Problem Set #4: Boxcar and Exponentially-weighted Averagers

DUE: Friday, 2014-02-28 at 5:00 PM **as a Sakai assignment attachment**

1. **Boxcar Averager** A sliding window averager has a window length of three samples.

   (a) Using Matlab, plot, on the same axes, the input and output of this averager given the input samples listed below. Assume that the input at times *before* and *after* these samples is equal to zero.

   0 0.2 0 -0.1 0.3 0.8 1.2 0.9 1.1 1.2 0.8 1.1 1.2 0.8 1.1 0 -0.2 -0.1 0 -0.2

   (b) If the actual signal trying to be represented by these samples is:

   0.0 0.0 0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

   then comment on the pros / cons of the sliding averager relative to recovering the actual signal.

2. **Exponentially-Weighted Averager** An exponentially-weighted averager that produces an output signal that is a weighted sum of a series of input signals described by:

$$S_o = \frac{1}{m}\left[S_n + \left(\frac{m-1}{m}\right) S_{n-1} + \left(\frac{m-1}{m}\right)^2 S_{n-2} + \ldots\right],$$

   where m is the exponential weighting factor.

   (a) Demonstrate how m affects the weighting of more recent or later samples in the exponential averager. You can do this numerically, and not analytically.

   (b) We discussed in lecture how the boxcar averager acted as a low-pass filter on the data. Sketch the transfer function of the boxcar averager relative to a sinusoidal signal oscillating at ω_o, and demonstrate how if the averaging window is too long, then you are just left with the DC offset of the sinusoidal signal in the output.

   (c) Compare and contrast the frequency-domain transfer function of a boxcar averager with that of an exponentially-weighted averager. How does the transfer function of the exponentially-weighted averager change as a function of m?

3. **Arduino Implementation of Averagers**

   (a) Build an Arduino-controlled circuit that modulates the brightness of an LED with a sine wave that has a 5 s period.

   (b) Add noise to your sinusoid using `random()` to achieve an SNR of 6 dB.

   (c) Run your code with the noisy sinusiod and note the difference in your LED illumination.

   (d) Implement a boxcar averager in your Arduino code to filter the noise sinusoid to illuminate the LED more like the pure sinusoid.

   (e) Include in your code comments your analysis / reasoning for choosing the size of your boxcar.

   (f) Submit your functioning code as part of your attachment to the Sakai assignment.