

EJ. 08

Dec 3. 17

1/11

# ADVANCED SIGNAL PROCESSING 2008

## Solutions:

### 1) [Bookwork]

$$z[n] = a_1 z[n-1] + a_2 z[n-2] + \dots + a_p z[n-p] + w[n]$$

where  $a_1, \dots, a_p$  are the model parameters and  $\{w[n]\}$  is the driving white noise.

a) i) The first order Markov process is given by

$$z[n] = az[n-1] + w[n]$$

By applying the expectation operator  $E\{\cdot\}$  to

$$z[n-k]z[n]$$

we have

$$\rho(k) = a\rho(k-1) \quad \text{or} \quad \rho(k) = a^k, \quad k \geq 0$$

where  $\rho(0) = 1$  and  $\rho(1) = a$ .

### ii) [worked example]

The ACF for  $a = \pm 0.9$  is  $\rho(k) = (\pm 0.9)^k, k \geq 0$ . The plots are a decaying function with or without alternating the sign (for a negative  $a$ ).

### iii) [bookwork and worked example]

For  $k = 0$  the variance becomes

$$\sigma_z^2 = \frac{\sigma_w^2}{1 - a\rho(1)} = \frac{\sigma_w^2}{1 - a^2}$$

The spectrum of an AR(1) process is given by

$$S(f) = \frac{2\sigma_w^2}{|1 - ae^{-j2\pi f}|^2} = \frac{2\sigma_w^2}{1 + a^2 - 2a \cos(2\pi f)}$$

For a negative  $a$  this represents a high-pass filter.

### iv) [bookwork and intuitive reasoning]

Initially we may not know which order of autoregressive process to fit to an observed time series. This problem is analogous to deciding on the number of independent variables to be included in a multiple regression.

The partial autocorrelation function is a device which exploits the fact that whereas an AR(p) process has an autocorrelation function which is infinite in extent, it can by its very nature be described in terms of  $p$  nonzero functions of

S 1

autocorrelations. Denote by  $a_{kj}$  the  $j$ th coefficient in an autoregressive representation of order  $k$ , so that  $a_{kk}$  is the last coefficient. The  $a_{kj}$  satisfy the set of equations

$$\rho(j) = a_{k1}\rho(j-1) + \dots + a_{kk}\rho(j-k) \quad j = 1, 2, \dots, k$$

leading to the Yule–Walker equations. The quantity  $a_{kk}$ , regarded as a function of lag  $k$  is called the partial autocorrelation function. The large values of the partial autocorrelation function may therefore indicate undermodelling.

b) **[new example]**

From the stationarity and invertibility conditions of AR(1) and MA(1) processes, process  $z[n]$  is stationary for  $-1 < a_1 < 1$  and invertible for  $-1 < b_1 < 1$ .

i) and ii) **[bookwork and new example]**

From question a) part i) we know that the autocorrelation coefficients of an ARMA(p,q) process are equal to the autocorrelation coefficients of an AR(p) process for  $k \geq q+1$ . From the autocovariance function, dividing by  $c(0)$ , for an ARMA(1,1) process we have

$$\begin{aligned} \rho_0 &= 1 \\ \rho_1 &= \frac{(1 + a_1 b_1)(a_1 + b_1)}{1 + b_1^2 + 2a_1 b_1} \\ \rho_k &= a_1 \rho_{k-1}, \quad k \geq 2 \end{aligned}$$

2) a) [bookwork and worked example]

The optimisation problem to find the BLUE is:

- minimise the variance of the BLUE  $\text{var}\{\hat{\theta}\} = \mathbf{a}^T \mathbf{C} \mathbf{a}$

subject to

- the unbiased constraint  $\sum_{n=0}^{N-1} E\{x[n]\} = \theta \Leftrightarrow \sum_{n=0}^{N-1} a_n s[n] \theta = \theta \Leftrightarrow \mathbf{a}^T \mathbf{s} = 1$ .

Since  $E\{x[n]\} = A$ , therefore  $s[n] = 1$  and  $\mathbf{s} = \mathbf{1}$

(Follows from  $E\{x[n]\}$  linear in  $\theta$  or  $E\{x[n]\} = s[n]\theta$ ).

The BLUE becomes

$$\hat{A} = \frac{\mathbf{1}^T \frac{1}{\sigma^2} I \mathbf{x}}{\mathbf{1}^T \frac{1}{\sigma^2} I \mathbf{1}} = \frac{1}{N} \sum_{n=1}^{N-1} x[n] = \bar{x}$$

and has minimum variance (for a linear estimator)

$$\text{Var}(\hat{A}) = \frac{1}{\mathbf{1}^T \frac{1}{\sigma^2} I \mathbf{1}} = \frac{\sigma^2}{N}$$

The sample mean is the BLUE independent of the PDF of the data.

