ECE 792-41 Statistical Foundations for Signal Processing and Machine Learning

Project 2: Linear Prediction and Frequency Estimation

The project should be completed individually. You must use the IEEE Transactions template (Word or LaTex) for writing your report. Your submission must be a concise write-up of the results and findings. It should include figures and tables, and the descriptions and discussions of them. There is no need to include detailed background reviews that mainly repeat the techniques taught in lectures. You are allowed to use off-the-shelf packages. You should prepare a README file (in txt or html) to describe your files for source codes, audio results, and report. Details can be found in the submission guideline.

You may use MATLAB or other programming languages such as Python, R, and C++ to do your project. You are allowed to use only primitive functions such as convolution, FFT, etc. You should implement your own functions that are closely related to the course materials such as autocorrelation, Levinson-Durbin recursion, etc. You may use more sophisticated functions only for verification purposes. You are not allowed to use or modify upon any speech encoder/decoder built by other people (such as those in the Internet and textbook CDs). Any resources used in your work should be cited at the end of your report.

You should sign the **Honor Pledge** at the beginning of the report: "*I pledge in my honor that I have not given or received any unauthorized assistance on this report*".

---

1. **[20 points] Simulating wide-sense stationary AR signals:** Implement a function capable of generating a simulated AR(p) signal $\{u(n)\}$. Let $p = 5$. Choose a set of parameters so that the AR process will be stable, and fix the parameters throughout this problem.

(1) In Problem 1 of Homework 5, we have shown that the mean and variance of a simulated random variable $u(n)$ converges to the mean and variance of the true AR process. Use this theoretical result to design a test that allows you determine how many elements need to be thrown away for your chosen AR parameters so that the remaining simulated values form a wide-sense stationary AR process. Modify your AR signal generation function accordingly for future use. (Hint: You may want to repeatedly generate AR signals using independent innovations. Calculate the sample variance at every time index.)

(2) Calculate the value of the theoretical autocorrelation sequence $\{r(k), k = 0, \ldots, 5\}$ for your chosen parameters. (Hint: Express the Yule-Walker equations in the matrix-vector form about the coefficient vector $\mathbf{a} = [a_1, \ldots, a_5]^T$, and then reformulate it into a linear system about an unknown vector $\mathbf{r} = [r(0), \ldots, r(5)]^T$.)

(3) Use the correlation ergodicity property to estimate the *unbiased* autocorrelation $\hat{r}(k)$. Plot its mean squared error as a function of the sequence length.

(4) Use the correlation ergodicity property to estimate a *biased* autocorrelation $\tilde{r}(k)$. Plot in one figure its mean squared error, bias squared, and sample variance as a function of the sequence length.

2. [20 points] Linear Prediction and Levinson-Durbin Recursion:

(1) Derive from scratch the normal equation (N.E.) and augmented N.E. for the order-5 forward linear prediction.

(2) Derive from scratch the Levinson-Durbin Recursion for the augmented N.E.

(3) Replace the theoretical quantities in the N.E. by their *biased* autocorrelation estimates, and solve the forward prediction coefficient vector by direct inverse. Repeat such procedure with different random seeds 100,000 times, and obtain 100,000 estimated forward prediction coefficient vectors. What are the sample mean and the

sample variance-covariance matrix for these 100,000 vectors? What do the diagonal elements and off-diagonal elements of the sample variance-covariance matrix measure, respectively?

(4) Repeat (3) using *unbiased* autocorrelation estimates. Compare the results with those of (3). Explain your observations.

(5) Replace the matrix inverse operation in (3) by Levinson-Durbin Recursion. Compare the results with those of (3). Explain your observations.

3. **[30 points] Analyzing Speech Signals**: This task lets you explore the basic analysis and synthesis of speech signal based on the concept of linear prediction. Some test clips are provided in course webpage, encoded at 8 kHz and 8 bits per sample.

(1) Design an automated procedure to determine the average duration of the speech signal within which the signal is approximately wide-sense stationary. Draw a block diagram or write pseudocode in the report for the designed procedure. We shall call the best average duration the "frame length." You may partition the speech signal into frames of this length in the rest of this problem.

(2) Build a 10^{th}-order linear predictive model and implement a function that can efficiently find the optimal prediction coefficients for a given frame of the speech signal. (You may reuse the functions written for Problem 1.) Show through simulation how much is the difference between the true speech signal and the predicted one from your model. Note that you can use different coefficients for different frames, and you should examine the difference both "objectively" (using some suitable quantitative measures) and "subjectively" (listen to the signals).

(3) Examine variations on the above speech analysis: through simulations of higher and lower order than 10, discuss how the selection of order affects the performance of the prediction.

4. [30 points] Frequency Estimation

4.1. Generate 9 groups of 8-second signals with SNR = −40, −30, …, 0, …, 30, 40 dB, respectively, with sampling frequency at 150 Hz. Each group of signals consists of 1,000 realizations of real-valued sinusoid with f = 60 Hz and random phase buried in the white Gaussian noise.

(a) Implement the following spectrum/frequency estimation methods, periodogram, averaged periodogram, MVSE and MUSIC.

(b) Write a peaking searching algorithm capable of outputting from the estimated spectrums generated in (a) the index of the peak, i, and its peak value $S(i)$, as well as the value of the left and right neighbors of the peak, $S(i-1)$ and $S(i+1)$. Derive a deterministic function that takes as input $\{i, S(i-1), S(i), S(i+1)\}$ and output an adjusted index of peak $i^* \in (i-1, i+1)$ that is the peak position of a quadratic curve passing through $S(i-1)$, $S(i)$, and $S(i+1)$. Use this deterministic function to improve the accuracy of all the algorithms in (a).

(c) Examine how the change of parameter values such as the length of window, the number of segments, the size of correlation matrix affects the mean squared error (MSE) between the frequency estimates and the ground truth. Is the effect of the change of parameter values the same at different SNR levels? Discuss your findings.

4.2. Generate one-hour long sinusoidal signals of time-varying frequency with frequency changing every T seconds. Assume that the time varying frequency has a nominal value of 60 Hz and its fluctuation follows an AR(1) process with $a_1 = -0.9$ and $\sigma_e = 0.05$ Hz. Assume the sampling frequency of the sinusoidal signal is f_s = 150 Hz, generate separate signals for T = 0.5, 1, 2, 4, 8 seconds. Answer the following questions.

(a) Using the equation for frequency modulation (FM) in the integral form, prove that in the discrete case

$$x(n) = \cos[\phi_0 + 2\pi T_s \sum_{\ell=1}^{n} f(\ell)],$$

where ϕ_0 is a random initial phase, T_s is the sampling period, and $f(\ell)$ is the averaged frequency during ℓth sampling period.

(b) Use the frequency estimation methods in 4.1 to estimate the instantaneous dominating frequency of the generated sinusoidal signals. Plot your results in terms of MSE as a function of *T*. What do you observe from your results? Discuss your findings.

(c) Now add white Gaussian noise at a SNR = −40, −30, …, 0, …, 30, 40 dB to your sinusoidal signal and repeat part (a). What is the effect of noise levels on the frequency estimates? Which method gives you the best frequency estimates? Why?