Applied Signal Processing: Ex 1

Or Tal 305793168

Question 1:

1. Refer to the noise process defined in Subsection 1.1 and appropriately use the notation shown in the lecture notes. Find the desired signal $\{D_n\}$, the measurements signal $\{U_n\}$. Write the matrix $\mathbf{R}$ and vector $\mathbf{p}$ in terms of the process parameters, α and σ_N^2.

   In this section we would like to estimate Z_n, hence we will define $\{D_n\} = Z_n$.

   Assuming we hold L samples $\{Z_{n-1},\cdots,Z_{n-L}\}$ we define $U_n = Z_{n-1}$. We may see that

$$\boldsymbol{R} = \mathbb{E}[\boldsymbol{U_n}\boldsymbol{U_n^T}] = \mathbb{E}\begin{bmatrix} U_nU_n & U_nU_{n-1} & \cdots & U_nU_{n-L+1} \\ U_{n-1}U_n & U_{n-1}U_{n-1} & \cdots & U_{n-1}U_{n-L+1} \\ \vdots & \vdots & \ddots & \vdots \\ U_{n-L+1}U_n & U_{n-L}U_n & \cdots & U_{n-L+1}U_{n-L+1} \end{bmatrix}$$

$$= \mathbb{E}\begin{bmatrix} Z_{n-1}Z_{n-1} & Z_{n-1}Z_{n-2} & \cdots & Z_{n-1}Z_{n-L} \\ Z_{n-2}Z_{n-1} & Z_{n-2}Z_{n-2} & \cdots & Z_{n-2}Z_{n-L} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{n-L}Z_{n-1} & Z_{n-L}Z_{n-1} & \cdots & Z_{n-L}Z_{n-L} \end{bmatrix} = \begin{bmatrix} R_{Z,0} & \cdots & R_{Z,L-1} \\ \vdots & \ddots & \vdots \\ R_{Z,L-1} & \cdots & R_{Z,0} \end{bmatrix}$$

   Hence:

$$\boldsymbol{R} = \begin{bmatrix} R_{Z,0} & \cdots & R_{Z,L-1} \\ \vdots & \ddots & \vdots \\ R_{Z,L-1} & \cdots & R_{Z,0} \end{bmatrix} = \begin{bmatrix} \frac{1}{1-\alpha^2}+\sigma_N^2 & \frac{\alpha}{1-\alpha^2} & \cdots & \frac{\alpha^{L-1}}{1-\alpha^2} \\ \frac{\alpha}{1-\alpha^2} & \frac{1}{1-\alpha^2}+\sigma_N^2 & & \vdots \\ \vdots & & \ddots & \frac{\alpha}{1-\alpha^2} \\ \frac{\alpha^{L-1}}{1-\alpha^2} & \cdots & \frac{\alpha}{1-\alpha^2} & \frac{1}{1-\alpha^2}+\sigma_N^2 \end{bmatrix}$$

   And the Auto-correlation vector:

$$\boldsymbol{P} = \mathbb{E}[D_n\boldsymbol{U}_n] = \mathbb{E}\begin{bmatrix} D_nU_n \\ D_nU_{n-1} \\ \vdots \\ D_nU_{n-L+1} \end{bmatrix} = \mathbb{E}\begin{bmatrix} Z_nZ_{n-1} \\ Z_nZ_{n-2} \\ \vdots \\ Z_nZ_{n-L} \end{bmatrix} = \begin{bmatrix} R_{Z,1} \\ R_{Z,2} \\ \vdots \\ R_{Z,L} \end{bmatrix} = \frac{1}{1-\alpha^2}\begin{bmatrix} \alpha \\ \alpha^2 \\ \vdots \\ \alpha^L \end{bmatrix}$$

2. Write the prediction for the optimal linear prediction filter

   We have seen in the lecture that the optimal linear estimator in terms of MSE would be:

$$\widehat{D_n} = \boldsymbol{w_n^T}\boldsymbol{U_n}\,;\;\; \boldsymbol{w}_n = \boldsymbol{R^{-1}P}$$

3. a. $0 < \alpha < 1,\; \sigma_N^2 = 0$ show that the optimal filter in this case is $\boldsymbol{w}^* = [w_0, 0,\cdots,0]^T$, give intuition to this solution

   Intuitively, when $\sigma_N^2 = 0$, there is no noise N_n meaning that $Z_n = X_n = \alpha X_n + G_n$

   The best prediction we could have for G_n is its mean (=0) hence our best estimation should be $\hat{X}_n = \alpha X_{n-1}$

   Therefore $\hat{X}_n = [X_{n-1},\cdots,X_{n-L}]\cdot\boldsymbol{w}^* = \alpha X_{n-1} \Rightarrow \boldsymbol{w}^* = [\alpha, 0,\cdots,0]^T$ as requested.

More formal proof would be:

In Lecture we saw that the optimal linear estimator in the WSS case would be given by $\widehat{D}_n = \boldsymbol{w_n^T U_n}$

