Problem Set #2 – Solutions
Due: Friday April 20, 2018 at 5 PM.

1. **Non-ideal sampling and recovery of ideal samples by discrete-time filtering** (30 pts)
   Consider a system to convert a continuous-time signal to discrete-time defined by the input-output relation
   \[ y[n] = \int_{(n-1)T_s}^{nT_s} x(\tau)d\tau. \]  
   (1)

   a) Show that the system in (1) can be obtained by passing \( x(t) \) through a filter with impulse response \( h(t) = u(t) - u(t-T_s) \) followed by a pointwise uniform sampler at times \( t = nT_s, n \in \mathbb{N} \): (6 pts)

   \[ x(t) \xrightarrow{H(j\omega)} y_c(t) = x(t) * h(t) \xrightarrow{\text{sampler}} y[n] = y_c(nT_s) \]

   **Solution:**

   \[ y_c(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau)(u(t-\tau) - u(t-T_s - \tau)) d\tau \]
   \[ = \int_{t-T_s}^{t} x(\tau)d\tau \]
   \[ y[n] = [y_c(t)]_{t=nT_s} = \int_{(n-1)T_s}^{nT_s} x(\tau)d\tau \]

   b) Find an expression for \( Y(e^{j\Omega}) \) in terms of \( X(j\omega) \) and \( T_s \). Your answer should involve a sum of frequency-shifted copies of \( X(j\omega) \), multiplied by other frequency-dependent factors. (6 pts)

   **Solution:**

   There are two approaches to this problem depending on how you choose to write \( H(j\omega) \). This leads to two expressions which are mathematically equivalent but look very different.

   **Approach 1:** \( h(t) = \Pi \left( \frac{t-T_s/2}{T_s} \right) \)

   \[ Y_c(j\omega) = X(j\omega)H(j\omega) = X(j\omega)CTFT \left\{ \Pi \left( \frac{t-T_s/2}{T_s} \right) \right\} \]
   \[ = X(j\omega)T_s \text{sinc} \left( \frac{T_s\omega}{2\pi} \right) e^{-j\omega T_s/2} = X(j\omega)\frac{2\sin(\omega T_s/2)}{\omega} e^{-j\omega T_s/2} \]
By DTFT properties of sampled signals:

\[ Y(e^{j\Omega}) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} Y_c(j(\Omega - 2\pi k)/T_s) \]

\[ = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X(j(\Omega - 2\pi k)/T_s) H(j(\Omega - 2\pi k)/T_s) \]

\[ = \sum_{k=-\infty}^{\infty} X(j(\Omega - 2\pi k)/T_s) \frac{2\sin((\Omega - 2\pi k)/2)}{\Omega - 2\pi k} e^{-j(\Omega - 2\pi k)/2} \]

(2)

\[ \text{Approach 2: } h(t) = u(t) - u(t - T_s) \]

\[ Y_c(j\omega) = X(j\omega)H(j\omega) = X(j\omega)\text{CTFT}\{u(t) - u(t - T_s)\} \]

\[ = X(j\omega) \left( \pi \delta(\omega) + \frac{1}{j\omega} - e^{j\omega T_s} \left( \pi \delta(\omega) + \frac{1}{j\omega} \right) \right) \]

\[ = X(j\omega) \left( \frac{1 - e^{-j\omega T_s}}{j\omega} \right) \]

(3)

c) Assume that

\[ X(j\omega) = \begin{cases} A & |\omega| \leq \pi/T_s, \\ 0 & \text{otherwise.} \end{cases} \]

Sketch a single period of the magnitude and phase of \( Y(e^{j\Omega}) \). Also, sketch a single period of the magnitude and phase of \( X_d(e^{j\Omega}) \), the DTFT of the sampled signal \( x_d[n] = x(nT_s) \). (6 pts)

\[ \text{Solution:} \]

From the sampling relation, we have

\[ X_d(e^{j\Omega}) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X(j(\Omega - 2\pi k)/T_s) . \]

