ECE 9023– Random Signals, Adaptive and Kalman Filtering

MATLAB Assignment1

Course Instructor: Dr. Vijay Parsa

Student Number: 251343797 .

Student Name: Yuanpeng Wang .

Table of Contents

1

(10 marks) Generate the following random signals in MATLAB. Plot the estimated probability density function (PDF) and power spectral density (PSD) for each. Comment on how close the estimated PDF and PSD are to their theoretical values. (see "WGN.mlx" and "WUN.mlx" for guidance).

a

White Gaussian Noise, mean = 0, variance = 0.75

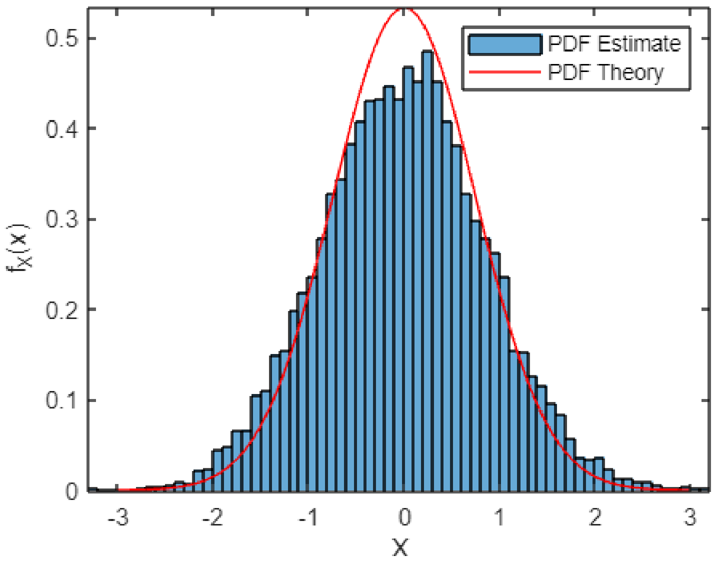


Fig. 1 The plot of theoretical and estimated PDF of the white Gaussian noise with zero mean, 0.75 variance

We can observe that PDF theory value of the white Gaussian noise matches PDF estimated of the white Gaussian noise. They are 0 mean and 0.75 variance in the time domain samples.

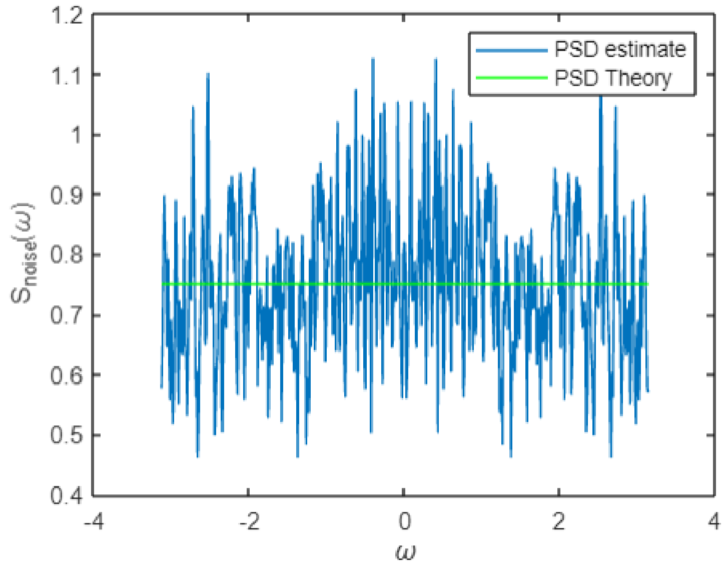


Fig. 2 The plot of theoretical and estimated PSD of the white Gaussian noise with zero mean, 0.75 variance

We can observe that the theory white Gaussian noise is a flat line, which located at the variance value of the white Gaussian noise. The estimated white Gaussian noise is just the single realization of the white Gaussian noise, therefore, it has deviated from the theoretical value of the white Gaussian noise.

