Adaptive Signal Processing
Assignment Report

Abhishek Sekar
Roll No: EE18B067

December 4, 2020

1 Problem 2

1.1 Introduction and Theory

1.1.1 Objective

To illustrate the construction of learning curves by looking at an example of channel estimation.

1.1.2 Theory

- **Channel Estimation:** The characteristic impulse response of a channel is very useful as it determines the input-output relationship of that channel.
  We have a finite impulse response channel with input u(i) and output d(i) being zero mean sequences and the characteristic finite impulse response being c. We wish to estimate c.
  The output d(i) can be modelled in terms of the input in the below manner:
  $$d(i) = u_i c + v(i) \tag{1}$$
  $$u_i = [u(i), u(i-1), ..., u(i-M+1)]$$
  Where, M is the number of taps in the channel,u_i is a vector holding the previous M values(i inclusive) of the input from the i^{th} state,v is a vector that models a zero mean noise sequence that is uncorrelated with u_i.
  Since we don't know c or the statistics of the input and the output, we can try estimating c by means of stochastic gradient algorithms,which is what we're doing in this problem.

- **Stochastic Gradient Algorithms:** This class of algorithms implement an altered gradient descent algorithm where the gradient and the hessian are approximated using the training data. We assume that the statistics change slowly with time.We implement three such algorithms for estimating the channel, namely LMS,ϵ -NLMS ,RLS.w is what we're using to estimate c with an initial guess w_{-1}.

LMS: This algorithm is a variant of the simple gradient descent algorithm where the approximation is instantaneous.
The i^{th} recursion:

$$w_i = w_{i-1} + \mu u_i^H [d(i) - u_i w_{i-1}] \tag{2}$$

ϵ **-NLMS:** This algorithm is a variant of Newton's method algorithm with regularization ϵ where the approximation is instantaneous.
The i^{th} recursion:

$$w_i = w_{i-1} + \frac{\mu}{\epsilon + ||u_i||^2} u_i^H [d(i) - u_i w_{i-1}] \tag{3}$$

RLS: This algorithm is a variant of Newton's method algorithm with regularization ϵ where the approximation is a mix of instantaneous and multiple observations(weighted using λ for R_u).
The i^{th} recursion:

$$w_i = w_{i-1} + P_i u_i^H [d(i) - u_i w_{i-1}] \tag{4}$$

$$P_i = \frac{1}{\lambda}[P_{i-1} - \frac{\frac{1}{\lambda} P_{i-1} u_i^H u_i P_{i-1}}{1 + \frac{1}{\lambda} u_i P_{i-1} u_i^H}] \tag{5}$$

$$P_{-1} = \frac{1}{\epsilon} I$$

- **Ensemble Average Learning Curves:**These are the average error curves that we obtain after repeating the given experiment several times. Mathematically, the error e(i) at i^{th} iteration is given as

  $$e(i) = d(i) - u_i w_{i-1} \tag{6}$$

  Looking at the cost function,

  $$J(i) = |e(i)^2| \tag{7}$$

  Repeating over L experiments,

  $$J_{avg}(i) = \frac{1}{L} \sum_{j=1}^{j=L} |e_j(i)^2| \tag{8}$$

  Plotting J_{avg} against the iterations give the ensemble average learning curves.

1.1.3 Problem Statement

A channel estimation problem with the following specifications.

- V is a white additive noise sequence with a variance of 0.01.
- 600 u(i) and d(i) pairs are generated using a 4 tap FIR filter.
- The filter's finite impulse response, c= $[1, 0.5, -1, 2]^T$ shown below in the two graphs.

