HOMEWORK ASSIGNMENT #1

Due Thurs. Sept. 16, 1999 (in class)

1. *Lloyd-Max Quantizer Design:*
Here you will design a Lloyd-Max quantizer for a random variable x with zero-mean Gaussian pdf:

$$p_x(x) \;=\; \frac{1}{\sqrt{2\pi\sigma_x^2}}\, e^{-\frac{x^2}{2\sigma_x^2}}.$$

(a) Show that

$$\int x p_x(x) dx \;=\; -\,\sigma_x^2 p_x(x).$$

(b) Show that

$$\int_a^\infty p_x(x) dx \;=\; \frac{1}{2}\text{erfc}\Big(a/\sqrt{2\sigma_x^2}\Big)$$

where the "complementary error function" `erfc` is defined as follows:

$$\text{erfc}(x) \;=\; \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt.$$

(c) Implement the Lloyd-Max algorithm described in class to determine $L+1$ decision thresholds $\{x_k\}$ and L reconstruction values $\{y_k\}$ for a zero-mean Gaussian r.v. with variance σ_x^2. You may find the following Matlab functions useful: `erfc, fzero, inline, optimset`. Be careful about how $\hat{y}_1$ is initialized and updated!

(d) Assuming $\sigma_x^2 = 1$, find the Lloyd-Max $\{x_k\}$ and $\{y_k\}$ for $L = 5$ and $L = 32$ and plot them superimposed on $p_x(x)$ as in Fig. 1.

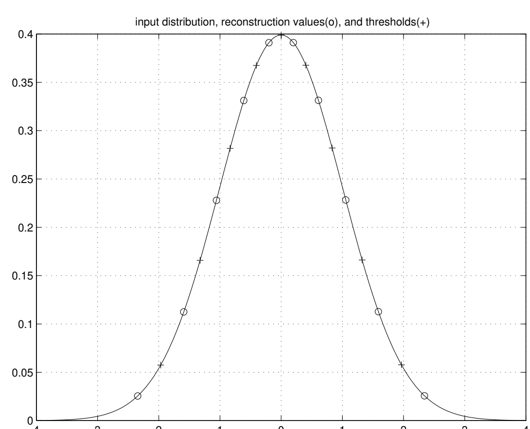

Figure 1: Lloyd-Max quantizer design.

2. *Memoryless Scalar Quantizer MSE Performance:*
Here you will analyze the performance of Lloyd-Max and uniform quantizers for large L.

   (a) Generate a 10000-sample vector of unit-variance zero-mean Gaussian random variables (using `randn`), quantize it with an $L = 32$ Lloyd-Max quantizer (try using the `find` command), and compute the mean-squared quantization error.

   (b) Show that, for zero-mean Gaussian x,

$$\left(\int_{-x_{\max}}^{x_{\max}} \sqrt[3]{p_x(x)}dx \right)^3 \;=\; 3^{3/2} \cdot 2\pi\sigma_x^2,$$

   and calculate the theoretical Lloyd-Max error variance using the formula derived in class.

   (c) Reusing the random vector from (a), compute the mean squared quantization errors resulting from uniform quantizers designed under a number (e.g., 20) of assumptions regarding $x_{\max}$ in the range $2\sigma_x \le x_{\max} \le 5\sigma_x$.

   (d) For each of the values of $x_{\max}$ used in (c), compute the theoretical uniform quantizer error variance using the formula derived in class.

   (e) Superimpose the results of (a)-(d) on one plot and discuss. For example, what do your results suggest about the choice of $x_{\max}$?

3. *Memoryless Scalar Quantizer Entropy-Rate vs. Distortion:*
Here you will analyze the output entropy for the various quantizers designed in problems 1 and 2. Recall that the quantizer output entropy determines the theoretically minimum transmission rate of a scheme that entropy encodes the quantizer outputs.

   (a) Using the $L = 32$ Lloyd-Max decision thresholds from 1(c), calculate the quantizer output entropy assuming zero-mean Gaussian input with $\sigma_x^2 = 1$. (You may find the 1(b) useful.)

