# EEL 6814
# Neural Networks for Signal Processing
# Homework 1-Adaptive Linear Systems

## I. Problem 1

The first problem is system identification of a nonlinear plant using a linear model. In some cases a linear model can capture the characteristics of a nonlinear plant; however, it depends on how well a projection from the input space can map to the desired system.

The system model given is

$$y(n+1) = \left(\frac{y(n)}{1+y^2(n)} + 1\right) \sin^2(u(n)). \tag{1}$$

The input is defined as independent samples from a standard normal distribution, i.e. the most convenient input defined for system identification in linear systems.

The difficulty in modeling this as a non-linear system is two-fold: first, the output is a product of the input and the previous output, meaning the impulse response is always zero, and the $sin^2(u(n))$ term cannot be approximated by a linear solution. Between (-1,1), where 68% of the input values fall, it may be approximated by the $|u(n)|$ which the best linear approximation is a constant. The trigonometric function also bounds the output to at most 1.5, see Fig. 1. As mentioned the impulse response is always zero, but the response to constant inputs of various amplitudes readily converges, Fig. 2. However, as the input is centered around zero, the output is constantly driven low, with few walks above the mean. The mean of the output was found empirically to be near 0.5667, the output was zero-centered before attempts at system identification, since the input had no mean a filter without an additional bias term performed worse since they were not able to even capture the mean.

### A. Wiener Filter Solution

As a result of the nature of the this system, the best FIR linear solution was the trivial solution of all zero weights. Both the Wiener and LMS readily converged to this value.

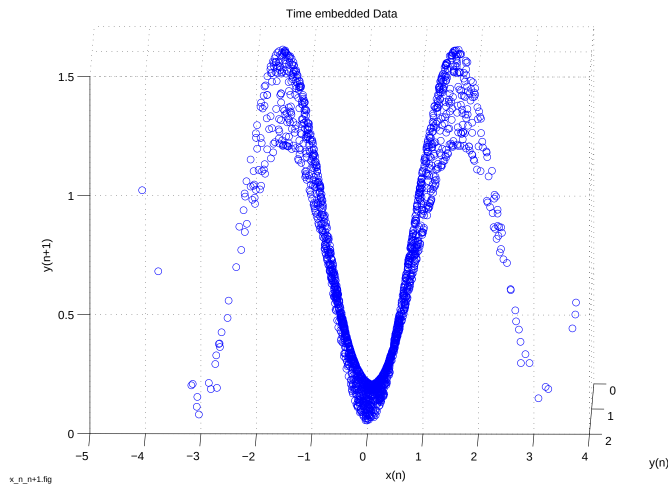



Fig. 1. The output $y(n+1)$ given the input.

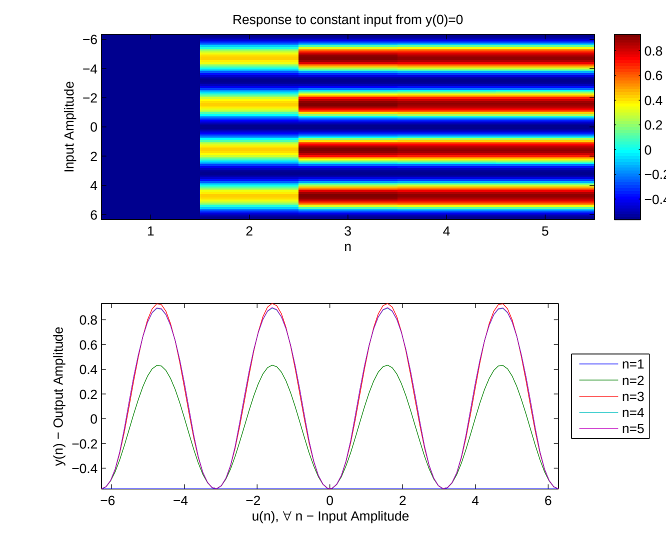



Fig. 2. The transient response to constant input values over a range of input.

