Lecture 2: Linear Prediction

Mark Hasegawa-Johnson

ECE 417: Multimedia Signal Processing, Fall 2021

Outline

All-Pole Filter

An all-pole filter has the system function:

$$H(z) = \frac{1}{(1 - p_1 z^{-1})(1 - p_1^* z^{-1})} = \frac{1}{1 - a_1 z^{-1} - a_2 z^{-2}},$$

so it can be implemented as

$$y[n] = x[n] + a_1 y[n-1] + a_2 y[n-2]$$

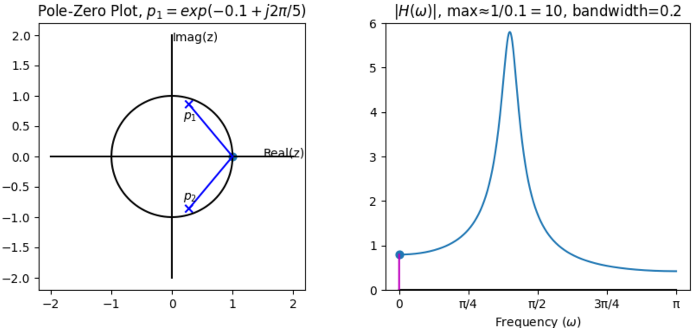
where

$$a_1 = (p_1 + p_1^*) = 2e^{-\sigma_1}\cos(\omega_1)$$
$$a_2 = -|p_1|^2 = -e^{-2\sigma_1}$$

Frequency Response of an All-Pole Filter

We get the magnitude response by just plugging in $z = e^{j\omega}$, and taking absolute value:

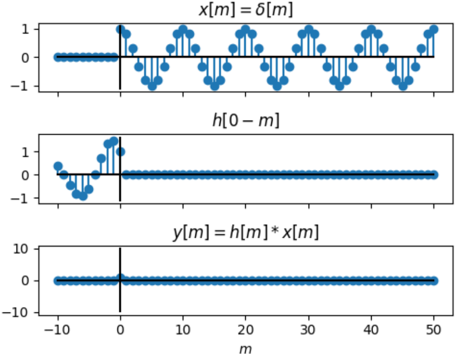
$$|H(\omega)| = |H(z)|_{z=e^{j\omega}} = \frac{1}{|e^{j\omega} - p_1| \times |e^{j\omega} - p_1^*|}$$



Impulse Response of an All-Pole Filter

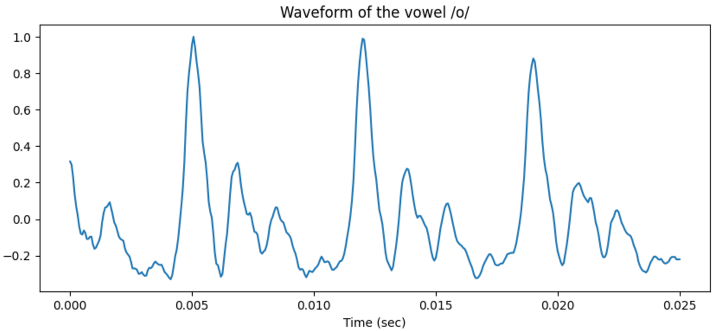
We get the impulse response using partial fraction expansion:

$$h[n] = \left(C_1 p_1^n + C_1^* (p_1^*)^n\right) u[n]$$
$$= \frac{1}{\sin(\omega_1)} e^{-\sigma_1 n} \sin\left(\omega_1 (n+1)\right) u[n]$$



Speech is made up of Damped Sinusoids

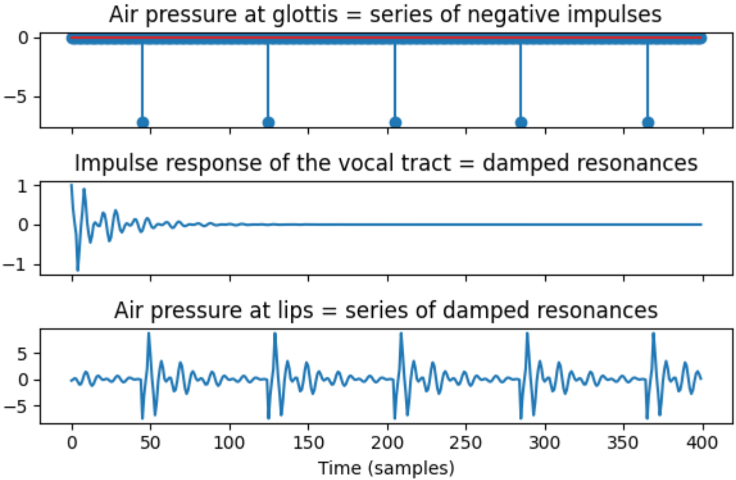
Resonant systems, like speech, trumpets, and bells, are made up from the series combination of second-order all-pole filters.