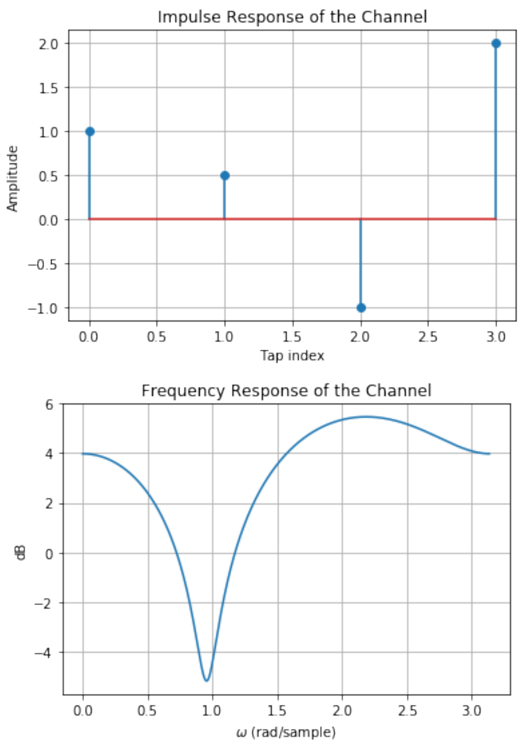

Figure 1: Stem and Frequency plots of c

- We use these input pairs to model our stochastic gradient algorithm with the 'initial guess' $w_{-1} = 0$
- Other algorithm specific values are given below
  - μ_{lms}=0.01
  - μ_{nlms}=0.2
  - $\epsilon_{nlms,rls}$=0.001

- λ_{rls}= 0.995

- We run this set of experiments 300 times and then plot the respective ensemble average learning curves for each algorithm and compare them.

1.2 The Plot

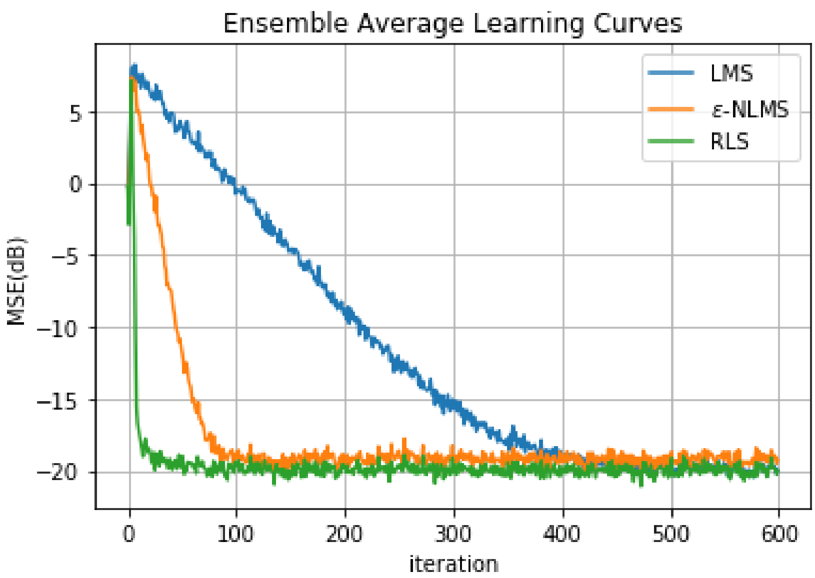

Figure 2: Ensemble averaging curves obtained by averaging over 300 experiments

1.3 Results,Observations and Inferences

- We observe from the graph that,the final errors in the ensemble average learning curves are very small,roughly in the order of 10^{-2}. So we can say that for the given specifications,all the algorithms do a good job at estimating the channel impulse response c.

- We also see that with the increase in the number of iterations, the approximation becomes even better which confirms what we expected in theory.

- If we look closer,we observe that towards the end,in the last few iterations,the MSE obtained by ϵ -NLMS seems to be the highest followed by the LMS and RLS variants. We can verify that this is indeed the case by taking an average of the MSE(J_{avg}) over the last 100 iterations Doing this yields the following values(rounded to 5 decimal places) for the different algorithms.This is also what we'd expect in theory given the expressions for EMSE(Excess MSE).

  - LMS : 0.01032
  - ϵ -NLMS :0.01207
  - RLS: 0.01007

- Looking at the values given above we can verify that the MSE converges to 0.01 which is the variance of the noise or the minimum MSE, J_{min}.

- By computing the theoretical MSE using the expressions for EMSE.

$$J_{theory} = J_{min} + EMSE \tag{9}$$

$$J_{min} = \sigma_v^2$$

  We observe that the theoretical MSE values are significantly(EMSE approx. 10^{-5}) lower than the values obtained above.This tells us that for matching the theoretical results, we have to run the algorithms for a lot more iterations.

- We observe that the convergence trends are in the fashion, LMS being the slowest followed by ϵ -NLMS and RLS as far as the rough number of iterations for convergence are concerned. This is what we'd expect in theory as well, as LMS is built on a basic gradient descent algorithm which is much slower than the Newton's Method algorithm which is implemented with regularization in ϵ -NLMS and RLS. However,ϵ -NLMS uses instantaneous approximation for the hessian and the gradient while RLS uses a mix of instantaneous approximation and a weighted approximation using more observations (for R_u) and due to this we'd expect RLS to perform better.

