AN ADAPTIVE SEQUENTIAL LEARNING ALGORITHM FOR ROBUST ESTIMATION USING THE FAIR PENALTY FUNCTION

Guang Deng

Department of Electronic Engineering, La Trobe University
Bundoora, Victoria 3086, Australia
d.deng@latrobe.edu.au

ABSTRACT

In this paper we propose an alternative way to developing a robust and adaptive sequential algorithm for estimating the unknown impulse response of a linear system. Our approach is based on formulating the problem as a maximum penalized likelihood (MPL) problem. We use the Fair penalty function as the generalized log-likelihood and a quadratic function to play a regularization role. The MPL formulation also leads naturally to adaptive schemes for learning the regularization and scale parameters. The robustness of the proposed algorithm to impulsive noise is demonstrated through mathematical analysis and numerical simulations.

Index terms: robust sequential learning, maximum penalized likelihood

1. INTRODUCTION

An application of supervised learning in signal processing is to identify a linear system of unknown impulse response. In an iterative formulation of the problem, we have at time n the available data, denoted $D_n = \{y_n, \mathbf{x}_n\}$, and the observation model

$$y_n = \mathbf{x}_n^T \mathbf{w} + r_n \tag{1}$$

where $\mathbf{w}$ is an $(M \times 1)$ vector, $\mathbf{x}_n$ is a $(M \times 1)$ input signal vector, y_n is the system output and r_n is the independent and identically distributed noise with a known distribution function. We also have the estimate of the system impulse response from the (n-1)th iteration, denoted $\mathbf{w}_{n-1}$. The problem is thus to determine an optimal estimation, denoted $\mathbf{w}_n$, based on the available information. A classical solution to this problem is the LMS algorithm [1]

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mu \bar{e}_n \mathbf{x}_n. \tag{2}$$

where μ is called the step size and $\bar{e}_n = y_n - \mathbf{x}_n^T \mathbf{w}_{n-1}$.

A potential problem for the LMS algorithm is that it is not robust to impulsive noise. Let us consider a case in which $\mathbf{w}_{n-1}$ is quite close to $\mathbf{w}$, and r_n is the impulsive noise. As such $\bar{e}_n$ is dominated by the noise and the second term of (2) is large. As a result $\mathbf{w}_n$ will be forced to move away from $\mathbf{w}_{n-1}$. Various types of robust adaptive filters, including robust LMS-type of algorithms [2–4], have been studied to tackle this problem. Robust LMS-type algorithms are in the following general form

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \Phi(|\bar{e}_n|)\text{sign}(\bar{e}_n)\mathbf{x}_n \tag{3}$$

where $\Phi(|\bar{e}_n|)$ is a nonlinear function of $|\bar{e}_n|$. One common characteristic of this nonlinear function is that its value is reduced or saturated when $|\bar{e}_n|$ is large indicating a possible case of impulsive noise. For example, the signed normalized LMS (NLMS) algorithm is given by $\Phi(|\bar{e}_n|) = \gamma/(\mathbf{x}_n^T\mathbf{x}_n)$ where γ is a constant. We can also see that when $\Phi(|\bar{e}_n|) = \gamma|\bar{e}_n|$ and $\Phi(|\bar{e}_n|) = \gamma|\bar{e}_n|/(\mathbf{x}_n^T\mathbf{x}_n)$, (3) becomes the LMS algorithm and the NLMS algorithm [1], respectively. A recent development of robust LMS-type of algorithm is the combination of two adaptive filters [5].

In this paper, we formulate the optimal estimation problem as a step-wise maximum penalized likelihood (MPL) problem and develop a robust sequential learning in which the parameters are adaptively updated. It is a step-wise algorithm, because at each time n when the training is received, we determine a new optimum estimate by solving an MPL problem. The robustness to impulsive noise is achieved by using a robust penalty function as the log-likelihood function. The so-called Fair penalty function is studied in this paper because it has only one parameter and has everywhere continuous derivatives of first three orders. It is recommended as the one of the best penalty functions [6]. Learning algorithm using Huber's penalty function is presented in [7]. On the other hand, we use a quadratic penalizing function as a regularization term to stabilize the algorithm. The algorithmic development is presented in section 2. In section 3, we present numerical examples which shows that the proposed algorithm has the desirable robust characteristics and its initial learning performance is close to that of a robust RLS-type of algorithm [8] and is better than that of an LMS-type of algorithm [5].

