Manatee Detection from Adaptive Linear Prediction

Mark Rosenberg

***Abstract*—The goal of manatee detection is to detect in a noisy signal when there is a manatee sound. Adaptive predictive filters were trained on manatee sounds and acoustic noise, and their prediction errors of the test data are used for binary classification. The design considerations include test data labeling, adaptive filter order, error smoothing window size, and feature design. Two free parameters, LMS step size and smoothing window length, are selected to optimize the area under the receiver operating characteristic. The optimal parameter choices result in a performance of $AUC = 0.9356$, which validates the capabilities of adaptive filters.**

***Index Terms*—Adaptive Filter, Binary Classification, Receiver Operating Characteristic**

I. Introduction

Adaptive filters can be used to solve a variety of problems, such as prediction. An adaptive system configuration consists of the following components: the input signal, $x(t)$; the adaptive filter; the filter output signal, $y(t)$; the desired signal, $d(t)$; the cost function; and the model parameter update algorithm. The design choices for the filter model include FIR, Gamma, and IIR. The design choices for the cost function of the error between the filter's desired signal and output signal include mean squared error (MSE) and correntropy. And the options for the update algorithm include methods called Least Mean Squares (LMS) and Recursive Least Squares (RLS). The goal of an adaptive filter is to evolve the model parameters towards values that optimize the cost function. The configuration of the adaptive filter in the context of the broader problem is designed such that optimizing the cost function coincides with the goal of the given task. In the prediction problem, the desired signal is specified as the next input sample:

$$d(t) = x(t + 1). \tag{1}$$

The consequence of this choice of desired signal is that when the model parameters evolve towards values that optimize the cost function, the resulting filter output becomes closer to the desired signal, which is chosen in Equation 1 to be the next input sample. This choice coincides with the definition of prediction. If the testing data is of the same class as the filter's training data, then the expected value of the difference between the filter output and the next sample is expected to be smaller than that of filters trained on data of a different class. This is because training optimizes the cost (e.g. minimizes the MSE) for data in that class.

Predictive adaptive filters can be used as one component of a classification problem. By training adaptive predictors on time series of a certain class label, the filter parameters evolve towards values that optimize the accuracy of the expected prediction of the next sample. One approach to classifying test data is first to train predictors for each class (or subclass), then to filter the test data through each predictor, and finally to label each test sample as the class associated with the predictor with the smallest prediction error. This approach is applied to the manatee detection problem, which is a binary classification problem with classes "manatee" and "noise."

M. Rosenberg is with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA e-mail: markrosenberg@ufl.edu

II. Data Description

The manatee detection problem uses training and testing data of sound waves recorded by hydrophone in order to classify whether the label at each time step is "manatee" or "noise." The first step towards solving this problem is to listen to the data and to view the spectrograms of the data. Human systems are very good at auditory discrimination of manatee calls from background noise after learning what manatee calls sound like. They also can easily visually detect manatee spectrogram patterns in testing data after learning from training data what manatee call spectrograms look like. This fact is utilized to label the testing data. Figure 1 shows 10 spectrograms from the 10 calls in the manatee training data, and Figure 2 shows the spectrograms from the testing data, with the beginning and end time boundaries noted in the subplot titles for both figures, and with the spectrogram window set to 100.

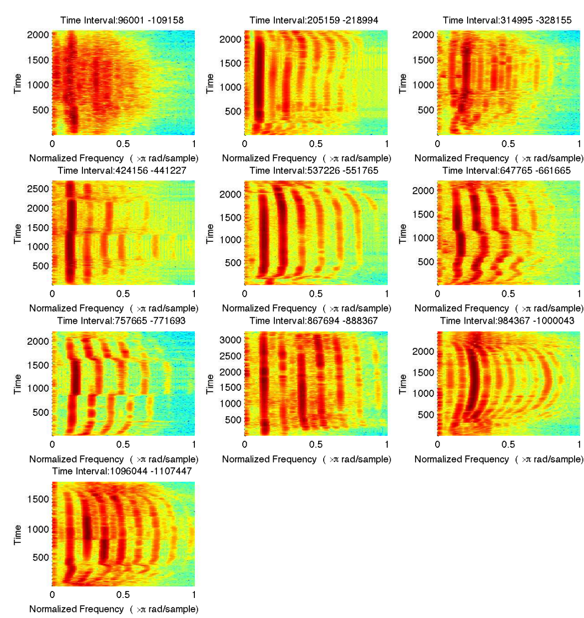


Fig. 1. Spectrograms of Manatee Training Data

III. Adaptive Filter Design

The role of adaptive filters in this problem consists of first determining the predictor parameter values from the training