- This also brings to light the trade off between computational complexity and performance since the trends for that are the exact opposite when compared to the convergence speeds.This trend is also observed in practice when you run the algorithms separately and use the tqdm tool in python to observe the time it takes for them to complete.

2 Problem 3

2.1 Introduction and Theory

2.1.1 Objective

Comparing how the steady state MSE of a 10 tap LMS filter (in a channel estimation problem) fares against the theoretically predicted one for different choices of step sizes and signal conditions.

2.1.2 Theory

- **EMSE:** EMSE is the excess mean square error that persists as the iterations of the algorithm tend to infinity.Calculating it requires the following assumptions.
  - There exists a w^0 such that, d(i) = u_i w^0 + v(i).
  - v(i) is iid and has variance,$\sigma_v^2 = E|v(i)|^2$.
  - v(i) is independent of u_j for all (i,j).
  - The initial guess w_{-1} can be chosen independently.
  - R_u is a positive definite matrix.
  - d,v,u are all zero mean random variables.

  For LMS,these assumptions hold here and the EMSE turns out to be,

  $$\zeta^{LMS} = \frac{\mu}{2}(E||u_i||^2|e_a(i)|^2 + \sigma_v^2 Tr(R_u)) \tag{10}$$

  Where ζ^{LMS} is the EMSE and e_a(i) is the a-priori error corresponding to iteration i.

- **Small Step Size approximation:** This is done to simplify the EMSE calculations and to get around calculating the expectation. The approximation assumes, that for a small enough μ

  $$E||u_i||^2|e_a(i)|^2 << \sigma_v^2 Tr(R_u)$$

  Therefore, the expression for ζ^{LMS} now becomes,

  $$\zeta^{LMS} = \frac{\mu}{2}(\sigma_v^2 Tr(R_u)) \tag{11}$$

- **Separation principle:**We get around calculating the expectation the hard way by coming up with the separation assumption which states that at steady state, norm$(u_i)^2$ is independent of e_a(i), thereby e(i) as well. This assumption converts the expectations of products into a product of expectations which is easier to evaluate. Therefore, the expression for ζ^{LMS} now becomes,

$$\zeta^{LMS} = \frac{\mu\sigma_v^2 Tr(R_u)}{2 - \mu Tr(R_u)} \tag{12}$$

- **Regressors with shift structure:** Shift structure implies that norm$(u_i)^2$ is kept constant since whatever u(i-M+1) is being removed from the vector u_i comes along as the next u(i).This is generated by feeding the correlated data u(i) into a tapped delay line. This correlated data is obtained by filtering a unit variance iid Gaussian random process s(i) through a first order auto regressive filter whose z-transform transfer function is given below.

$$\frac{\sqrt{1-a^2}}{1-az^{-1}}$$

A frequency plot of the transfer function looks like the below for a=0.8.

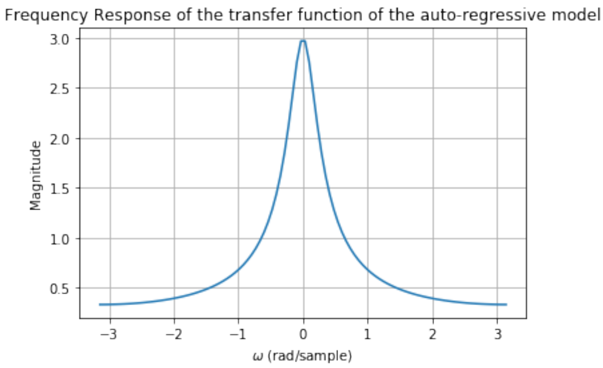

Figure 3: Frequency plot of the auto regressive filter with the given transfer function

The auto-correlation of the generated process is shown below.

$$r(k) = E[u(i) * u(i-k)] = a^{|k|}$$

Therefore, the covariance matrix R_u can then be worked out to an MxM Toeplitz matrix with entries,

$$R_u[i,j] = a^{|i-j|}$$

2.1.3 Problem Statement

The problem specifications are very similar to the previous problem with one major addition.

- V is a white additive noise sequence with a variance of 0.001.
- $4\text{x}10^5$ u(i) and d(i) pairs are generated using a 10 tap FIR filter.
- The filter's finite impulse response is left for us to choose and I've chosen c= $[2, 4, 3, 0.6, 1.4, 0.15, 0.3, 0.75, 0.9, 1]^T$ shown below in the two graphs.

