Chap 9 : Adaptative Filtering

Ex 9.1 : Adaptative Linear Prediction Using the LMS and NLMS Algorithm

Let x[n] be a second order autoregressive process AR(2) that is generated according the difference equation :

$$\begin{aligned} x[n] &= -a[1] \cdot\ x[n-1] - a[2] \cdot\ x[n-2] + v[n] \\ &= 1.2728 \cdot\ x[n-1] - 0.81 \cdot\ x[n-2] + v[n] \end{aligned}$$

Where v[n] is unit variance. So the coefficients are:

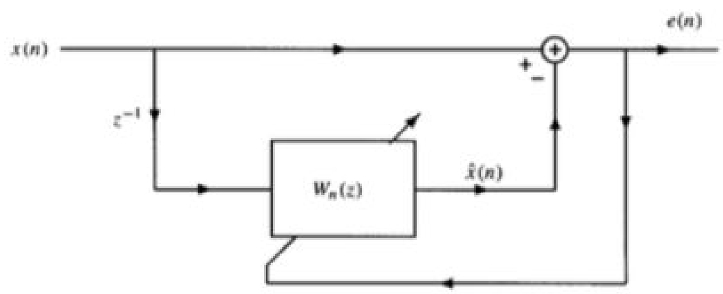
$$\begin{aligned} a[1] &= \ -1.2728 \\ a[2] &= 0.81 \end{aligned}$$

It gives a transfert function:

$$\frac{X(z)}{V(z)} = \frac{1}{1 + a[1]z^{-1} + a[2]z^{-2}} = \frac{1}{1 - 1.2728z^{-1} + 0.81z^{-2}}$$

The goal is to design a one step predictor using the adaptive LMS algorithm and to compare it to a Wiener filter. This is possible because the process is stationary.

Below the scheme of the one step adaptive predictor.



Therefore suppose we consider an adaptive linear predictor of the form:

$$\hat{x}[n] = w_n[1]x[n-1] + w_n[2]x[n-2]$$

Therefore, when $w_n[1] = 1.2728$ and $w_n[2] = -0.81$, the eror becomes $e[n] = v[n]$ and the minimum square error is

$$\xi_{\min} = \sigma_v^2 = 1$$

Data Generation

Parameter

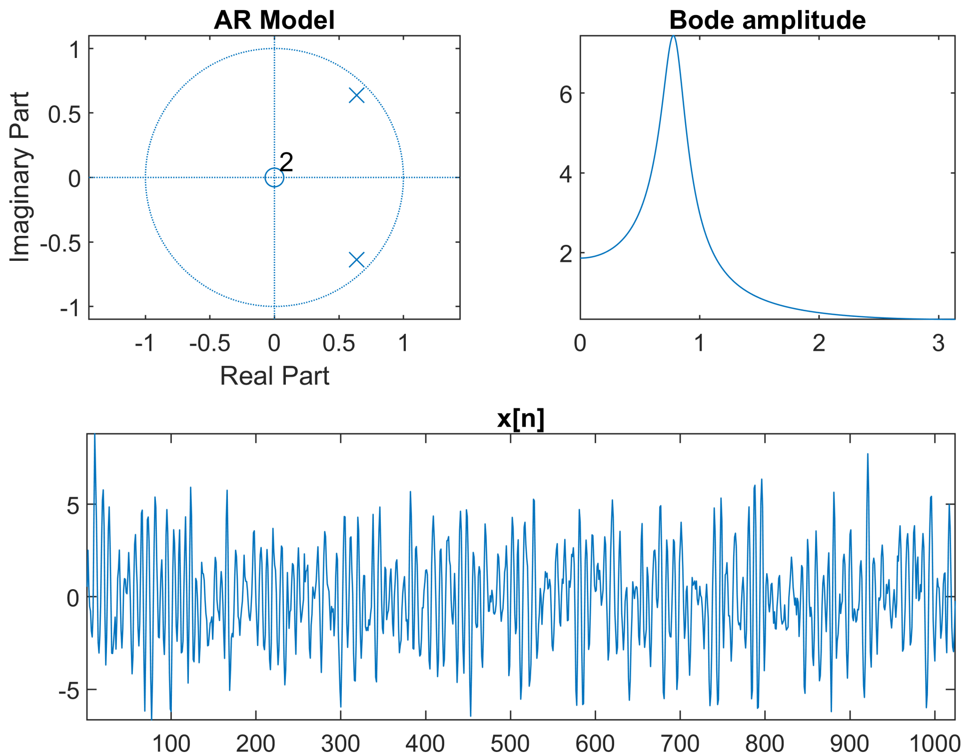
Let's set the parameters:

a	N

```
   1    -1.2728       0.81    1024
```

Generation

Now the autoregressive process can be generated. The poles/zeros and the bode are also displayed.