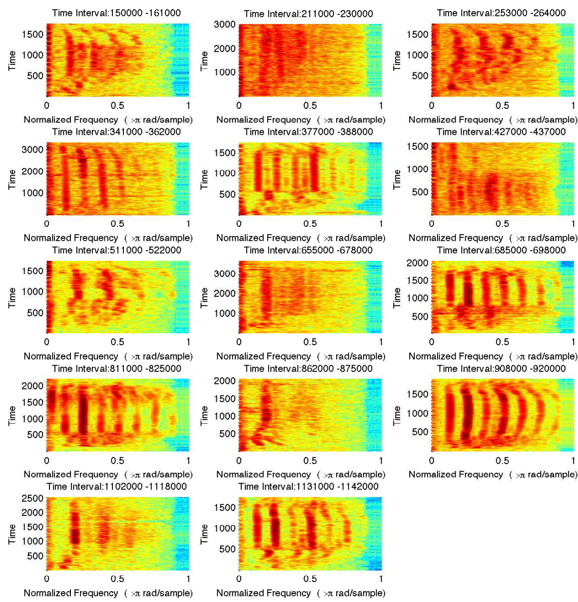


Fig. 2. Spectrograms of Manatee Test Data

data, and then filtering the testing data. Afterwards, features are generated from the filter errors for classification. Determining the predictor model parameter values is an optimization problem. In this study, the FIR filter model, the MSE cost function and the LMS algorithm were used to investigate the effect of two free parameters (LMS step size μ, described in Section III, and smoothing window L, described in Section IV) on classification performance AUC, described in Section V. The LMS algorithm is an online search method that iteratively updates the model parameters in the direction of steepest descent of the cost function, over the space of the model parameters. At every time step, the LMS algorithm consists of three steps. First, the filter output is computed as

$$y(t) = \mathbf{w}^T \mathbf{x}(t), \tag{2}$$

where $\mathbf{w} = [w_1, ..., w_M]^T$ is called the tap-weight vector, $\mathbf{x}(t) = [x(t-M+1), ..., x(t)]^T$ is called the tap-input vector, and M is the filter order. Second, the error is computed as

$$e(t) = d(t) - y(t). \tag{3}$$

Third, the tap-weight vector is updated as

$$\mathbf{w}(t+1) = \mathbf{w}(t) + 2\mu e(t)\mathbf{x}(t), \tag{4}$$

where μ is the step size parameter. The free parameters of this algorithm are model order, M, and LMS step size parameter, μ. Because the manatee calls and background noise are highly variable, small model orders generalize to test data better, so $M = 3$ is selected.

The training data of each class may be subdivided into multiple subclasses (e.g. specific types of manatee calls), with a predictor trained on data from each subclass. The manatee training data contains 10 manatee calls, and a predictor was trained on each manatee call. A single predictor was trained on the noise data, for a total of 11 predictors. To determine the predictor parameter values, the LMS algorithm is applied to the training data. The initial conditions of the weights are set to zero. Since it takes multiple passes through the training data for the weights to converge to steady state, 10 copies of each predictor's training data are concatenated for the predictor to adapt to. The desired signal is also constructed likewise with the final value of each copy set to zero. The step size parameter remains to be optimized with respect to the AUC measure of performance, discussed in Section V. For step size parameter $\mu = 0.81$, which will be shown to be optimal, the weight tracks are shown in Figure 3, where the first 10 subplots are from the manatee predictors and the final subplot is from the noise predictor. Finally, the mean weight tracks over the last pass through the training data are used as the fixed weights of the corresponding predictor. Figure 3 shows that the 10 manatee predictors have similar weights, which implies that the training manatee calls do not represent substantially different subclasses. It follows that a single manatee predictor may suffice to capture the variability in the training data.

