ECE 6123
Advanced Signal Processing
Linear Prediction

Peter Willett

Fall 2017

1 AR Models and Wiener Prediction

1.1 Relationship

Suppose we wish to "predict" $u[n]$ linearly, based on $\{u[n-1],\ u[n-2],\ \ldots,\ u[n-M]\}$ via

$$\hat{u}[n] \ = \ \mathbf{w}^H \mathbf{u}_{n-1} \tag{1}$$

Then the Wiener filter is straightforward:

$$\mathbf{R}\mathbf{w} \ = \ \mathbf{r} \tag{2}$$

where

$$\mathbf{r} \ \equiv \ \mathcal{E}\left\{ \begin{pmatrix} u[n-1] \\ u[n-2] \\ \vdots \\ u[n-M] \end{pmatrix} u[n]^* \right\} \ = \ \begin{pmatrix} r[-1] \\ r[-2] \\ \vdots \\ r[-M] \end{pmatrix} \tag{3}$$

Why should we want to make such a prediction when all we need is to wait a sample to see the real value $u[n]$? The answer is that if $u[n]$ were what we wanted we should do just that: wait for it. The reason to pose this as a prediction problem is that the structure is important and useful.

Equation (2) should look familiar. In fact, to repeat from the first lecture, we have the autoregressive (AR) model:

$$u[n] \ = \ \nu[n] - \sum_{k=1}^{M-1} a_k^* u[n-k] \tag{4}$$

$$u[n] \ = \ x[n] - \mathbf{a}^H \mathbf{u}_{n-1} \tag{5}$$

where the input $\nu[n]$ is assumed to be white (and usually but not necessarily Gaussian) with power σ_ν^2. In the introduction we found that we could recover the AR model (i.e., the a_k's) from knowledge of the autocorrelation lags (the $r[m]$'s) via

$$\mathbf{R}\mathbf{a} \ = \ -\mathbf{r} \tag{6}$$

which are the "Yule-Walker" equations. That is: $\mathbf{w} = -\mathbf{a}$. And that does make a good amount of sense: according to (4), using $\hat{u}[n] = \mathbf{w}^H \mathbf{u}_{n-1}$ eliminates all the "randomness" in $u[n]$ except that from $\nu[n]$; and $\nu[n]$ can't be removed.

1.2 Augmented Yule-Walker Equations

Some helpful insight provided by the Wiener approach to AR modeling is from the generic Wiener equation

$$J_{min} \;=\; \sigma_d^2 \;-\; \mathbf{w}_o^H \mathbf{p} \tag{7}$$

In the AR case (2) we write this as

$$\sigma_\nu^2 \;=\; r[0] \;-\; \mathbf{w}^H \mathbf{r} \tag{8}$$

We can concatenate (8) and (2) to get

$$\begin{pmatrix} r[0] & \mathbf{r}_M^H \\ \mathbf{r}_M & \mathbf{R}_M \end{pmatrix} \begin{pmatrix} 1 \\ -\mathbf{w} \end{pmatrix} \;=\; \begin{pmatrix} P_M \\ 0 \end{pmatrix} \tag{9}$$

where we have stressed the dimension of the matrix and vector in the subscripts. It is probably useful to note the full matrix in long form

$$\begin{pmatrix} r[0] & r[1] & r[2] & \dots & r[M] \\ r[-1] & r[0] & r[1] & \dots & r[M-1] \\ \vdots & \vdots & \ddots & \vdots & \\ r[-M] & r[-(M-1)] & r[-(M-2)] & \dots & r[0] \end{pmatrix} \begin{pmatrix} 1 \\ -\mathbf{w} \end{pmatrix} \;=\; \begin{pmatrix} P_M \\ 0 \end{pmatrix} \tag{10}$$

or

$$\mathbf{R}_{M+1} \begin{pmatrix} 1 \\ -\mathbf{w} \end{pmatrix} \;=\; \begin{pmatrix} P_M \\ 0 \end{pmatrix} \tag{11}$$

to see the naturalness of this concatenation.