Wiener-Hopf

As it is a WSS, the coefficients can be found with the Wiener-Hopf equations. The Wiener filter is designed with the estimation done on the data.

The optimum causal linear predictor for x[n] is:

$$\hat{x}[n] = w[1]x[n-1] + w[2]x[n-2]$$

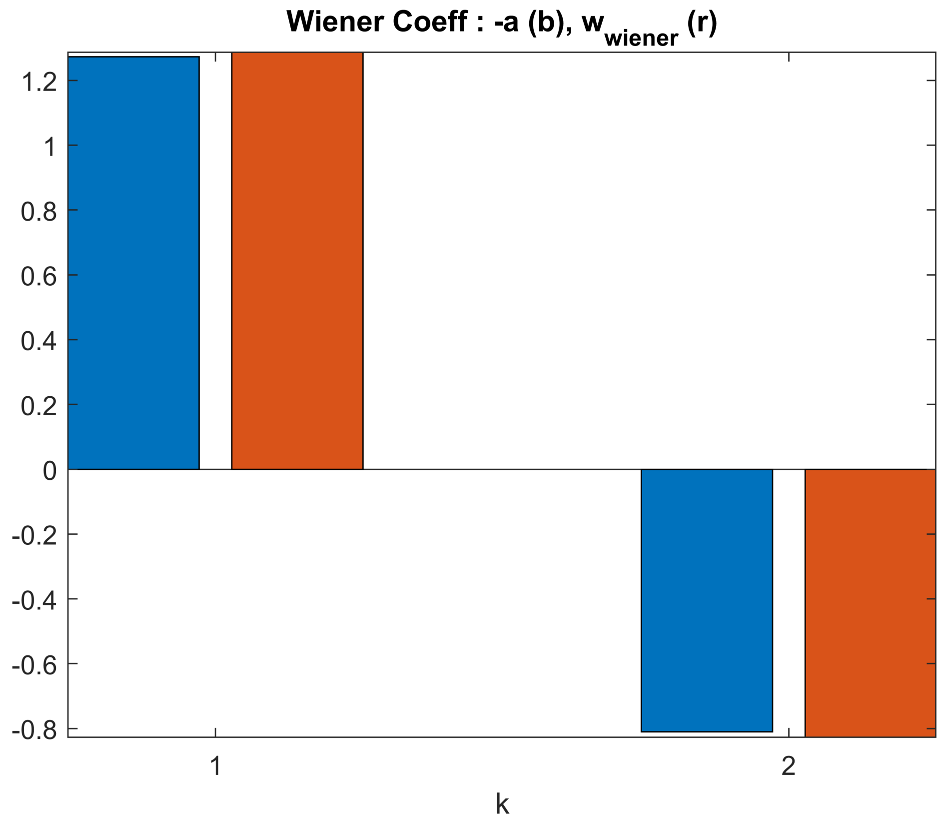
$$\hat{x}[n] = 1.2728x[n-1] - 0.81x[n-2]$$

Let's see what Wiener-Hopf provide:

Design

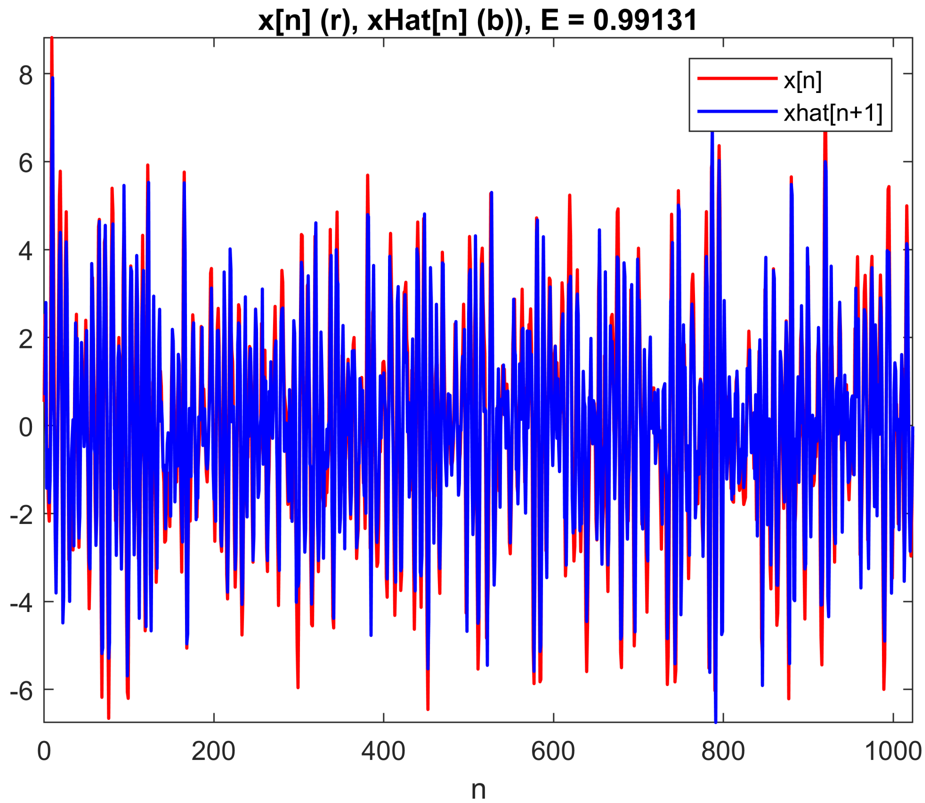
It gives predictor coefficient very close of the auto-regressive coefficients.

```
Wiener-Hopf results : w[k] =
    1.2874
   -0.8276
```



Prediction

If we apply the prediction and we compute the average squared error, it corresponds to the theory $\xi_{\min} = \sigma_v^2 = 1$.



LMS

The predicitve parameters can be also found using the LMS algorithm.

With the LMS algorithm, the predictor coefficients $w_n[k]$ are updated as follows:

$$w_{n+1}[k] = w_n[k] + \mu\, e[n] x^*[n-k]$$

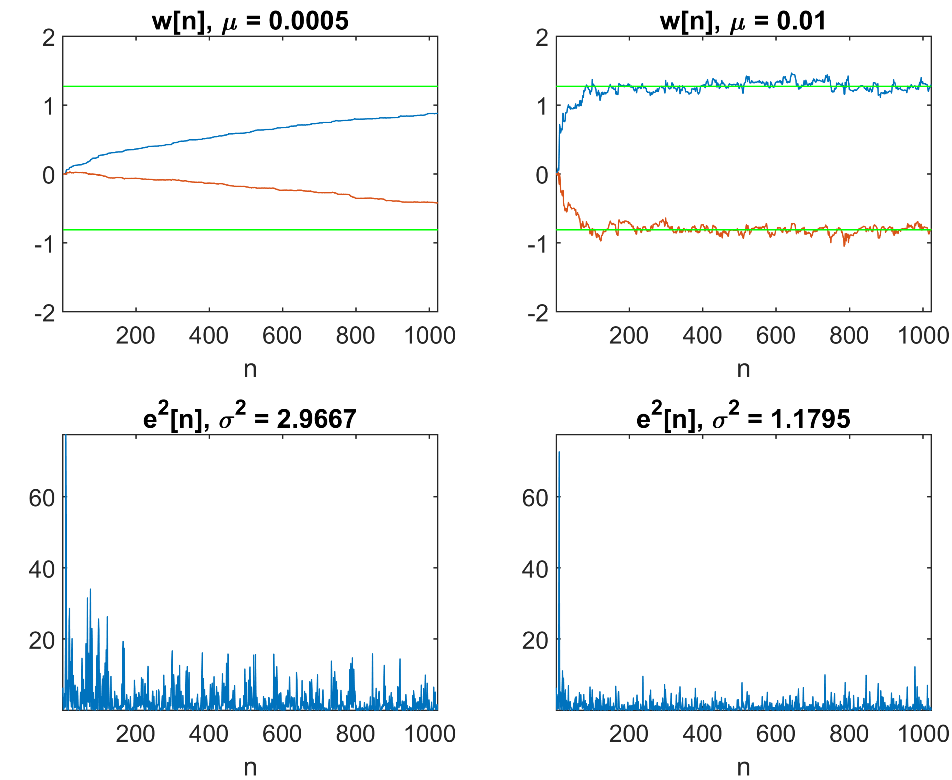
Design

For the design, we will compare two different rate μ. One too small ($\mu = 5 \cdot 10^{-4}$) and one good ($\mu = 1 \cdot 10^{-2}$). The following is showing the progression of the coefficient as well the squared error. To achieve that, you need to implement first the LMS algorithm (myLMS). Then, you can compute the estimated filter coefficient.

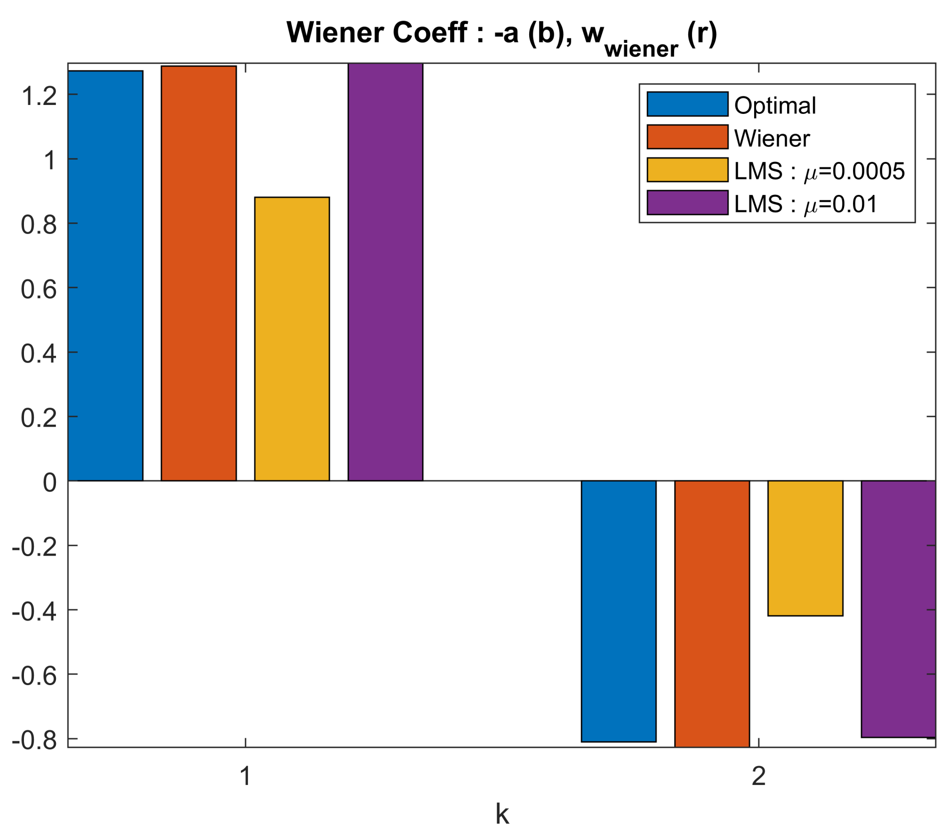
```
xinput = x(1:end-1) % input
doutput =  x(2:end) % desired output
nord   = 2 % number of coefficient
[A, E] = myLMS (xinput, doutput, mu, nord);
```

```
LMS results (mu=0.0005) :
    0.8804   -0.4187

LMS results (mu=0.01) :
    1.2971   -0.7968
```



