E8: Least Squares

Consider a signal model that involves measuring a signal parameterized by θ with noise.

$$x[n] = s[n;\theta] + \epsilon[n] \qquad n = 0, 1, \ldots, (N-1)$$

Suppose that $\epsilon[n]$ contains a mixture of noise and modelling inaccuracies in general (but we will call it noise or error).

A measure of closeness between $x[n]$ and $s[n;\theta]$ is the sum of squared errors (L_2-norm-squared of the error).

$$J(\theta) = \sum_{n=0}^{N-1} \left(x[n] - s[n;\theta] \right)^2 \quad \text{(least squares criterion)}$$

The value of θ that minimizes $J(\theta)$ is the least squares estimator (LSE).

$$\hat{\theta}_{LSE} = \underset{\theta}{\arg\min}\ J(\theta).$$

While so far we did not make any assumptions about the distribution of $x[n]$ (or equivalently that of $\epsilon[n]$, since $s[n;\theta]$ is deterministic), clearly the performance of $\hat{\theta}_{LSE}$ will depend on the statistical properties of $x[n]$. Recall that in the case of additive Gaussian noise, we encountered $J(\theta)$, especially in MLE.

Ex] DC Level Signal

Let $s[n;A] = A$. Assume we observe $x[n]$, $n=0,\ldots,(N-1)$. Then, $J(A) = \sum_{n=0}^{N-1}(x[n]-A)^2$ and $\hat{A}_{LSE} = \frac{1}{N}\sum_{n=0}^{N-1} x[n] = \bar{x}$ by finding the solution to $\frac{\partial J(A)}{\partial A} = 0$. While at this point, we can only claim that $\hat{A}_{LSE}$ minimizes the sum of squared errors, we have seen previously that if $x[n] = A + w[n]$ with 0-mean WGN $w[n]$, $\hat{A}_{LSE}$ will coincide with/become the MVU estimator.

Suppose that $E[x[n]] \neq A$ in this example. Then $E[\hat{A}_{LSE}] = E[\bar{x}] \neq A$; so in the event that the noise is non-zero mean, we might consider redefining $s[n]$ to take into account this systematic error (bias), if there is confidence that the error should be zero mean and iid.

Assuming that $\epsilon[n]$ is 0-mean and iid, we see that as $N \to \infty$, $J(A) \to var(\epsilon[n])$. In practice, systematic errors are opportunities for model improvement.

Ex] Sinusoidal Frequency Estimation

Let $s[n] = \cos(2\pi f_0 n)$ and $J(f_0) = \sum_{n=0}^{N-1}(x[n] - \cos(2\pi f_0 n))^2$ for measurement $x[n]$, $n=0,\ldots,N-1$. Here, $J(f_0)$ is <u>not</u> a quadratic function of the unknown parameter f_0, so numerical optimization is needed. This is a <u>nonlinear least squares</u> problem

Ex] Sinusoidal Amplitude Estimation

Let $s[n] = A\cos(2\pi f_0 n)$ where f_0 is known and $x[n]$, $n=0,\ldots,N-1$ are measured.

Then $J(A) = \sum_{n=0}^{N-1}\left(x[n] - A\cos(2\pi f_0 n)\right)^2$ is a quadratic function of the unknown parameter A; so this is a <u>linear least squares (LS) problem.</u>

If both A and f_0 are unknown, then we have a nonlinear LS problem with

$$J(A, f_0) = \sum_{n=0}^{N-1}(x[n] - A\cos(2\pi f_0 n))^2$$

Here, since $J(A, f_0)$ is quadratic in A and nonquadratic in f_0, specialized alternating algorithms can be used and these types of problems with parameters in two groups (J quadratic wrt group 1 and nonquadratic wrt the other) are called <u>separable LS</u> problems.

<u>Linear Least Squares</u>

Assume (for scalar θ) that $s[n] = \theta h[n]$ where $h[n]$ is known. $J(\theta) = \sum_{n=0}^{N-1}(x[n] - \theta h[n])^2$. Solving for the root of $\frac{\partial J(\theta)}{\partial \theta} = 0$, we get $\hat{\theta}_{LSE} = \frac{\sum_{n=0}^{N-1} x[n]h[n]}{\sum_{n=0}^{N-1} h^2[n]}$.

$$J_{min} \triangleq J(\hat{\theta}_{LSE}) = \cdots = \sum_{n=0}^{N-1} x^2[n] - \frac{\left(\sum_{n=0}^{N-1} x[n]h[n]\right)^2}{\sum_{n=0}^{N-1} h^2[n]} = \sum_{n=0}^{N-1} x^2[n] - N\bar{x}^2.$$

after some work

Clearly, $0 \le J_{min} \le \sum_{n=0}^{N-1} x^2[n]$.