b

White Uniform Noise, mean = 0, variance = 0.75

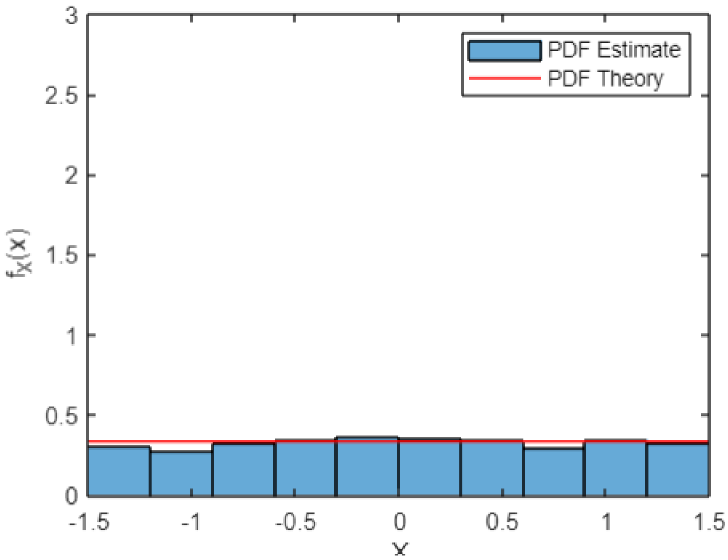


Fig. 3 The plot of theoretical and estimated PDF of the white uniform noise with zero mean, 0.75 variance

We can observe that the PDF theory value of the white uniform noise matches the PDF estimated value of the white uniform noise. They are 0 mean and 0.75 variance in the time domain.

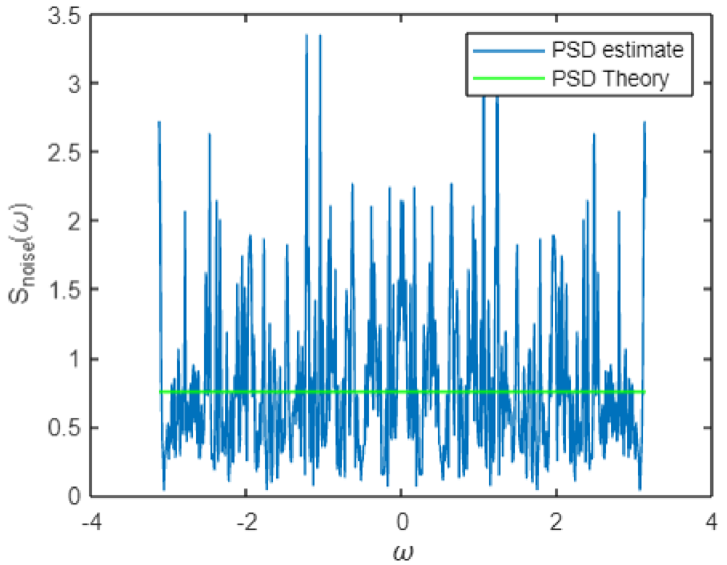


Fig. 4 The plot of theoretical and estimated PSD of the white uniform noise with zero mean, 0.75 variance

Similar to the PSD result of the white Gaussian noise. We can observe that the theoretical uniform noise is a flat line, which is located at the variance value of the white uniform noise. The estimated value of white uniform noise is the single realization of the white uniform noise.

c

Pink Gaussian Noise, mean = 0, variance = 1

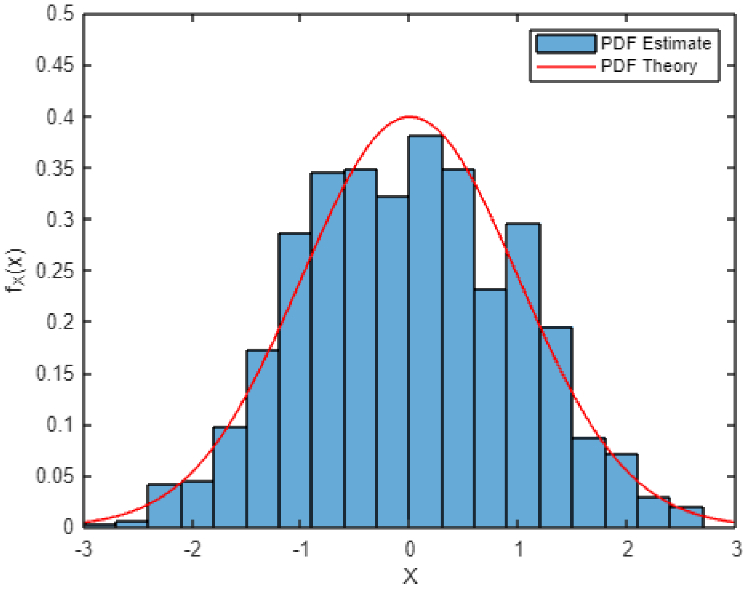


Fig. 5 The plot of theoretical and estimated PDF of the pink Gaussian noise with zero mean, standard variance

We can learn that PDF theory value of the pink Gaussian noise matches PDF estimated of the pink Gaussian noise.