   (b) Using the 20 sets of uniform quantizer thresholds from 1(c), calculate the quantizer output entropies assuming zero-mean Gaussian input with $\sigma_x^2 = 1$.

   (c) Given that

$$h_x \;=\; \frac{1}{2}\log_2(2\pi e\,\sigma_x^2)$$

   for zero-mean Gaussian x, what is the relationship between entropy and SNR in dB ($= 10\log_{10}\sigma_x^2/\sigma_q^2$) for large-$L$ uniform quantizers? What is the relationship for the optimal *non-memoryless* quantizer?

   (d) Plot, on a single graph, as a function of SNR in dB, the entropies from parts (a) and (b) together with the two bounds from (c). Discuss. For example, what do your results suggest about the choice of $x_{\max}$?

   *(e) Extra credit: Show that differential entropy $h_x = \frac{1}{2}\log_2(2\pi e\,\sigma_x^2)$ for zero-mean Gaussian x.

4. *Adaptive Quantization:*
Here you will implement AQF and AQB adaptive quantizers in Matlab and study their performance on an audio file. You have the choice of implementing block *or* exponential quantizers. *If you choose exponential, substitute the block length "N" below for a forgetting factor "α".*

   (a) Write a routine of the form $[\mathbf{y}, \boldsymbol{\sigma}] = \texttt{encode\_aqf}(\mathbf{x}, \phi, B, N)$ and a complementary routine of the form $\hat{\mathbf{x}} = \texttt{decode\_aqf}(\mathbf{y}, \boldsymbol{\sigma}, \phi, B, N)$, where

   - $\mathbf{x}$ is the input vector (of, say, length M),

- $\mathbf{y}$ is a length-M vector of quantizer outputs whose elements y_k take on integer values in the range $-2^{B-2} < y_k \leq 2^{B-2}$,
- $\boldsymbol{\sigma}$ is a length-$\lceil M/16 \rceil$ vector of forward standard-deviation estimates whose elements σ_k take on integer values in the range $0 < \sigma_k \leq 2^{16}$,
- N is the number of samples used to compute each forward standard-deviation estimate,
- ϕ is a scale factor used to multiply the variance estimate before quantization so that signal amplitudes $|x| \leq \phi\sigma_k$ are not clipped, and
- $\hat{\mathbf{x}}$ is the reconstructed version of $\mathbf{x}$.

Each standard-deviation estimate in $\boldsymbol{\sigma}$ scales a (future) block of 16 decoder inputs. Note that the total average transmission rate equals B bits per sample.

(b) Write a routine of the form $[\mathbf{y}, \boldsymbol{\sigma}] =$ `encode_aqb`$(\mathbf{x}, \phi, B, N)$ and a complementary routine of the form $\hat{\mathbf{x}} =$ `decode_aqb`$(\mathbf{y}, \phi, B, N)$, where

- $\mathbf{x}$, $\hat{\mathbf{x}}$, ϕ, and B are as above,
- $\mathbf{y}$ is a length-M vector of quantizer outputs whose elements y_k take on integer values in the range $-2^{B-1} < y_k \leq 2^{B-1}$, and
- N is the number of samples used to compute each backward standard-deviation estimate.

Note that the total average transmission rate equals B bits per sample, and that `decode_aqb` does not have access to $\boldsymbol{\sigma}$.

(c) For the soundfile `sco_29.wav`[1], $\phi = 4$, $B = 6$, the two choices $N = \{256, 1024\}$ (or $\alpha = \{0.9, 0.99\}$), and the two encoders, calculate the average SNR, i.e.,

$$10 \log_{10} \left(\frac{\sum_{i=1}^{M} x_k^2}{\sum_{i=1}^{M} (x_k - \hat{x}_k)^2} \right),$$

and plot

i. the input signal $\mathbf{x}$,
ii. the standard-deviation estimates $\boldsymbol{\sigma}$,
iii. the quantized output $\mathbf{y}$,
iv. the vector of instantaneous absolute errors $|x_k - \hat{x}_k|$.

so that they line up vertically (e.g., using the `subplot(41x)` commands and `orient tall`) for ease in comparison. This should give you four plots that *each* look somewhat like Fig. 2.