### B. LMS Solution

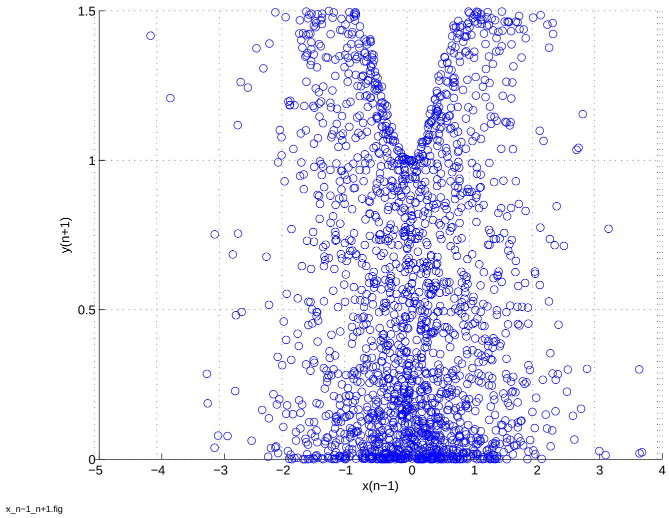



Fig. 3. The output $y(n+1)$ given the previous input.

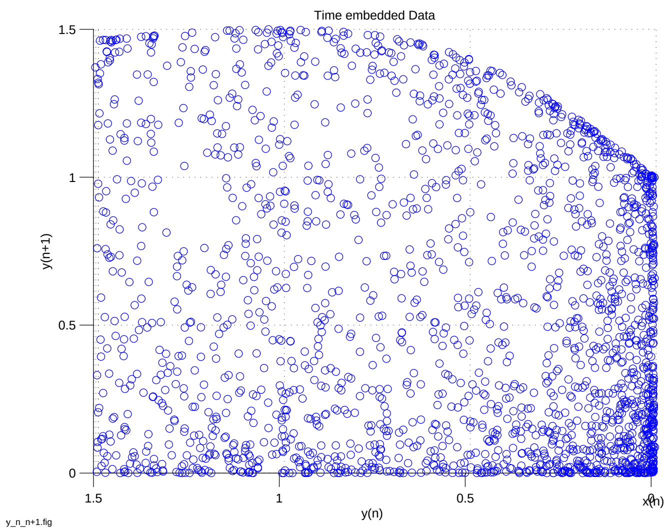



Fig. 4. The output $y(n+1)$ is dependent on the previous output $y(n)$ but not in a linear fashion.

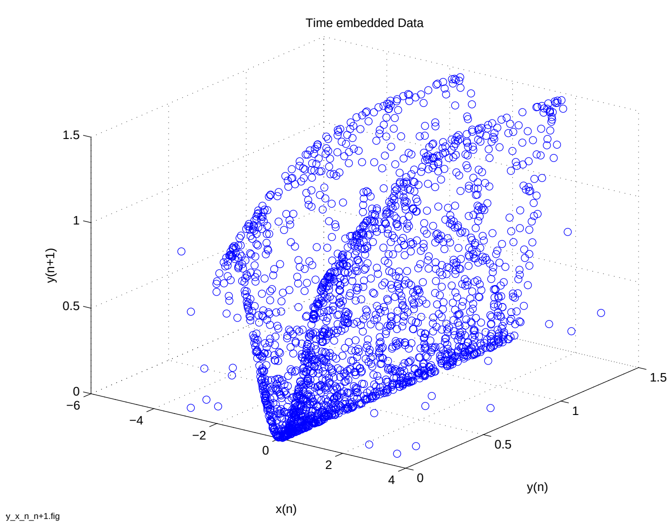



Fig. 5. A time embedding space with the input and previous values.

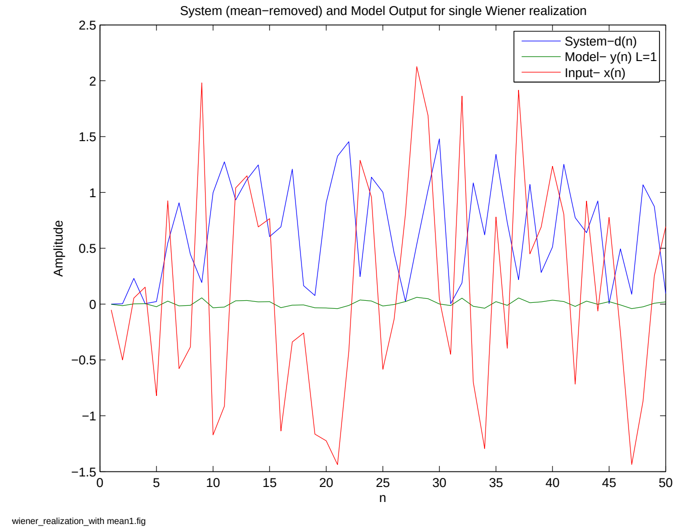



Fig. 6. Realization with mean.

