Least Mean Square Algorithm

B Swaroop Reddy and Dr G V V Sharma*

Contents

***Abstract*—This manual provides an introduction to the LMS algorithm.**

1 Source Files

1) Get the git source and enter the local directory

```
git clone https://github.com/gadepall/adsp.git
cd adsp/audio_source
```

2) Play the **signal_noise.wav** and **noise.wav** file.

2 Problem Formulation

The **signal_noise.wav** $d(n)$ contains a human voice along with an instrument sound in the background. This sound is captured in **noise.wav** $X(n)$. The goal is to suppress $X(n)$ in d_n. Let

$$d(n) = e(n) + y(n) \tag{2.1}$$

where $e(n)$ is the desired signal. We want an estimate of $I(n)$ from $X(n)$. This can be done by considering

$$y(n) = W^T(n)X(n) \tag{2.2}$$

where

$$X(n) = \begin{bmatrix} X(n) \\ X(n-1) \\ X(n-2) \\ .. \\ .. \\ X(n-M+1) \end{bmatrix}_{MX1} \tag{2.3}$$

$$W(n) = \begin{bmatrix} w_1(n) \\ w_2(n) \\ w_3(n) \\ .. \\ .. \\ w_{n-M+1}(n) \end{bmatrix}_{MX1} \tag{2.4}$$

and estimating $W(n)$. The human voice can be characterized as

$$e(n) = d(n) - W^T(n)X(n) \tag{2.5}$$

The goal is to find $W(n)$ that will allow $W^T(n)X(n)$ to mimic the instrument sound in $d(n)$. This is possible if $e(n)$ is minimum. This problem can be expressed as

$$\min_{W(n)} e^2(n) \tag{2.6}$$

3 Gradient Descent Method

Consider the problem of finding the square root of a number c. This can be expressed as the equation

$$x^2 - c = 0 \tag{3.1}$$

Problem 3.1. Show that (3.1) results from

$$\min_x f(x) = x^3 - 3xc \tag{3.2}$$

Problem 3.2. Find a numerical solution for (3.1).

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in. All content in this manual is released under GNU GPL. Free and open source.

Solution: A numerical solution for (3.1) is obtained as

$$x_{n+1} = x_n - \mu f'(x) \tag{3.3}$$
$$= x_n - \mu\left(3x_n^2 - 3c\right) \tag{3.4}$$

where x_0 is an inital guess.

Problem 3.3. Write a program to implement (3.4).

Solution: Execute **square_root.py** in the **lms** directory.

4 LMS Algorithm

Problem 4.1. Show using (5.1) that

$$\nabla_{W(n)} e^2(n) = \frac{\partial e^2(n)}{\partial W(n)} \tag{4.1}$$
$$= -2X(n)d(n) + 2X(n)X^T(n)W(n) \tag{4.2}$$

Problem 4.2. Use the gradient descent method to obtain an algorithm for solving

$$\min_{W(n)} e^2(n) \tag{4.3}$$

Solution: The desired algorithm can be expressed as

$$W(n+1) = W(n) - \bar{\mu}[\nabla_{W(n)} e^2(n)] \tag{4.4}$$
$$W(n+1) = W(n) + \mu X(n)e(n) \tag{4.5}$$

where $\mu = \bar{\mu}$.

Problem 4.3. Write a program to suppress $X(n)$ in $d(n)$.

Solution: Execute **LMS_NC_SPEECH.py** in the **lms** directory.

5 Wiener-Hopf Equation

Problem 5.1. Let

$$e(n) = d(n) - W^T(n)X(n) \tag{5.1}$$

Show that

$$E[e^2(n)] = r_{dd} - W^T(n)r_{xd} - r_{xd}^T W(n) + W^T(n)RW(n) \tag{5.2}$$

where

$$r_{dd} = E[d^2(n)] \tag{5.3}$$
$$r_{xd} = E[X(n)d(n)] \tag{5.4}$$
$$R = E[X(n)X^T(n)] \tag{5.5}$$

Problem 5.2. By computing

$$\frac{\partial J(n)}{\partial W(n)} = 0, \tag{5.6}$$

show that the optimal solution for

$$W^*(n) = \min_{W(n)} E\left[e^2(n)\right] = R^{-1} r_{xd} \tag{5.7}$$

This is the Wiener optimal solution.