Outline

Speech

Speech is made when we take a series of impulses, one every 5-10ms, and filter them through a resonant cavity (like a bell).



Speech

Speech is made when we take a series of impulses, one every 5-10ms, and filter them through a resonant cavity (like a bell).

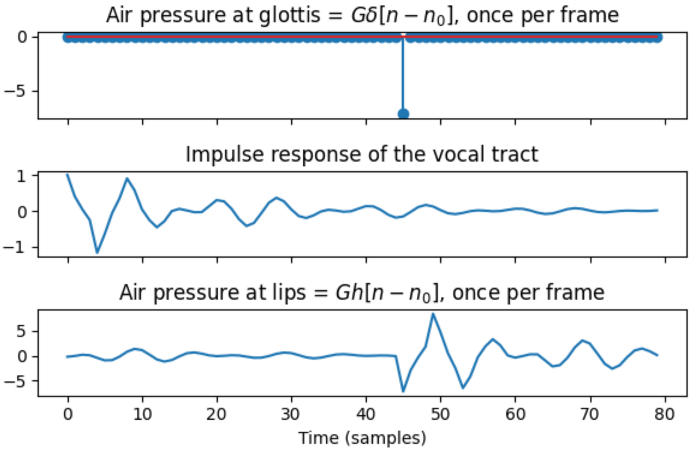
$$S(z) = H(z)E(z) = \frac{1}{A(z)}E(z)$$

where the excitation signal is a set of impulses, maybe only one per frame:

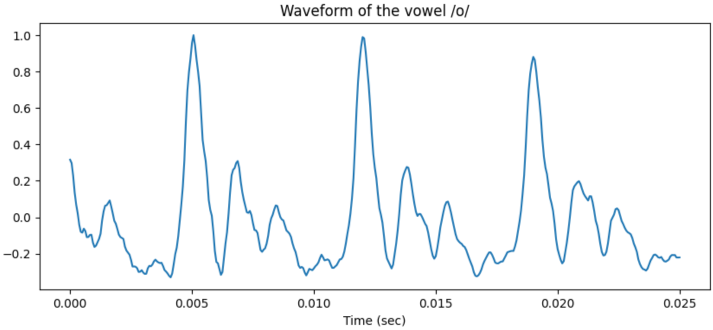
$$e[n] = G\delta[n - n_0]$$

The only thing we don't know, really, is the amplitude of the impulse (G), and the time at which it occurs (n_0). Can we find out?

Speech: The Model



Speech: The Real Thing



Inverse Filtering

If $S(z) = E(z)/A(z)$, then we can get $E(z)$ back again by doing something called an **inverse filter:**

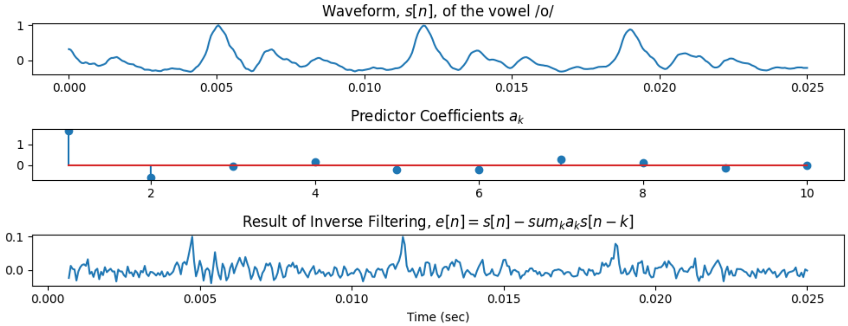
$$\textbf{IF: } S(z) = \frac{1}{A(z)}E(z) \quad \textbf{THEN: } E(z) = A(z)S(z)$$

The inverse filter, $A(z)$, has a form like this:

$$A(z) = 1 - \sum_{k=1}^{p} a_k z^{-k}$$

where p is twice the number of resonant frequencies. So if speech has 4-5 resonances, then $p \approx 10$.