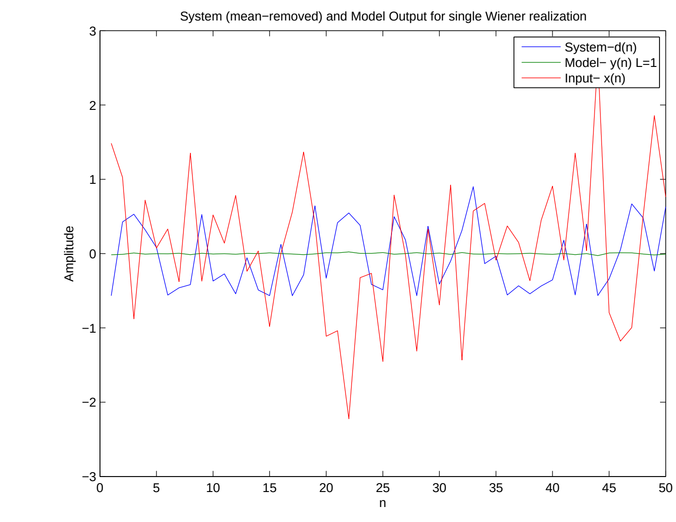



Fig. 7. Realization with mean removed.

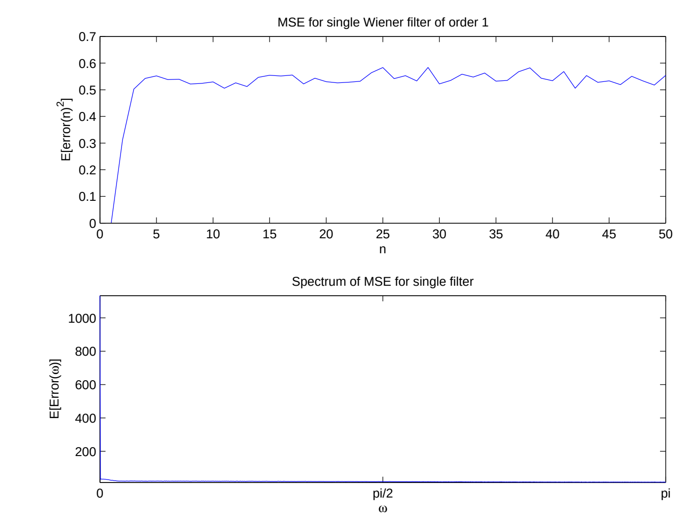



Fig. 8. Performance with mean, notice huge DC term in error.

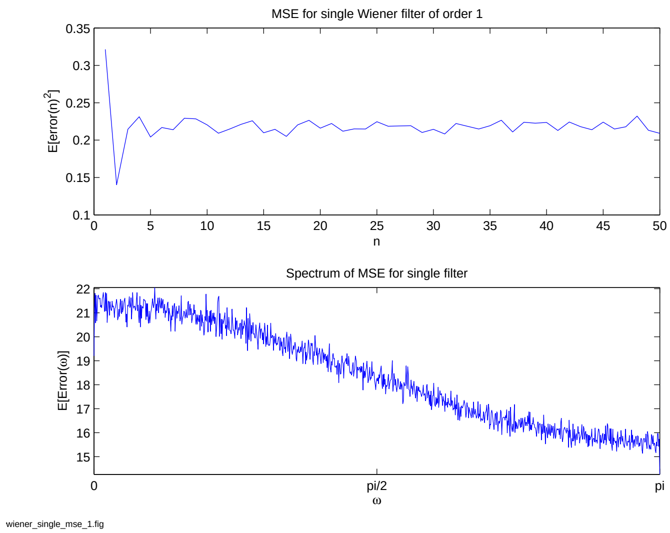



Fig. 9. Performance without mean.

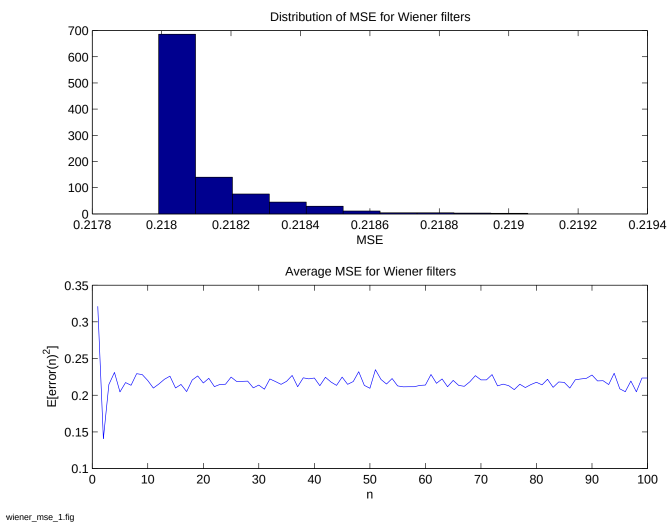



Fig. 10. Performance without mean with order 1.

