = 2, 3, \ldots, 100\}$. The target basis weight depends upon the grade of the paper being made. We assume that the target basis weight for the paper is 40lb/3300 sq ft. and our tolerance level is ± 0.5. Therefore, we consider the basis weight, B_t in the range of 39.5 and 40.5 as "good" and assign the value of "1" for the response variable, Y_t. Otherwise, the basis weight is "bad" and Y_t is assigned "0". For each such Y_t, the corresponding X_t is $0.7B_{t-1} + 0.25S_{t-1}$. Thus, we have 99 data points of (X_t, Y_t)'s from the given 100 values of basis weight and stock flow. We used (X_t, Y_t)'s with odd t as the training set and the remaining even-index set as the validation set.

By identifying the discriminator function, we hope to be able to predict whether the future basis weight will be "good" or "bad" at the measured basis weight and stock flow. In addition, we want to make the output basis weight maintained at the "good" range of target value by manipulating the stock flow. For example, by looking at the measured basis weight and stock flow at time t, we can guess the basis weight at time $t + 1$ and from this future basis weight, we know which range of stock flow rate we should have to get a "good" basis weight at time $t + 2$.

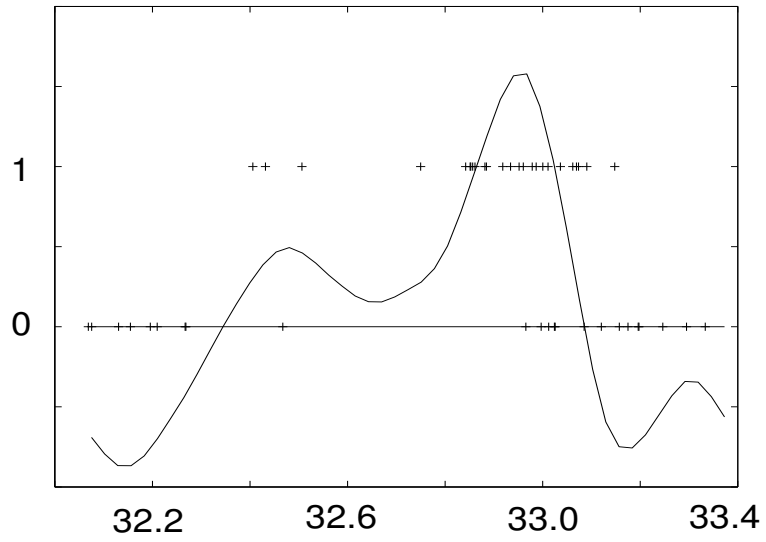

Figure 3: Classifier function

We make a regularized estimator (6) for paper-making process using 50 validation data points. The wavelet based classifier and the validation data, $(X_{2t}, Y_{2t}),\ t = 1, \ldots, 50$ are shown in Figure 3.
The empirical error of the classifier, $\tilde{g}_{49,7,\lambda}$ in (9) is

$$\tilde{L}_{49}(7, 50, \lambda) = \frac{1}{50} \sum_{t=1}^{50} I(\tilde{g}_{49,7,\lambda}(X_{2t}) \neq Y_{2t}) = 0.18.$$

Thus, the error rate of the wavelet-based discriminator in this applied context is 18%, which given the noise in the data is good performance.

5 Appendices

5.1 A. Proofs of Consistency

Proof of Theorem 1. First we show that the $\hat{c}^n_{J,k}$'s are unbiased estimates of the $c_{J,k}$'s.

$$\begin{aligned}
\mathbb{E}[\hat{c}^n_{J,k}] &= \mathbb{E}[(2Y-1)\phi_{J,k}(X)] = \mathbb{E}\Big[\mathbb{E}[(2Y-1)\phi_{J,k}(X)|X]\Big] \\
&= \mathbb{E}\Big[\phi_{J,k}(X)\mathbb{E}[(2Y-1)|X]\Big] = \mathbb{E}[(2\eta(X)-1)\phi_{J,k}(X)] \\
&= \int (2\eta(x)-1)\phi_{J,k}(x)f(x)dx = c_{J,k},
\end{aligned}$$

and that there exists an upper bound on variance of $\hat{c}^n_{J,k}$:

$$\begin{aligned}
\mathbf{Var}(\hat{c}^n_{J,k}) &= \frac{1}{n}\mathbf{Var}(\phi_{J,k}(X_1)(2\eta(X_1)-1)) \\
&= \frac{1}{n}\left(\int \phi^2_{J,k}(x)(2\eta(x)-1)^2 f(x)dx - c^2_{J,k}\right) \\
&\leq \frac{1}{n}\left(B\int \phi^2_{J,k}(x)dx - c^2_{J,k}\right) \leq \frac{1}{n}(B - c^2_{J,k}) \\
&\leq \frac{B}{n}
\end{aligned}$$

where we used $f(x) \leq B$ and $\int \phi^2_{J,k}(x)dx = 1$. By Parseval's identity,

$$\int \alpha^2(x)dx = \sum_k c^2_{J,k} + \sum_{j\geq J}\sum_k d^2_{J,k}.$$

Using orthonormality of $\phi_{J,k}$'s and $\psi_{j,k},\ j \geq J$ we have

$$\begin{aligned}
& \int \Big(\alpha(x) - \sum_k \hat{c}^n_{J,k}\phi_{J,k}(x)\Big)^2 dx \\
= & \int \alpha^2(x)dx + \int \Big(\sum_k \hat{c}^n_{J,k}\phi_{J,k}(x)\Big)^2 dx \\
& -2\int \alpha(x)\sum_k \hat{c}^n_{J,k}\phi_{J,k}(x)dx \\
= & \sum_k c^2_{J,k} + \sum_{j\geq J,k} d^2_{j,k} + \sum_k (\hat{c}^n_{J,k})^2 - 2\sum_k c_{J,k}\hat{c}^n_{J,k} \\
= & \sum_k (\hat{c}^n_{J,k} - c_{J,k})^2 + \sum_{j\geq J,\ k} d^2_{j,k}.
\end{aligned}$$

Thus, the expected L_2-error is bounded as follows:

$$\begin{aligned}
& \mathbb{E}\Big\{ \int \Big(\alpha(x) - \sum_k \hat{c}^n_{J,k}\phi_{J,k}(x)\Big)^2 dx \Big\} \\
= & \mathbb{E}\Big\{ \sum_k (\hat{c}^n_{J,k} - c_{J,k})^2 \Big\} + \sum_{j\geq J,\ k} d^2_{j,k} \\
= & \sum_{k=1}^{K} \mathbf{Var}(\hat{c}^n_{J,k}) + \sum_{j\geq J,\ k} d^2_{j,k} \\
\leq & \frac{KB}{n} + \sum_{j\geq J,\ k} d^2_{j,k}.
\end{aligned}$$

Since α is in L_2, $\sum_{j\geq J,\ k} d^2_{j,k}$, goes to zero if $J \to \infty$. If $K/n \to 0$, then the expected L_2-error converges to zero, which means the estimate is L_2 consistent. $\square$

Proof of Theorem 2. By mimicking the proof of Theorem 1 and taking into account that

$$\tilde{\alpha}(x) = \sum_k \hat{c}^n_{J_0,k}\phi_{J_0,k}(x) + \sum_{J_0\leq j<J,k} d^*_{j,k}\psi_{j,k}(x),$$

we obtain

$$\begin{aligned} &\int \Big(\alpha(x) - \tilde{\alpha}_{n,J,\lambda}(x)\Big)^2 dx \\ &\leq \frac{KB}{n} + \sum_{j \geq J_0,k} d_{j,k}^2 + \sum_{J_0 \leq j < J,k} (d_{j,k}^*)^2 - 2 \sum_{J_0 \leq j < J,k} d_{j,k} d_{j,k}^*. \end{aligned} \tag{10}$$

Since $|d_{j,k} d_{j,k}^*| \leq d_{j,k}^2$, an upper bound of (10) is

$$\frac{KB}{n} + 4 \sum_{j \geq J_0,k} d_{j,k}^2,$$

which goes to 0 when $J_0 \to \infty$. The level $J_0 = J_0(n)$ can be selected in such a way that

$$K^*\Big(J - J_0\Big) \cdot \Big(\max_{J_0 \leq j < J,k} d_{j,k}^2\Big) \to 0, \quad n \to \infty \tag{11}$$

where $K^* = \sum_{J_0 \leq j < J} K(j)$, and $K(j)$ is the number of coefficients in the level j. □

5.2 B. Daubechies-Lagarias Algorithm

A challenge in exhibiting a wavelet-based classificator is calculational. Namely, except for the Haar wavelet, all compactly supported orthonormal families of wavelets (e.g., Daubechies, Symmlet, Coiflet, etc.) scaling and wavelet functions have no a closed form. A non-elegant solution is to have values of the mother and father wavelet given in a table. Evaluation of $\phi_{jk}(x)$ or $\psi_{jk}(x)$, for given x, then can be performed by interpolating the table values.