Since \( X(j\omega) \) is bandlimited to \(|\omega| < \pi/T_s\), for \(-\pi < \Omega < \pi\) the only non-zero term in the sum above (and in the \( Y(e^{j\Omega}) \) found in (b)) corresponds to \( k = 0 \). As a result we have

\[ X_d(e^{j\Omega}) = \frac{1}{T_s} X(j\Omega/T_s) = \frac{A}{T_s}, \quad -\pi \leq \Omega \leq \pi, \]
and (Approach 1 for (b))

\[ Y(e^{j\Omega}) = X(j\Omega/T_s) \frac{2\sin(\Omega/2)}{\Omega} e^{-j\Omega/2} = \frac{2\sin(\Omega/2)}{\Omega} A e^{-j\Omega/2}, \quad -\pi \leq \Omega \leq \pi. \]  

or (Approach 2 for (b))

\[ Y(e^{j\Omega}) = X(j\Omega/T_s) \frac{1 - e^{-j\Omega}}{j\Omega} = \frac{1 - e^{-j\Omega}}{j\Omega} A, \quad -\pi \leq \Omega \leq \pi. \]  

\( d \) We now filter \( y[n] \) using \( g[n] \) to obtain \( z[n] = y[n] * g[n] \). Find the frequency response \( G(e^{j\Omega}) \) such that \( z[n] = x[n] = x(nT_s) \). Can the system \( g[n] \) be causal? explain. (6 pts)

\textit{Solution:}

From our solution to (c) we see that the filter \( g[n] \) must invert the effect of the term \( \frac{1 - e^{-j\Omega}}{j\Omega} \) in the DTFT of \( y[n] \). We thus conclude that

\[ G(e^{j\Omega}) = \frac{1}{T_s} \frac{\Omega}{2\sin(\Omega/2)} e^{j\Omega/2} \quad \text{or} \quad G(e^{j\Omega}) = \frac{1}{T_s} \frac{j\Omega}{1 - e^{-j\Omega}}, \]

for \(-\pi \leq \Omega \leq \pi\) (and periodically extend the last expression to all \( \Omega \)s).

The filter is not causal. One way to see this is to observe that \( G(e^{j\Omega}) \) can be expressed as \( G(e^{j\Omega}) = 0.5 e^{+j(\Omega/2)} \frac{\Omega}{\sin(\Omega/2)} \). This corresponds to a left shift (in time) of the inverse transform of a real function of \( \Omega \). Thus it is a left shift of a sequence which is hermitian in time and cannot be non causal. Any correct reasoning which justifies why the signal might be causal also given full credit.
e) Sketch a single period of the magnitude and phase of $G(e^{j\Omega})$. (6 pts)

**Solution:**

If we use the first from from (d), then the answer is immediate. From the second form, some additional trig manipulation is required, but the result is the same.

$$|G(e^{j\Omega})| = \frac{\Omega}{2\sin(\Omega/2)}.$$

$$\angle G(e^{j\Omega}) = \Omega/2$$

2. **Effect of upsampling on reconstruction filter requirements** (20 pts)

A continuous music signal is bandlimited to $\frac{\omega_m}{2\pi} = 10kHz$ and sampled at a frequency $\frac{\omega_s}{2\pi} = 32kHz$.

a) Using zero-order hold, a continuous signal is reconstructed from the samples. Sketch the magnitude of the reconstruction filter, $|H_r(j\omega)|$ that enables perfect reconstruction. What is the range of frequencies for which the output spectrum of the zero-order hold is 0? What (if any) restrictions are there on the reconstruction filter response at these frequencies? (10 pts)

**Solution:**

The output $x_0(t)$ of a ZOH at rate $\omega_s = 2\pi/T_s$ is defined to be

$$x_0(t) = \sum_{n=-\infty}^{\infty} x_d[n]h_0(t-nT_s),$$

where $h_0(t)$ is the impulse response of a causal rect signal of length $T_s$. That is,

$$H_0(j\omega) = CTFT \left\{ \Pi \left( \frac{t-T_s/2}{T_s} \right) \right\} = T_s e^{-j\omega T_s/2} \text{sinc} \left( \frac{\omega}{\omega_s} \right), \quad \omega_s = 2\pi/T_s.$$