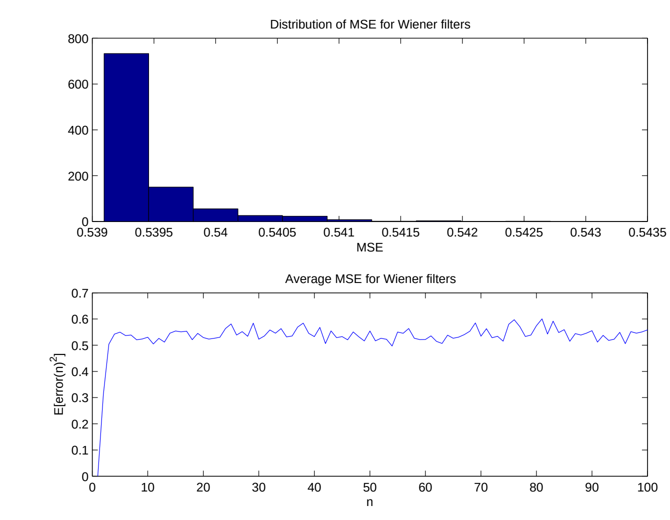



Fig. 11. Performance with mean with order 1.

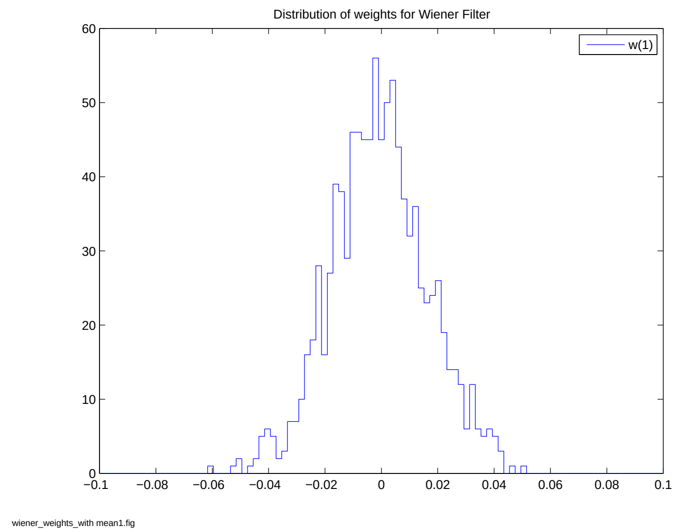



Fig. 12. Single tap weight distribution

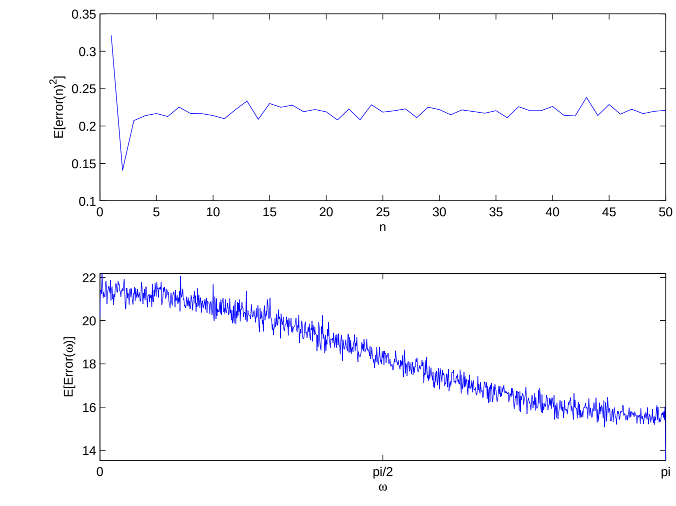



Fig. 13. Performance of 2 tap with stepsize of 0.0001.