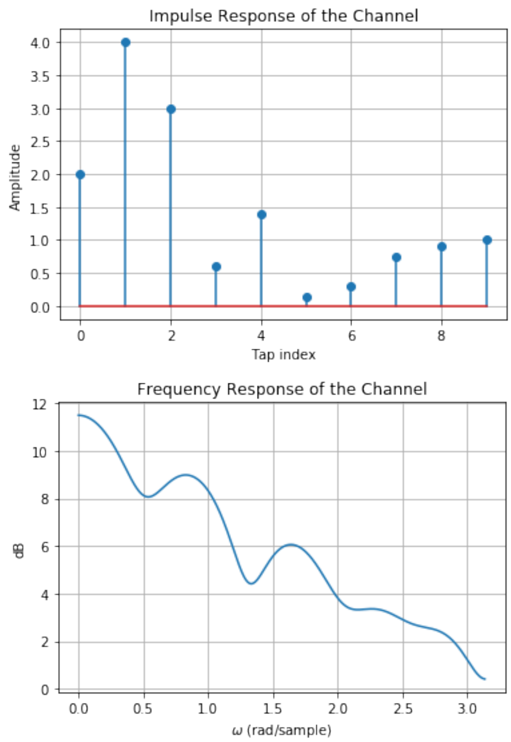

Figure 4: Stem and Frequency plots of c

- We use these input pairs to model our LMS algorithm with the 'initial guess' $w_{-1} = 0$

- In problem 16.2, we will be generating the regressors by means of a diagonal R_u with an eigenvalue spread of 5 as shown below.

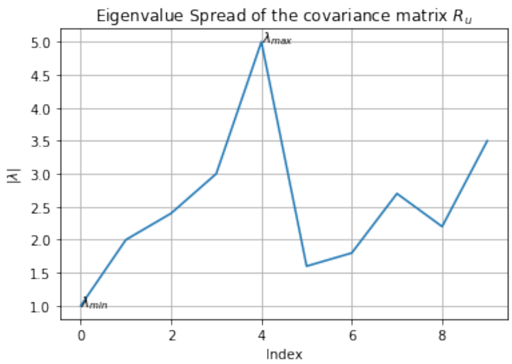

Figure 5: Eigenvalue spread of R_u

- In problem 16.3, we will follow the process of generating shift structure regressors discussed in theory with a=0.8.
- We then compute the ensemble average learning curves by repeating this 100 times and then take the average of the last 5000 values as the experimental MSE.
- Now,coming to the addition,we would be performing this experiment over 20 values of μ, with 10 between μ=10^{-4} and 10^{-3} and the other 10 between μ=10^{-3} and 10^{-2}. For each μ, we find the theoretical and experimental MSE and plot them vs μ and try comparing and analyzing them.