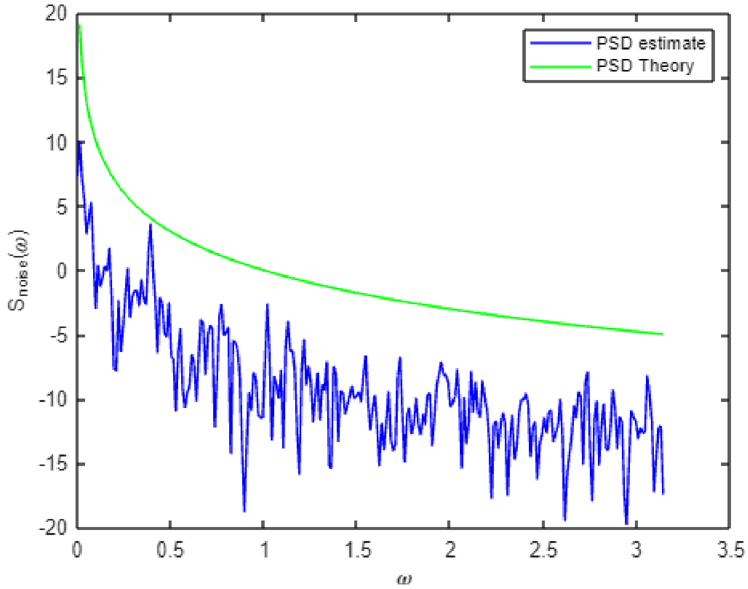


Fig. 5 The plot of theoretical and estimated PSD of the pink Gaussian noise with zero mean standard variance

We can learn that PSD of the pink Gaussian noise has a different distribution for its estimated value compared to the white Gaussian and white uniform noise. We can clearly see the theoretical PSD of the pink noise is a curve instead of a flat line. It decreases with increasing of the frequency.

2.

(10 marks) In this exercise, we will apply AR modeling to speech samples. Download "m01ae.wav", "w01ae.wav", "w01ih.wav", and "w01uw.wav" from "Course Content -> Week 4" under the "Asynchronous" section. Complete the following:

a

For each speech sample, plot the estimated variance of the white noise input against the model order, with the model order ranging from 1 to 25. See the documentation for "aryule" command for accessing the estimated variance. Comment on the results. What would be a good model order for modeling these waveforms?