Moreover, we saw that it is optimal only if $\mathbb{E}[e_n \boldsymbol{U}_n^T] = \boldsymbol{0}$, then when $\boldsymbol{w}^* = [w_0, 0, \cdots, 0]$ we see that

$$\mathbb{E}[e_n \boldsymbol{U}_n^T] = \mathbb{E}\left[\left(D_n - \widehat{D}_n\right)\boldsymbol{U}_n^T\right] = \mathbb{E}\left[\left(X_n - \boldsymbol{w^{*T}}\boldsymbol{X}_{n-1}\right)\boldsymbol{X}_{n-1}^T\right] = \mathbb{E}[(X_n - w_0 X_{n-1})\boldsymbol{X}_{n-1}^T] = \mathbb{E}\begin{bmatrix}(X_n - w_0 X_{n-1})X_{n-1} \\ \vdots \\ (X_n - w_0 X_{n-1})X_{n-L}\end{bmatrix}^T$$

$$= \mathbb{E}\begin{bmatrix}X_n X_{n-1} \\ \vdots \\ X_n X_{n-L}\end{bmatrix}^T - w_0 \mathbb{E}\begin{bmatrix}X_{n-1}X_{n-1} \\ \vdots \\ X_{n-1}X_{n-L}\end{bmatrix}^T = \begin{bmatrix}R_{X,1} \\ \vdots \\ R_{X,L}\end{bmatrix}^T - w_0 \begin{bmatrix}R_{X,0} \\ \vdots \\ R_{X,L-1}\end{bmatrix}^T = \begin{bmatrix}\frac{\alpha}{1-\alpha^2} - \frac{w_0 \alpha^0}{1-\alpha^2} \\ \vdots \\ \frac{\alpha^L}{1-\alpha^2} - \frac{w_0 \alpha^{L-1}}{1-\alpha^2}\end{bmatrix}^T$$

And for $w_0 = \alpha$: $\mathbb{E}[e_n \boldsymbol{U}_n^T] = \begin{bmatrix}\frac{\alpha}{1-\alpha^2} - \frac{\alpha \cdot \alpha^0}{1-\alpha^2} \\ \vdots \\ \frac{\alpha^L}{1-\alpha^2} - \frac{\alpha \cdot \alpha^{L-1}}{1-\alpha^2}\end{bmatrix}^T = \begin{bmatrix}0 \\ \vdots \\ 0\end{bmatrix} \Rightarrow$ as requested.

b. $\alpha = 0, \sigma_N^2 > 0$ what is the number of non-zero filter coefficients? Give intuition for the solution

Intuitively, when $\alpha = 0$: $Z_n = X_n + N_n = G_n + N_n$, hence, Z_n is simply the sum of two gaussian noise signals with zero mean. Therefore the optimal estimation would be the expectation of these gaussians = $\boldsymbol{0}$.

More formally, we would like to find the num. of non-zero values in $\boldsymbol{w_n}$ s.t $\boldsymbol{R w_n} - \boldsymbol{p} = \boldsymbol{0}$ holds.

$\boldsymbol{P} = \frac{1}{1-\alpha^2}\begin{bmatrix}\alpha \\ \alpha^2 \\ \vdots \\ \alpha^L\end{bmatrix} = \boldsymbol{0}$, so we may look for $\boldsymbol{R w_n} = \boldsymbol{0}$.

$$\boldsymbol{R} = \begin{bmatrix}\frac{1}{1-\alpha^2} + \sigma_N^2 & \frac{\alpha}{1-\alpha^2} & \cdots & \frac{\alpha^{L-1}}{1-\alpha^2} \\ \frac{\alpha}{1-\alpha^2} & \frac{1}{1-\alpha^2} + \sigma_N^2 & & \vdots \\ \vdots & & \ddots & \frac{\alpha}{1-\alpha^2} \\ \frac{\alpha^{L-1}}{1-\alpha^2} & \cdots & \frac{\alpha}{1-\alpha^2} & \frac{1}{1-\alpha^2} + \sigma_N^2\end{bmatrix} = \sigma_N^2 \boldsymbol{I}_{L\times L}$$, is of full rank meaning that the homogeneous case has only one solution, and because $\boldsymbol{R}$ is diagonal we may conclude that in this case the only solution is $\boldsymbol{0}$

4. Write a Matlab code that generates the WSS signal. Set the number of samples of the process to correspond to 10 seconds of audio in 48KHz sampling frequency. You can create the signal by drawing white Gaussian process using 'randn' command, and passing it through a filter for which $b = [1]$ and $a = [1, -\alpha]$ (make sure you understand why α should have a minus sign!).

   (a) For $\alpha = 0.5$ and $\sigma_N^2 = 1$, calculate the empirical mean:

$$\frac{1}{N}\sum_{\ell=1}^{N} Z_\ell, \tag{10}$$

   and the empirical second moment:

$$\frac{1}{N}\sum_{\ell=1}^{N} Z_\ell^2, \tag{11}$$

   where N is the number of samples in the process. Compare the values you calculated and compare them to the expected theoretical values.

   (b) Find the appropriate scaling coefficient β, so that $\mathbb{E}(\beta Z_n)^2 = \frac{1}{2}$.

We would expect

$$\mathbb{E}(Z_n) = \mathbb{E}(X_n) + \mathbb{E}(N_n) = \alpha^n \mathbb{E}(X_0) = 0; \quad \mathbb{E}(Z_n^2) = R_{z,0} = \frac{1}{1-\alpha^2} \approx 2\frac{1}{3}$$

$$\mathbb{E}((\beta Z_n)^2) = \beta^2 \mathbb{E}(Z_n^2) = \beta^2 \cdot \frac{7}{3} \Rightarrow \beta = \sqrt{\frac{3}{14}} \approx 0.4629$$

Output for sections a, b:

```
Q1 section 4.a:
ampirical mean = -0.0029390017596332032
ampirical 2nd moment = 2.333445212681118
Q1 section 4.b:
beta = 0.46289980917286405
testing: mean((beta*Z_n)^2) = 0.500001850860537
```

(c) Play the scaled signal using Matlab sound command. Explain why the scaling is necessary.

After listening to both signals, I noticed that they are different. This difference holds in the fact that the "play" function expects [-1,1] values, therefore it clips all values >1 to 1, and all values <-1 to -1

for the first case, that has 2.26 variance , more values are outside $[-1,1]$ and were passed to -1 or 1 accordingly. And as for the case of 0.5 variance, significantly less values were outside $[-1,1]$, hence the difference.