Let us re-define[1] the AR vector $\mathbf{a}$ as

$$\mathbf{a} \;\equiv\; \begin{pmatrix} 1 \\ -\mathbf{w} \end{pmatrix} \;=\; \begin{pmatrix} 1 \\ \mathbf{a}_{old} \end{pmatrix} \;=\; \begin{pmatrix} 1 \\ a_1 \\ a_2 \\ \vdots \\ a_M \end{pmatrix} \tag{12}$$

[1] Sorry, but this has to be done.

where $\mathbf{a}_{old}$ is as in (2) and (4). Then

$$\mathbf{R}_{M+1}\mathbf{a}_M \;=\; \begin{pmatrix} P_M \\ 0 \end{pmatrix} \tag{13}$$

is a way to re-write (11). Here the subscript on $\mathbf{a}$ denotes the order of the predictor – it is a vector of length $M+1$.

1.3 Backwards Prediction

Suppose we want to "predict" $u[n-M]$ based on $\{u[n-M+1],\, , u[n-m+2], \ldots,\, u[n-1,\, u[n]\}$ (i.e., $\mathbf{u}_n$). The Wiener solution $\hat{u}[n-M] = \mathbf{g}^H\mathbf{u}_n$ is pretty simple:

$$\mathbf{Rg} \;=\; \mathbf{p} \;=\; \mathcal{E}\{u[n-M]^*\mathbf{u}_n\} \;=\; \begin{pmatrix} r[M] \\ r[M-1] \\ \vdots \\ r[1] \end{pmatrix} \;=\; \mathbf{r}^{B*} \tag{14}$$

and that does indeed mean "backwards and conjugated". We can concatenate this and the Wiener error equation as

$$\begin{pmatrix} \mathbf{R}_M & \mathbf{r}^{B*} \\ (\mathbf{r}^B)^T & r[0] \end{pmatrix} \begin{pmatrix} -\mathbf{g} \\ 1 \end{pmatrix} \;=\; \begin{pmatrix} 0 \\ P_M^{backwards} \end{pmatrix} \tag{15}$$

where $P_M^{backwards}$ is the Wiener error for the backward prediction. Now let's write the Wiener solution another way, using $\hat{u}[n-M] = (\mathbf{g}^B)^H\mathbf{u}_n^B$, which is identical in effect to the "normal" ordering. Now we have

$$\begin{aligned} \mathcal{E}\{\mathbf{u}^B(\mathbf{u}^B)^H\}\mathbf{g}^B &= \mathcal{E}\{\mathbf{u}^B u[n-M]^*\} && (16)\\ \mathbf{R}^*\mathbf{g}^B &= \mathbf{r}^* && (17)\\ \mathbf{R}\mathbf{g}^{B*} &= \mathbf{r} && (18) \end{aligned}$$

Compare (18) to (2) and it clear that we can write

$$\begin{aligned} \mathbf{g}^{B*} &= \mathbf{w} && (19)\\ \begin{pmatrix} \mathbf{g}^{B*} \\ 1 \end{pmatrix} &= \mathbf{a} && (20)\\ P_M^{backwards} &= P_M && (21) \end{aligned}$$

That is: the AR process "looks the same" whether viewed in forward time or reverse time. That's a cute point, but the main by-product of this analysis

is that we can write

$$\begin{pmatrix} \mathbf{R}_M & \mathbf{r}^{B*} \\ \mathbf{r} & r[0] \end{pmatrix} \begin{pmatrix} -\mathbf{w}^{B*} \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ P_M \end{pmatrix} \tag{22}$$

or

$$\mathbf{R}_{M+1}\mathbf{a}_M^{B*} = \begin{pmatrix} 0 \\ P_M \end{pmatrix} \tag{23}$$

as an alternate way to write the augmented Yule-Walker equations.

2 The Levinson-Durbin Algorithm

Solution of $M+1$ linear equations requires $\mathcal{O}((M+1)^3)$ operations. But the YW equations are special: they actually contain only $M+1$ unique "inputs" $\{r[0], r[1], \ldots, r[M]\}$ whereas the general complexity applies to $(M+1)^2 + (M+1)$. Can we exploit this structure? Note that the YW equations are rather unusual, in that there are M unknowns on the LHS and one (P_M) on the RHS.