2. THE SEQUENTIAL LEARNING ALGORITHM

2.1. Step-wise maximum penalized likelihood

At time n, we receive the training data $D_n = \{y_n, \mathbf{x}_n\}$ and have the optimal estimate at time $n-1$, denoted $\mathbf{w}_{n-1}$. Then the objective function of maximum penalized likelihood problem can be stated as follows

$$\mathcal{P}_n(\mathbf{w}) = -\log p(y_n|\mathbf{w}) + \frac{1}{\alpha}\mathcal{F}(\mathbf{w};\mathbf{w}_{n-1}) \quad (4)$$

where $\log p(y_n|\mathbf{w})$ is the log-likelihood, α is a parameter and $\mathcal{F}(\mathbf{w};\mathbf{w}_{n-1})$ is the penalty function. The optimum estimate at time n is determined by solving the following optimization problem

$$\mathbf{w}_n = \arg\min_{\mathbf{w}} \mathcal{P}_n(\mathbf{w}) \quad (5)$$

In this paper, we choose the L_2 norm as the penalty function to simplify the development

$$\mathcal{F}(\mathbf{w};\mathbf{w}_{n-1}) = \frac{1}{2}(\mathbf{w}-\mathbf{w}_{n-1})^T(\mathbf{w}-\mathbf{w}_{n-1}) \quad (6)$$

The penalty function plays a role of regularization. The parameter α balances the two potentially conflicting requirements from the log-likelihood function and the penalty function: finding an estimate that fits the training data reasonably well and an estimate that is not too far away from the previous estimate.

To develop a robust algorithm, we regard the robust penalty function, denoted $\rho(r_n)$, as the generalized negative log-likelihood such that

$$\rho(r_n) = -\log p(y_n|\mathbf{w}) \quad (7)$$

In this paper, we consider the Fair penalty function [6] given by

$$\rho(t) = \sigma^2\left[|\frac{t}{\sigma}| - \log(1+|\frac{t}{\sigma}|)\right] \quad (8)$$

The gradient of the objective function is given by

$$\nabla\mathcal{P}_n(\mathbf{w}) = -\psi(r_n)\mathbf{x}_n + \frac{1}{\alpha}(\mathbf{w}-\mathbf{w}_{n-1}) \quad (9)$$

where $\psi(t) = \rho^{'}(t) = \dfrac{t}{1+|t|/\sigma}$. The solution to the optimization problem stated in (5) satisfies

$$\nabla\mathcal{P}_n(\mathbf{w})|_{\mathbf{w}=\mathbf{w}_n} = -\psi(\hat{e}_n)\mathbf{x}_n + \frac{1}{\alpha}(\mathbf{w}_n-\mathbf{w}_{n-1}) = 0 \quad (10)$$

where $\hat{e}_n = y_n - \mathbf{x}_n^T\mathbf{w}_n$. From (10), it is easy to show that

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \alpha\psi(\hat{e}_n)\mathbf{x}_n \quad (11)$$

Since $\mathbf{w}_n$ depends on $\psi(\hat{e}_n)$, we need to determine $\hat{e}_n$ which is given by

$$\hat{e}_n = \bar{e}_n - \alpha\psi(\hat{e}_n)\mathbf{x}_n^T\mathbf{x}_n \quad (12)$$

where $\bar{e}_n = y_n - \mathbf{x}_n^T\mathbf{w}_{n-1}$. When $\hat{e}_n$ is determined, we can use (11) to calculate the optimum estimate.

2.2. An approximate solution

Although there is a closed form solution for $\hat{e}_n$ by solving (12), the solution is quite complicated and does not have a clear interpretation. In this section, we propose to use an approximation of the penalty function $\rho(r_n)$ in order to have a simpler solution. More specifically, we can Taylor expand $\rho(r_n)$ around $\bar{e}_n$ and use a quadratic approximation as follows

$$\hat{\rho}(r_n) = \rho(\bar{e}_n) + \psi(\bar{e}_n)(r_n-\bar{e}_n) + \frac{1}{2}\varphi(\bar{e}_n)(r_n-\bar{e}_n)^2 \quad (13)$$

where $\varphi(\bar{e}_n) = \rho^{''}(\bar{e}_n) = (1+|\bar{e}_n|/\sigma)^{-2}$. An approximation of the cost function $\mathcal{P}_n(\mathbf{w})$ is then given by

$$\hat{\mathcal{P}}_n(\mathbf{w}) = \hat{\rho}(r_n) + \frac{1}{2\alpha}(\mathbf{w}-\mathbf{w}_{n-1})^T(\mathbf{w}-\mathbf{w}_{n-1}) \quad (14)$$