Scaling shifted values to fit the "play" range better.

5. For $\alpha = 0.9$ and $\sigma_N^2 = 0.5$ do the following:

(a) Calculate the optimal filters coefficients of orders $L = 1$ to $L = 5$. Hint: You can use Matlab 'toeplitz' command for the calculation of $\mathbf{R}$. Write the filter coefficients in your solution.

calculated according to $\boldsymbol{w}^* = \boldsymbol{R}^{-1}\boldsymbol{P}$

```
-- section 5.a. --
coef vec of order: 1 is: [0.82191781]
coef vec of order: 2 is: [0.65934066 0.1978022 ]
coef vec of order: 3 is: [0.64954128 0.16513761 0.04954128]
coef vec of order: 4 is: [0.64892506 0.16308354 0.04146192 0.01243857]
coef vec of order: 5 is: [0.64888621 0.16295403 0.04095253 0.01041166 0.0031235 ]
```

(b) Implement the filters in Matlab and calculate $\hat{Z}_n$. Hint: If you use Matlab 'filter' command, and set the first coefficient of the b vector to zero. Make sure you understand why this is necessary.
Calculate the prediction error signal $e_n = Z_n - \hat{Z}_n$ for all values of L. Calculate the appropriate scaling coefficient β, such that $\mathbb{E}(\beta Z_n)^2 = \frac{1}{2}$. Use Matlab 'sound' command to listen to the scaled original noise process βZ_n and to the scaled residual prediction error βe_n. Does the prediction error sound lower?

I ran a few experiments, and in all of them the prediction sound pitch sounded higher, and weaker.

(c) Calculate the average estimation error, $\frac{1}{N}\sum_{\ell=1}^{N} e_\ell^2$, for all values of L. Explain why this value cannot be smaller than $1 + \sigma_N^2$.

```
-- section 5.d. --
average estimation error = 1.7763134551888509, L = 1
average estimation error = 1.5973204474143203, L = 2
average estimation error = 1.589719276345053, L = 3
average estimation error = 1.589668734755126, L = 4
average estimation error = 1.58962915280736, L = 5
```

As G_n, N_n are iid, our best estimation for them would be their mean, which is zero, hence our best estimator should estimate $\hat{Z}_n = \alpha X_{n-1}$ in the best case scenario.

Meaning that the MSE would then be computed by:

$$\frac{1}{N}\sum_{l=1}^{N} e_l^2 = \frac{1}{N}\sum_{l=1}^{N}\left(Z_l - \hat{Z}_l\right)^2 \geq \frac{1}{N}\sum_{l=1}^{N}(\alpha X_{l-1} + G_n + N_n - \alpha X_{l-1})^2 = var(G_n + N_n) = 1 + \sigma_N^2$$

(d) For all values of L, calculate the noise reduction (in dB scale) given in the following formula:

$$\mathrm{NR}_{\mathrm{dB}} \triangleq 10\log_{10} \frac{\frac{1}{N}\sum_{\ell=1}^{N} Z_\ell^2}{\frac{1}{N}\sum_{\ell=1}^{N} e_\ell^2}. \tag{12}$$

Explain the operative meaning of $\mathrm{NR}_{\mathrm{dB}}$.

```
-- section 5.e. --
NRdb = 14.934234075477908, L = 1
NRdb = 16.816963193321996, L = 2
NRdb = 17.248427013121702, L = 3
NRdb = 17.354613502395747, L = 4
NRdb = 17.381177747614075, L = 5
```

The NRdb relation above is a $10\log_{10}$ scale between the empirical variances of the original signal and the error

This expresses a relation between the variance of the noise, and the variance of noise that "remained" after subtracting our prediction, quantifying a ratio for the reduction of noise, where higher is better.

Question 2:

For the process with parameters $\alpha = 0.9$ and $\sigma_N^2 = 0.5$, calculate the filter for $L = 4$ using the steepest descent algorithm as follows:

1. Calculate the eigenvalues of $\mathbf{R}$ (you can use Matlab 'eigs' function). What is the largest eigenvalue?

```
-- section 1 --
eigenvalues of R are:
[19.06124298  2.12967383  1.03875702  0.82295775]
largest eigenvalue = 19.06124297752537
```

2. For every $\mu \in \{0.001, 0.01, 0.1, 0.2\}$ calculate the weight vector $\mathbf{w}_n$ over 100 iterations of the steepest descent algorithm. For every iteration (and every value of μ) calculate the error norm:

$$||\mathbf{c}_n||^2 = ||\mathbf{w}_n - \mathbf{w}^*||^2. \tag{13}$$

3. Plot a figure whose X axis is the iteration (starting from 0, in which $\mathbf{w}$ is a zero vector) and whose Y axis is the relative normalized weight error in dB scale. Namely:

$$10 \log_{10} \frac{||\mathbf{c}_n||^2}{||w^*||^2} \tag{14}$$

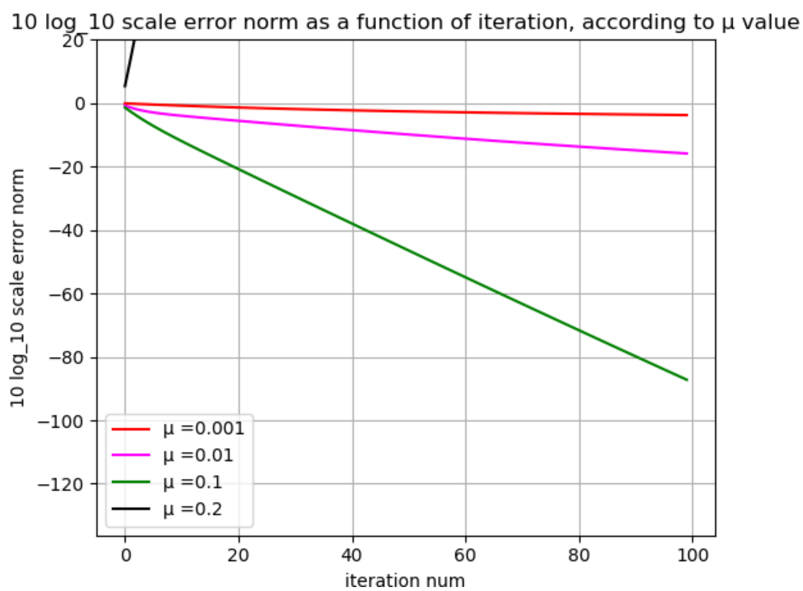
4. Plot all the graphs, corresponding to all values of μ on a single figure (using the 'hold' command). Set the maximal value in the Y axis to 20dB.

5. Explain the results. Relate the maximal μ value for which the error decreases to the bound seen in class ($\frac{2}{\lambda_{\max}}$).

The largest eigenvalue is ~ 19.0613

Hence, as we have seen in the lecture, for any $0 < \mu < \frac{2}{\lambda_{max}}$ the algorithm will eventually converge to the correct solution.

$\frac{2}{\lambda_{max}} \approx \frac{2}{19.0613} \approx 0.105$, meaning that for all of these μ values, except for 0.2, μ is within the convergence range, and we may see that as closer we get to $< \frac{2}{\lambda_{max}}$, we see a faster convergence.

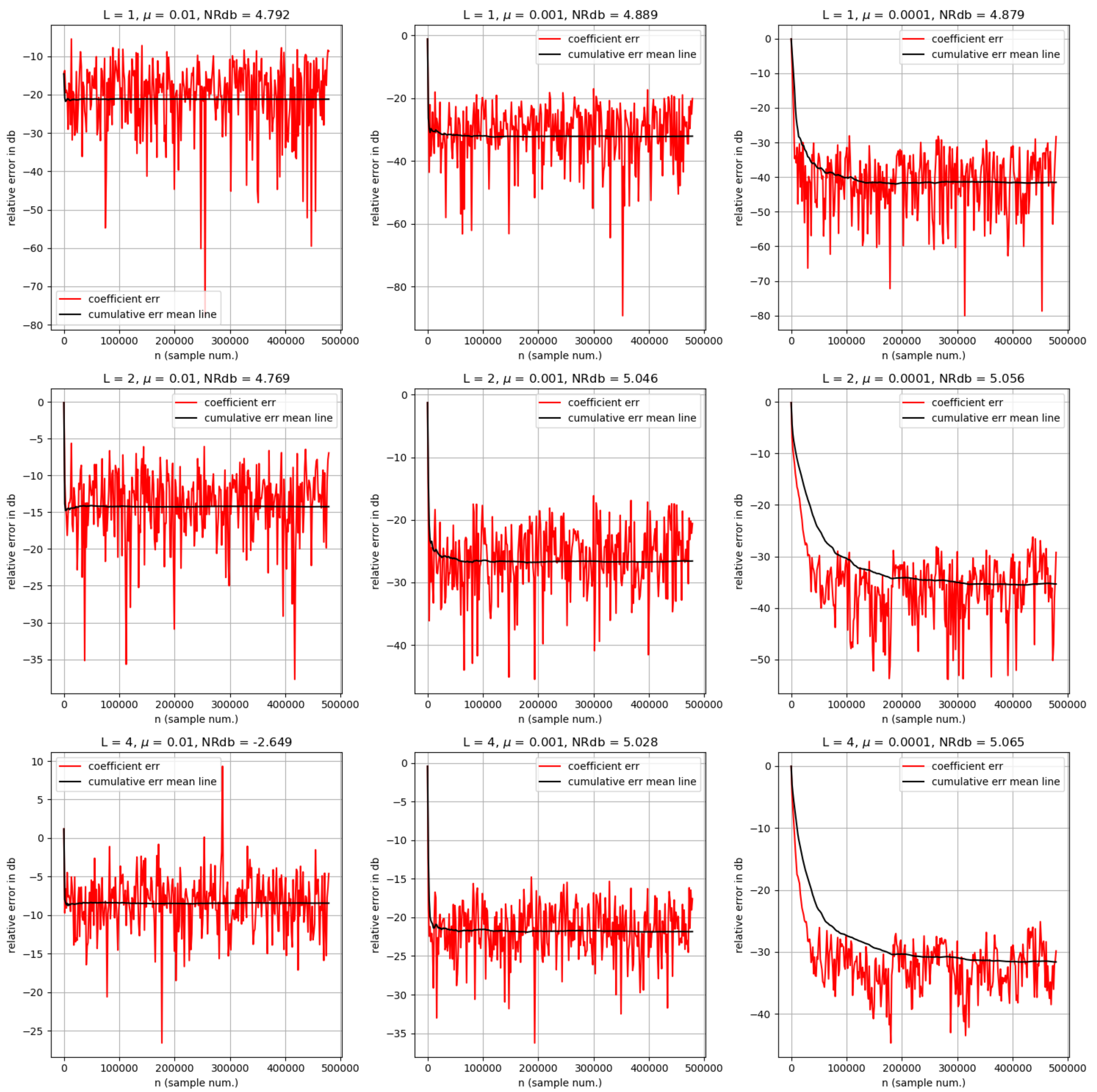


Question 3:

In this question we shall design filters for the process with parameters $\alpha = 0.9$ and $\sigma_N^2 = 0.5$, using the LMS algorithm as follows:

1. Implement the LMS algorithm for the filter calculation for all the combinations of $L \in \{1, 2, 4\}$ and $\mu \in \{0.01, 0.001, 0.0001\}$. For every setting of L and μ, plot the error in the filter coefficients (14) in dB scale as a function of the iterations, and plot the relative prediction error as in (12). Note that this time, the number of iterations is the number of samples in the process.