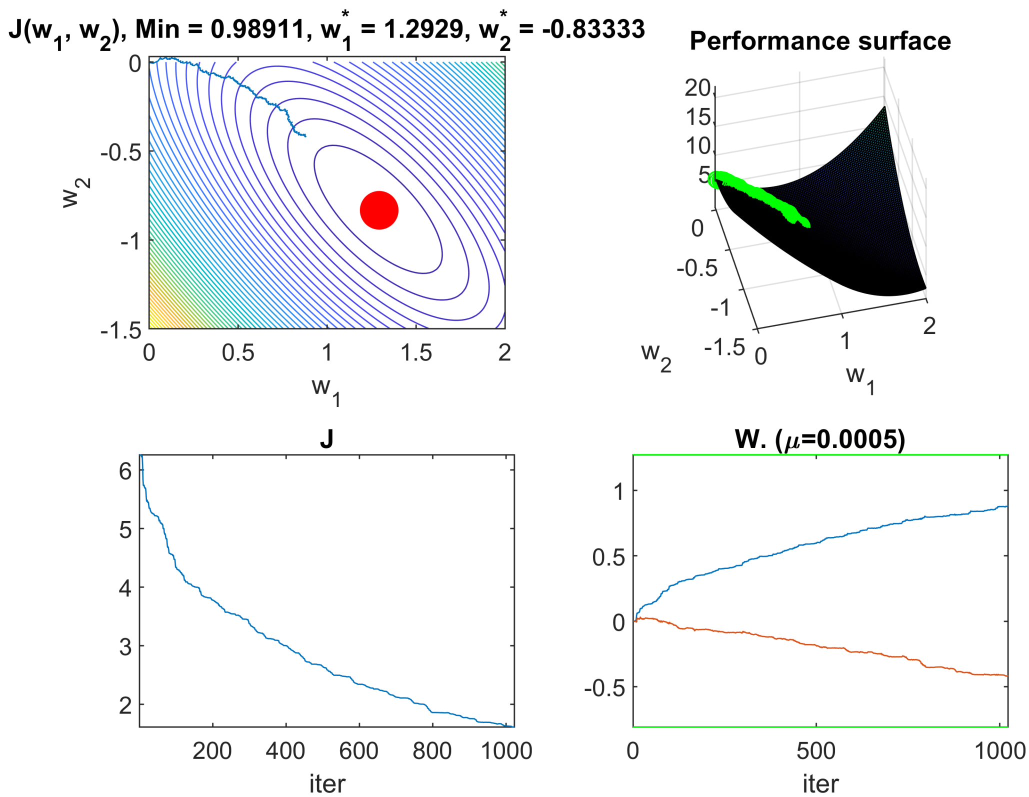
The coefficients obtained are very close of the optimal coefficients after 1000 steps of adaptation for the the good μ.