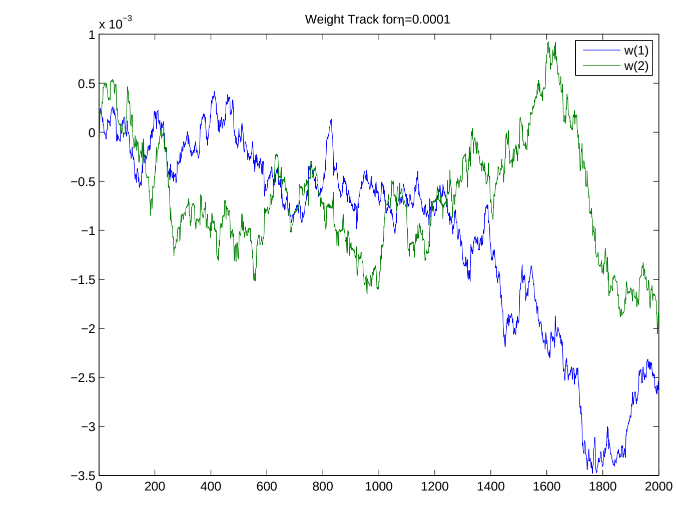



Fig. 14. LMS Weight track of 2 tap with stepsize of 0.0001.

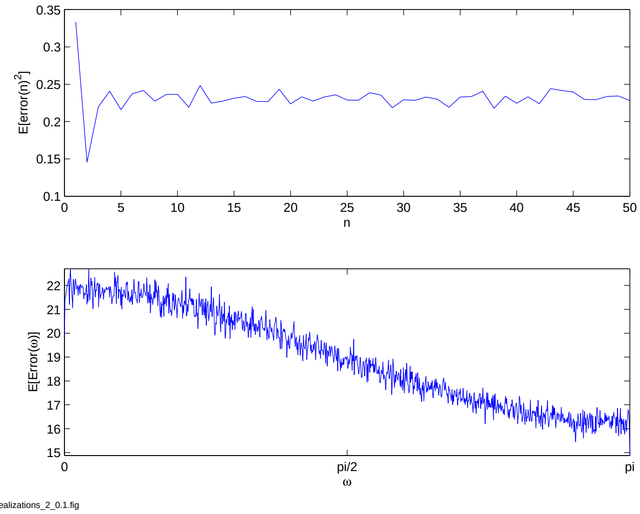



Fig. 15. Performance of 2 tap LMS filter with stepsize of 0.1.

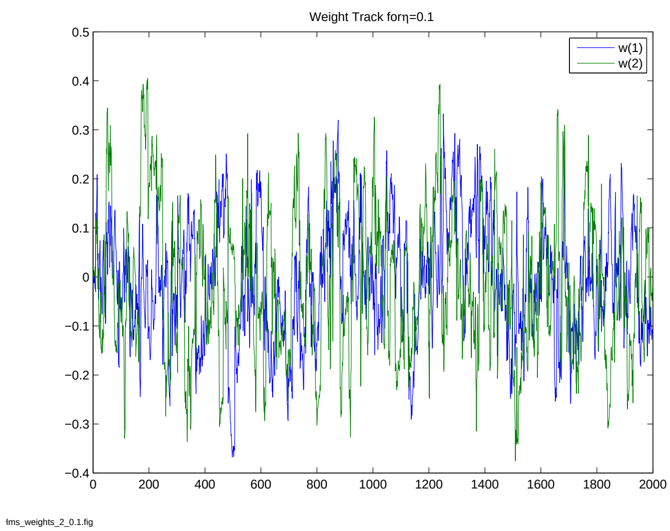



Fig. 16. LMS Weight track of 2 tap with stepsize of 0.1.

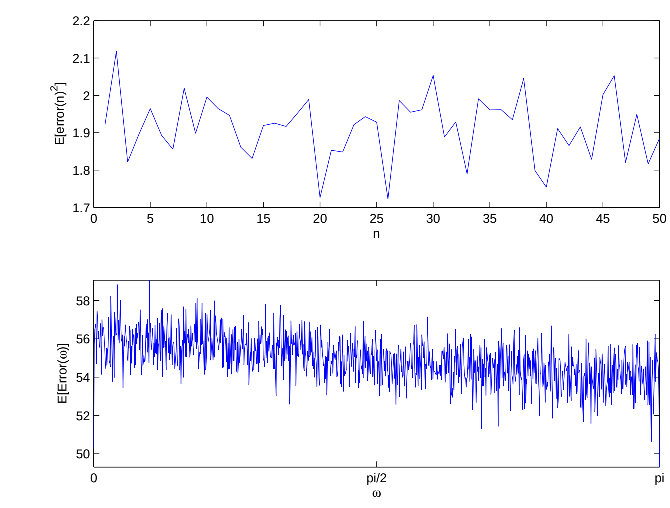



Fig. 17. Divergence of 2 tap with stepsize of 0.5.