Step size for visibility on graphs: 1000 samples



2. Explain the trade-off between μ and L. What do you think is a reasonable setting for μ and L? Justify your answer.

<u>Hint:</u> Check what happens to the largest eigenvalue of $\mathbf{R}$ when L increases.

Largest eigenvalues matching L values:

```
section 2:
L = 1, R's largest eigenvalue is ~ 5.7632
L = 2, R's largest eigenvalue is ~ 10.5
L = 4, R's largest eigenvalue is ~ 19.0612
```

As seen here, as L goes up, λ_{max} goes up, and $\frac{2}{\lambda_{max}}$ gets smaller $\left(\frac{2}{\lambda_{max}} \approx \begin{cases} 0.347, & L=1 \\ 0.19, & L=2 \\ 0.105, & L=4 \end{cases}\right)$

Due to that,

In order to consider more information in the estimation process we would like to have a larger L value, yet it increases the influence of the step μ as well (closer to the upper bound) and therefore the signal gets “noisier”.

In addition, when L is relatively large $(\gg 4)$ it could have impact on computation time, and the estimation would start to converge in delay – say we estimate using $L = 100$, hence estimation should start to converge starting from the 101^{th} sample.

As we decrease μ, the errors’ variance seems to decrease, the mean seems to converge slower, and the mean coefficient error seems to get lower.

According these observations, we see that we have a trade-off between L, μ regarding convergence opposed to relative prediction error and the prediction error variance.

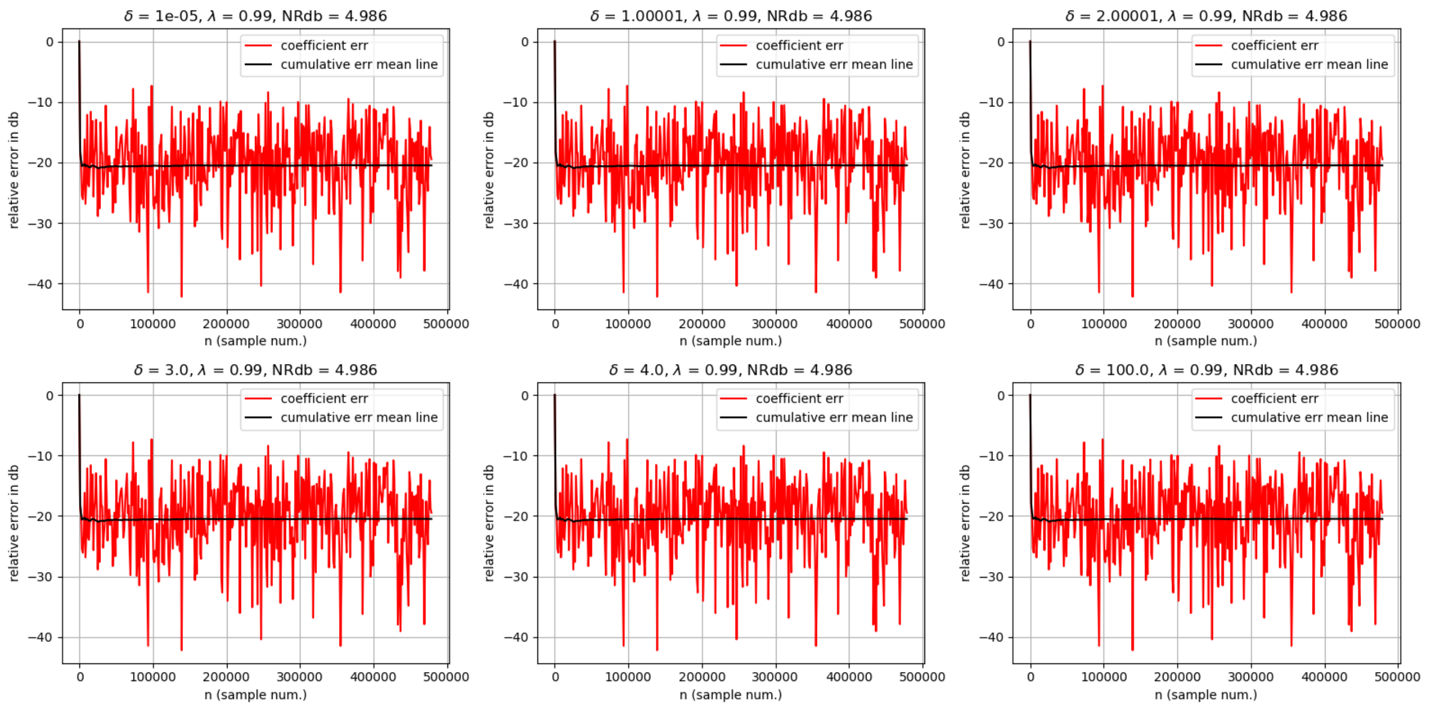
For this model, we would like the error to have a fast convergence and a small variance, aside to larger NRdb value and smaller relative coefficient value.