Extension to vector θ

Let θ be a $p \times 1$ parameter vector and $s = [s[0], s[1], \ldots, s[N-1]]^T$ where $s = H\theta$ with a known $N \times p$ $(N > p)$ rank p matrix H (the observation matrix).

$$J(\theta) = \sum_{n=0}^{N-1} (x[n] - s[n])^2 = (x - H\theta)^T (x - H\theta) = \|x - H\theta\|_2^2$$

Note that $J(\theta)$ is quadratic in θ: $J(\theta) = x^T x - 2x^T H\theta + \theta^T H^T H\theta$.

$$\frac{\partial J(\theta)}{\partial \theta^T} = -2H^T x + 2H^T H\theta = 0 \Big|_{\theta = \hat{\theta}_{LSE}}$$

$\Rightarrow \hat{\theta}_{LSE} = (H^T H)^{-1} H^T x$. The equations $H^T H\theta = H^T x$ are called the normal equations and appear in different names and forms (e.g. Wiener-Hopf equations in adaptive filters).

$$J_{min} \triangleq J(\hat{\theta}_{LSE}) = \cdots = x^T (x - H\hat{\theta}_{LSE}).$$

Weighted Linear LS

In some cases, we may want to weigh the errors in samples differently (due to different scales, noise level, etc.).

Then $J_W(\theta) = (x - H\theta)^T W (x - H\{\theta\}) = \|x - H\theta\|_W^2$ (Mahalanobis distance squared)

$\hat{\theta}_{WLSE} = (H^T W H)^{-1} H^T W x$ ($W > 0$ is required).

$J_{min} \triangleq J(\hat{\theta}_{WLSE}) = x^T (W - WH(H^T W H)^{-1} H^T W) x$.

If W is diagonal, $J_W(\theta)$ weighs each sample error by w_n. similar sample to the case with linear model under varying variance

Geometrical Interpretations

Consider $s = H\theta = [h_1 \; h_2 \; \cdots \; h_p] \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_p \end{bmatrix} = \sum_{i=1}^{p} \theta_i h_i$

as a linear combination of "signal" vectors $\{h_i\}$.

The LS ~~squared~~ error is $J(\theta) = (x - H\theta)^T(x - H\theta) = \|x - H\theta\|_2^2$

(will use $\|\cdot\|$ to mean $\|\cdot\|_2$) $= \|x - \sum_{i=1}^{p} h_i \theta_i\|_2^2$.

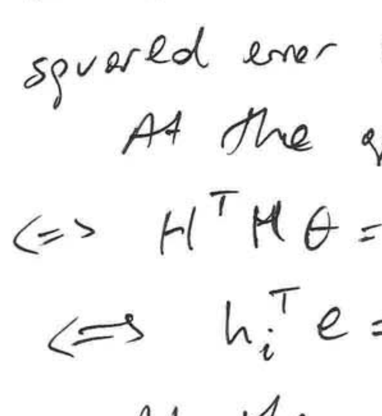
The linear LS problem tries to model the data x as a linear combination of $\{h_i\}_{i=1}^{p}$ such that the squared error is minimized.

At the optimal solution, $\frac{\partial J(\theta)}{\partial \theta^T} = -2H^T x + 2H^T H\theta = 0$

$\iff H^T H \theta = H^T x \iff H^T s = H^T x \iff H^T(s - x) = 0$

$\iff h_i^T e = 0$ (where $e = x - s$) $\forall i \in \{1, \ldots, p\}$.

$\therefore$ At the optimal linear LS solution, the error (between the data and its approximation) is orthogonal to the columns of H (or $e \perp h_i \; \forall i$).



In a two-dimensional subspace the situation could be visualized as illustrated here.

Recall that $\hat{\theta}_{LS} = (H^T H)^{-1} H^T x$. If $h_i \perp h_j \; \forall i \neq j$, then $H^T H$ is diagonal. If $h_i^T h_j = \delta_{ij}$, then $H^T H = I$. In this latter case ($H^T H = I$), we have $\hat{\theta}_{LS} = H^T x$.

Ex) Fourier Analysis

$s[n] = a\cos(2\pi f_0 n) + b\sin(2\pi f_0 n) \quad n = 0, 1, \ldots, (N-1)$

where f_0 is known and $\theta = \begin{bmatrix} a \\ b \end{bmatrix}$ is to be estimated.

$$\begin{bmatrix} s[0] \\ s[1] \\ \vdots \\ s[N-1] \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \cos(2\pi f_0) & \sin(2\pi f_0) \\ \vdots & \vdots \\ \underbrace{\cos(2\pi f_0(N-1))}_{h_1} & \underbrace{\sin(2\pi f_0(N-1))}_{h_2} \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \Rightarrow s = a h_1 + b h_2$$

Let $f_0 = k/N$ where k is an integer, $k \in \{1, 2, \ldots, \frac{N}{2} - 1\}$.

Then $h_1^T h_2 = \sum_{n=0}^{N-1} \cos(2\pi \frac{k}{N} n)\sin(2\pi \frac{k}{N} n) = 0.$ (orthogonal).

Also $h_1^T h_1 = h_2^T h_2 = \frac{N}{2}$ (not normal).

$\therefore H^T H = \frac{N}{2} I$, so $\hat{\theta} = \frac{2}{N} H^T x = \begin{bmatrix} \frac{2}{N} \sum_{n=0}^{N-1} x[n]\cos(2\pi \frac{k}{N} n) \\ \frac{2}{N} \sum_{n=0}^{N-1} x[n]\sin(2\pi \frac{k}{N} n) \end{bmatrix}$

If our model was $s[n] = a' \sqrt{\frac{2}{N}} \cos(2\pi \frac{k}{N} n) + b' \sqrt{\frac{2}{N}} \sin(2\pi \frac{k}{N} n)$

then $H'^T H' = I$ would make $\hat{\theta}' = H'^T x$.