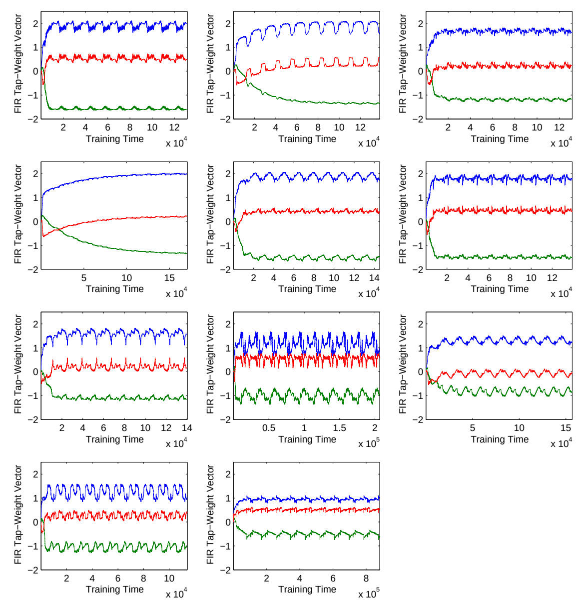


Fig. 3. Training Tap-Weight Vector, $\mu = 0.81$

IV. Feature Construction

The goal of binary classification is to label the test data as one of two classes, "manatee" or "noise." First, the test data is filtered by the predictors to generate the error sequences. Since prediction accuracy depends on the magnitude of the error, the log squared error is computed. After this step, if there are multiple predictors associated with the same class but with different subclasses, then a new sequence, $e_C(t)$, for that class is constructed as in Equation 5,

$$e_C(t) = \min_{s \in C} \log\left(e_s(t)^2\right), \tag{5}$$

where $e_s(t)$ is the error sequence from subclass predictor s, and class $C \in \{manatee, noise\}$. A portion of the the two sequences, $e_{manatee}$ and e_{noise}, is shown in Figure 4, with the true classification labels overlaid (manatee class high, noise class low).

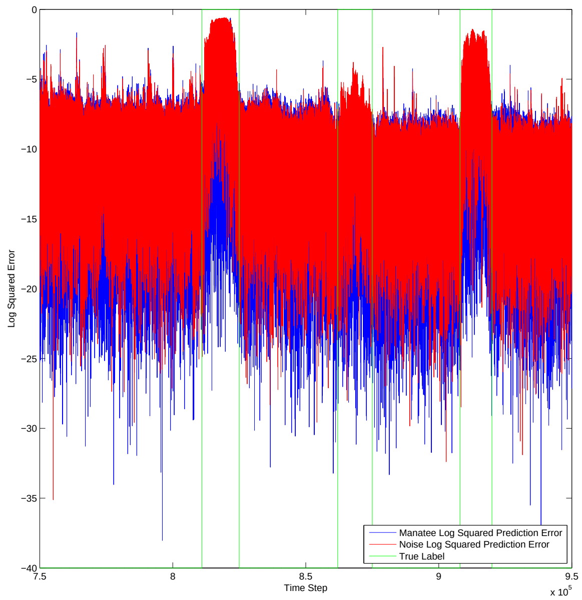


Fig. 4. Log Squared Errors without Smoothing

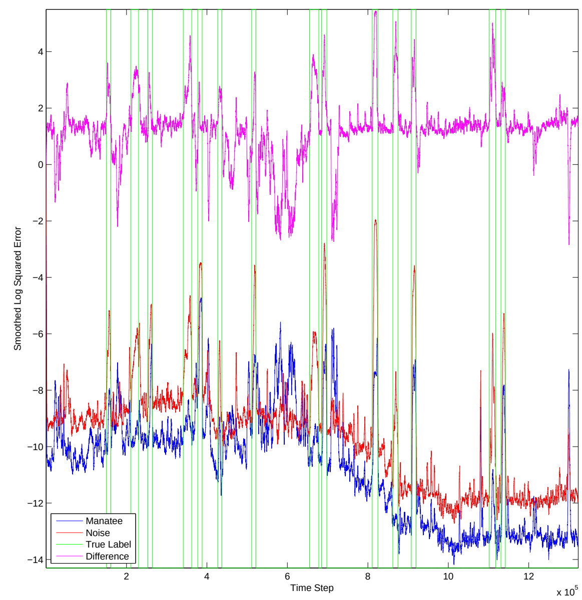


Fig. 5. Smoothed Log Squared Errors and their Difference

Figure 4 shows that the sequences $e_C(t)$ are noisy, which motivates smoothing. Smoothing is done as in Equation 6,