Based on Daubechies and Lagarias (1991, 1992) *local pyramidal algorithm* a solution is proposed. A brief theoretical description and MATLAB program are provided.

Let ϕ be the scaling function of a compactly supported wavelet generating an orthogonal MRA. Suppose the support of ϕ is $[0, N]$. Let $x \in (0, 1)$, and let $dyad(x) = \{d_1, d_2, \dots d_n, \dots\}$ be the set of 0-1 digits in dyadic representation of x $(x = \sum_{j=1}^{\infty} d_j 2^{-j})$. By $dyad(x, n)$ we denote the subset of the first n digits from $dyad(x)$, i.e., $dyad(x, n) = \{d_1, d_2, \dots d_n\}$.

Let $h = (h_0, h_1, \dots, h_N)$ be the vector of wavelet filter coefficients. Define two $N \times N$ matrices as

$$T_0 = \sqrt{2}(h_{2i-j-1})_{1 \le i,j \le N}, \text{and} \quad T_1 = \sqrt{2}(h_{2i-j})_{1 \le i,j \le N}. \tag{12}$$

Then

Theorem 3 *(Daubechies and Lagarias, 1992.)*

$$\lim_{n \to \infty} T_{d_1} \cdot T_{d_2} \cdot \cdots \cdot T_{d_n} = \begin{bmatrix} \phi(x) & \phi(x) & \dots & \phi(x) \\ \phi(x+1) & \phi(x+1) & \dots & \phi(x+1) \\ \vdots & & & \\ \phi(x+N-1) & \phi(x+N-1) & \dots & \phi(x+N-1) \end{bmatrix}. \tag{13}$$

The convergence of $||T_{d_1} \cdot T_{d_2} \cdot \cdots \cdot T_{d_n} - T_{d_1} \cdot T_{d_2} \cdot \cdots \cdot T_{d_{n+m}}||$ to zero, for fixed m, is exponential and constructive, i.e., effective bounds, that decrease exponentially to 0, can be established.

Example: Consider the DAUB 2 wavelet basis (N=3). The corresponding filter is $(\frac{1+\sqrt{3}}{4\sqrt{2}}, \frac{3-\sqrt{3}}{4\sqrt{2}}, \frac{3+\sqrt{3}}{4\sqrt{2}}, \frac{1-\sqrt{3}}{4\sqrt{2}})$. According to (12) the matrices T_0 and T_1 are given as:

$$T_0 = \begin{bmatrix} \frac{1+\sqrt{3}}{4} & 0 & 0 \\ \frac{3-\sqrt{3}}{4} & \frac{3+\sqrt{3}}{4} & \frac{1+\sqrt{3}}{4} \\ 0 & \frac{1-\sqrt{3}}{4} & \frac{3-\sqrt{3}}{4} \end{bmatrix}, \text{and} \quad T_1 = \begin{bmatrix} \frac{3+\sqrt{3}}{4} & \frac{1+\sqrt{3}}{4} & 0 \\ \frac{1-\sqrt{3}}{4} & \frac{3-\sqrt{3}}{4} & \frac{3+\sqrt{3}}{4} \\ 0 & 0 & \frac{1-\sqrt{3}}{4} \end{bmatrix}.$$

If, for instance, $x = 0.45$, then $dyad(0.45, 20) = \{$ 0, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1 $\}$. The values $\phi(0.45), \phi(1.45)$, and $\phi(2.45)$ are calculated as

$$\prod_{i \in dyad(0.45,20)} T_i = \begin{bmatrix} 0.86480582 & 0.86480459 & 0.86480336 \\ 0.08641418 & 0.08641568 & 0.08641719 \\ 0.04878000 & 0.04877973 & 0.04877945 \end{bmatrix}.$$

By using so-called two-scale equations, it is possible to give an algorithm for calculating values of mother wavelet, the function ψ_{jk}, see Vidakovic (1999). For our purposes direct calculation of wavelet coefficients is unnecessary since, having scaling coefficients at some level J, all wavelet coefficients at coarser levels can be obtained utilizing fast Mallat's algorithm.