We first calculate the gradient and the Hessian as follows

$$\nabla\hat{\mathcal{P}}_n(\mathbf{w}) = -\psi(\bar{e}_n)\mathbf{x}_n - \varphi(\bar{e}_n)(r_n-\bar{e}_n)\mathbf{x}_n + \frac{1}{\alpha}(\mathbf{w}-\mathbf{w}_{n-1}) \quad (15)$$

and

$$\nabla\nabla\hat{\mathcal{P}}_n(\mathbf{w}) = \varphi(\bar{e}_n)\mathbf{x}_n\mathbf{x}_n^T + \frac{1}{\alpha}\mathbf{I} \quad (16)$$

Since $\varphi(\bar{e}_n) > 0$, the Hessian is positive definite. Thus, this objective function is strictly convex and the minimizer of $\hat{\mathcal{P}}_n(\mathbf{w})$ is a global minimum. Let $\mathbf{w}_n$ be the minimizer of $\hat{\mathcal{P}}_n(\mathbf{w})$ such that $\nabla\hat{\mathcal{P}}_n(\mathbf{w}_n) = 0$. From (15), we can write

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \alpha[\psi(\bar{e}_n) + \varphi(\bar{e}_n)(\hat{e}_n - \bar{e}_n)]\mathbf{x}_n \quad (17)$$

Left-multiplying both sides of the above equation by $\mathbf{x}_n^T$, then subtracting both sides by y_n, we obtain

$$\hat{e}_n = \bar{e}_n - \frac{\alpha\psi(\bar{e}_n)\mathbf{x}_n^T\mathbf{x}_n}{1+\alpha\varphi(\bar{e}_n)\mathbf{x}_n^T\mathbf{x}_n} \quad (18)$$

Substitute (18) into (17), we have

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \frac{\alpha\psi(\bar{e}_n)\mathbf{x}_n}{1+\alpha\varphi(\bar{e}_n)\mathbf{x}_n^T\mathbf{x}_n} \quad (19)$$

Comparing (11) with (19), we can see that using the quadratic approximation for $\rho(r_n)$ results in an approximation of $\psi(\hat{e}_n)$ by $\psi(\bar{e}_n)/(1+\alpha\varphi(\bar{e}_n)\mathbf{x}_n^T\mathbf{x}_n)$.

2.3. Robustness

To see the robustness of this learning algorithm, we rewrite (19) in the same form as (3). In this case, we have

$$\Phi(|\bar{e}_n|) = \alpha|\bar{e}_n|/\left(\epsilon + \frac{\alpha}{\epsilon}\mathbf{x}_n^T\mathbf{x}_n\right) \quad (20)$$

where $\epsilon = 1+|\bar{e}_n|/\sigma$. Since $\Phi^{'}(|\bar{e}_n|) = \frac{d\Phi(|\bar{e}_n|)}{d|\bar{e}_n|} > 0$ and $\lim_{|\bar{e}_n|\to\infty}\Phi(|\bar{e}_n|) = 1$, we can see that the value of $\Phi(|\bar{e}_n|)$ nonlinearly increases as $|\bar{e}_n|$ increases. Its value saturated when $|\bar{e}_n|$ is very large. As such, the learning algorithm avoids making large mis-adjustment when $|\bar{e}_n|$ is large.

2.4. An adaptive learning algorithm

We propose an adaptive update for the regularization parameter α and the scaling parameter σ. We can rewrite (19) as

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \frac{\alpha_{n-1}\bar{e}_n \mathbf{x}_n}{\epsilon + \frac{\alpha_{n-1}}{\epsilon}\mathbf{x}_n^T\mathbf{x}_n} \qquad (21)$$

where $\epsilon = 1 + |\bar{e}_n|/\sigma_{n-1}$. To update the regularization parameter, we can regard the penalty function as the logarithm of a Gaussian distribution. As such, α is naturally interpreted as the variance. We propose the update for it as follows

$$\alpha_n = \beta\alpha_{n-1} + (1-\beta)\frac{\alpha_{n-1}|\bar{e}_n|\sum_{i=1}^{M}|x_n(i)|}{\epsilon + \frac{\alpha_{n-1}}{\epsilon}\mathbf{x}_n^T\mathbf{x}_n} \qquad (22)$$

where $x_n(i)$ is the ith element of the input vector $\mathbf{x}_n$. We can see that as the learning process proceeds, it is expected that in an impulsive noise-free case, $\bar{e}_n$ becomes smaller. From (21) we can see that when $\alpha_{n-1} \to 0$, there is very little update. This may be undesirable in the early stage of the learning process. Thus to prevent it from approaching zero too early, after the update given by (22), we can force it to a pre-determined value α_{min} if $\alpha_n < \alpha_{\text{min}}$. This is implemented as: $\alpha_n = \max\{\alpha_n, \alpha_{\text{min}}\}$.