(d) Finally, process the longer file `sco_160.wav` and listen to the four outputs. Which one "sounds" the best? Is it the one with the best average SNR?

Hints:

- Be careful how your encoders deal with zero-valued portions of the input.
- You may need to zero-pad the front of the input vector in order to compute the first N variance estimates.
- If using the `wavwrite` command, you will need to make sure that output samples do not exceed absolute value one.

[1] This is a 29 kbyte (about 1/3 second) 16-bit mono excerpt from track 3 of John Scofield's *A Go Go*, ©1998 Polygram Records, Inc., with backup from the popular trio "Martin, Medeski, and Wood."

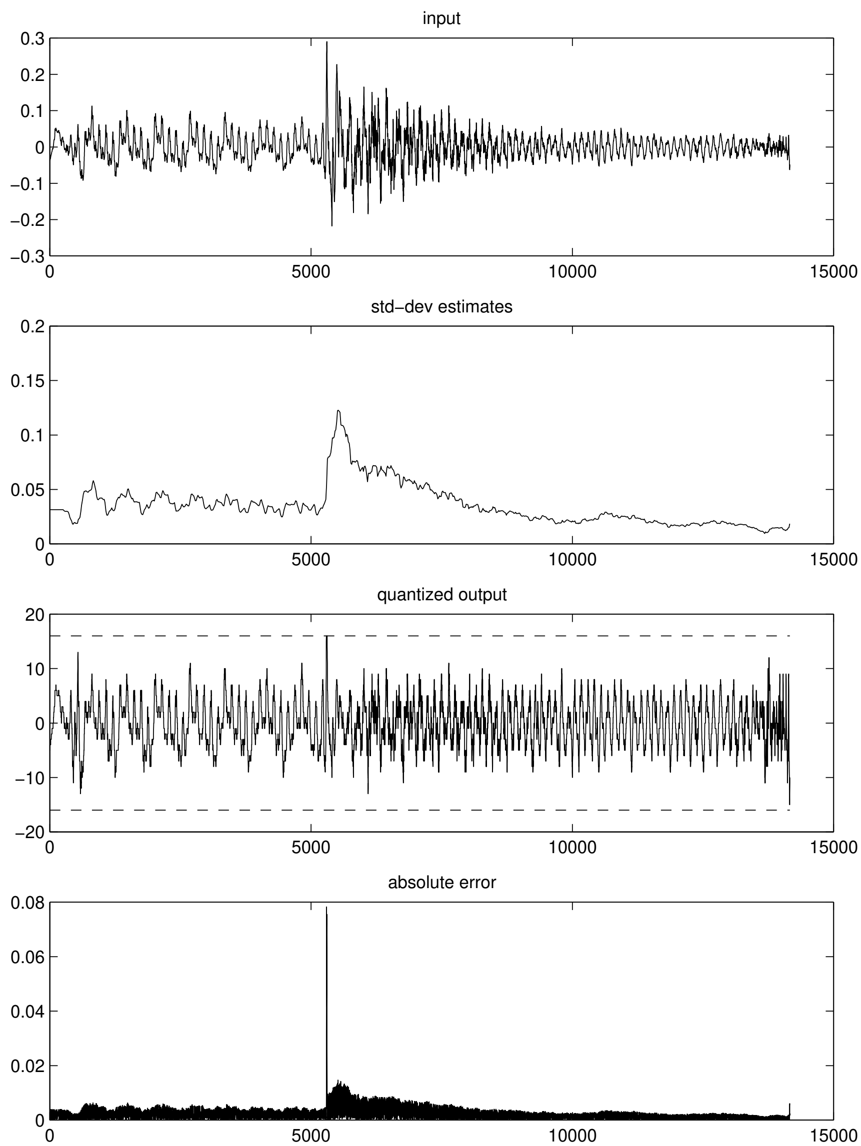

Figure 2: Example Matlab output plot for block AQF on `sco_29.wav`.