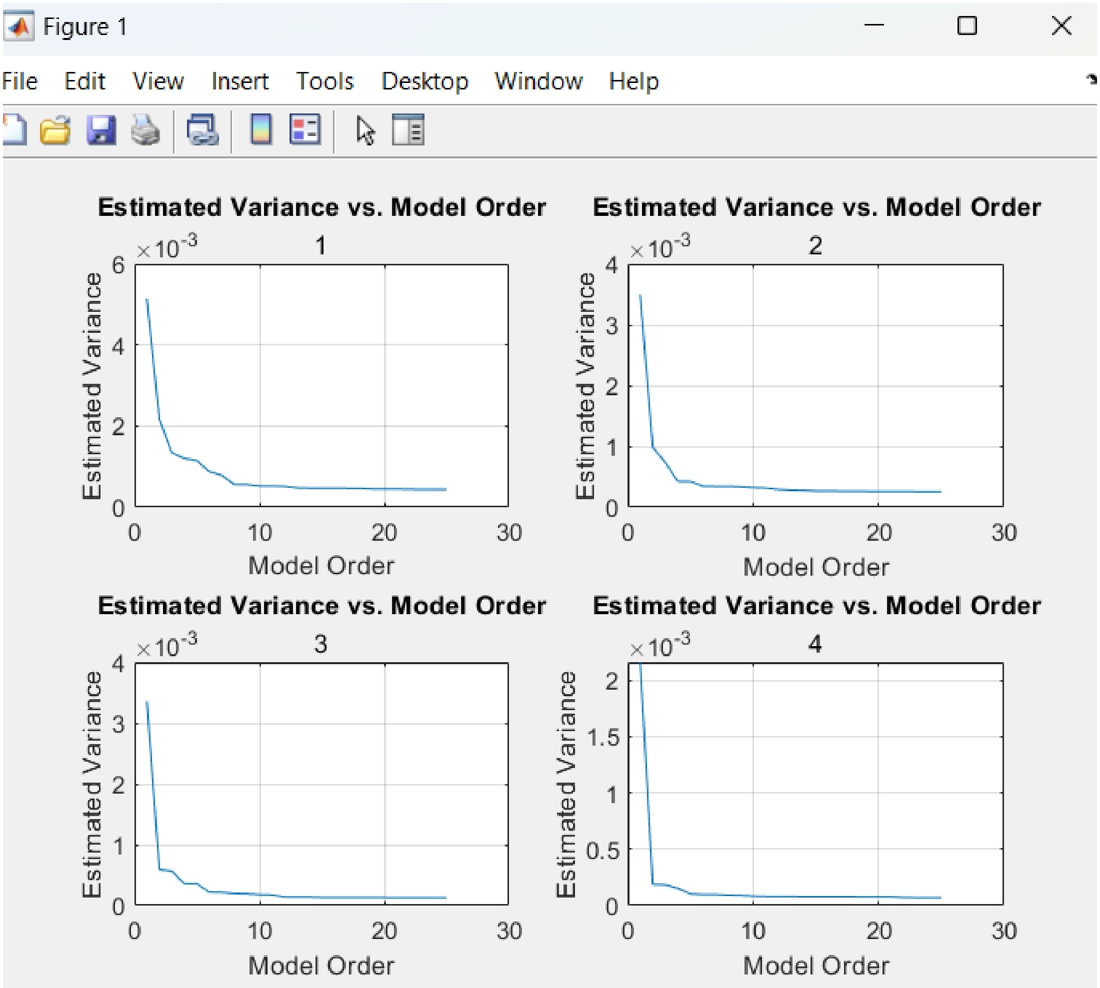


Fig. 6 The estimated variance of the white noise input against the model order

We can observe the variance of input white noise has a significant reduction by increasing the order of the AR model at a certain order. Then the variance of the input white noise remains almost unchanged by increasing the order of the AR model.

For the speech sample **"m01ae.wav",** we should choose the order at 13

For the speech sample **"w01ae.wav",** we should choose the order at 12

For the speech sample **"w01ih.wav",** we should choose the order at 12

For the speech sample **"w01uw.wav"** we should choose the order at 10

b

For each speech sample and the chosen model order, compute and plot the periodogram and AR spectral estimates. See "LinearPredictionExample.mlx" for guidance. Comment on the results. In particular, what is the AR spectral estimate trying to model? Are the AR spectral estimates the same across the four speech samples?