It is the MVU estimator for Gaussian noise.

i) and ii) [bookwork]

When the PDF of the data is **unknown**, or **cannot be assessed**, the MVU estimator, even if it exists, cannot be found

⇒ Methods which rely on the CRLB cannot be applied

⇒ Resort to a sub-optimal estimator - check its variance and ascertain whether this meets the required specifications

⇒ Common approach - assume estimator to be **linear in the data**, which is **unbiased** and has **minimum variance**

□ This estimator is termed the Best Linear Unbiased Estimator (BLUE) which **requires only knowledge of the first two moments of the PDF**.

Note if the data are Gaussian, the BLUE and MVUEs are equivalent.

b) [bookwork and worked example] D.C. level in WGN

$$x[n] = A + w[n] \quad n=0,1,\dots,N-1$$

↑  
A to be estimated

$$w[n] \sim N(0, \sigma^2)$$

$$\text{PDF } p(\underline{x}; A) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp \left[ -\frac{1}{2\sigma^2} \sum_{n=0}^{N-1} (x[n] - A)^2 \right]$$

Take the derivative of the log-likelihood function

$$\frac{\partial \ln p(\underline{x}; A)}{\partial A} = \frac{1}{\sigma^2} \sum_{n=0}^{N-1} (x[n] - A)$$

Set the result to zero to yield the MLE

$$\hat{A} = \frac{1}{N} \sum_{n=0}^{N-1} x[n]$$

Which is clearly the MVU estimator which yields the CRLB, hence it is efficient.

i) and ii) [**bookwork and worked example**]

If an efficient estimator exists the maximum likelihood procedure will produce it

When an efficient estimator does not exist, the MLE has the desirable feature that it yields "an asymptotically efficient" estimator - namely one, that for sufficiently large datasets, that

- is unbiased
- achieves the CRLB
- has a Gaussian PDF,  $\hat{\theta}^{asy} \sim \mathcal{N}(\theta, I^{-1}(\theta))$

Provided the PDF  $p(\underline{x}; \theta)$  satisfies the regularity conditions, the derivatives of the log-likelihood function exist and the Fisher information is non-zero.

c) [**bookwork and worked example**]

$$J(A) = \sum_{n=0}^{N-1} (x[n] - A)^2$$

$$\frac{dJ(A)}{dA} = 2 \sum_{n=0}^{N-1} (x[n] - A)$$

Set the derivative to zero to yield the LS estimator

$$\hat{A} = \frac{1}{N} \sum_{n=0}^{N-1} x[n]$$

i) and ii) [**bookwork**]

- There are no assumptions on the pdf of the noise.

- Familiar sample mean, but it cannot be claimed to be optimal in the MVU sense, except if

$$x[n] = A + w[n]$$

with  $w[n] \sim N(0, \sigma^2)$ ; all we can say is that it minimises the sum squared error.

- applies only to zero-mean noise.

d) [bookwork and worked example]

Determine the CRLB for A

$$\begin{aligned} p(\underline{x}; \theta) &= \prod_{n=0}^{N-1} \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left[ -\frac{1}{2\sigma^2} (x[n] - A)^2 \right] \\ &= \frac{1}{(2\pi\sigma^2)^{N/2}} \exp \left[ -\frac{1}{2\sigma^2} \sum_{n=0}^{N-1} (x[n] - A)^2 \right] \end{aligned}$$

Taking the first derivative

$$\begin{aligned} \frac{\partial \ln p(\underline{x}; A)}{\partial A} &= \frac{\partial}{\partial A} \left[ -\ln [(2\pi\sigma^2)^{N/2}] - \frac{1}{2\sigma^2} \sum_{n=0}^{N-1} (x[n] - A)^2 \right] \\ &= \frac{1}{\sigma^2} \sum_{n=0}^{N-1} (x[n] - A) = \frac{N}{\sigma^2} (\bar{x} - A) \end{aligned}$$

where  $\bar{x}$  is the sample mean.

Differentiating again

$$\frac{\partial^2 \ln p(\underline{x}; A)}{\partial A^2} = -\frac{N}{\sigma^2}$$

Therefore  $\text{Var}(\hat{A}) \geq \frac{\sigma^2}{N}$  is the CRLB, which implies that the sample mean estimator attains the bound and must, therefore, be the MVU estimator in the WGN.