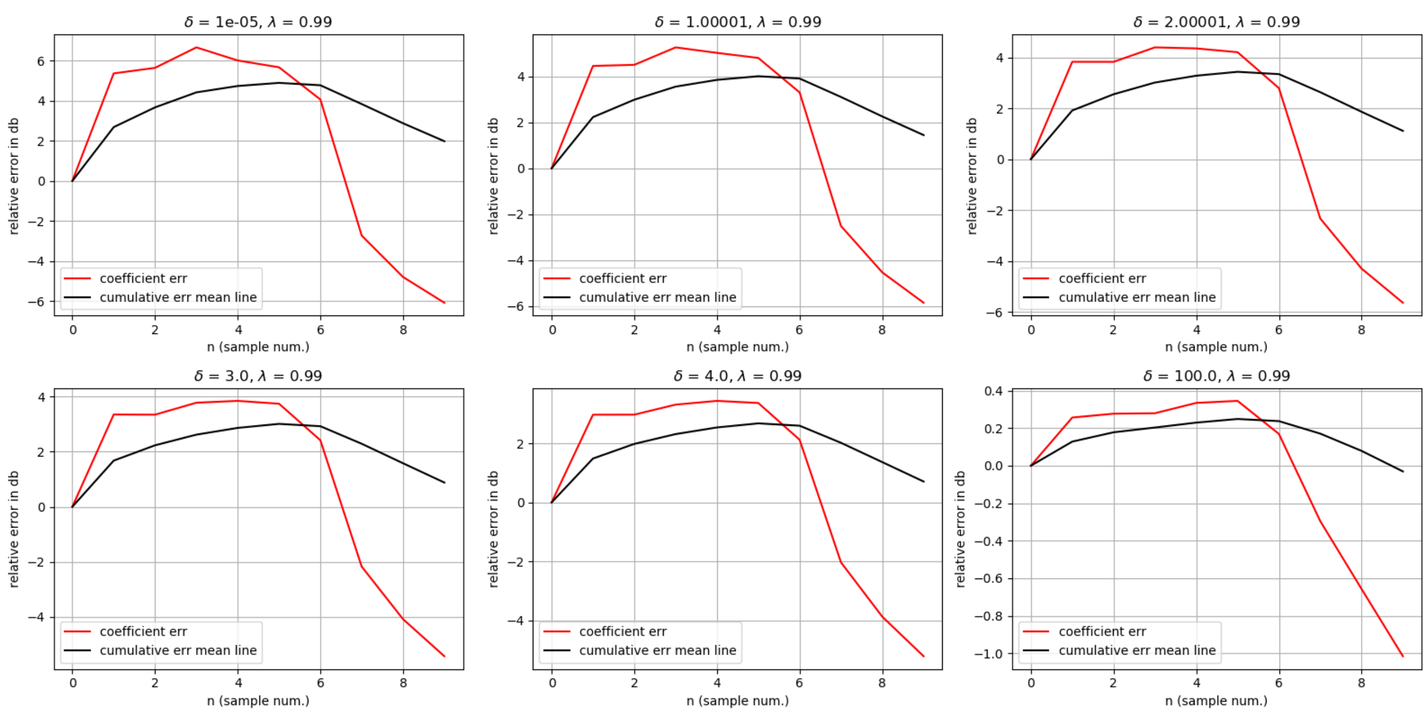
Hence, it appears that a fair choice of parameters for the model would be $\mu = 0.001,\ L = 2$

Question 4 - Recursive Least Squares (RLS) Algorithm

In this question we shall design filters for the process with parameters $\alpha = 0.9$ and $\sigma_N^2 = 0.5$, using the RLS algorithm as follows:

1. Implement the RLS algorithm as presented in the lecture notes for $L = 2$, $\lambda = 0.99$ and various values of δ . Plot the figure of the relative coefficient error (14) and the prediction error (12).





2. suggest a good setting for δ explain your choice

As one may notice, impact of choice of δ is insignificant as $num\ of\ samples \gg L$,and as specified in lecture, in this case the choice of δ is neglectable. We may choose $\delta = 10^{-2}$, or any other value.

3. explain why λ should be set close to 1

λ represents the "forgetfulness" factor in this estimation, hence as we are anticipating a signal that is distorted with a WSS noise, we would like the filter to adapt, yet we would still like to consider the close past events.

When sampling in 48 Khz, we should have 480 samples in $\frac{1}{100}$ second, and $0.99^{480} \approx 8 \cdot 10^{-3}$ hence the relevant time for this estimation drops significantly within the $\frac{1}{100}$ second range.

Having $\lambda = 1$ means simply minimizing the MSE of the entire input, and as we wish to have our filter adapt to the noise conditions, we would like to have some forgetfulness factor, and yet some amount of relevant samples. We may see that even $\lambda = 0.95$ drops way more within $\frac{1}{100}$ second range as $0.95^{480} \approx 2 \cdot 10^{-11}$

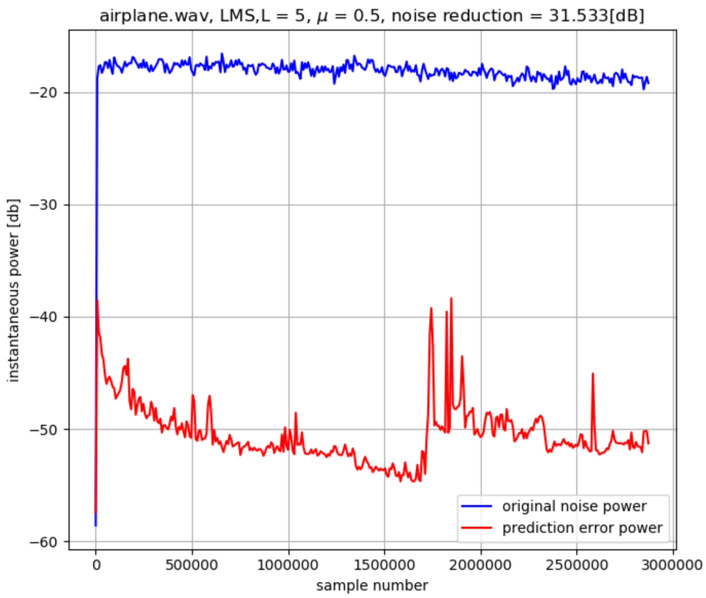
We may conclude that we would like our filter to be close to one, and yet less than one to have some portion of relevant samples, and yet still have adaptivity.