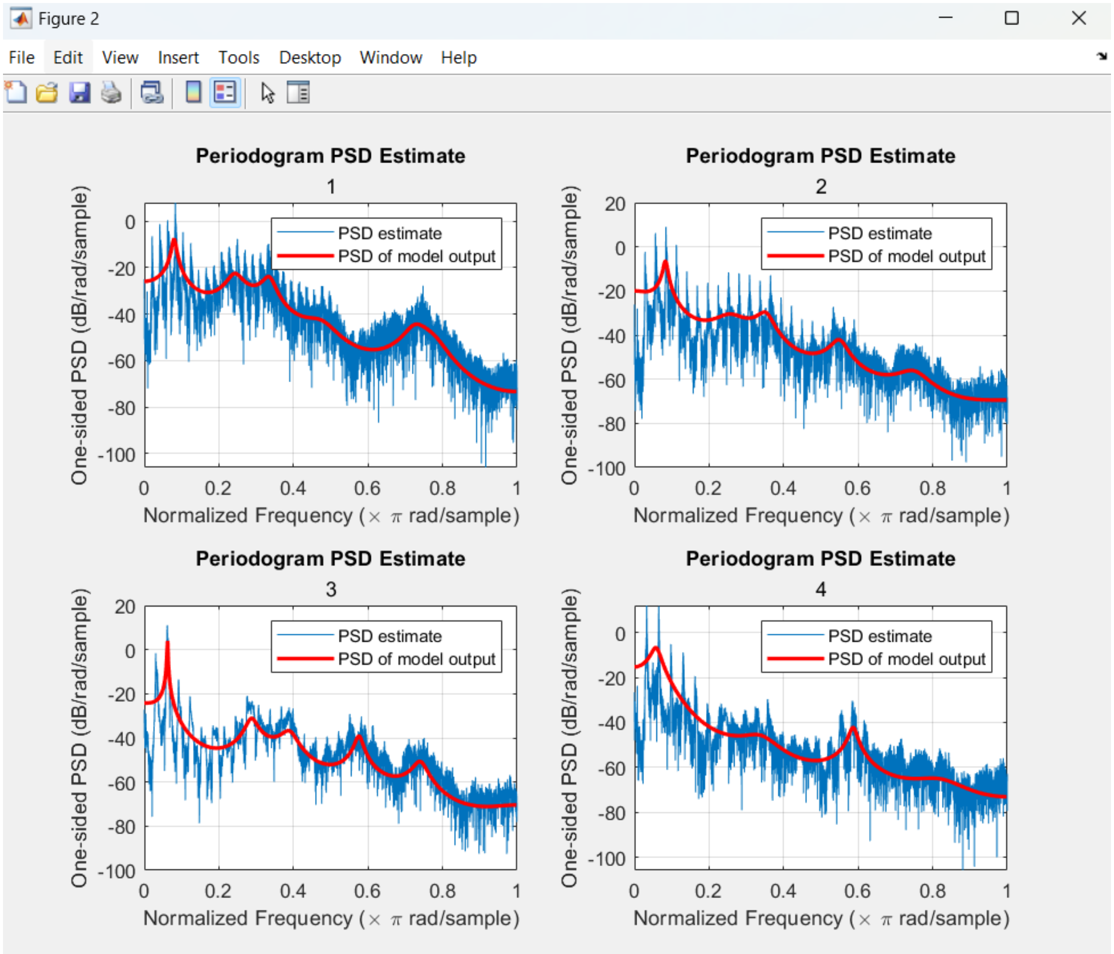


Fig. 7 the periodogram and AR spectral estimates for each speech sample

Plot 1 shows the smoothly match between PSD of the speech sample 1 and AR model.

Plot 2 shows the AR model can perfectly capture some of the speech sample 2 characteristics in the frequent domain.

Plot 3 has further improvement in matching the periodogram's peaks, especially the waveform of speech sample 3.

Plot 4 shows the perfect matches between the PSD of the speech sample 4 and the AR model.

In particular, we use an AR model to statistically match any given signal in the spectral domain allowing for the representation of how the signal's power is distributed across different frequencies. This approach estimates the power spectral density (PSD) of the signal, which provides a detailed view of its frequency content. By fitting the AR model to the signal, we can approximate the signal's spectrum, highlighting the distribution of power among various frequency components.

No, They are not the same estimation.

3

(10 marks) The input to a Wiener filter of length two is described by the difference equation, u(n) = x(n) + v2(n), where x(n) = 0.56x(n-1) + 0.32 x(n-2) + v1(n), and v1(n) and v2(n) are zero-mean white noise processes of variances 0.5 and 0.2 respectively. The desired input is given by the difference equation, d(n) = 0.65 x(n) + 0.94 x(n-1). Derive the equations for the error performance surface and the Wiener filter for this example. Do the following in MATLAB (see "WienerExample.mlx" for guidance):

a

Plot the error performance surface as function of the weights.

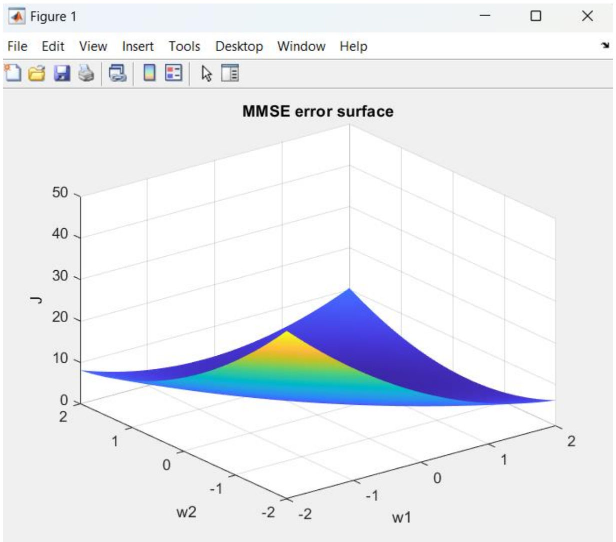


Fig. 8 the error performance surface

b

Plot the contours of the error performance surface and indicate the Wiener solution on this plot.