Inverse Filtering



Inverse Filtering

This one is an all-pole (feedback-only) filter:

$$S(z) = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}} E(z)$$

That means this one is an all-zero (feedfoward only) filter:

$$E(z) = \left(1 - \sum_{k=1}^{p} a_k z^{-k}\right) S(z)$$

which we can implement just like this:

$$e[n] = s[n] - \sum_{k=1}^{p} a_k s[n-k]$$

Outline

Linear Predictive Analysis

This particular feedforward filter is called **linear predictive analysis**:

$$e[n] = s[n] - \sum_{k=1}^{p} a_k s[n-k]$$

It's kind of like we're trying to predict $s[n]$ using a linear combination of its own past samples:

$$\hat{s}[n] = \sum_{k=1}^{p} a_k s[n-k],$$

and then $e[n]$, the glottal excitation, is the part that can't be predicted:

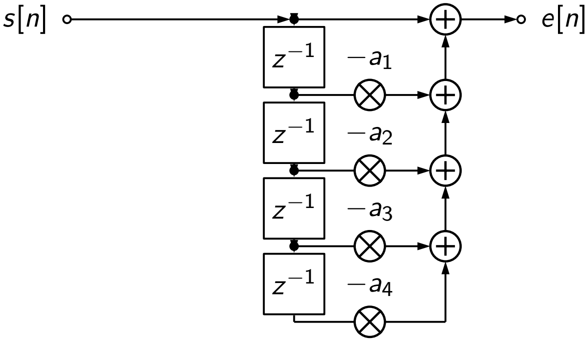
$$e[n] = s[n] - \hat{s}[n]$$

Linear Predictive Analysis

Actually, linear predictive analysis is used a lot more often in finance, these days, than in speech:

- In finance: detect important market movements = price changes that are not predictable from recent history.
- In health: detect EKG patterns that are not predictable from recent history.
- In geology: detect earthquakes = impulses that are not predictable from recent history.
- . . . you get the idea. . .

Linear Predictive Analysis Filter



Linear Predictive Synthesis

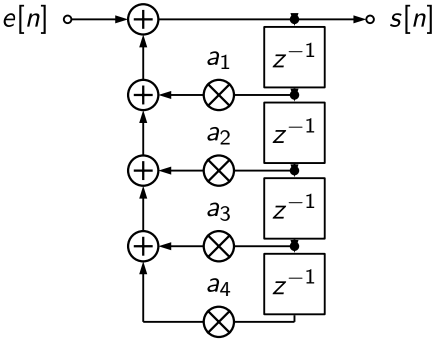
The corresponding feedback filter is called **linear predictive synthesis**. The idea is that, given $e[n]$, we can resynthesize $s[n]$ by adding feedback, because:

$$S(z) = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}} E(z)$$

means that

$$s[n] = e[n] + \sum_{k=1}^{p} a_k s[n-k]$$

Linear Predictive Synthesis Filter



Outline

Finding the Linear Predictive Coefficients

Things we don't know:

- The timing of the unpredictable event (n_0), and its amplitude (G).
- The coefficients a_k.

It seems that, in order to find n_0 and G, we first need to know the predictor coefficients, a_k. How can we find a_k?

Finding the Linear Predictive Coefficients

Let's make the following assumption:

- Everything that can be predicted is part of $\hat{s}[n]$. Only the unpredictable part is $e[n]$.

Finding the Linear Predictive Coefficients

Let's make the following assumption:

- Everything that can be predicted is part of $\hat{s}[n]$. Only the unpredictable part is $e[n]$.
- So we define $e[n]$ to be:

$$e[n] = s[n] - \sum_{k=1}^{p} a_k s[n-k]$$

- ...and then choose a_k to make $e[n]$ as small as possible.

$$a_k = \text{argmin} \sum_{n=-\infty}^{\infty} e^2[n]$$

Finding the Linear Predictive Coefficients

So we've formulated the problem like this: we want to find a_k in order to minimize:

$$\mathcal{E} = \sum_{n=-\infty}^{\infty} e^2[n] = \sum_{n=-\infty}^{\infty} \left(s[n] - \sum_{m=1}^{p} a_m s[n-m] \right)^2$$

Finding the Linear Predictive Coefficients

We want to find the coefficients a_k that minimize $\mathcal{E}$. We can do that by differentiating, and setting the derivative equal to zero:

$$\frac{d\mathcal{E}}{da_k} = 2\sum_{n=-\infty}^{\infty}\left(s[n]-\sum_{m=1}^{p}a_m s[n-m]\right)s[n-k],\quad \text{for all } 1\le k\le p$$

$$0 = \sum_{n=-\infty}^{\infty}\left(s[n]-\sum_{m=1}^{p}a_m s[n-m]\right)s[n-k],\quad \text{for all } 1\le k\le p$$

This is a set of p different equations (for $1 \le k \le p$) in p different unknowns (a_k). So it can be solved.