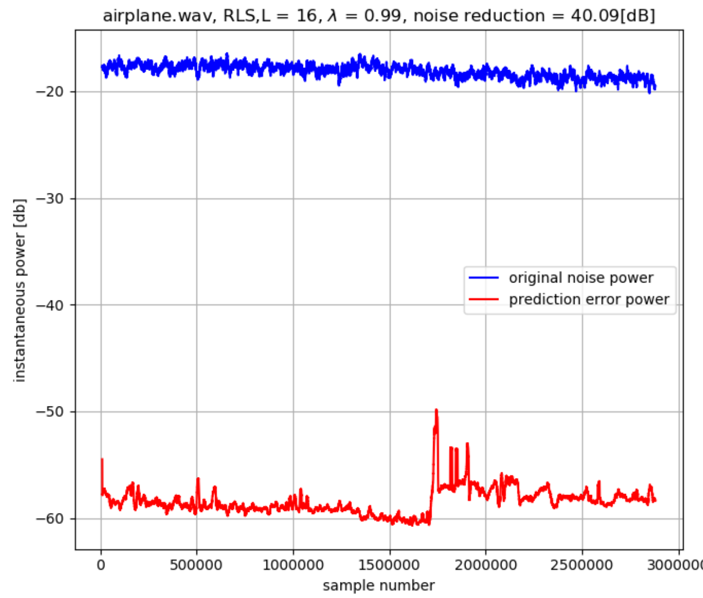
Question 5 - Real Life Signals

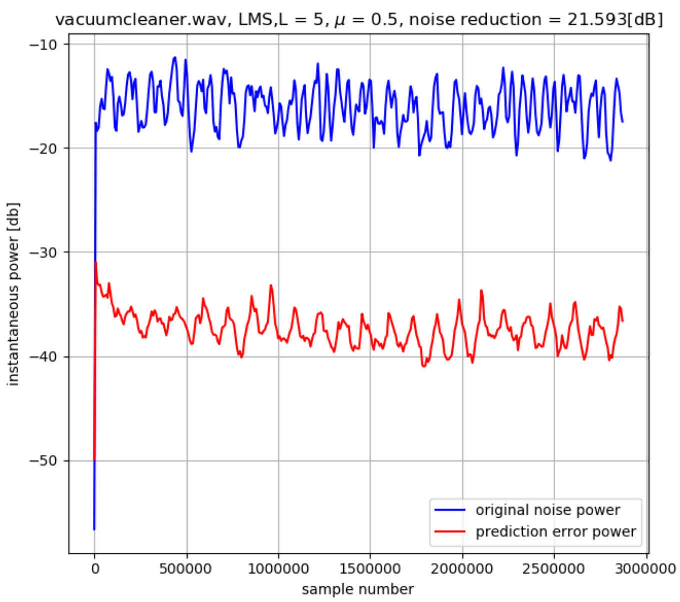
In this question we will use both the LMS and RLS algorithms on the four audio files described in Subsection 1.2. For every file, use both RLS and LMS, and several settings of the algorithm parameters (L and either μ or λ). For any set of parameters you choose, do the following (no need to submit the outcome of Sections (a)-(d). The submission does need to include the answers Sections (1)-(2)):

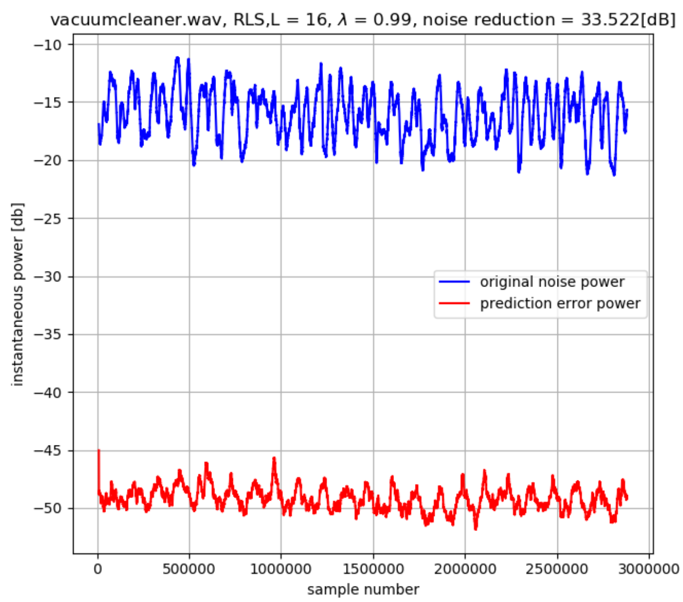
1. For each of the four audio files, include a figure according to the explanation in item (d) above, for at least one good setting of the RLS algorithm and one good setting of the LMS algorithm.