Note that $x_0(t)$ an also be written as

$$x_0(t) = \int_{-\infty}^{\infty} h_0(t-\tau) \sum_{n=-\infty}^{\infty} x(nT_s) \delta(\tau - nT_s)$$

$$= h_0(t) * x_p(t),$$
so

\[ X_0(j\omega) = H_0(j\omega)X_p(j\omega) = \left( T e^{-j\omega T_s / 2} \text{sinc} \left( \frac{\omega}{\omega_s} \right) \right) \left( \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X(j(\omega - k\omega_s)) \right) \]

\[ = \left( e^{-j\omega T_s / 2} \text{sinc} \left( \frac{\omega}{\omega_s} \right) \right) \left( \sum_{k=-\infty}^{\infty} X(j(\omega - k\omega_s)) \right) \]

We now apply a reconstruction filter \( H_r(j\omega) \) to the output of a ZOH:

\[ X_r(j\omega) = H_r(j\omega)X_0(j\omega). \]

In order to optimally recover \( x(t) \), we require \( X_r(j\omega) = X(j\omega) \). Thus to ensure that only the \( k = 0 \) term survives, \( H_r \) must be a LPF with cutoff \( \omega_s / 2 \); further, in the passband, it must invert the scaling in \( X_0(j\omega) \). This leads to

\[ H_r(j\omega) = \Pi \left( \frac{\omega}{\omega_s} \right) \frac{e^{-j\omega T_s / 2}}{\text{sinc}(\omega / \omega_s)}. \]

The magnitude of \( H_r(j\omega) \) is given by

\[ |H_r(j\omega)| = \Pi \left( \frac{\omega}{\omega_s} \right) \frac{1}{\text{sinc}(\omega / \omega_s)}. \]

Ideally \( |H_r(j\omega)| \) is zero for \( |\omega| > \omega_s / 2 \). In practice, however, it is hard to implement such an abrupt transition to zero at \( \omega = \pm \omega_s / 2 \). Since \( x(t) \) is bandlimited to \( \omega_m \), the condition \( |\omega| > \omega_s / 2 \) can be relaxed to \( H_r(j\omega) = 0 \) for \( |\omega| > \omega_s - \omega_m = 2\pi(32 - 10) \cdot 10^3 \), allowing for a gradual roll-off between \( \omega_s / 2 = 2\pi(16 \cdot 10^3) \) and \( \omega_s - \omega_m = 2\pi(22 \cdot 10^3) \)

b) Ideal upsampling by a factor of \( L = 2 \) (as described in the reader) is performed and a continuous signal is reconstructed from a zero-order hold. Repeat part (a) for this new sampling rate. (10 pts)

Solution:

Since the original sampling was done above the Nyquist frequency \( \omega_s \geq 2\omega_m \), upsampling by \( L = 2 \) is equivalent to having done our original sampling at twice the sampling frequency. Thus we need only replace \( \omega_s = 2\pi(32 \cdot 10^3) \) in the final solutions for (a) with \( 2\omega_s = 2\pi(64 \cdot 10^3) \)

3. *Discrete time upsampling and downsampling* (20 pts)

(Adapted from OWN, Prob. 7.19) Consider the system shown below with input \( x[n] \) and output \( y[n] \). The zero insertion system inserts two points with zero amplitude between each of the sequence values in \( x[n] \). The decimation is defined by \( y[n] = w[5n] \), where \( w[n] \) is the input sequence for the decimation system, and \( H(e^{j\Omega}) = 1 \), if \( \Omega \in [-\pi/5, \pi/5] \) and zero otherwise.

\[ x[n] \xrightarrow{\text{Zero insertion}} t[n] \xrightarrow{H(e^{j\Omega})} \xrightarrow{\text{Decimation}} y[n] \]
(a) If the input is of the form
\[ x[n] = \frac{\sin \omega_1 n}{\pi n}, \]
determine the output \( y[n] \), and plot the DTFT of \( x[n], t[n], w[n] \) and \( y[n] \) for the following values of \( \omega_1 \).

i. \( \omega_1 \leq \frac{3\pi}{5} \)
ii. \( \omega_1 > \frac{3\pi}{5} \).