$\hat{s}_{LS} = H\hat{\theta}_{LS} = H(H^T H)^{-1} H^T x$ is an orthogonal projection of x to the space spanned by the columns of H. Let $P = H(H^T H)^{-1} H^T$ be an orthogonal projection matrix. Then.

1) P is symmetric $(P^T = P)$
2) P is idempotent $(P^2 = P)$
3) P is singular (rank p) unless $n = p$.

The error $e_{LS} = x - \hat{s}_{LS} = (I-P)x = P^{\perp}x$. Clearly $P^{\perp}$ is also an orthogonal projection matrix.

$$J_{min} = x^T(I-P)x = x^T P^{\perp} x = x^T P^{\perp T} P^{\perp} x = \|P^{\perp}x\|^2 = \|e\|^2.$$

Order-Recursive LS

In many cases, we assume that the model is in a parametric family, possible with a nested structure in terms of model complexity. In that case, we refer to the number of components (~ parameters) as the model order. For instance, assuming that the data is a polynomial of time plus noise $x[n] = p_{P-1}[n] + w[n]$ results in a nested parametric family where increasing the polynomial order by one improves our ability to better fit to experimental data, since the lower order polynomial remains as a special case. In model (order) selection, over-fitting becomes a concern. For example, given a finite number of samples, we can always increase the model (e.g. polynomial) order to achieve zero error. This leads to poor <u>generalization</u>, which is the approximation performance on novel test data not used in model fitting.

Suppose that, in linear LS modelling, we would like to increase the model order by one by adding a new column to H and a new parameter to θ. Let H_{k+1} and θ_{k+1} represent the basis matrix and parameter vector with $(k+1)$ columns and elements, respectively (omitting $\cdot_{LS}$)

$$\hat{\theta}_{k+1} = (H_{k+1}^T H_{k+1})^{-1} H_{k+1}^T x \text{ is the new LS solution.}$$

Now, $H_{k+1}^T H_{k+1}^T = \begin{bmatrix} H_k^T \\ h_{k+1}^T \end{bmatrix} \begin{bmatrix} H_k & h_{k+1} \end{bmatrix} = \begin{bmatrix} H_k^T H_k & H_k^T h_{k+1} \\ h_{k+1}^T H_k & h_{k+1}^T h_{k+1} \end{bmatrix}$

Inverse of a partitioned (symmetric) matrix is given by:

$$(A \text{ is symmetric}) \quad \begin{bmatrix} A & b \\ b^T & c \end{bmatrix} = \begin{bmatrix} (A - \frac{bb^T}{c})^{-1} & -\frac{1}{c}(A - \frac{bb^T}{c})^{-1} b \\ -\frac{b^T}{c}(A - \frac{bb^T}{c})^{-1} & \dfrac{1}{c - b^T A^{-1} b} \end{bmatrix}$$

Also from the matrix inversion lemma (Woodbury identity)

$$(A - \frac{bb^T}{c})^{-1} = A^{-1} + \frac{A^{-1} b b^T A^{-1}}{c - b^T A^{-1} b}$$

Using these:

$$(H_{k+1}^T H_{k+1})^{-1} = \begin{bmatrix} D_k + \dfrac{D_k H_k^T h_{k+1} h_{k+1}^T H_k D_k}{h_{k+1}^T h_{k+1} - h_{k+1}^T H_k D_k H_k^T h_{k+1}} & -\dfrac{E_k H_k^T h_{k+1}}{h_{k+1}^T h_{k+1}} \\ -\dfrac{h_{k+1}^T H_k E_k}{h_{k+1}^T h_{k+1}} & \dfrac{1}{h_{k+1}^T h_{k+1} - h_{k+1}^T H_k D_k H_k^T h_{k+1}} \end{bmatrix}$$

$$\triangleq D_{k+1}$$

where $D_k = (H_k^T H_k)^{-1}$ and $E_k = D_k + \dfrac{D_k H_k^T h_{k+1} h_{k+1}^T H_k D_k}{h_{k+1}^T h_{k+1} - h_{k+1}^T H_k D_k H_k^T h_{k+1}}$.

$$\therefore\ D_{k+1} = \begin{bmatrix} E_k & -\dfrac{E_k H_k^T h_{k+1}}{h_{k+1}^T h_{k+1}} \\ -\dfrac{h_{k+1}^T H_k E_k}{h_{k+1}^T h_{k+1}} & \underbrace{\left(h_{k+1}^T h_{k+1} - h_{k+1}^T H_k D_k H_k^T h_{k+1}\right)^{-1}}_{= \dfrac{1}{h_{k+1}^T P_k^\perp h_{k+1}}} \end{bmatrix}$$

since $P_k^\perp = I - H_k D_k H_k^T$ as defined earlier.