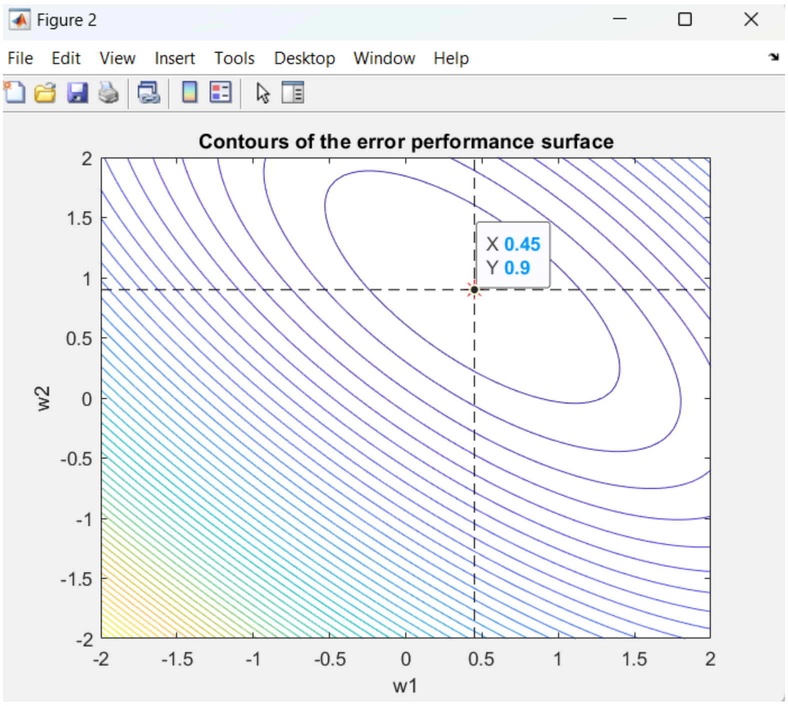


Fig. 9 The contours of the error performance with the indication of the Wiener solution

c

Plot the gradient vectors and comment on their orientation.

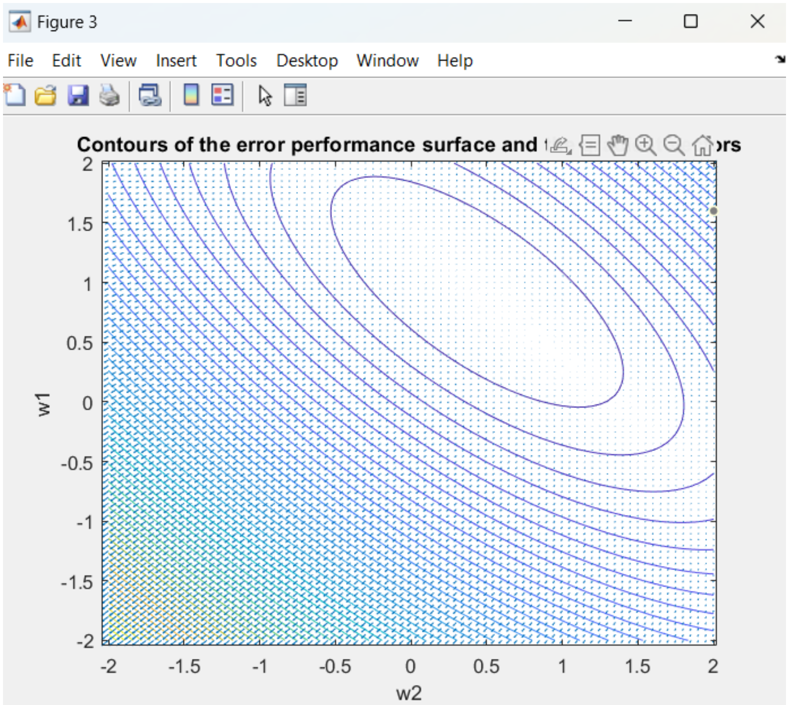


Fig. 10 The gradient vectors of the contours of the error performance

The arrows indicate the direction of the gradient vectors, it always pointing towards to the direction of gradient vectors changes. At any given point, gradient vectors always point away from the minimum solution.