Solution:
First, we need the DTFT of \( x[n] \):
\[
x[n] = \frac{\sin \omega_1 n}{\pi n} = \frac{\omega_1}{\pi} \text{sinc} \left( \frac{\omega_1 n}{\pi} \right)
\]
\[
X(e^{j\Omega}) = \sum_{k=-\infty}^{\infty} \Pi \left( \frac{\Omega - 2\pi k}{2\omega_1} \right) = \Pi \left( \frac{\Omega}{2\omega_1} \right)
\]
The last equality holds because periodicity in \( \Omega \) with period \( 2\pi \) is implicit by definition of \( \Omega \).
Up-sampling with \( L = 3 \), we get \( T(e^{j\Omega}) = X(e^{j3\Omega}) = \Pi(\frac{3\Omega}{2\omega_1}) \). So far, (i) and (ii) are identical.

\[
\begin{array}{c|c}
\hline
\text{\( \omega_1 \)} & \text{\( \Omega \)} \\
\hline
-\frac{\pi}{6} & 0 \\
0 & \frac{\pi}{2} \\
\frac{\pi}{6} & \frac{\pi}{2} \\
\hline
\end{array}
\]

Now apply \( H(e^{j\Omega}) \), an LPF with cutoff frequency \( \omega_c = \pi/5 \). Thus we have two cases:

i. \( \omega_1 \leq 3\pi/5 \Rightarrow \omega_1/3 \leq \pi/5 \Rightarrow W_i(e^{j\Omega}) = \Pi(\frac{\Omega}{2\omega_1/3}) = \Pi(3\Omega/2\omega_1) \)

ii. \( \omega_1 > 3\pi/5 \Rightarrow \omega_1/3 > \pi/5 \Rightarrow W_{ii}(e^{j\Omega}) = \Pi(\frac{\Omega}{3\pi/5}) = \Pi(5\Omega/2\pi) \)

\[
\begin{array}{c|c}
\hline
\text{\( \omega_1 \)} & \text{\( \Omega \)} \\
\hline
-\frac{\pi}{6} & 0 \\
0 & \frac{\pi}{2} \\
\frac{\pi}{6} & \frac{\pi}{2} \\
\hline
\end{array}
\]

Passing \( w[n] \) through the decimator gives us \( Y(e^{j\Omega}) = \frac{1}{3} W(e^{j\Omega/5}) \):

i. \( Y_i(e^{j\Omega}) = \frac{1}{5} \Pi(3\Omega/10\omega_1) \)

ii. \( Y_{ii}(e^{j\Omega}) = \frac{1}{5} \Pi(\Omega/2\pi) \)
Finally, we take the inverse transform to revert to the time domain:

i. \( y_1[n] = \frac{5\omega_1}{15\pi} \text{sinc} \left( \frac{5\omega_1 n}{3\pi} \right) \)

ii. \( y_{ii}[n] = \frac{1}{5} \text{sinc}[n] = \frac{1}{5} \delta[n] \)

(b) Assume \( \omega_1 \leq \frac{3\pi}{5} \). Suppose the low pass filter \( H \) was replaced with a filter \( \tilde{H}(e^{j\Omega}) \) that is constant over bandwidth \([-W_1, W_1]\), takes any value between \([-W_2, -W_1]\) and \([W_1, W_2]\) and is zero outside \([-W_2, W_2]\). For what values of \( W_1 \) and \( W_2 \) are \( W(e^{j\Omega}) \) and \( Y(e^{j\Omega}) \) the same as with the ideal LPF of part a)?

Solution:

Any combination of \( W_1, W_2 \) satisfying \( W_1 \geq \frac{\omega_1}{3} \), \( W_2 \leq \left( \frac{2}{5} \pi - \frac{\omega_1}{3} \right) \) works. This can be found out by looking at the plot for \( T(e^{j\Omega}) \) for a general \( \omega_1 \).