Autocorrelation

In order to write the solution more easily, let's define something called the "autocorrelation," $R[m]$:

$$R[m] = \sum_{n=-\infty}^{\infty} s[n]s[n-m]$$

In terms of the autocorrelation, the derivative of the error is

$$0 = R[k] - \sum_{m=1}^{p} a_m R[k-m] \quad \forall\ 1 \le k \le p$$

or we could write

$$R[k] = \sum_{m=1}^{p} a_m R[k-m] \quad \forall\ 1 \le k \le p$$

Matrices

Since we have p linear equations in p unknowns, let's write this as a matrix equation:

$$\begin{bmatrix} R[1] \\ R[2] \\ \vdots \\ R[p] \end{bmatrix} = \begin{bmatrix} R[0] & R[1] & \cdots & R[p-1] \\ R[1] & R[0] & \cdots & R[p-2] \\ \vdots & \vdots & \ddots & \vdots \\ R[p-1] & R[p-2] & \cdots & R[0] \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}$$

where I've taken advantage of the fact that $R[m] = R[-m]$:

$$R[m] = \sum_{n=-\infty}^{\infty} s[n]s[n-m]$$

Matrices

Since we have p linear equations in p unknowns, let's write this as a matrix equation:

$$\vec{\gamma} = R\vec{a}$$

where

$$\vec{\gamma} = \left[\begin{array}{c} R[1] \\ R[2] \\ \vdots \\ R[p] \end{array}\right], \quad R = \left[\begin{array}{cccc} R[0] & R[1] & \cdots & R[p-1] \\ R[1] & R[0] & \cdots & R[p-2] \\ \vdots & \vdots & \ddots & \vdots \\ R[p-1] & R[p-2] & \cdots & R[0] \end{array}\right].$$

Matrices

Since we have p linear equations in p unknowns, let's write this as a matrix equation:

$$\vec{\gamma} = R\vec{a}$$

and therefore the solution is

$$\vec{a} = R^{-1}\vec{\gamma}$$

Finding the Linear Predictive Coefficients

So here's the way we perform linear predictive analysis:

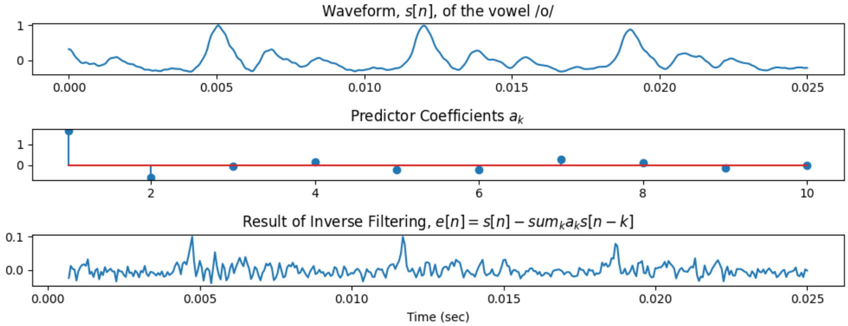
1. Create the matrix R and vector $\vec{\gamma}$:

$$\vec{\gamma} = \begin{bmatrix} R[1] \\ R[2] \\ \vdots \\ R[p] \end{bmatrix}, \quad R = \begin{bmatrix} R[0] & R[1] & \cdots & R[p-1] \\ R[1] & R[0] & \cdots & R[p-2] \\ \vdots & \vdots & \ddots & \vdots \\ R[p-1] & R[p-2] & \cdots & R[0] \end{bmatrix}$$

2. Invert R.

$$\vec{a} = R^{-1}\vec{\gamma}$$

Inverse Filtering



Outline

Inverse Filtering

If $S(z) = E(z)/A(z)$, then we can get $E(z)$ back again by doing something called an **inverse filter:**

$$\textbf{IF: } S(z) = \frac{1}{A(z)}E(z) \quad \textbf{THEN: } E(z) = A(z)S(z)$$

which we implement using a feedfoward difference equation, that computes a linear prediction of $s[n]$, then finds the difference between $s[n]$ and its linear prediction:

$$e[n] = s[n] - \sum_{k=1}^{p} a_k s[n-k]$$

Linear Predictive Analysis

Actually, linear predictive analysis is used a lot more often in finance, these days, than in speech:

- In finance: detect important market movements = price changes that are not predictable from recent history.
- In health: detect EKG patterns that are not predictable from recent history.
- In geology: detect earthquakes = impulses that are not predictable from recent history.
- . . . you get the idea. . .

Finding the Linear Predictive Coefficients

Let's make the following assumption:

- Everything that can be predicted is part of $\hat{s}[n]$. Only the unpredictable part is $e[n]$.
- So we define $e[n]$ to be:

$$e[n] = s[n] - \sum_{k=1}^{p} a_k s[n-k]$$

- ...and then choose a_k to make $e[n]$ as small as possible.

$$a_k = \operatorname{argmin} \sum_{n=-\infty}^{\infty} e^2[n]$$

which, when solved, gives us the simple equation $\vec{a} = R^{-1}\vec{\gamma}$.