Also
$$\frac{E_k H_k^T h_{k+1}}{h_{k+1}^T h_{k+1}} = \frac{D_k H_k^T h_{k+1}}{h_{k+1}^T h_{k+1}} + \frac{D_k H_k^T h_{k+1} h_{k+1}^T H_k D_k H_k^T h_{k+1}}{(h_{k+1}^T P_k^\perp h_{k+1})(h_{k+1}^T h_{k+1})}$$

$$= \frac{D_k H_k^T h_{k+1} \left[h_{k+1}^T P_k^\perp h_{k+1} + h_{k+1}^T (I - P_k^\perp) h_{k+1} \right]}{(h_{k+1}^T P_k^\perp h_{k+1})(h_{k+1}^T h_{k+1})}$$

$$= \frac{D_k H_k^T h_{k+1}}{h_{k+1}^T P_k^\perp h_{k+1}} \qquad \frac{h_{k+1}^T \left[P_k^\perp + I - P_k^\perp \right] h_{k+1}}{h_{k+1}^T h_{k+1}} = 1$$

$$\therefore\ D_{k+1} = \begin{bmatrix} D_k + \dfrac{D_k H_k^T h_{k+1} h_{k+1}^T H_k D_k}{h_{k+1}^T P_k^\perp h_{k+1}} & -\dfrac{D_k H_k^T h_{k+1}}{h_{k+1}^T P_k^\perp h_{k+1}} \\ -\dfrac{h_{k+1}^T H_k D_k}{h_{k+1}^T P_k^\perp h_{k+1}} & \dfrac{1}{h_{k+1}^T P_k^\perp h_{k+1}} \end{bmatrix}$$

and
$$\hat{\theta}_{k+1} = D_{k+1} H_{k+1}^T x = D_{k+1} \begin{bmatrix} H_k^T x \\ h_{k+1}^T x \end{bmatrix}$$

$$= \begin{bmatrix} D_k H_k^T x + \dfrac{D_k H_k^T h_{k+1} h_{k+1}^T H_k D_k H_k^T x}{h_{k+1}^T P_k^\perp h_{k+1}} - \dfrac{D_k H_k^T h_{k+1} h_{k+1}^T x}{h_{k+1}^T P_k^\perp h_{k+1}} \\ -\dfrac{h_{k+1}^T H_k D_k H_k^T x}{h_{k+1}^T P_k^\perp h_{k+1}} + \dfrac{h_{k+1}^T x}{h_{k+1}^T P_k^\perp h_{k+1}} \end{bmatrix}$$

$$= \begin{bmatrix} \hat{\theta}_k - \dfrac{D_k H_k^T h_{k+1} h_{k+1}^T (x - H_k D_k H_k^T x)}{h_{k+1}^T P_k^\perp h_{k+1}} \\ \dfrac{h_{k+1}^T (x - H_k D_k H_k^T x)}{h_{k+1}^T P_k^\perp h_{k+1}} \end{bmatrix}$$

$$= \begin{bmatrix} \hat{\theta}_k \\ 0 \end{bmatrix} + \begin{bmatrix} -[D_k H_k^T h_{k+1} h_{k+1}^T P_k^\perp x] / [h_{k+1}^T P_k^\perp h_{k+1}] \\ [h_{k+1}^T P_k^\perp x] / [h_{k+1}^T P_k^\perp h_{k+1}] \end{bmatrix}$$

The minimum LS after the update is

$$J_{min,k+1} = (x - H_{k+1}\hat{\theta}_{k+1})^T (x - H_{k+1}\hat{\theta}_{k+1})$$

$$= \cdots \text{ after some work}$$

$$= J_{min,k} - \frac{(h_{k+1}^T P_k^\perp x)^2}{(h_{k+1}^T P_k^\perp h_{k+1})}$$

The projection matrix after the update becomes

$$P_{k+1} = H_{k+1}(H_{k+1}^T H_{k+1})^{-1} H_{k+1} = H_{k+1} D_{k+1} H_{k+1}^T$$

$$= \cdots \text{ after some work}$$

$$= P_k + \frac{(I - P_k) h_{k+1} h_{k+1}^T (I - P_k)}{h_{k+1}^T P_k^\perp h_{k+1}}$$

(where $P_k^\perp = I - P_k$). Let $u_{k+1} = \dfrac{(I-P_k) h_{k+1}}{\|(I-P_k) h_{k+1}\|}$.

Then $P_{k+1} = P_k + u_{k+1} u_{k+1}^T$ (is a rank-one update).

$\Rightarrow \hat{s}_{k+1} = P_{k+1} x = P_k x + u_{k+1} u_{k+1}^T x = \hat{s}_k + u_{k+1}(u_{k+1}^T x)$.

Sequential LS

In online application of linear LS parameter estimation, as samples arrive one at a time, we would like to update the model parameters in a recursive fashion.