4

(15 marks) Consider a one-step adaptive predictor for a generic second order real AR process defined by the difference equation u(n) + a1 u(n-1) +a2 u(n-2) = v(n), where v(n) is a zero-mean white noise process with variance, σ^2.

a

Derive the equations for r(0), r(1), and r(2).

$$\begin{bmatrix} r_u(0) & r_u(1) \\ r_u(1) & r_u(0) \end{bmatrix} \begin{bmatrix} -a_1 \\ -a_2 \end{bmatrix} = \begin{bmatrix} r_u(1) \\ r_u(2) \end{bmatrix}$$

$$-a_1 r_u(0) - a_2 r_u(1) = r_u(1)$$

$$-a_1 r_u(1) - a_2 r_u(0) = r_u(2)$$

According to Yule-Walker condition:

$$r_u(0) + a_1 r_u(1) + a_2 r_u(2) = \sigma^2$$

Therefore:

$$\begin{bmatrix} -a_1 & -(a_2+1) & 0 \\ -a_2 & -a_1 & -1 \\ 1 & a_1 & a_2 \end{bmatrix} \begin{bmatrix} r_u(0) \\ r_u(1) \\ r_u(2) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \sigma^2 \end{bmatrix}$$

Solving the matrix we can represent r(0), r(1), r(2) as:

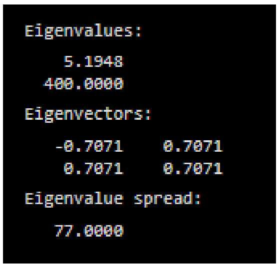
$$r(0) = \frac{\sigma^2(a_2+1)}{a_1^2 a_2 - a_1^2 - a_2^3 - a_2^2 + a_2 + 1}$$

$$r(1) = \frac{-\sigma^2 a_1}{a_1^2 a_2 - a_1^2 - a_2^3 - a_2^2 + a_2 + 1}$$

$$r(2) = \frac{\sigma^2(a_1^2 - a_2^2 - a_1)}{a_1^2 a_2 - a_1^2 - a_2^3 - a_2^2 + a_2 + 1}$$

b

If a1 = -1.9, a2 = 0.95, and $\sigma^2 = 1$, compute the eigenvalues, eigenvectors, and the eigenvalue spread in MATLAB.



```
Eigenvalues:
     5.1948
   400.0000
Eigenvectors:
   -0.7071    0.7071
    0.7071    0.7071
Eigenvalue spread:
    77.0000
```

Fig. 11 The eigenvalues, and eigenvectors, and the eigenvalue spread

The dimension of R is 2x2 since the difference equation is second order.

c

Define $e(n) = u(n) - \hat{u}(n)$, $\varepsilon 1\ (n) = 1.9 - w1(n)$, and $\varepsilon 2\ (n) = -0.95 - w2(n)$. Using the LMS algorithm with the convergence parameter, μ = 0.003, plot the power spectral densities for e(n), $\varepsilon 1(n)$, and $\varepsilon 2(n)$. Include these plots in your assignment, clearly labeling the axes. What observations can you make from these plots?

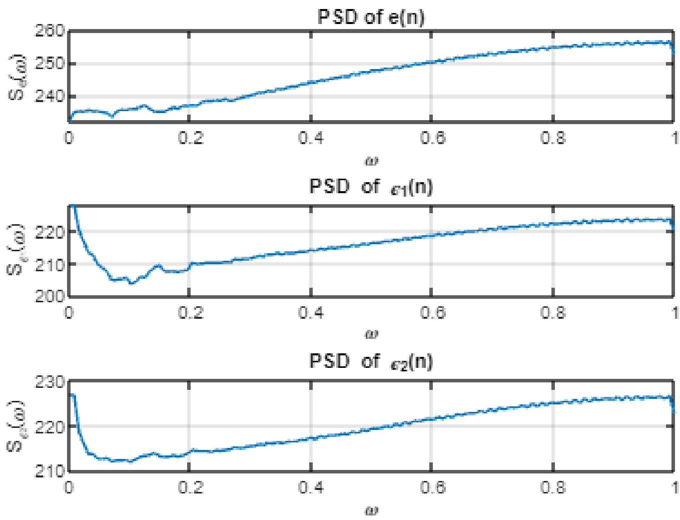


Fig. 12 The PSD for e(n), $\varepsilon 1(n)$, and $\varepsilon 2(n)$.

