ELEC-E5521 Speech and language processing methods

Assignment I: Linear prediction
Spring 2022

Each team is given the following speech signals, produced by a male talker, to work on. Data are in the wav format, sampled at 8 kHz.

“male_a.wav”: clean vowel /a/
“male_i.wav”: clean vowel /i/
“male_u.wav”: clean vowel /u/
“male_a_noisy_20dB.wav”: noise-corrupted vowel /a/, SNR 20 dB
“male_a_noisy_10dB.wav”: noise-corrupted vowel /a/, SNR 10 dB

Draw all figures clearly. Avoid color images. Scale figure axes so that the data relevant to the given questions can be clearly seen. Return the assignment (as pdf file), including the MATLAB code that you wrote. You can use MATLAB functions listed on page 2 (or write your own code).

Compute Exercises 1-3 for **all three clean vowels**:

Exercise 1

1.1 Compute LP analysis (with the autocorrelation criterion) using prediction order p = 10, 25 ms frame length, and Hamming window. Insert the analysis frame in the middle of the signal.
1.2 Compute the impulse response of the LP synthesis filter and draw the impulse response for interval 0-25 ms.
1.3 Compute and draw the LP spectrum and FFT spectrum of the signal into the same figure with their energies normalized to the same level (use the FFT length of 1024 and draw the spectra on the logarithmic dB scale).
1.4 Draw normalized autocorrelation functions of the following two signals: original speech signal (over the 25 ms analysis frame) and impulse response of the LP filter. Draw the autocorrelation functions (overlapping in the same figure) for time index values 0-20. Compare these two autocorrelation functions: what is similar and what is different in them?

Exercise 2

2.1 Compute the residual. Draw spectra of the speech signal and residual (on the dB scale) and compare them visually.
2.2 Draw the autocorrelation function of the residual as the third plot. Estimate the fundamental frequency of the speech signal based on the autocorrelation function.
2.3 Compute the normalized residual energy (also known as the prediction gain).

Exercise 3

3.1 Create an excitation signal with duration of 300 ms using an impulse train. Ensure that the impulse interval corresponds to the period length estimated in Exercise 2.

3.2 Synthesize a signal by filtering the impulse train with the LP-filter. Save the signal for later listening (signals are to be returned together with the exercise report).

Exercise 4

This exercise is made **only** for signals “male_a.wav”, “male_a_noisy_20dB.wav”, and “male_a_noisy_10dB.wav”.

4.1 Compute LP analysis as in 1.1 for the clean and noise-corrupted vowels and draw the three LP spectra (on the dB scale) into the same figure with their energies normalized to the same level (FFT length 1024). Describe how noise-corruption affects LP spectrum.

Exercise 5

This exercise is made **only** for signal “male_a_8kHz.wav”.

5.1 Using the Durbin method, compute LP inverse filter (order p = 10) in both direct form and lattice structure. Return the filter coefficients for both structures in a table. Prove that the obtained solutions are equal by drawing their amplitude spectra (on the dB scale).

5.2 Quantize the reflection coefficients using 8, 6, and 4 bits per coefficient. Use uniform quantization in a form that guarantees the quantized filter to be of minimum phase. Return the original reflection coefficients and the quantized coefficients in a table.

5.3 Draw the amplitude response of the original and quantized LP filters. Draw all responses into the same figure.

Exercise 6

This exercise is made **only** for signal “male_a_8kHz.wav”.

6.1 Compute the pole-zero representation of the LP filter (order p = 10) and draw it in the z-plane.

6.2 Compute the LSP polynomials of the filter and draw them in the z-plane using the pole-zero representation (you need to construct P(z) and Q(z) polynomials).

6.3 Define the LSF frequencies (in Hz).

6.4 Quantize the LSF frequencies using 5, 4, and 3 bits. Return the quantized frequencies together with the original LSF frequencies in a table. Make sure that the obtained LP filter is always stable (e.g., using differential quantization).

6.5 Draw frequency responses of the original and quantized LP filters into the same figure.

Useful MATLAB functions: audioread, resample, lpc, hamming, filter, fft, real, impz, xcorr, audiowrite, zplane, poly2lsf, quantiz, freqz, lsf2poly