Of course the answer is yes: the LD algorithm. We begin by proposing a structure.

$$\mathbf{a}_m = \begin{pmatrix} \mathbf{a}_{m-1} \\ 0 \end{pmatrix} + \Gamma_m \begin{pmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{pmatrix} \tag{24}$$

where we are noting the order as m rather than the *true* (or at least we assume it's true) model order M, since we will start with $m = 0$ and work up to $m = M$. This will be an inductive development, so we need to show that the structure replicates. The structure might be suggested by (13) and (23); but we need to show that it works.

We multiply (24) by $\mathbf{R}_{M+1}$; we want this product to be consistent with (13). We have

$$\begin{aligned} \mathbf{R}_{m+1}\mathbf{a}_m &= \begin{pmatrix} \mathbf{R}_m & \mathbf{r}_m^{B*} \\ (\mathbf{r}^B)^T & r[0] \end{pmatrix} \begin{pmatrix} \mathbf{a}_{m-1} \\ 0 \end{pmatrix} \\ &\quad + \Gamma_m \begin{pmatrix} r[0] & \mathbf{r}_m^H \\ \mathbf{r}_m & \mathbf{R}_m \end{pmatrix} \begin{pmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{pmatrix} && (25) \\ &= \begin{pmatrix} \mathbf{R}_m\mathbf{a}_{m-1} \\ (\mathbf{r}^B)^T\mathbf{a}_{m-1} \end{pmatrix} + \Gamma_m \begin{pmatrix} \mathbf{r}_m^T\mathbf{a}_{m-1}^{B*} \\ \mathbf{R}_m\mathbf{a}_{m-1}^{B*} \end{pmatrix} && (26) \\ &= \begin{pmatrix} P_{m-1} \\ 0 \\ \Delta_{m-1} \end{pmatrix} + \Gamma_m \begin{pmatrix} \Delta_{m-1}^* \\ 0 \\ P_{m-1} \end{pmatrix} && (27) \end{aligned}$$

where

$$\Delta_{m-1} \equiv (\mathbf{r}^B)^T \mathbf{a}_{m-1} \tag{28}$$

So, how do we make (27) into (13)? Easy: choose

$$\Gamma = \frac{-\Delta_{m-1}}{P_{m-1}} \tag{29}$$

Once we do that, we have

$$\mathbf{a}_m = \begin{pmatrix} \mathbf{a}_{m-1} \\ 0 \end{pmatrix} + \Gamma_m \begin{pmatrix} 0 \\ \mathbf{a}^{B*}_{m-1} \end{pmatrix} \tag{30}$$

as desired, and

$$P_m = P_{m-1}(1 - |\Gamma|^2) \tag{31}$$

The LD algorithm consists of starting with $P_0 = r[0]$ & $\mathbf{a}_0 = 1$, and iterating on m: (28), (29), (30) then (31). Notice that the missing RHS of (13) – that is, P_m – is created as is needed.

3 Other Neat Things

3.1 The Lattice Structure

Consider (4). We can turn this into a "prediction error filter" (actually a *forward* PEF) as

$$u[n] + \sum_{k=1}^{m} a_k^* u[n-k] = \nu[n] \tag{32}$$

$$f_m[n] = u[n] - \left(\sum_{k=1}^{m} w_k^* u[n-k] \right) \tag{33}$$

$$f_m[n] = \sum_{k=0}^{m} a_{m,k}^* u[n-k]) \tag{34}$$

$$F_m(z) = H_{f,m}(z) U(z) \tag{35}$$

where (32) is a restatement of the AR model, the second is (33) is the same in Wiener filtering notation, where $f_m[n]$ denotes the prediction error for the m^{th}-order predictor, (34) uses the new formulation for $\mathbf{a}_m$ and (35) is the z-transform of (33). This is basically presented to suggest what is meant by $f_m[n]$ and $H_{f,m}(z)$. We do the same thing for "backward" prediction errors

$b_m[n]$ and $H_{b,m}(z)$. We'll use

$$\begin{aligned} b_m[n] &= u[n-m] \; - \; \left(\sum_{k=1}^{m} g_k^* u[n+1-k] \right) && (36) \\ b_m[n] &= u[n-m] \; - \; \left(\sum_{k=1}^{m} w_k u[n-m+k] \right) && (37) \\ b_m[n] &= \sum_{k=0}^{m} a_{m,k} u[n-m+k]) && (38) \\ B_m(z) &= H_{b,m}(z) U(z) && (39) \end{aligned}$$

and it should be noted closely that $b_m[n]$ refers to the "error" in predicting $u[n-m]$.