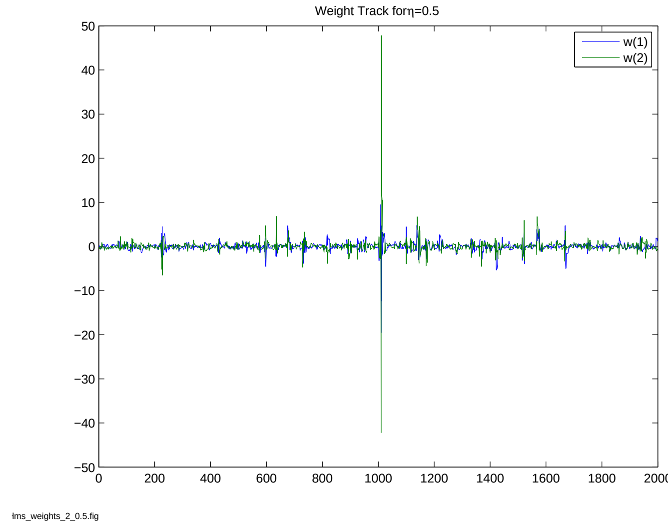



Fig. 18. Weight track of 2 tap with stepsize of 0.5. Diverges, but remains near zero.

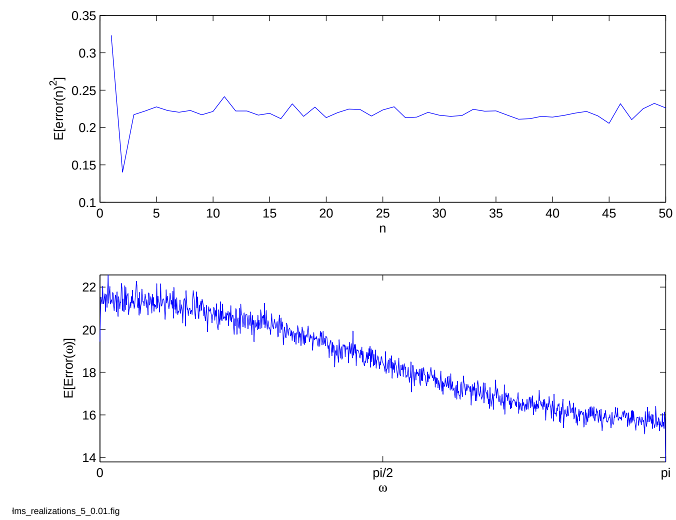



Fig. 19. Performance of 5 tap with stepsize of 0.01.

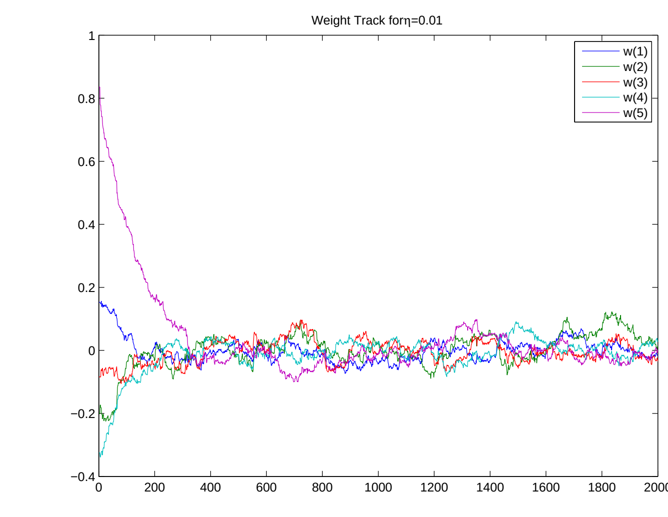



Fig. 20. Weight track of 5 tap with stepsize of 0.01 from random initial weights. Rattles around trivial solution.

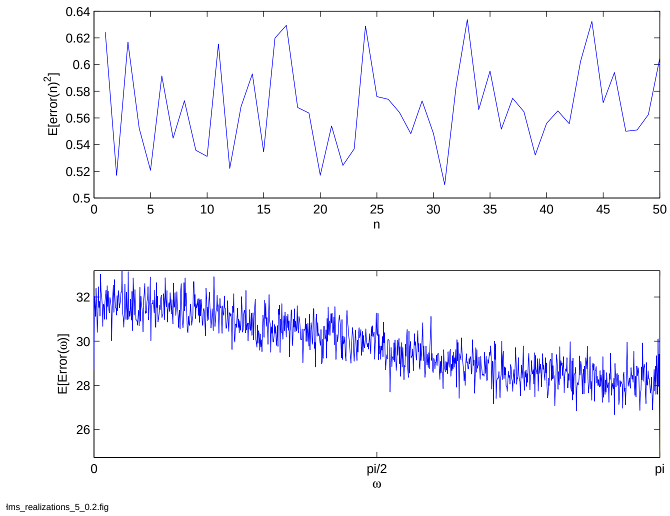



Fig. 21. Divergence of 5 tap with stepsize of 0.2.