Similarly, we propose the update of the scaling parameter as follows

$$\sigma_n = \beta\sigma_{n-1} + (1-\beta)\min\{3\sigma_{n-1}, |\bar{e}_n|\} \qquad (23)$$

where the term $\min\{3\sigma_{n-1}, |\bar{e}_n|\}$ takes the smaller value of the two as the output. This operation provides the protection against the effect of impulsive noise in estimating the scaling parameter.

3. NUMERICAL EXAMPLES

3.1. Settings of the simulations

To study the robustness of the proposed algorithm, we set up the following simulation experiments. The impulse response ($\mathbf{w}$) of the system to be identified is given by $\mathbf{w} = [1\ 2\ 3\ 4\ 5\ 4\ 3\ 2\ 1]^T/10$. At time n, a random input signal vector $\mathbf{x}_n$ is generated as $\mathbf{x}_n = randn(9,1)$ and y_n is calculated using (1). The noise r_n is generated from a mixture of two zero mean Gaussian distributions with variance $s_1 = 0.1$ and $s_2 = 5$, respectively. This is simulated in Matlab by: rn = s1 * `randn(4000,1)` + s2 * `randn(4000,1)` .* (`abs(randn(4000,1)` > T). The threshold T controls the percentage of impulsive noise. In our simulations, we set $T = 2.5$ which corresponds to about 1.2% of impulsive noise. In Fig. 1, we plot the noise used in our simulations. The performance of an algorithm is measured by the distance between the two vectors $||\mathbf{w} - \mathbf{w}_n||^2$ which is a function of n and is called the learning curve. Each learning curve presented in the following is the result of averaging of 100-run of the program with the same additive noise and impulse response.

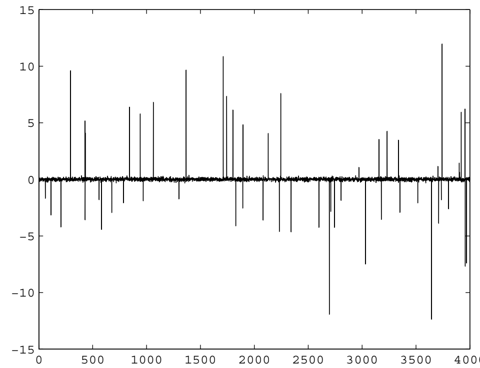

Fig. 1. The noises used in our simulations.

3.2. Simulation results

There are a number of parameters in the proposed algorithm. Through simulation study, we observe that the performance of the algorithm is not sensitive to variations in setting the initial values of the scaling parameter σ_0 and the regularization parameter α_0. We set $\sigma_0 = \alpha_0 = 5$ and $\mathbf{w}_0 = \mathbf{0}$. The parameter β controls the rate of adaptation for σ_n and α_n. This indirectly controls the learning performance of proposed algorithm. In Fig. 2, we test the case of fixing $\beta = 0.01$ and using three different settings of the parameter $\alpha_{\text{min}} supervisor.$. We observe that this parameter controls a trade-off between the learning rate and the steady state with $\alpha_{\text{min}} = 0.01$ being a good choice. In Fig. 3, we test the case of fixing $\alpha_{\text{min}} = 0.01$ and using three different settings of the parameter β. We observe that $\beta = 0.9$ is a good choice.

Next, we compare the learning performance of the proposed algorithm with a recently published RLM algorithm [8] using the suggested values of parameters. The RLM algorithm is a robust RLS type of algorithm. It is computationally more expensive than the proposed algorithm. We also compare the performance of the proposed algorithm with that of a recently published algorithm (called CAF algorithm) which adaptively combines the NLMS and the signed NLMS algorithms using the suggested settings of parameters [5]. Results are presented in Figure 4. We can see that the performance of the proposed algorithm is comparable to that of the CAF and the initial learning rate is close to that of the RLM algorithm.

