CS578: Project Vector Quantization within LPC

March 20th 2020

Delivery: April 20th 2020

Questions: yannis@csd.uoc.gr, kafentz@csd.uoc.gr, or hy578-list@csd.uoc.gr

During this project you will explore the quantization process. More specifically you will develop a uniform scalar quantization and a vector quantization. You will apply this into the Linear Prediction algorithm studied during the 1st project.

Here are the steps.

1. **Material of the project:**

   For this project you will use 30 speech files (15 from males and 15 from females) (**dataset 1**) for building and training your quantization functions you will develop.

   You will also use two speech files (1 from male and 1 from female, **dataset 2**) for performing on them quantization in the context of Linear Prediction, using the matlab code you created during the 1st project.

2. **Linear prediction coefficients:**

   As we learned during the corresponding lectures (LPC and Quantization) it is not a good idea to perform quantization directly on the computed linear prediction coefficients (line 45). Instead, you must convert these into a set of coefficients like reflection coefficients k_i or the companding of k_i:
   $$g_i = \log \frac{1 - k_i}{1 + k_i}$$
   which are more appropriate for speech coding.

   Therefore, you will need to develop a function which will transform linear prediction coefficients to k_i or to g_i. Then you must design and develop another matlab function which will transform these coefficients back to the usual linear prediction coefficients.

3. **Scalar Quantization:**

   In the liner prediction matlab code you compute the gain of the linear prediction filter. Design and develop a scalar quantization matlab function which will take as input the value of the gain, as this is computed in line 46, and the number of bits for the quantization

(i.e., 6 bits). It will return the quantized value for this gain. Synthesis then will performed using this quantized version.

For this purpose you have to put in comments the lines where the energy of the output signal is normalized according to the energy of the analyzed signal (lines 51-52).

4. **Vector Quantization (VQ):**

   You must develop a matlab function (maybe called VQ) which will take as input many vectors (from **dataset 1**) and will construct a codebook of a given size. For this, you will follow the instructions for developing an LBG-VQ. If you create 512 size codebook, it means you will have 512 codewords.

5. **Using your quantization scheme:**

   Using speech files from **dataset 2** you will perform analysis-quantization-synthesis based on the quantization function developed above and the linear prediction matlab function you used during project 1.

   Using the above VQ function and given an input (reflection or companding reflection coefficients, from **dataset 2**) you will get the closest to the input codeword (i.e., using the Euclidean distance) and then make the transformation back to the linear prediction coefficients. You need this in order to use these coefficients during synthesis (line 50).

   Compute the excitation signal using the original linear prediction coefficients and gain (so, don't change line 47) and use that excitation into line 50. Otherwise you will not be able to check the efficiency and the problems introduced by your quantization scheme.

6. **Experiments:**

   Construct two codebooks with different size. Do analysis, quantization, and synthesis and then listen to the outputs. Compare the quality of the outputs as a function of the size of the codebook.

   Listen to the original and the processed speech signal using the Matlab command **soundsc** or **sound**.

   Write down very briefly your listening impressions.

7. **Give us your voice:**

   Record a speech signal using your voice and perform analysis, quantization and synthesis using the linear prediction model. You may use 16000 Hz as sampling frequency during your recording and 16 bits resolution. Use the bit rate you prefer for your voice (note changing the size of the codebook will require to re-build your codebook).

   Save both signals and give them to us.

   Answers may be given in Greek or in English. Return the functions you wrote by yourself plus the original (initial) Matlab file with the requested lines filled in.