Let $\hat{\theta}[n] = \left(H^T[n] C^{-1}[n] H[n] \right)^{-1} H^T[n] C^{-1}[n] x[n]$ be the LS estimate using n samples (with noise covariance $C[n]$). Then

assume uncorr noise

$$\hat{\theta}[n] = \left(\begin{bmatrix} H^T[n-1] & h[n] \end{bmatrix} \begin{bmatrix} C[n-1] & 0 \\ 0^T & \sigma_n^2 \end{bmatrix} \begin{bmatrix} H[n-1] \\ h^T[n] \end{bmatrix} \right)^{-1} \cdot$$

$$\left(\begin{bmatrix} H^T[n-1] & h[n] \end{bmatrix} \begin{bmatrix} C[n-1] & 0 \\ 0^T & \sigma_n^2 \end{bmatrix} \begin{bmatrix} x[n-1] \\ x[n] \end{bmatrix} \right)$$

new datum.

$$= \left(H^T[n-1] C^{-1}[n-1] H[n-1] + \frac{1}{\sigma_n^2} h[n] h^T[n] \right)^{-1} \cdot$$

$$\left(H^T[n-1] C^{-1}[n-1] x[n-1] + \frac{1}{\sigma_n^2} h[n] x[n] \right)$$

Let $\Sigma[n-1] = \left(H^T[n-1] C^{-1}[n-1] H[n-1] \right)^{-1}$. This is $\mathrm{Cov}(\hat{\theta}[n-1])$.

Then $\hat{\theta}[n] = \left(\Sigma^{-1}[n-1] + \frac{1}{\sigma_n^2} h[n] h^T[n] \right)^{-1} \left(H^T[n-1] C^{-1}[n-1] x[n-1] + \frac{1}{\sigma_n^2} h[n] x[n] \right)$

Using Woodbury identity

$$\Sigma[n] = \left(\Sigma^{-1}[n-1] + \frac{1}{\sigma_n^2} h[n] h^T[n] \right)^{-1}$$

$$= \Sigma[n-1] - \frac{\Sigma[n-1] h[n] h^T[n] \Sigma[n-1]}{\sigma_n^2 + h^T[n] \Sigma[n-1] h[n]}$$

Covariance Update

$$= \left(I - K[n] h^T[n] \right) \Sigma[n-1]$$

where $K[n] \triangleq \dfrac{\Sigma[n-1]h[n]}{\sigma_n^2 + h^T[n]\Sigma[n-1]h[n]}$ (the Kalman gain).

Then $\hat{\theta}[n] = (I - K[n]h^T[n])\Sigma[n-1]\left(\Sigma^{-1}[n-1]\hat{\theta}[n-1] + \frac{1}{\sigma_n^2}h[n]x[n]\right)$

since $\hat{\theta}[n-1] = (H^T[n-1]C^{-1}[n-1]H[n-1])^{-1}H^T[n-1]C^{-1}[n-1]x[n-1]$

$= \Sigma[n-1]H^T[n-1]C^{-1}[n-1]x[n-1]$.

Expanding the expression for $\hat{\theta}[n]$ ---

$$\hat{\theta}[n] = \hat{\theta}[n-1] + \frac{1}{\sigma_n^2}\Sigma[n-1]h[n]x[n] - K[n]h^T[n]\hat{\theta}[n-1]$$
$$- \frac{1}{\sigma_n^2}K[n]h^T[n]\Sigma[n-1]h[n]x[n]$$

Note that

$$\frac{1}{\sigma_n^2}\Sigma[n-1]h[n] - \frac{1}{\sigma_n^2}K[n]h^T[n]\Sigma[n-1]h[n]$$
$$= \frac{1}{\sigma_n^2}\left(\sigma_n^2 + h^T[n]\Sigma[n-1]h[n]\right)K[n]$$
$$- \frac{1}{\sigma_n^2}K[n]h^T[n]\Sigma[n-1]h[n]$$

$$= K[n]$$

$\therefore \hat{\theta}[n] = \hat{\theta}[n-1] + K[n]x[n] - K[n]h^T[n]\hat{\theta}[n-1]$

$\boxed{\text{Estimator update}}$ $= \hat{\theta}[n-1] + K[n](x[n] - h^T[n]\hat{\theta}[n-1])$

(This is the recursive least squares update.)

With some work, we get

$$J_{min}[n] = (x[n] - H[n]\hat{\theta}[n])^T C^{-1}[n](x[n] - H[n]\hat{\theta}[n]) = \cdots$$
$$= J_{min}[n-1] + \frac{e^2[n]}{\sigma_n^2 + h^T[n]\Sigma[n-1]h[n]}$$

Constrained Linear LS

In some cases, the parameters of the linear model need to be constrained. Suppose that we knew the true parameters to satisfy $A\theta = b$. Then our optimization problem becomes

$$\min_{\theta} (x - H\theta)^T (x - H\theta) \quad \text{s.t.} \quad \underbrace{A\theta - b = 0}_{r \text{ constraints}}$$

The Lagrangian is

$$J_c(\theta, \lambda) = (x - H\theta)^T (x - H\theta) + \lambda^T (A\theta - b)$$

$$\frac{\partial J_c}{\partial \theta^T} = -2H^T x + 2H^T H\theta + A^T \lambda = 0 \Big|_{\theta = \hat{\theta}_c}$$

Then $\hat{\theta}_c = (H^T H)^{-1} H^T x - \frac{1}{2} (H^T H)^{-1} A^T \lambda$

$$= \hat{\theta} - (H^T H)^{-1} A^T \frac{\lambda}{2}$$

The unconstrained LSE

$A\hat{\theta}_c = A\hat{\theta} - A (H^T H)^{-1} A^T \frac{\lambda}{2} = b$ is required.