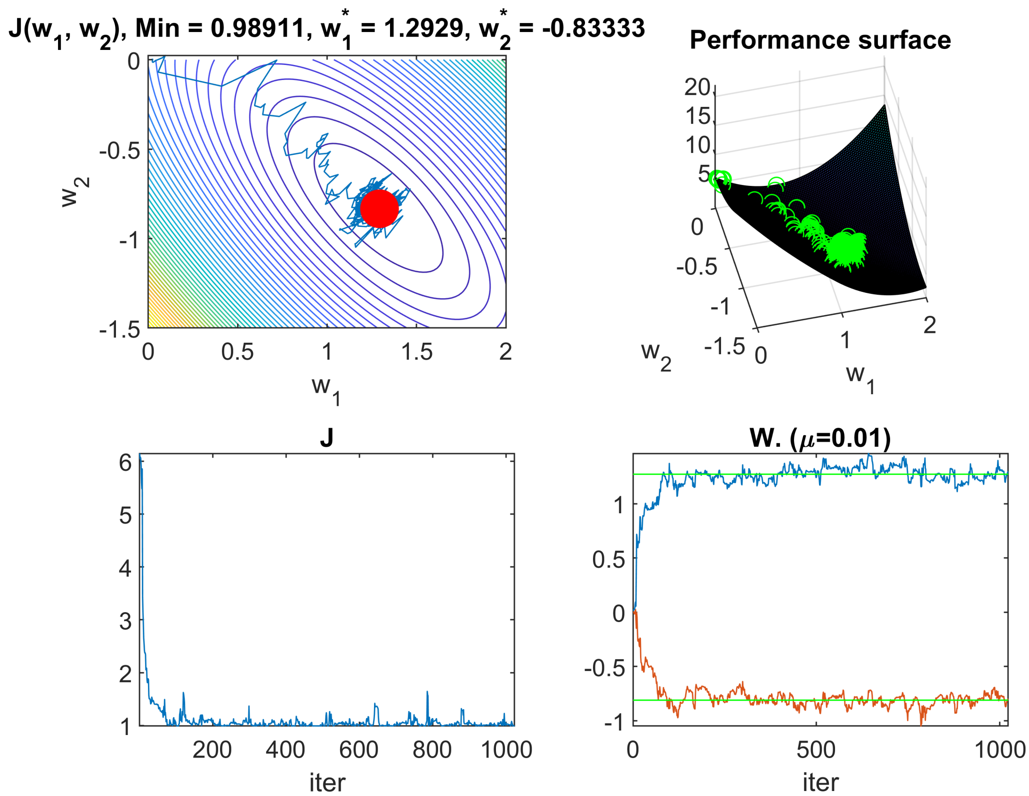


Performance surface

Finally, a performance surface is drawn with the two rates. It is showing the path realized on the error surface by the optimizer.

It can be observed that with the good rate, the optimizer is reaching the optimum coefficients and then oscillate there while with the slow rate, the optimizer is never reaching it. To help you ot build the following figures, you can create a function *J = ComputeJFigure (x, a1, a2)* which will compute the average squared error of the prediction for the given parameters.





NLMS

The predicitve parameters can be also found using the normalized LMS algorithm also called NLMS.

With the NLMS algorithm, the predictor coefficients $w_n[k]$ are updated as follows:

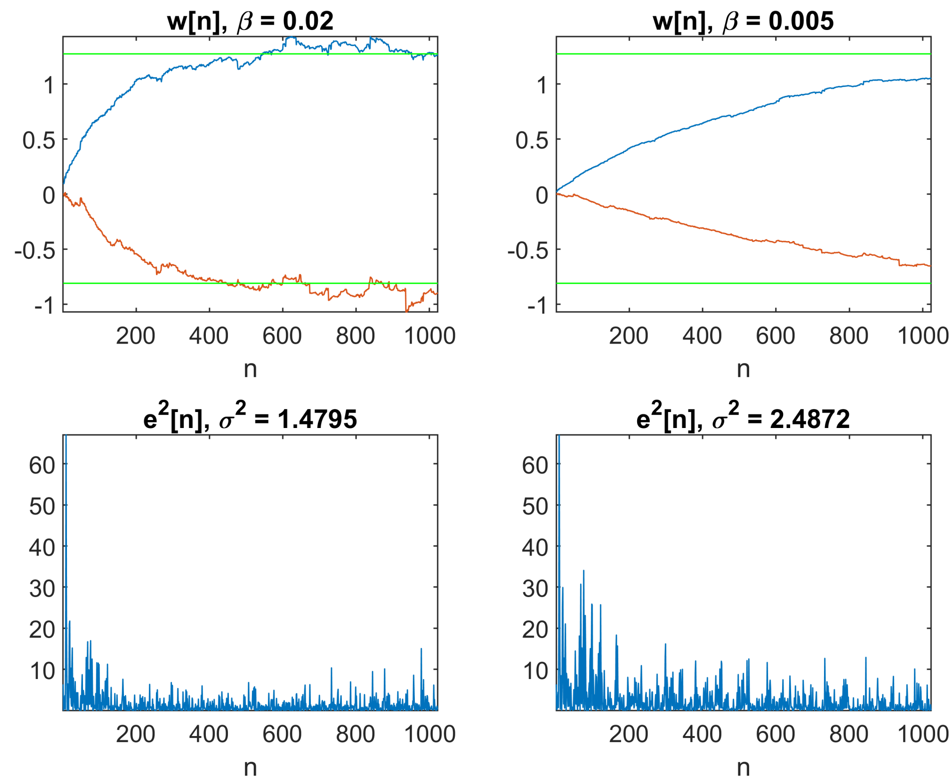
$$w_{n+1}[k] = w_n[k] + \beta \frac{x^*[n-k]}{\varepsilon + x^2[n-1] + x^2[n-2]}\, e[n]$$

Design

Lets design the NLMS by using the normalized step sizes of $\beta = 0.02$ and $\beta = 0.005$, with $\epsilon = 0.0001$.

```
NLMS results (beta=0.02) :
    1.2653   -0.8965

NLMS results (beta=0.005) :
    1.0517   -0.6513
```



Finally, we can compare the found coefficients between all the methods.