3) a) [bookwork] • Signal,  $s[n]$  is assumed to be generated by the signal

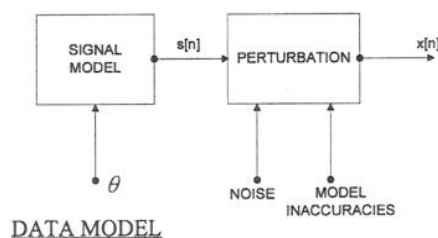


Figure 1: Data model for least squares estimation

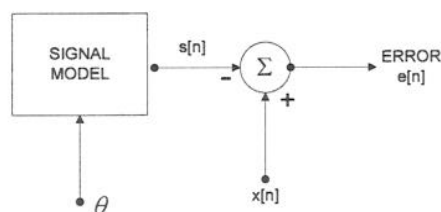
model which is a function of  $\theta$

- The observation noise/model inaccuracies perturb  $s[n]$  to yield the measurement  $x[n]$
- The Least Squares Estimator of  $\theta$  chooses the value that makes  $s[n]$  closest to the observed data  $x[n]$ , where closeness is measured by the LS error criterion

$$J(\theta) = \sum_{n=0}^{N-1} \underbrace{(x[n] - s[n])^2}_{e[n]}$$

$$\text{LSE: } \min_{\theta} J(\theta)$$

Note, no probabilistic assumptions have been made about the data  $x[n]$



b) [new example]

The MSE we want to minimise is

$$\xi = E\{[x(n+1) - \hat{x}(n+1)]^2\} = E\{x^2(n+1) - 2x(n+1)\hat{x}(n+1) + \hat{x}^2(n+1)\}$$

i) Since the estimate of  $x(n+1)$  is

$$\hat{x}(n+1) = ax(n) + bx(n-1)$$

then setting the derivative of  $\xi$  wrt  $a$  and  $b$  equal to zero, we have

$$\begin{aligned}\frac{\partial \xi}{\partial a} &= -2E\{x(n+1)x(n) + E\{2\hat{x}(n+1)x(n)\}\} = 0 \\ \frac{\partial \xi}{\partial b} &= -2E\{x(n+1)x(n-1) + E\{2\hat{x}(n+1)x(n-1)\}\} = 0\end{aligned}$$

Dividing by 2 and substituting for  $\hat{x}(n+1)$  gives

$$\begin{bmatrix} r_x(0) & r_x(1) \\ r_x(1) & r_x(0) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} r_x(1) \\ r_x(0) \end{bmatrix}$$

Solving for  $a$  and  $b$  we find

$$\begin{bmatrix} a \\ b \end{bmatrix} = \frac{1}{r_x^2(0) - r_x^2(1)} \begin{bmatrix} r_x(0)r_x(1) - r_x(1)r_x(2) \\ r_x(0)r_x(2) - r_x^2(1) \end{bmatrix}$$

ii)

If  $x(n)$  and  $x(n+1)$  are uncorrelated, then  $r_x(1) = 0$  and the values for  $a$  and  $b$  become

$$a = 0 \quad b = r_x(2)/r_x(0)$$

In this case, the linear predictor is

$$\hat{x}(n+1) = \frac{r_x(2)}{r_x(0)}x(n-1)$$

Similarly, if  $x(n+1)$  is uncorrelated with both  $x(n)$  and  $x(n-1)$ , then the values for  $a$  and  $b$  are

$$a = b = 0$$

and the linear predictor is

$$\hat{x}(n+1) = 0$$

which is equal to the expected value of  $x(n+1)$

$$\hat{x}(n+1) = E\{x(n+1)\}$$

c) [new example]

The AR(2) model for process  $x(n+1)$  is

$$x(n+1) = a_1x(n) + a_2x(n-1) + w(n)$$

where  $a_1 = a$  and  $a_2 = b$ .

By applying the expectation operator, we have

$$\hat{x}(n+1) = E\{x(n+1)\} = a_1x_n + a_2x(n-1)$$

since  $E\{w\} = 0$ . To find  $a_1$  and  $a_2$  we can use standard Yule-Walker equations.

i) [new example]

The value of prediction error is proportionate to the noise variance. The prediction error power is equal to the variance of the driving noise of the AR model.