We write

$$\begin{aligned} H_{f,m}(z) &= \sum_{k=0}^{m} a_{m,k}^* z^{-k} && (40) \\ &= \sum_{k=0}^{m-1} a_{m-1,k}^* z^{-k} \; + \; \Gamma_m^* \sum_{k=0}^{m-1} a_{m-1,m-k-1}^* z^{-k-1} && (41) \\ &= H_{f,m-1}(z) \; + \; \Gamma_m^* z^{-1} H_{b,m-1}(z) && (42) \end{aligned}$$

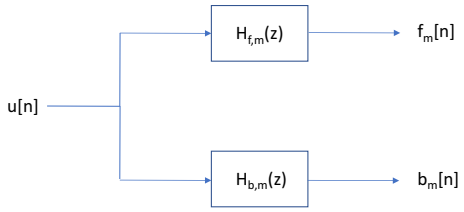
where (40) leads to (41) via (30). We also have

$$\begin{aligned} H_{b,m}(z) &= \sum_{k=0}^{m} a_{m,m-k}^* z^{-k} && (43) \\ &= \sum_{k=0}^{m} a_{m,k}^* z^{-(m-k)} && (44) \\ &= z^{-m} \left(H_{f,m}(1/z^*) \right)^* && (45) \end{aligned}$$

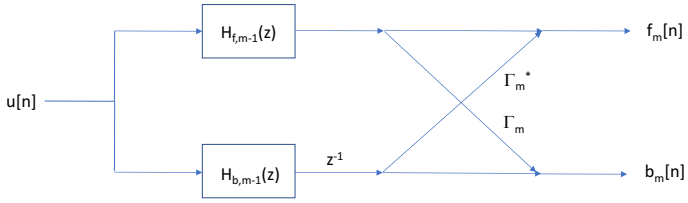
Substituting (42) into (45) we have

$$\begin{aligned} H_{b,m}(z) &= z^{-m} \left(H_{f,m-1}(1/z^*) \right)^* \; + \; \Gamma_m^* z z^{-m} \left(H_{b,m-1}(1/z^*) \right)^* && (46) \\ &= z^{-1} H_{b,m-1}(z) \; + \; \Gamma_m^* z z^{-m} \left(z^{m-1} H_{b,m-1}(1/z^*) \right) && (47) \\ &= \Gamma_m^* H_{f,m-1}(z) z^{-1} \; + \; z^{-1} H_{b,m-1}(z) && (48) \end{aligned}$$

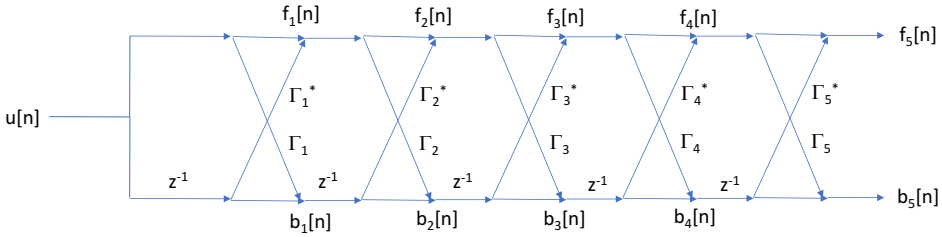
Then we have



being equivalent to



and overall we have the nice structure



exemplified for a fifth-order PEF.

3.2 Orthogonality

This is pretty simple once you remember that the *p.o.o.* governs all this optimal filtering. Let's assume that $i < j$ and remember that $b_i[n]$ is a linear function of $\{u[n-i],\ u[n-i+1],\ \ldots,\ u[n]\}$. By the *p.o.o.*, $b_j[n]$ is orthogonal to $\{u[n-j+1],\ u[n-j+2], \ldots,\ u[n]\}$, and hence it is orthogonal to its subset $\{u[n-i+1],\ u[n-i+2], \ldots,\ u[n]\}$. Case closed: $b_i[n]$ and $b_j[n]$ are orthogonal

$$\mathcal{E}\{b_i[n]b_j[n]^*\} \ = \ 0 \tag{49}$$

for all $i \neq j$ – and this expectation is by definition P_i for the case $i = j$.

