Week 10 & 11

Table of Contents

Filters that minimize the output mean square error

Problem: using a linear estimator, we want to guess the value of random variable y from a set of observations of a related set of p random variables $x_1, x_2, \cdots, x_p$:

$$\hat{y} = \sum_{k=1}^{p} h_k x_k = \boldsymbol{h}^{\mathrm{T}} \boldsymbol{x}. \qquad (14.100)$$

This is basically a multivariate linear regression problem where $\boldsymbol{h}$ is the coefficients of the model. In the multivariate regression problem, we hypothesise that the response random variable Y can be computed as a linear combination of the cofficients of the model and the predictor variables $X_1, X_2, \cdots, X_p$

$$\widehat{y}_h(\boldsymbol{x}) = \boldsymbol{h}^T \boldsymbol{x}$$

The mean squared error loss function is:

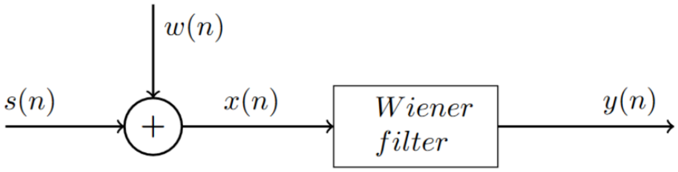
$$J_h = \frac{1}{2N} \sum_{i=1}^{N} \left(\widehat{y}_h(\boldsymbol{x}_i) - y_i \right)^2$$

We can minimise the loss function by taking the derivative and solving it.

Wiener filter

A special case of the multivariate linear regression model is the Wiener filter, which is used to estimate the original signal $s[n]$ given p observations of a noise distorted signal $x[n] = s[n] + w[n]$.

We assume that the noise process $w[n]$ is uncorrelated with the desired process that generated the original signal $s[n]$. We also assume that $x[n]$ and $w[n]$ are wide-sense stationary.



An p'order Wiener filter for estimating the original signal $s(n)$ is given by Eq. 14.112:

$$\hat{y}[n] = \sum_{k=1}^{p} h_k x[n+1-k] \qquad (14.112)$$

The normal equation for the Wiener filter is given by Eq. 14.113

$$\begin{bmatrix} r_x[0] & r_x[1] & \dots & r_x[p-1] \\ r_x[1] & r_x[0] & \dots & r_x[p-2] \\ \vdots & \vdots & \ddots & \vdots \\ r_x[p-1] & r_x[p-2] & \dots & r_x[0] \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_p \end{bmatrix} = \begin{bmatrix} r_{yx}[0] \\ r_{yx}[1] \\ \vdots \\ r_{yx}[p-1] \end{bmatrix}, \qquad (14.113)$$

where $h_k = h_o[k-1]$

The minimum square error for a pth Wiener filter is given by Eq. 14.115:

$$J_o = r_y[0] - \boldsymbol{h}_o^T \boldsymbol{g} = r_y[0] - \sum_{k=0}^{p-1} h_o[k] r_{yx}[k]. \qquad (14.115)$$

The optimum Wiener filter "passes" the input at bands with high SNR and "blocks" the input at bands with low SNR (see Eq. 14.120)

Linear prediction

Problem: we want predict the current signal value $\hat{x}[n]$ given a set of p previous samples of a wide-sense stationary process:

$$\hat{x}[n] = \sum_{k=1}^{p} h_k x[n-k] = \boldsymbol{h}^T \boldsymbol{x}[n-1], \qquad (14.122)$$

This is called one-step *forward linear predictor.*

The normal equations for the optimum linear predictor are:

$$\boldsymbol{R}\boldsymbol{h} = \boldsymbol{r}, \tag{14.125}$$

where $\boldsymbol{R}$, which is a symmetric Toeplitz matrix, and the vector $\boldsymbol{r}$ are defined by

$$\boldsymbol{R} \triangleq \begin{bmatrix} r[0] & r[1] & \dots & r[p-1] \\ r[1] & r[0] & \dots & r[p-2] \\ \vdots & \vdots & \ddots & \vdots \\ r[p-1] & r[p-2] & \dots & r[0] \end{bmatrix} \quad \text{and} \quad \boldsymbol{r} \triangleq \begin{bmatrix} r[1] \\ r[2] \\ \vdots \\ r[p] \end{bmatrix}. \tag{14.126}$$

The minimum squared error is given by Eq. 14.127:

$$J_0 = r[0] - \boldsymbol{h}^{\mathrm{T}}\boldsymbol{r} = r[0] - \boldsymbol{r}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{r}. \tag{14.127}$$

All-pole signal modelling

Suppose the input signal $x[n]$ is an AR(p) model i.e.:

$$x[n] = -\sum_{k=1}^{p} a_k x[n-k] + z[n] = -\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}[n-1] + z[n], \tag{14.128}$$

where $z[n] \sim \mathrm{WN}\left(0, \sigma_z^2\right)$

The value that we want to predict $x[n]$ can written as:

$$x[n] = \sum_{k=1}^{p} h_k x[n-k] + e[n] = \boldsymbol{h}^{\mathrm{T}}\boldsymbol{x}[n-1] + e[n]. \tag{14.131}$$

Setting $\boldsymbol{h} = -\boldsymbol{a}$, the prediction error can be expressed as:

$$e[n] = x[n] + \sum_{k=1}^{p} a_k x[n-k] = x[n] + \boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}[n-1], \tag{14.132}$$

This shows that the prediction error $e[n]$ is the output of a filter with the following system function:

$$A(z) = 1 + \sum_{k=1}^{p} a_k z^{-k}. \tag{14.133}$$

This system $A(z)$ is known as the ***prediction error filter*** or the ***analysis filter***.

If the input $x[n]$ is an AR(p) process, the output of $A(z)$ is a white noise process $z[n]$

The Levinson–Durbin algorithm