$$e_{C,L}(t) = \frac{1}{L} \sum_{\tau=t-L+1}^{t} e_C(\tau), \tag{6}$$

where L is the smoothing window length, which is a free parameter. Both the LMS step size μ and the smoothing window length L are free parameters that are selected as the optimizers of the AUC classification performance measure, discussed in Section V. For $L = 2100$, which will be shown to be optimal, the results of smoothing are shown in Figure 5. Also shown is the difference sequence,

$$\delta(t) = e_{noise,L}(t) - e_{manatee,L}(t). \tag{7}$$

Figure 5 shows that whenever there is a manatee call, $\delta(t)$ has high values. Accordingly, $\delta(t)$ is used as a feature for classification.

V. Classification

Binary classification of the feature $\delta(t)$ is done by comparing $\delta(t)$ to a discrimination threshold, where values higher than the threshold are labeled as "manatee" and values lower than the threshold are labeled as "noise." For a fixed threshold, δ_0, the resulting classification can be compared to the true class labels by computing the true positive rate (TPR) and the false positive rate (FPR), where

$$TPR = \frac{\sum_t \delta(t) > \delta_0}{\sum_t l(t) = manatee}, \tag{8}$$

$$FPR = \frac{\sum_t \delta(t) > \delta_0}{\sum_t l(t) = noise}, \tag{9}$$

and $l(t)$ is the true classification label sequence. For a threshold that varies as $\delta_0 \in (-\infty, \infty)$, the plot of $TPR(\delta_0)$ vs. $FPR(\delta_0)$ is called the Receiver Operating Characteristic (ROC). The area under the ROC curve (AUC) ranges from $[0, 1]$, with 1 corresponding to separable features. The AUC is the performance measure that the free parameters μ and L are optimized with respect to.

Figure 6 shows the effect of varying μ on the AUC with fixed optimal $L^* = 2100$, and Figure 7 shows the effect of varying L on the AUC with fixed optimal $\mu^* = 0.81$. The range of μ tested was $[0.75, 0.90]$ with interval 0.01, and the range of L tested was $[1800, 2200]$ with interval 10. For optimal μ and L, the ROC plot is shown in Figure 8. The highest AUC over the free parameters was $AUC^* = 0.9356$.

VI. Conclusion

This study used adaptive predictive filters to detect manatee calls in a noisy signal. The capability of human systems to detect manatee calls both from auditory and visual training data was used to label the test data. Order 3 FIR predictors were trained with LMS and applied to test data. The minimum log squared prediction errors over each subclass were smoothed, and their difference was the feature used to generate an ROC curve to obtain the AUC performance measure. Two free parameters, LMS step size μ and smoothing window L, were optimized with respect to the AUC performance measure, resulting in optimal performance of $AUC^* = 0.9356$. These results validate the capabilities of adaptive filters.