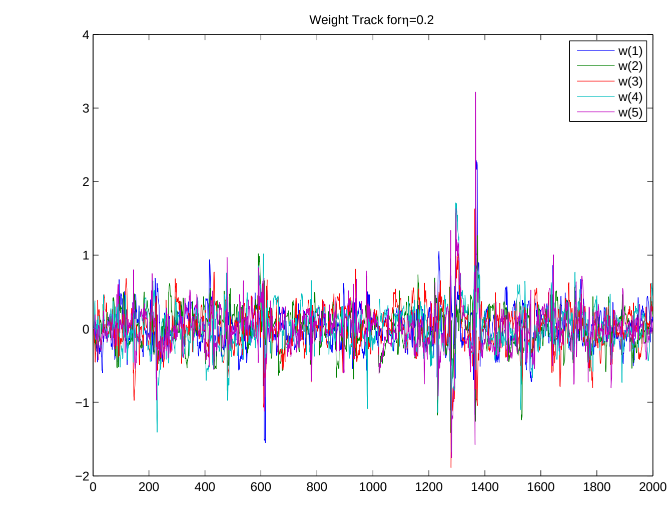



Fig. 22. Weight track of 5 tap with stepsize of 0.2. Lack of convergence, but centered around zero.

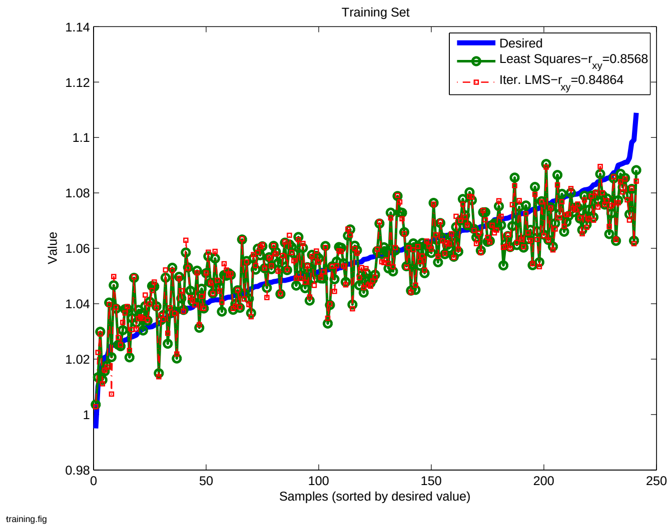



Fig. 23. Training set performance for both Least Squares and Batch LMS.

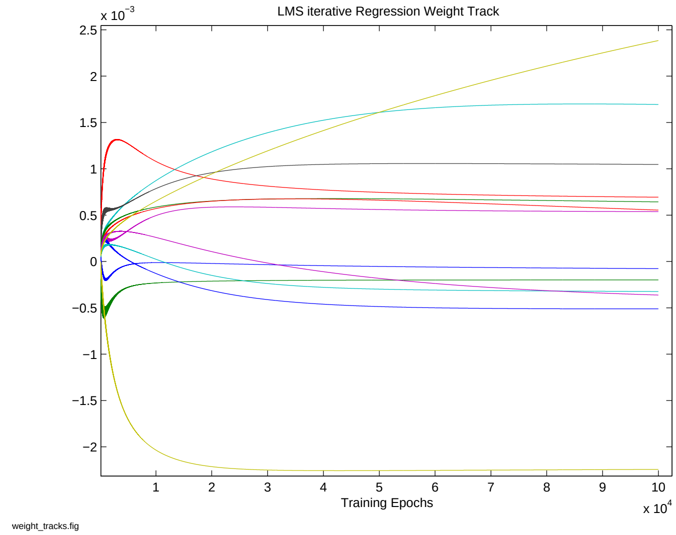



Fig. 24. Weight tracks for multiple training epochs of the LMS solution.