So $\frac{\lambda}{2} = \left[A (H^T H)^{-1} A^T\right]^{-1} (A\hat{\theta} - b)$

$$\Rightarrow \hat{\theta}_c = \hat{\theta} - (H^T H)^{-1} A^T \left[A (H^T H)^{-1} A^T\right]^{-1} (A\hat{\theta} - b)$$

where $\hat{\theta} = (H^T H)^{-1} H^T x$.

Notice that if $A\hat{\theta} = b$, then the second (correction) term becomes zero and $\hat{\theta}_c = \hat{\theta}$.

Nonlinear LS

Assume that the data/signal x can be modeled by $s(\theta)$. For instance, if $x = s(\theta) + w$ where $w \sim \mathcal{N}(0, \sigma^2 I)$ (WGN), then we have a correspondence between LSE and MLE. In linear LS, $s(\theta) = H\theta$. Linear LS results in a quadratic objective that is easy to optimize. For nonlinear LS, iterative or brute-force methods must be used (such as gradient or Newton descent & grid search).

$$J(\theta) = (x - s(\theta))^T (x - s(\theta))$$

and $\hat{\theta}_{LS} = \arg\min_\theta J(\theta)$. If we can find an invertible function g such that $\alpha = g(\theta)$ results in $s(\theta(\alpha)) = s(g^{-1}(\alpha)) = H\alpha$ then we can solve for $\hat{\alpha}_{LS}$ using linear LS and determine $\hat{\theta}_{LS} = g^{-1}(\hat{\alpha}_{LS})$ as the desired solution. This parameter transformation approach only works for invertible g.

Ex] $s[n] = A\cos(2\pi f_0 n + \phi)$ $n = 0, 1, \ldots, (N-1)$ and we need to estimate $\theta = \binom{A}{\phi}$ with f_0 known ($A > 0$). The LSE is

$$\hat{\theta}_{LS} = \arg\min_\theta J(\theta) = \arg\min_\theta \sum_{n=0}^{N-1} \left(x[n] - A\cos(2\pi f_0 n + \phi)\right)^2.$$

Note that $A\cos(2\pi f_0 n + \phi) = A\cos\phi\cos(2\pi f_0 n) - A\sin\phi\sin(2\pi f_0 n)$

Let $\alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} A\cos\phi \\ -A\sin\phi \end{bmatrix}$. Then $s[n] = \alpha_1 \cos(2\pi f_0 n) + \alpha_2 \sin(2\pi f_0 n)$.

So $s = H\alpha$ where $H = \begin{bmatrix} 1 & 1 \\ \cos(2\pi f_0 n) & \sin(2\pi f_0 n) \\ \vdots & \vdots \\ \cos(2\pi f_0 (N-1)) & \sin(2\pi f_0 (N-1)) \end{bmatrix}$.

$\hat{\alpha}_{LS} = (H^T H)^{-1} H^T x$ and $\hat{\theta}_{LS} = \begin{bmatrix} (\hat{\alpha}_1^2 + \hat{\alpha}_2^2)^{1/2} \\ \arctan\left(\frac{-\hat{\alpha}_2}{\hat{\alpha}_1}\right) \end{bmatrix}$

In some cases, the problem may be separable in parameters and the model could be linear in some parameters while nonlinear in the remaining ones. Consider

$s = H(\alpha)\beta$ where $\theta = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$ with $\alpha \sim (p-q)$ parameters, $\beta \sim q$ parameters.

Now, $J(\alpha, \beta) = (x - H(\alpha)\beta)^T (x - H(\alpha)\beta)$ and for a given α the optimal β estimate is $\hat{\beta}(\alpha) = (H^T(\alpha) H(\alpha))^{-1} H^T(\alpha) x$.

For this $(\alpha, \hat{\beta}(\alpha))$ pair, we have

$$J(\alpha, \hat{\beta}(\alpha)) = x^T \left[I - H(\alpha)(H^T(\alpha) H(\alpha))^{-1} H^T(\alpha) \right] x$$

which needs to be minimized w.r.t. α. Equivalently

$$\hat{\alpha}_{LS} = \arg\max_{\alpha} \; x^T P(\alpha) x \quad \text{where } P(\alpha) = H(\alpha)[H^T(\alpha) H(\alpha)]^{-1} H^T(\alpha).$$

This reduces the parameter dimensionality in which we have to solve a nonlinear LS problem.

Ex] $s[n] = A_1 r^n + A_2 r^{2n} + A_3 r^{3n}$ and $\theta = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \\ r \end{bmatrix}$, $0<r<1$.

The model is linear in $\beta = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix}$ and nonlinear in $\alpha = r$.

We need to solve $\hat{r} = \underset{0<r<1}{\arg\max}\; x^T H(r) [H^T(r) H(r)]^{-1} H^T(r) x$

where $H(r) = \begin{bmatrix} 1 & 1 & 1 \\ r & r^2 & r^3 \\ \vdots & \vdots & \vdots \\ r^{(N-1)} & r^{2(N-1)} & r^{3(N-1)} \end{bmatrix}$. Once $\hat{r}$ is found, for instance, using grid search in $r \in (0,1)$, we get

$$\hat{\beta} = (H^T(\hat{r}) H(\hat{r}))^{-1} H^T(\hat{r}) x.$$

For nonlinear LS with $J(\theta) = (x - s(\theta))^T (x - s(\theta))$, at the optimal solution, we need $\left.\frac{\partial J(\theta)}{\partial \theta}\right|_{\theta = \hat{\theta}_{LS}} = 0$. More explicitly,

$$\frac{\partial J(\theta)}{\partial \theta_j} = -2 \sum_{i=0}^{N-1} (x[i] - s[i]) \frac{\partial s[i]}{\partial \theta_j} = 0 \quad \forall j \in \{1, \dots, p\}.$$

Let $\left[\frac{\partial s(\theta)}{\partial \theta}\right]_{ij} = \frac{\partial s[i]}{\partial \theta_j}$ $\quad i = 0, \dots, (N-1)$, $j = 1, \dots, p$ (the Jacobian matrix).