6 Convergence of the LMS Algorithm

6.1 Convergence in the Mean

Problem 6.1. Show that R in (5.5) is symmetric as well as positive definite.

Let

$$\tilde{W}(n) = W(n) - W_* \tag{6.1}$$

where W_* is obtained in (5.7). Also, according to the LMS algorithm,

$$W(n+1) = W(n) + \mu X(n)e(n) \tag{6.2}$$
$$e(n) = d(n) - X^T(n)W(n) \tag{6.3}$$

Problem 6.2. Show that

$$E\left[\tilde{W}(n+1)\right] = [I - \mu R]E\left[\tilde{W}(n)\right] \tag{6.4}$$

Problem 6.3. Show that

$$R = U\Lambda U^T \tag{6.5}$$

for some U, Λ, such that Λ is a diagonal matrix and $U^T U = I$.

Problem 6.4. Show that

$$\lim_{n\to\infty} E\left[\tilde{W}(n+1)\right] = 0 \iff \lim_{n\to\infty} [I - \mu\Lambda]^n = 0 \tag{6.6}$$

Problem 6.5. Using (6.6), show that

$$0 < \mu < \frac{2}{\lambda_{\max}} \tag{6.7}$$

where $\lambda_{\max}$ is the largest entry of Λ.

6.2 Convergence in Mean-square sense

Let

$$X(n) = \begin{bmatrix} X_1(n) \\ X_2(n) \end{bmatrix} \tilde{W}(n) = \begin{bmatrix} \tilde{W}_1(n) \\ \tilde{W}_2(n) \end{bmatrix} \tag{6.8}$$

Problem 6.6. Show that

$$E[\tilde{W}^T(n)X(n)X^T(n)\tilde{W}(n)] = E[\tilde{W}^T(n)R\tilde{W}(n)] \tag{6.9}$$

for R defined in (5.5).

Problem 6.7. Show that

$$J(n) = E[e^2(n)] = E[e_*^2(n)] + E[\tilde{W}(n)X(n)X(n)^T\tilde{W}(n)^T] - E[\tilde{W}(n)X(n)e_*(n)] - E[e_*(n)X^T(n)\tilde{W}^T(n)] \quad (6.10)$$

where

$$\tilde{W}(n) = W(n) - W_* \quad (6.11)$$

$$e_*(n) = d(n) - W_*X(n) \quad (6.12)$$

Problem 6.8. Show that

$$E\left[\tilde{W}(n)X(n)e_*(n)\right] = E\left[e_*(n)X^T(n)\tilde{W}^T(n)\right] = 0 \quad (6.13)$$

Problem 6.9. Show that

$$E\left[\tilde{W}^T(n)R\tilde{W}(n)\right] = \mathrm{trace}\left(E\left[\tilde{W}^T(n)R\tilde{W}(n)\right]\right) \quad (6.14)$$

$$= \mathrm{trace}\left(E\left[\tilde{W}(n)\tilde{W}^T(n)\right]R\right) \quad (6.15)$$

Problem 6.10. Using (6.11), (6.2) and (6.12), show that

$$\tilde{W}(n+1) = \left[I - \mu X(n)X^T(n)\right]\tilde{W}(n) + \mu X(n)e_*(n) \quad (6.16)$$

Problem 6.11. Let $\mu^2 \to 0$. Using (6.5) and (5.7), show that

$$E\left[\tilde{W}(n+1)\tilde{W}^T(n+1)\right] = (I - 2\mu R)\,E\left[\tilde{W}(n)\tilde{W}^n(n)\right] \quad (6.17)$$

Problem 6.12. Show that

$$\lim_{n\to\infty} E\left[\tilde{W}(n)\tilde{W}^T(n)\right] = 0 \iff 0 < \mu < \frac{1}{\lambda_{max}} \quad (6.18)$$

Problem 6.13. Find the value of the cost function at infinity i.e. $J(\infty)$

Problem 6.14. How can you choose the value of μ from the convergence of both in mean and mean-square sense?