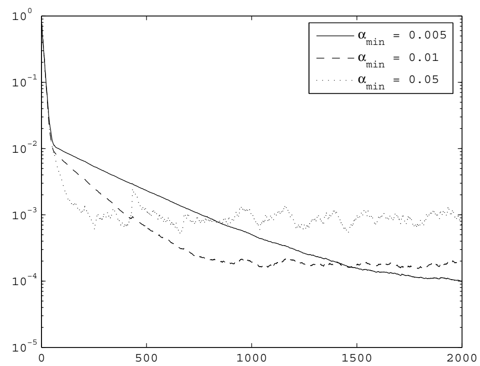

Fig. 2. The learning curves for the proposed algorithm under three different settings of the parameter α_{min} when $\beta = 0.01$.

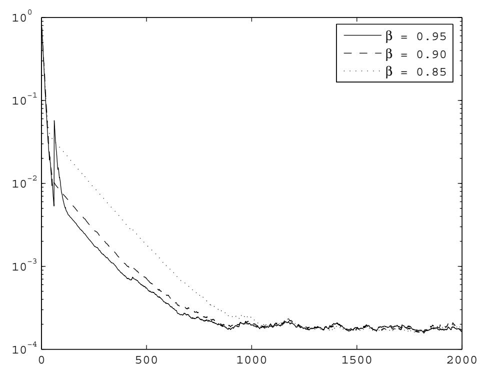

Fig. 3. The learning curves for the proposed algorithm under three different settings of the parameter β when $\alpha_{\text{min}} = 0.01$.

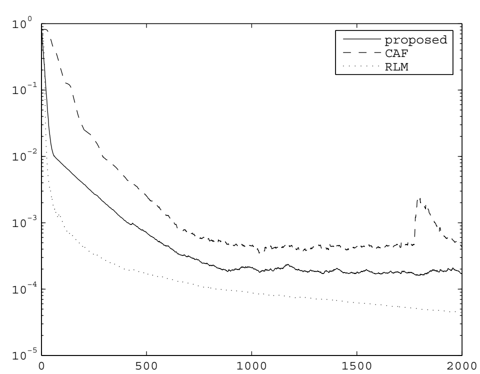

Fig. 4. A comparison of the learning curves of the proposed algorithm with those of a recently proposed RLM algorithm and the CAF algorithm.

4. CONCLUSIONS

In this paper we have presented an adaptive sequential learning algorithm which is robust to impulsive noise. The development is based on formulating the problem as a maximum penalized likelihood (MPL) problem which also permits us to develop an adaptive scheme to update the two model parameters. We have demonstrated the robustness of the proposed algorithm through mathematical analysis and numerical simulations. We note that the MPL approach is a general approach for developing sequential learning algorithms. It is thus important to extend this work to other type of robust penalty functions. Since the Bayesian approach is more general than the MPL approach, it is also important to study robust sequential learning algorithm within the full Bayesian framework.

5. REFERENCES

[1] S. Haykin, *Adaptive Filter Theory*, forth ed. Englewood Cliffs, New Jersey, USA: Prentice Hall Inc., 2002.

[2] S. C. Bang and S. Ann, "A robust algorithm for adaptive FIR filtering and its performance analysis with additive contaminated-Gaussian noise," *IEEE Trans. Circuits Syst. I*, vol. 43, no. 5, pp. 361–369, May 1996.

[3] J. Chambers and A. Avlonitis, "A robust mixed norm adaptive filter algorithm," *IEEE Signal Processing Lett.*, vol. 4, no. 2, pp. 46–48, Feb. 1997.

[4] Y. X. Zou, S. C. Chan, and T. S. Ng, "Least mean M-estimate algorithms for robust adaptive filtering in impulse noise," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 12, pp. 1564–1569, Dec. 2000.

[5] J. Arenas-García and A. R. Figueiras-Vidal, "Adaptive combination of normalised filters for robust system identification," *Electronics Letters*, vol. 41, pp. 874–875, 2005.

[6] W. J. J. Rey, *Introduction to Robust and Quasi-Robust Statistical Methods*. Berlin ; New York: Springer-Verlag, 1983.

[7] G. Deng and W.-Y. Ng, "A model-based approach for the development of LMS algorithms," in *Proc. IEEE ISCAS 2005, CDROM*.

[8] S. C. Chan and Y. X. Zou, "A recursive least M-estimate algorithm for robust adaptive filtering in impulsive noise: Fast algorithm and convergence performance analysis," *IEEE Trans. Signal Processing*, vol. 52, no. 4, pp. 975–991, Apr. 2004.