5.3 C: Matlab Program Calculating Scaling Function by Daubechies-Lagarias Algorithm

As a rule, in compactly supported orthogonal wavelets, wavelet and scaling function are of no closed form. We give the matlab program based on Daubechies-Lagarias algorithm (Daubechies and Lagarias, 1991; 1992), that calculates a value of scaling function at an arbitrary design point with a prescribed precision.

```
function yy=Phijk(z, j, k, filter, n)
%------------------------------------------------------------------
%  inputs:  z -- the argument
%           j -- scale
%           k -- shift
%           filter -- ON finite wavelet filter, might be an
%                     output of WaveLab's: MakeONFilter
%           n -- precision of approximation (default n=20)
%------------------------------------------------------------------
```

```
%  output:  yy -- value of father wavelet (j,k) corresponding to
%                  'filter' at z.
%-------------------------------------------------------------------
    if (nargin == 4)
        n=20
    end
    daun=length(filter)/2;
    N=length(filter)-1;
    x=(2^j)*z-k;
    if(x<=0|x>=N) yy=0;
    else
    int=floor(x);
    dec=x-int;
    dy=d2b(dec,n);
    t0=t0(filter);
    t1=t1(filter);
    prod=eye(N);
    for i=1:n
            if dy(i)==1 prod=prod*t1;
            else prod=prod*t0;
            end
        end
    y=2^(j/2)*prod;
      yyy = mean(y');
      yy = yyy(int+1);
    end

  %--------------------functions needed----------------------------

  function a = d2b(x,n)
            a=[];
             for i = 1:n
             if(x <= 0.5) a=[a 0]; x=2*x;
              else a=[a 1]; x=2*x-1;
             end
            end
 %-------------------------------------------------------------------
 function t0 = t0(filter)

 n = length(filter); nn = n - 1;

 t0 = zeros(nn); for i = 1:nn
    for j= 1:nn
        if (2*i - j > 0 & 2*i - j <= n)
            t0(i,j) = sqrt(2) * filter( 2*i - j );
        end
    end
end
%-------------------------------------------------------------------
function t1 = t1(filter)
```

```
n = length(filter); nn = n - 1;

t1 = zeros(nn); for i = 1:nn
    for j= 1:nn
        if (2*i -j+1 > 0 & 2*i - j+1 <= n)
            t1(i,j) = sqrt(2) * filter( 2*i - j+1 );
        end
    end
end
%-----------------------------------------------------------------
```

References

[1] Daubechies, I. and Lagarias, J. (1991). Two-scale difference equations I. Existence and global regularity of solutions, *SIAM J. Math. Anal.*, **22**, 5, 1388–1410.

[2] Daubechies, I. and Lagarias, J. (1992). Two-scale difference equations II. Local regularity, infinite products of matrices and fractals, *SIAM J. Math. Anal.*, **23**, 4, 1031–1079.

[3] Devroye, L., Györfi, L., and Lugosi, G. (1996). *A Probabilistic Theory of Pattern Recognition.* Springer-Verlag, New York, NY.

[4] Greblicki, W. (1981). Asymptotic efficiency of classifying procedures using the Hermite series estimate of multivariate probability densities, *IEEE Transactions on Information Theory,* 27, 3, 364–366.

[5] Greblicki, W. and Rutkowski, L. (1981). Density-free Bayes risk consistency of nonparametric patters recognition procedures, *Proceedings of the IEEE,* 69, 4, 482–483.

[6] Greblicki, W. and Pawlak, M. (1982). A classification procedure using the multiple Fourier series, *Information Sciences,* **26,** 115–126.

[7] Greblicki, W. and Pawlak, M. (1983). Almost sure convergence of classification procedures using Hermite series density estimates,*Pattern Recognition Letters,* **2,** 13–17.

[8] Kohler, M. (2001). Nonlinear orthogonal series estimates for random design regression. Technical Report, Department of Mathematics, University of Stutgart, Germany. http://www.mathematik.uni-stuttgart.de/mathA/lst3/kohler/papers-en.html

[9] Pandit, S. and Wu, S-M. (1993). *Time Series and System Analysis with Applications.* Krieger Publishing Company, Malabar, Fl.

[10] Van Ryzin, J. (1966). Bayes risk consistency of classification procedures using density estimates, *Sankhyā*, Ser. A 28, 161–170.

[11] Vidakovic, B. (1999). *Statistical Modeling by Wavelets.* John Wiley & Sons, Inc., New York, 384 pp.

Woojin Chang, Seong-Hee Kim, and Brani Vidakovic
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332-0205
woojin|skim|brani@isye.gatech.edu