## II. Part 2-Linear Regression

The goal of this part was to fit a linear model for a 13-dimension input space to a scalar value pertaining to body fat measurements. No preprocessing of the data was performed,and a least squares fit was performed with a bias term added. For the training set the LS fit had a correlation coefficient of 0.857, Fig. 23. Then the solution for the training set was found again by iterative least squares (LMS) in batch mode. Due, to the eigenvalues ranging from 56 to $1.85 \times 10^7$ the eigenvalue spread is above $10^5$ and there needed to be many epochs of training to converge to the least squares solutions and a stepsize of only $2.6 \times 10-5$ (a much more liberal value than the theoretical stability bound of $1 \times 10-7$), Fig. 24. However, the LMS converged to within the misadjustment of the LS, see Fig. 25 for weights and Fig. 26 for bias. The training set performance had a correlation coefficient of 0.847, Fig. 23.

The test set performance was not as successful, pointing to overfitting, Fig. 27.

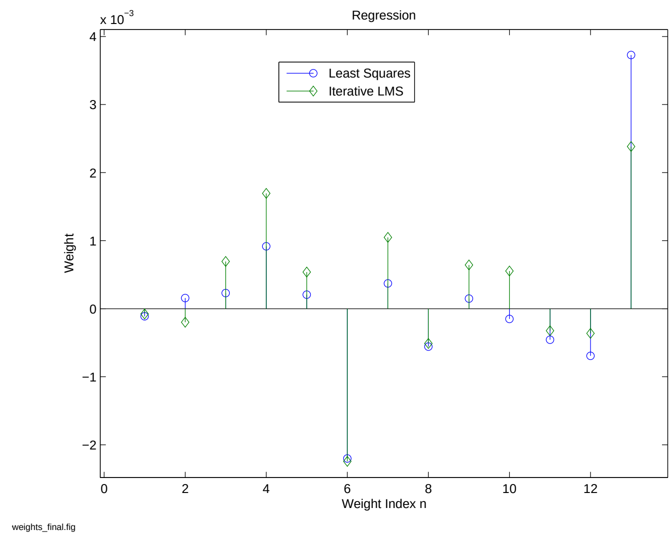



Fig. 25. Final weights (excluding DC bias) for least squares and LMS solution.

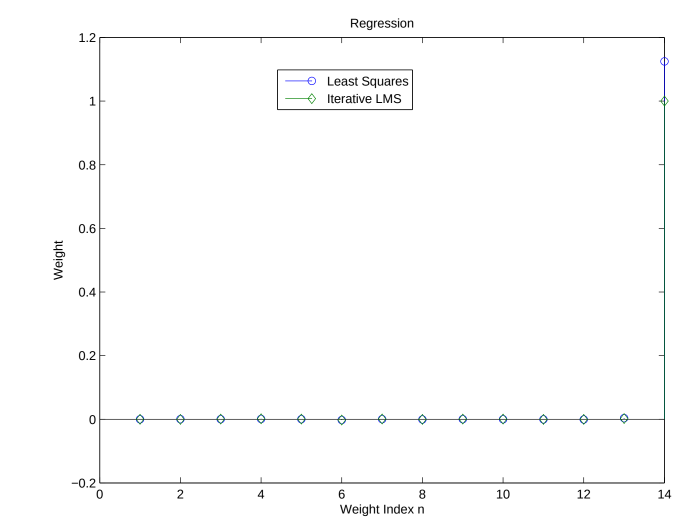
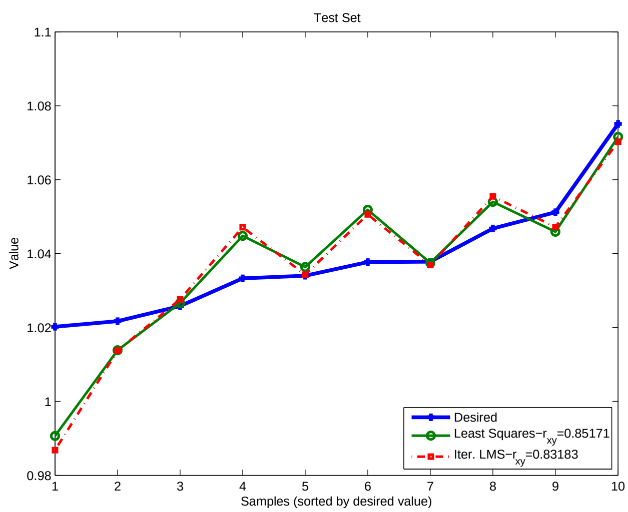



Fig. 26. Weights including DC bias for least squares and LMS solution.




test.fig

Fig. 27. Test set performance for Least Squares and LMS solution.