Then in matrix form: $\frac{\partial s(\theta)}{\partial \theta}^T (x - s(\theta)) = 0$. If the model is linear ($s = H\theta$), then $\frac{\partial s}{\partial \theta} = H$ as we saw before.

The roots of the nonlinear system of equations could be iteratively searched by starting from an initial estimate and using Newton-Raphson or Secant methods.

N-R iteration: $\hat{\theta}_{k+1} = \hat{\theta}_k - \left[\frac{\partial g(\theta)}{\partial \theta}\right]^{-1} g(\theta) \Big|_{\theta = \hat{\theta}_k}$

where $g(\theta) \triangleq \frac{\partial s(\theta)^T}{\partial \theta}(x - s(\theta))$. The Jacobian of g is

$$\frac{\partial [g(\theta)]_i}{\partial \theta_j} = \frac{\partial}{\partial \theta_j}\left[\sum_{n=0}^{N-1}(x[n]-s[n])\frac{\partial s[n]}{\partial \theta_i}\right]$$

$$= \sum_{n=0}^{N-1}\left[(x[n]-s[n])\frac{\partial^2 s[n]}{\partial\theta_i \partial\theta_j} - \frac{\partial s[n]}{\partial\theta_j}\frac{\partial s[n]}{\partial\theta_i}\right]$$

Letting $[H(\theta)]_{ij} = \left[\frac{\partial s(\theta)}{\partial \theta}\right]_{ij} = \frac{\partial s[i]}{\partial \theta_j}$ and $[G_n(\theta)]_{ij} = \frac{\partial^2 s[n]}{\partial\theta_i \partial\theta_j}$

we can obtain $\frac{\partial g(\theta)}{\partial \theta} = \sum_{n=0}^{N-1} G_n(\theta)(x[n]-s[n]) - H^T(\theta)H(\theta)$

as a compact expression. Then the N-R iteration is

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \left[H^T(\hat{\theta}_k)H(\hat{\theta}_k) - \sum_{n=0}^{N-1} G_n(\hat{\theta}_k)(x[n]-[s(\hat{\theta}_k)]_n)\right]^{-1} \cdot H^T(\hat{\theta}_k)(x - s(\hat{\theta}_k))$$

In particular for the linear model, this reduces to $\hat{\theta}_{k+1} = \hat{\theta}_k + (H^TH)^{-1}H^T(x - H\theta_k) = (H^TH)^{-1}H^Tx$, since $s(\theta) = H(\theta)$, $G_n(\theta) = 0$, and $H(\theta) = H$. In this case, the iteration converges to the linear LS solution in one step from any initial condition.

Ex] AR Parameter Estimation for the ARMA Model

For a WSS process $x[n]$ generated by white noise $u[n]$ passing through an ARMA model, the PSD is given by $P_{xx}(f) = \sigma_u^2 \frac{|B(f)|^2}{|A(f)|^2}$ where σ_u^2 is the power of $u[n]$, and the ARMA system has

$$B(f) = 1 + \sum_{k=1}^{q} b[k]e^{-j2\pi fk}, \quad A(f) = 1 + \sum_{k=1}^{p} a[k]e^{-j2\pi fk}$$

in its transfer function.

In z-transform domain, $P_{xx}(z) = \sigma_u^2 \frac{\mathcal{B}(z)\mathcal{B}(z^{-1})}{\mathcal{A}(z)\mathcal{A}(z^{-1})}$
where $B(f) = \mathcal{B}(e^{j2\pi f})$, $A(f) = \mathcal{A}(e^{j2\pi f})$.

In this setting, it can be shown that the autocorrelation sequence of $x[n]$ satisfies

$$\sum_{k=0}^{p} a[k] r_{xx}[n-k] = 0 \quad \text{for } n > q$$

where $a[0] = 1$. This is true because the MA part of the model, $\mathcal{B}(z)$ only influences output autocorrelation upto lag $\mp q$ in either direction, away from lag zero. The autocorr. lags beyond $\mp q$ are only dependent on $\mathcal{A}(z)$. These equations are called the modified Yule-Walker equations. The autocorr. estimates

$$\hat{r}_{xx}[k] = \frac{1}{N} \sum_{n=0}^{N-1-|k|} x[n] x[n+|k|]$$

could be used and due to statistical estimation errors we will have $\sum_{k=0}^{p} a[k] \hat{r}_{xx}[n-k] = \epsilon[n]$, $n > q$.

This yields $\hat{r}_{xx}[n] = -\sum_{k=1}^{p} a[k] \hat{r}_{xx}[n-k] + \epsilon[n]$ $n > q$.