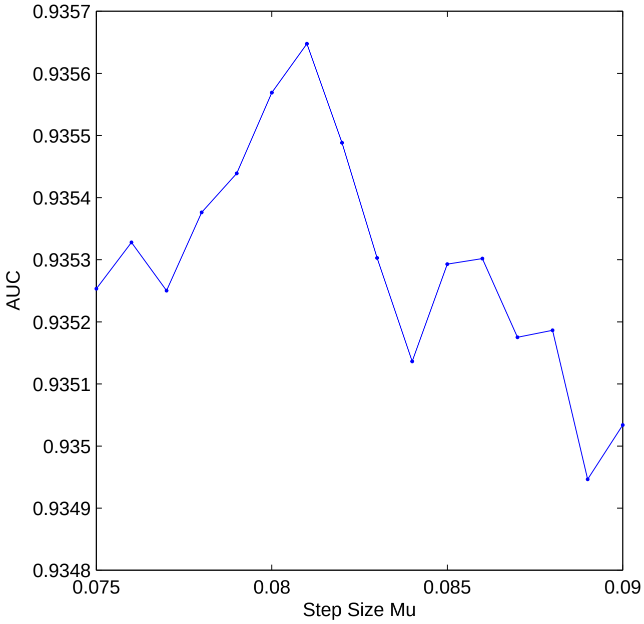


Fig. 6. Effect of μ on AUC with fixed optimal $L^* = 2100$

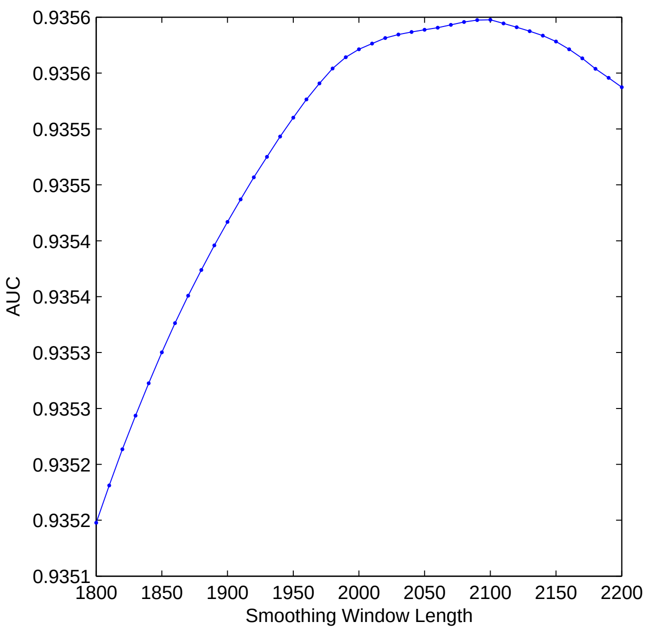


Fig. 7. Effect of L on AUC with fixed optimal $\mu^* = 0.081$

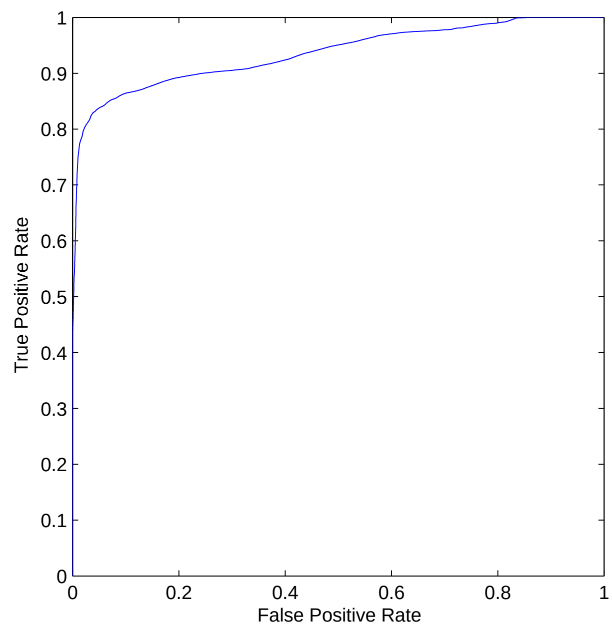
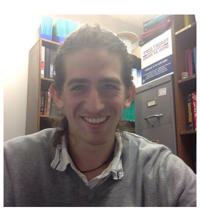


Fig. 8. ROC with optimal $\mu^* = 0.081$ and optimal $L^* = 2100$



Mark Rosenberg is pursuing his Ph.D. in Electrical Engineering at the University of Florida in the Computational NeuroEngineering Lab. He received his B.S. degree in Electrical Engineering from Washington University in St. Louis, St. Louis, MO, USA in 2012, and his M.S. degree in Electrical Engineering from the University of Florida, Gainesville, FL, USA in 2014.