Extra Credit Reconstruction of a band-limited signal from nonuniform sampling. (10 pts)

(Taken from OWN, Problem 7.37) A signal limited in bandwidth to \(|\omega| < W\) can be recovered from non-uniformly spaced samples as long as the average sample density is \(2(W/2\pi)\) samples per second. This problem illustrates a particular example of nonuniform sampling. Assume that in Figure (a):
• $x(t)$ is band-limited; $X(j\omega) = 0, |\omega| > W$.
• $p(t)$ is a non-uniformly spaced periodic pulse train, as shown in Figure (b).
• $f(t)$ is a periodic waveform with period $T = 2\pi/W$. Since $f(t)$ multiplies an impulse train, the only values that are significant are $f(0) = a$ and $f(\delta) = b$ at $t = 0$ and $t = \Delta$, respectively.
• $H_1(j\omega)$ is a 90 degree phase shifter defined as follows:

$$H_1(j\omega) = \begin{cases} j & \omega > 0 \\ -j & \omega < 0 \end{cases}$$

• $H_2(j\omega)$ is an ideal lowpass filter defined as follows:

$$H_2(j\omega) = \begin{cases} K & 0 < \omega < W \\ K^* & -W < \omega < 0 \\ 0 & |\omega| < W \end{cases}$$

where $K$ is a (possibly complex) constant.

**Answer the following questions:**

a) Express the CTFT of $p(t), y_1(t), y_2(t),$ and $y_3(t)$ in terms of $X(j\omega)$.
   **Solution:**
\[
P(j\omega) = CTFT \left\{ \sum_{k=-\infty}^{\infty} \delta(t - \frac{2\pi k}{W}) + \delta(t - \frac{2\pi k}{W} - \Delta) \right\}
\]
\[
= W \sum_{n=-\infty}^{\infty} \delta(\omega - Wn) + W \sum_{n=-\infty}^{\infty} e^{-j\omega \Delta} \delta(\omega - Wn)
\]
\[
= W \sum_{n=-\infty}^{\infty} \delta(\omega - Wn)(1 + e^{-j\omega \Delta}).
\]

Since \(y_1(t) = x(t)f(t)p(t)\), \(p(t)\) is an impulse train and \(f(t)\) is periodic, we have
\[
y_1(t) = \begin{cases} 
  ax(t) & t = \frac{2\pi}{W} k, \ k \in \mathbb{Z}, \\
  bx(t) & t = \frac{2\pi}{W} k + \Delta, \ k \in \mathbb{Z}, \\
  0 & \text{otherwise}.
\end{cases}
\]

This last expression implies that \(y_1(t)\) is a sum of two shifted impulse trains scaled according to \(a, b\) and the value of the sampled signal \(x(t)\). In order to obtain its CTFT, we first used similar arguments as in the expression for \(P(j\omega)\) above for the CTFT of \(p(t)f(t)\). We then apply the convolution-multiplication relation which leads to
\[
Y_1(j\omega) = \frac{1}{2\pi} X(j\omega) * \left( W \sum_{n=-\infty}^{\infty} \delta(\omega - nW)(a + be^{-jnW\Delta}) \right)
\]
\[
= \tilde{W} \sum_{n=-\infty}^{\infty} X(j(\omega - nW))(a + be^{-jnW\Delta}),
\]
where \(\tilde{W} = \frac{W}{2\pi}\).

Since \(y_2(t)\) is a filtered version of \(y_1(t)\) we have
\[
Y_2(j\omega) = H_1(j\omega)Y_1(j\omega) = H_1(j\omega)\tilde{W} \sum_{n=-\infty}^{\infty} X(j(\omega - Wn))(a + be^{-jWn\Delta})
\]
\[
= j\text{sign}(\omega)\tilde{W} \sum_{n=-\infty}^{\infty} X(j(\omega - nW))(a + be^{-jWn\Delta}).
\]

Finally,
\[
Y_3(j\omega) = \tilde{W} \sum_{n=-\infty}^{\infty} X(j(\omega - nW))(1 + e^{-jnW\Delta}).
\]

b) Specify the values of \(a, b,\) and \(K\) as functions of \(\Delta\) such that \(z(t) = x(t)\) for any band-limited \(x(t)\) and any \(\Delta\) such that \(0 < \delta < \pi/W\).