Now let's write this out in full:

$$b_0[n] \ \ = \ \ u[n] \tag{50}$$

$$
\begin{aligned}
b_1[n] &= u[n-1] + a_{1,1}u[n] && (51)\\
b_2[n] &= u[n-2] + a_{2,1}u[n-1] + a_{2,2}u[n] && (52)\\
\vdots &= \vdots\\
b_m[n] &= u[n-m] + a_{m,1}u[n-m+1] + \ldots a_{m,m}u[n] && (53)
\end{aligned}
$$

This can be written as

$$\mathbf{b}_n \;=\; \mathbf{L}\mathbf{u}_n \tag{54}$$

where $\mathbf{L}$ is a lower triangular matrix containing in the i^{th} row the backwards PEF. Since orthogonality tells us that the b's are uncorrelated, we have

$$\mathbf{D} \;=\; \mathbf{L}\mathbf{R}\mathbf{L}^H \tag{55}$$

where $\mathbf{D}$ is a diagonal matrix with $(i,i)^{th}$ element P_i. We can also write (55) as

$$\mathbf{R} \;=\; \mathbf{L}^{-1}\mathbf{D}\mathbf{L}^{-H} \tag{56}$$

indicating that the PEF's and the correspond error powers are actually the LDU decomposition of the correlation matrix of the data.

3.3 Stability

It's obvious that the PEFs are stable – they're FIR. But one reason to create a PEF structure is to be able to recreate the corresponding AR model. Since that involves the reciprocal of the PEF, we need to know if the zeros of the PEF are inside the unit circle. If not the AR model is unstable and trying to use one would be hopeless.

We repeat (42)

$$H_{f,m}(z) \;=\; H_{f,m-1}(z) \;+\; \Gamma_m^* z^{-1} H_{b,m-1}(z) \tag{57}$$

and plug in (45)

$$H_{b,m-1}(z) \;=\; z^{-(m-1)}\left(H_{f,m-1}(1/z^*)\right)^* \tag{58}$$

to get

$$H_{f,m}(z) \;=\; H_{f,m-1}(z) \;+\; \Gamma_m^* z^{-m}\left(H_{f,m-1}(1/z^*)\right)^* \tag{59}$$

We convert this to the DTFT (discrete-time Fourier transform) as

$$H_{f,m}(e^{j\omega}) \;=\; H_{f,m-1}(e^{j\omega}) \;+\; \Gamma_m^* e^{-jm\omega} H_{f,m-1}(e^{j\omega})^* \tag{60}$$

Since (60) comprises the sum of two complex vectors, the first one of magnitude $|H_{f,m-1}(e^{j\omega})|$ and the second one, since $|\Gamma_m| < 1$, of magnitude less

than $|H_{f,m-1}(e^{j\omega})|$, we can see that as ω travels from zero to 2π the total phase change of $H_{f,m}(e^{j\omega})$ must be the same as of $H_{f,m-1}(e^{j\omega})$ – the second term in the sum can have no effect. As such $H_{f,m}(e^{j\omega})$ begins and ends its phase at the same point $\forall m$.

We turn now to a generic FIR model

$$\begin{aligned} H(z) &= \prod_{i=1}^{m}(1 - z_i z^{-1}) && (61) \\ &= z^{-m}\prod_{i=1}^{m}(z - z_i) && (62) \\ H(e^{j\omega}) &= e^{-jm\omega}\prod_{i=1}^{m}(e^{j\omega} - z_i) && (63) \end{aligned}$$

It is easy to see that a NASC for all zeros to be inside the unit circle is that the total phase change as ω travels from zero to 2π must be zero. That is what we have, hence the FPEF is indeed stable – the FPEF is *minimum-phase*. It can be shown that all zero are *outside* the unit circle for the BPEF (it is maximum-phase).