Letting $x = \begin{bmatrix} \hat{r}_{xx}[q+1] \\ \vdots \\ \hat{r}_{xx}[M] \end{bmatrix}$ where $M \le N-1$, $\theta = \begin{bmatrix} a[1] \\ \vdots \\ a[p] \end{bmatrix}$, and

$H = -\begin{bmatrix} \hat{r}_{xx}[q] & \hat{r}_{xx}[q-1] & \cdots & \hat{r}_{xx}[q-p+1] \\ \hat{r}_{xx}[q+1] & \hat{r}_{xx}[q] & \cdots & \hat{r}_{xx}[q-p+2] \\ \vdots & \vdots & & \vdots \\ \hat{r}_{xx}[M-1] & \hat{r}_{xx}[M-2] & \cdots & \hat{r}_{xx}[M-p] \end{bmatrix}$, we can define $J(\theta) = (x - H\theta)^T (x - H\theta)$ as the LS objective.

Ex| Phase Lock Loop (PLL)

This is used for carrier recovery in a communication system.

The noise-free carrier is $s[n] = \cos(2\pi f_0 n + \phi)$, $n = -M, \ldots, 0, \ldots, M$ where f_0 and ϕ need to be estimated. Let $\theta = \begin{bmatrix} f_0 \\ \phi \end{bmatrix}$, then

$$\frac{\partial s[n]}{\partial f_0} = -n 2\pi \sin(2\pi f_0 n + \phi) \quad \& \quad \frac{\partial s[n]}{\partial \phi} = -\sin(2\pi f_0 n + \phi)$$

which yields
$$H(\theta) = -\begin{bmatrix} -M 2\pi \sin(-2\pi f_0 M + \phi) & \sin(-2\pi f_0 M + \phi) \\ -(M-1) 2\pi \sin(-2\pi f_0 (M-1) + \phi) & \sin(-2\pi f_0 (M-1) + \phi) \\ \vdots & \vdots \\ M 2\pi \sin(2\pi f_0 M + \phi) & \sin(2\pi f_0 M + \phi) \end{bmatrix}$$

and
$$H^T(\theta) H(\theta) = \begin{bmatrix} 4\pi^2 \sum_{n=-M}^{M} n^2 \sin^2(2\pi f_0 n + \phi) & \text{same} \\ 2\pi \sum_{n=-M}^{M} n \sin^2(2\pi f_0 n + \phi) & \sum_{n=-M}^{M} \sin^2(2\pi f_0 n + \phi) \end{bmatrix}$$

We have the following identities:

$$\sum_{n=-M}^{M} n^2 \sin^2(2\pi f_0 n + \phi) = \sum_{n=-M}^{M} \left[\frac{n^2}{2} - \frac{n^2}{2} \cos(4\pi f_0 n + 2\phi) \right]$$

$$\sum_{n=-M}^{M} n \sin^2(2\pi f_0 n + \phi) = \sum_{n=-M}^{M} \left[\frac{n}{2} - \frac{n}{2} \cos(4\pi f_0 n + 2\phi) \right]$$

$$\sum_{n=-M}^{M} \sin^2(2\pi f_0 n + \phi) = \sum_{n=-M}^{M} \left[\frac{1}{2} - \frac{1}{2} \cos(4\pi f_0 n + 2\phi) \right]$$

as well as the approximation

$$\frac{1}{(2M+1)^{i+1}} \sum_{n=-M}^{M} n^i \cos(4\pi f_0 n + 2\phi) \approx 0 \quad \text{for } i = 0, 1, 2.$$

Then
$$H^T(\theta) H(\theta) \approx \begin{bmatrix} 8\pi^2 M^3/3 & 0 \\ 0 & M \end{bmatrix} \quad \text{for } M \gg 1.$$

From the N-R iteration formula:

$$f_{0_{k+1}} = f_{0_k} - \frac{3}{4\pi M^3} \sum_{n=-M}^{M} n \sin(2\pi f_{0_k} n + \phi_k)\left(x[n] - \cos(2\pi f_{0_k} n + \phi_k)\right)$$

$$\phi_{k+1} = \phi_k - \frac{1}{M} \sum_{n=-M}^{M} \sin(2\pi f_{0_k} n + \phi_k)\left(x[n] - \cos(2\pi f_{0_k} n + \phi_k)\right)$$

Since $\frac{1}{(2M+1)} \sum_{n=-M}^{M} \sin(2\pi f_{0_k} n + \phi_k) \cos(2\pi f_{0_k} n + \phi_k) \approx 0$,

$$f_{0_{k+1}} \approx f_{0_k} - \frac{3}{4\pi M^3} \sum_{n=-M}^{M} n\, x[n] \sin(2\pi f_{0_k} n + \phi_k)$$

$$\phi_{k+1} \approx \phi_k - \frac{1}{M} \sum_{n=-M}^{M} x[n] \sin(2\pi f_{0_k} n + \phi_k)$$

For sufficiently high SNR, $\hat{f}_0 \approx f_0$ and $\hat{\phi} \approx \phi$.

Suggested Problems: 6, 7, 9, 11, 12, 15, 17, 20, 26, 27
Implement in Matlab: 15, 20 (consider 23)