e(n): The PSD the error rate appears relatively flat. The PSD of the error rate is similar to the PSD of white noise. The power the error rate increases as increasing the frequency.

$\varepsilon 1(n)$: The $\varepsilon 1$ has a apparent low pass properties. The power of $\varepsilon 1$ increases as increasing the frequency after reaching its bottom around $w = 0.1$.

$\varepsilon 2(n)$: Similar to the $\varepsilon 1$, $\varepsilon 1$ also has a apparent low pass properties. The power of $\varepsilon 1$ increases as increasing the frequency after reaching its bottom around $w = 0.1$.

d

Compute the ensemble-average learning curves for the NLMS algorithm using $\delta = 0.05$. Include these plots with your assignment, along with your observations comparing the NLMS and LMS results

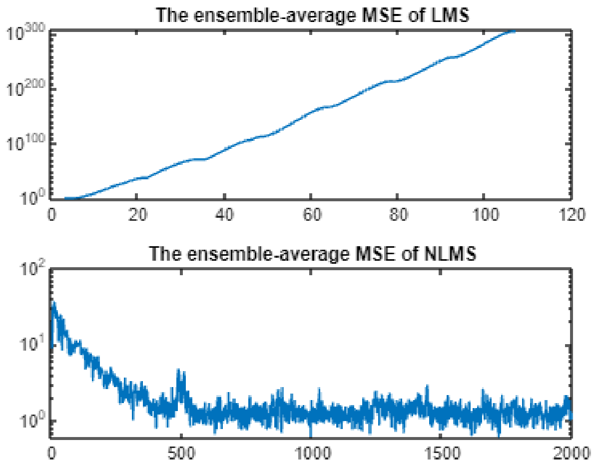


Fig. 13 The ensemble-average learning curves for LMS and NLMS algorithm when step size is 0.05

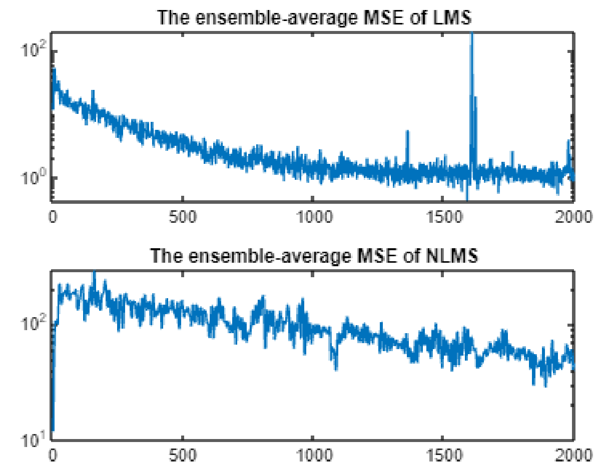


Fig. 13 The ensemble-average learning curves for LMS and NLMS algorithm when step size is 0.0005

I found that when I used u=0.05 for the LMS filter. Its MSE increased over time. This could indicate an instability of the LMS filter due to the inappropriately high step size. While I used u=0.0005 for the LMS filter. Its MSE decreased over time as the filter coefficients adapted to minimize the error, eventually reaching a steady state.

For the NLMS, its MSE decreased when u=0.05 and finally reached a steady state, while it increased over time when u=0.0005 due to the inappropriately low step size.

This question underscores the critical lesson you taught in class about the importance of choosing an appropriate step size—it should neither be too large nor too small.

Conclusion

Assignment 1 provided us with a valuable hands-on learning experience on the PDF, PSD of different noises, AR modeling, MSSE error surface, the one-step adaptive predictor, the effect of different learning rates for NLMS and LMS.

I extend my deepest appreciation for Dr. Parsa in-depth teaching and invaluable guidance.

I am looking forward to applying more knowledge in hands on labs.