2.2 The Plots

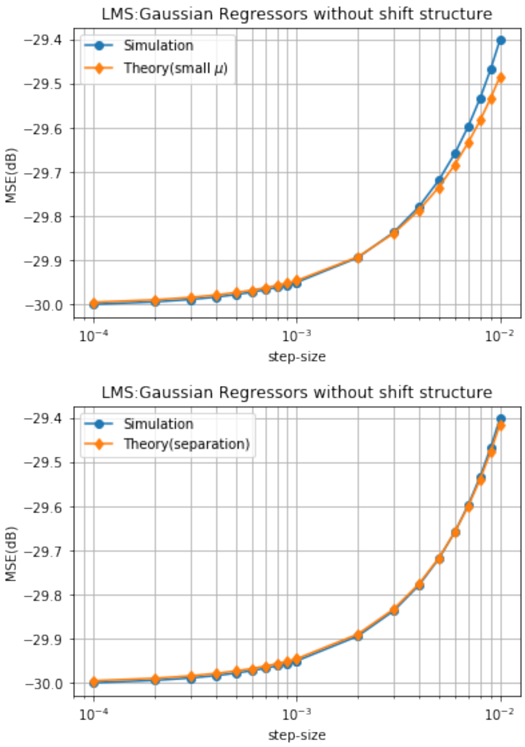

Figure 6: Plots corresponding to problem 16.2

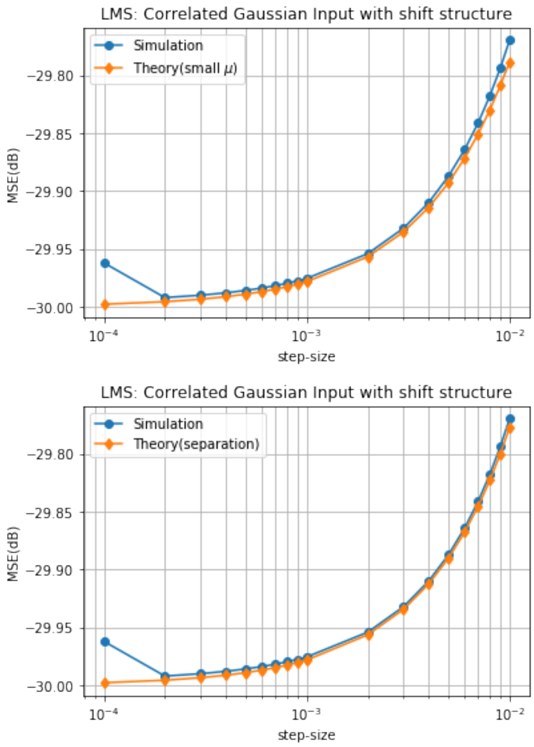

Figure 7: Plots corresponding to problem 16.3

2.3 Results,Observations and Inferences

- We observe that for a smaller step size, the MSE is lower.This resembles what we'd expect in theory as the LMS algorithm provides the best estimate when the step size is very small.

- We observe that the MSE values all are around (-30dB) which is the variance of the noise or J_{min}

- On the whole, the theoretical values of MSE perform well since the curves for theoretical and experimental MSE almost coincide in the graphs. We can observe that the small μ approximation doesn't hold as well for larger values of μ around 5x10^{-3}. This is because the approximation doesn't hold good for higher values of μ.

- On the other hand, the separation approximation works pretty good over the range of μ and this confirms that norm$(u_i)^2$ behaves as if it's independent with e_a(i) over the range of μ.

- If we compare the plots of 16.2 and 16.3, we observe the queer behaviour of the latter at small values of μ. There seems to be a minute variation between the simulation and the experimental curves, a difference of approximately -0.05dB.This can be explained by an analysis of the spread of the covariance matrix. By a python code, we can find that the Toeplitz matrix Ru, has

- λ_{max}= 5.50186 (rounded to 5 decimal places)
- λ_{min}= 0.1138
- Therefore the spread is their ratio, s = 48.347.(rounded to 3 decimal places)

The spread of this matrix is large compared to the spread of R_u in problem 16.2. A higher spread implies that the convergence speed will take a hit and therefore if we increase the number of iterations, we will be able to observe a significant match between experimental and theory behaviour as in the regressor without shift structure case.
I tried this for N=6x10^5 iterations (I couldn't go higher due to space limitations on Jupyter Notebook) and the graph below confirms the theory discussed above.

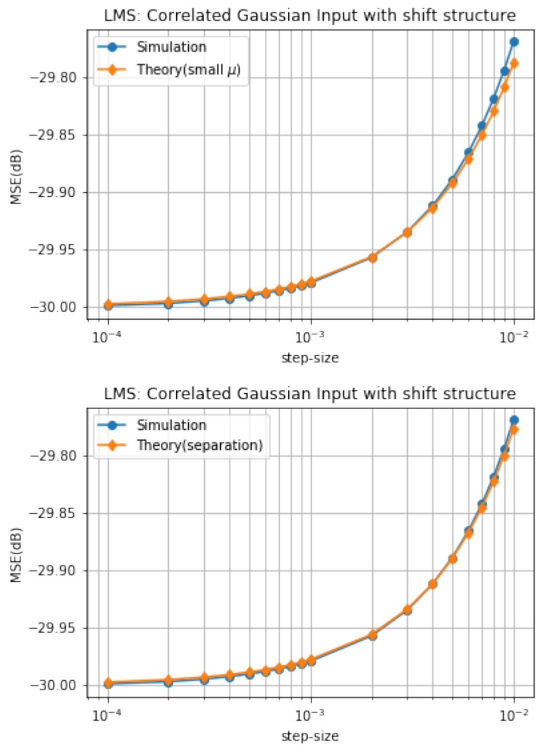

Figure 8: Plots corresponding to problem 16.3 but with N= 6x10^5

- Lastly, we observe that the the regressor with shift structure has a better behaviour with the theoretical curves(especially the separation one) when compared to the ones without.This is because the approximation we make in the separation case becomes exact for the regressor with shift structure since it has constant euclidean norm and therefore will always be independent with the a-priori error.