4) a) The problem: Given we know the LSE  $\hat{\theta}$  based on  $\{x[0]x[1] \dots x[N-1]\}$ , and then we observe  $x[N]$ , can we update  $\hat{\theta}$  (in time) without having to solve the normal equations?

b) Data can be collected sequentially, namely one point at a time. We can either wait until all the data points (samples) are collected and make an estimate of the unknown parameter, namely the block-based approach or least squares, or refine our estimate in time as each new sample arrives, the sequential approach or sequential least squares.

c) Such an estimator is computationally much less demanding and can be run on-line.

d) i)

$$\hat{A}[N-1] = \frac{1}{N} \sum_{n=0}^{N-1} x[n]$$

If we now observe,  $x[N]$ , we can rewrite the LSE

$$\begin{aligned} \hat{A}[N] &= \frac{1}{N+1} \sum_{n=0}^N x[n] = \frac{1}{N+1} \left( \sum_{n=0}^{N-1} x[n] + x[N] \right) \\ \Rightarrow \hat{A}[N] &= \frac{N}{N+1} \hat{A}[N-1] + \frac{1}{N+1} x[N] \end{aligned}$$

Clearly  $\hat{A}[N]$  can be calculated from  $\hat{A}[N-1]$  together with the new observation  $x[N]$ .

ii) Equation I) can be rewritten as

$$\underbrace{\hat{A}[N]}_{\text{New estimate}} = \underbrace{\hat{A}[N-1]}_{\text{Old estimate}} + \frac{1}{N+1} \left( \underbrace{x[N] - \hat{A}[N-1]}_{\text{correction term - error in prediction } x[n] \text{ by the previous sample}} \right)$$

iii) The minimum LS error may also be computed recursively

$$J_{\min}[N] = J_{\min}[N-1] + \frac{N}{N+1} (x[N] - \hat{A}[N-1])^2$$

e) This filter is a moving average filter, working on only four signal samples at a time. It is fast to estimate the unknown parameter value but is more prone to noise since it is not adaptive and is very short.

In fact it is a low pass filter, with

$$H(z) = \frac{1 - z^{-4}}{1 - z^{-1}}$$



which filters out the low pass component of the additive WGN.

Therefore, both the least squares and sequential least squares provide a better estimate of a DC level in WGN and they are asymptotically unbiased and consistent.

5) a) [bookwork] To avoid the matrix inversion involved in the Wiener filter, we can make the weights time varying and adapt them in an iterative fashion. In that case

$$\Delta w_k(n) = -\eta \nabla_{w_k} J(n), \quad k = 1, \dots, N$$

and

$$w_k(n+1) = w_k(n) + \Delta w_k(n)$$

Therefore

$$w_k(n+1) = w_k(n) + \eta \left[ r_{dx} - \sum_{j=1}^N w_j(n) r_{xx}(j, k) \right]$$

The method of steepest descent is exact in the sense that there are no approximations made in its derivation.

b) If we use the instantaneous estimates instead of the exact estimates of the autocorrelation and crosscorrelation functions, we have

$$\begin{aligned} r_{xx}(j, k; n) &= x_j(n)x_k(n) \\ r_{dx}(k; n) &= x_k(n)d(n) \end{aligned}$$

and the LMS algorithm becomes

$$w_k(n) = w_k(n-1) + \eta e(n)x_k(n), \quad k = 1, \dots, N$$

i) The learning rate defines the step towards the global minimum on the error surface. The convergence in the mean is attained provided that

$$0 < \eta < \frac{2}{\lambda_{max}}$$

where  $\lambda_{max}$  is the largest eigenvalue of the autocorrelation matrix. The error is stemming from the use of the instantaneous estimates of the quantities from the steepest descent algorithm, and also comes from the use of large learning rate.

ii) The LMS is computationally much simpler, can deal with both the stationary and nonstationary data and is very robust. However, due to the instantaneous estimates of the second order quantities from the Wiener filter, its steady state error can be considerable and the convergence is relatively slow. It is based on the “stochastic” cost function, that is, an instantaneous estimate. The least squares solution is based on the deterministic cost function, i.e. on the minimisation of the sum of squared errors for the whole signal. There are no approximation and

the method is best suited for stationary data.

c) [new example]

This is straightforward from the LMS derivation from b). You can either assume two different coefficients  $w_1$  and  $w_2$  and then simplify in order to find  $w$ , or derive the update directly for  $w$ .

i) The convexity helps to have the estimate somewhere between estimates  $x_1$  and  $x_2$ . This can also help to constrain the solutions, since the LMS update is coming from an unconstrained optimisation problem. In the limit the value of  $w$  determines which input is to be more trusted. The binary values of  $w$  select only one input  $x_1$  or  $x_2$ , which can be realised by using a hard limiter after the LMS adaptation.