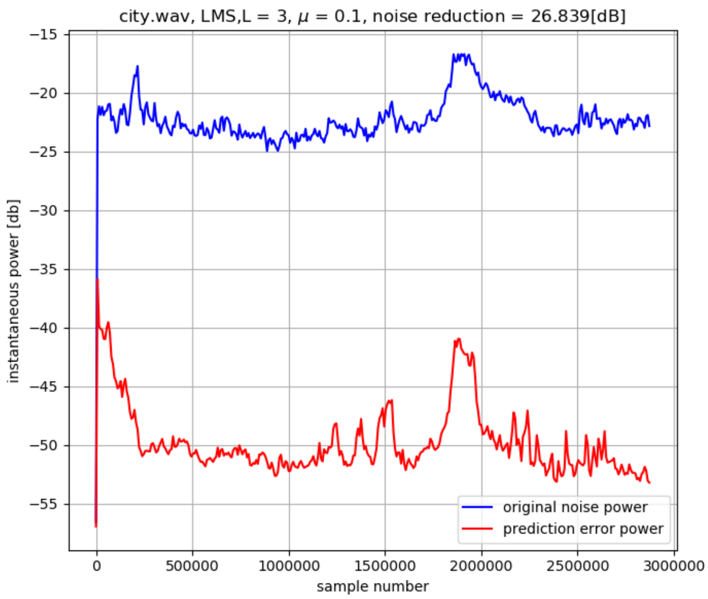
2. Discuss which noise processes are more predictable and which noises processes are less predictable.

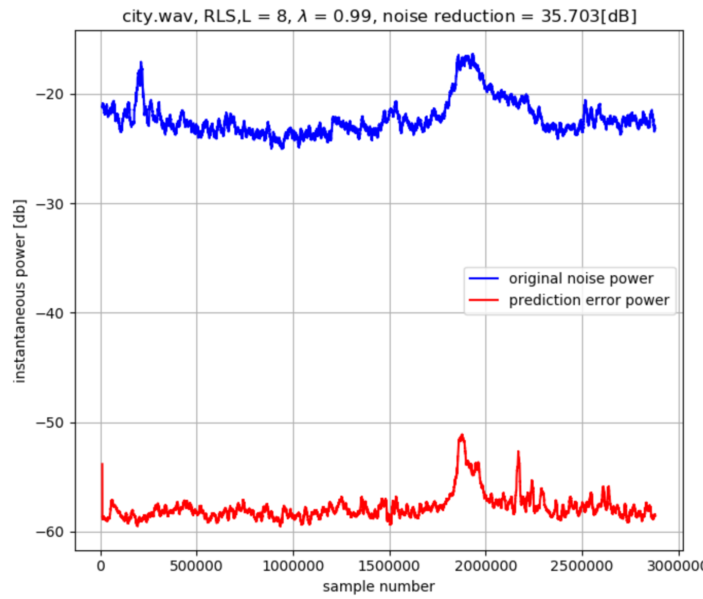


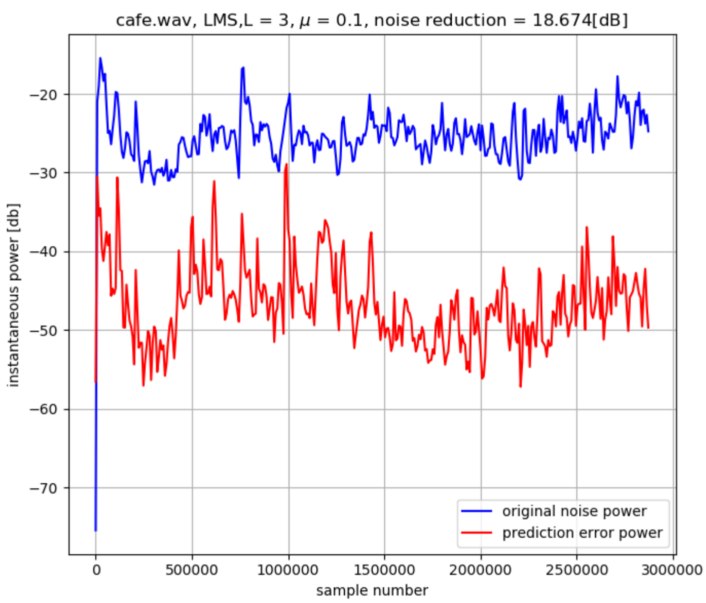


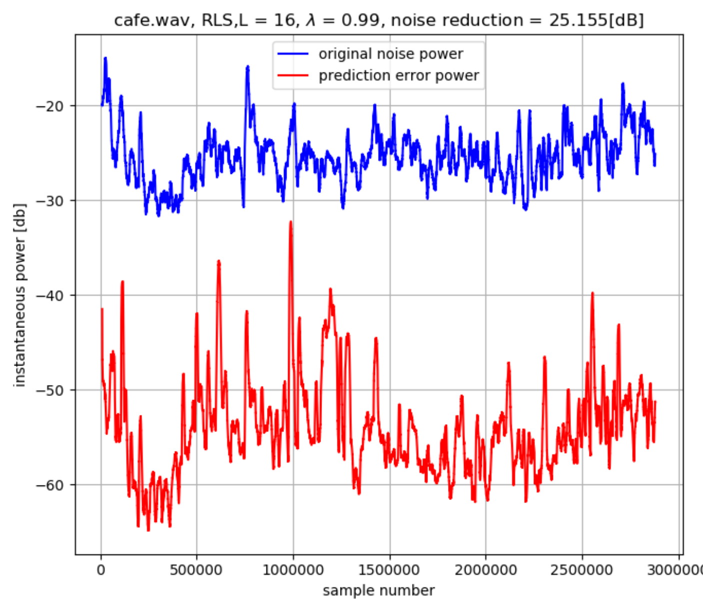












Two of these signals, airplane.wav and vacuumcleaner.wav, appear to be highly periodic (except for some sudden noises) and therefore we would expect to approximate the noise in high precision, and it appears so.

The two other files, and particularly the café file, are less periodic, hence we would expect the RLS algorithm to preform noticeably better on those, as it does.

Listening to the error signal, we may hear the convergence time of these algorithms on different parameters

In general, RLS showed better performance as it does not assume anything on the statistics of the noise.