**Solution:**

In this question, unless stated otherwise, we focus on the part of the signal corresponding to \(\omega \in [-W, W]\) because the low pass filtering removes everything else.
The CTFT of the signal at the input to the LPF is

\[
Y_3(j\omega) + Y_2(j\omega) = \begin{cases} 
\tilde{W}X(j\omega)(ja + jb + 2) + \tilde{W}X(j(\omega - W))(1 + e^{-jW\Delta} + j(a + be^{-jW\Delta})) & \text{if } \omega \in [0, W] \\
\tilde{W}X(j\omega)(-ja - jb + 2) + \tilde{W}X(j(\omega + W))(1 + e^{jW\Delta} - j(a + be^{jW\Delta})) & \text{if } \omega \in [-W, 0]
\end{cases}
\]

The CTFT of the output of the LPF is

\[
Z(j\omega) = \begin{cases} 
\tilde{W}KX(j\omega)(ja + jb + 2) + \tilde{W}KX(j(\omega - W))(1 + e^{-jW\Delta} + j(a + be^{-jW\Delta})) & \text{if } \omega \in [0, W] \\
\tilde{W}K^*X(j\omega)(-ja - jb + 2) + \tilde{W}K^*X(j(\omega + W))(1 + e^{jW\Delta} - j(a + be^{jW\Delta})) & \text{if } \omega \in [-W, 0]
\end{cases}
\]

For \(X(j\omega) = Z(j\omega)\), we need

\[
1 + e^{-jW\Delta} + j(a + be^{-jW\Delta}) = 0 \\
1 + e^{jW\Delta} - j(a + be^{jW\Delta}) = 0 \\
K(ja + jb + 2) = K^*(2 - ja - jb) = 1/\tilde{W}.
\]

Solving the set of equations, we get that

\[
a = \cos(W\Delta/2), \quad b = -a, \quad K = 0.5/\tilde{W} = \pi/W.
\]

### MATLAB Assignment

**General Instructions**

Answer all questions asked. Your submission should include all m-file listings and plots requested. All plots should have a title and x and y-axes properly labeled.

### 1 Upsampling and Downsampling

In this exercise, you will examine how upsampling and downsampling a discrete-time signal affects its discrete-time Fourier transform (DTFT).

**a)** For most of this exercise, you will be working with finite segments of the two signals

\[
x_1[n] = \left( \frac{\sin(0.4\pi(n - 62))}{0.4\pi(n - 62)} \right)^2 \\
x_2[n] = \left( \frac{\sin(0.2\pi(n - 62))}{0.2\pi(n - 62)} \right)^2
\]

Define \(x_1\) and \(x_2\) to be these signals for \(0 \leq n \leq 124\) using the `sinc` command. Plot both of these signals using `stem`. If you defined the signals properly, both plots should show that the signals are symmetric about their largest sample, which has height 1.
b) Analytically compute the DTFTs $X_1(e^{j\Omega})$ and $X_2(e^{j\Omega})$ of $x_1[n]$ and $x_2[n]$ as given in Eqs. (1) and (2), ignoring the effect of truncating the signals. Use `fft` to compute the samples of the DTFT of the truncated signals in $x_1$ and $x_2$ at $\Omega_k = 2\pi k / 2048$ for $0 \leq k \leq 2047$ and store the results in $X_1$ and $X_2$. Generate appropriately labeled plots of the magnitudes of $x_1$ and $x_2$. How do these plots compare with your analytical expressions?

c) Define the expansion of the signal $x[n]$ by $L$ to be the process of inserting $L - 1$ zeros between each sample of $x[n]$ to form

$$x_e[n] = \begin{cases} 
  x[n/L], & n = kL, \ k \text{ integer}, \\
  0, & \text{otherwise}.
\end{cases}$$

If $x$ is a row vector containing $x[n]$, the following commands implement expanding by $L$

```
>> xe = zeros(1, L*length(x));
>> xe(1:L:length(xe)) = x;
```

Based on this template, define $x_{e1}$ and $x_{e2}$ to be $x_1$ and $x_2$ expanded by a factor of 3. Also, define $X_{e1}$ and $X_{e2}$ to be 2048 samples of the DTFT of these expanded signals computed using `fft`. Generate appropriately labeled plots of the magnitude of these DTFTs. Expanding by $L$ should give a DTFT $X_e(e^{j\Omega}) = X(e^{j\Omega L})$. Do your plots agree with this?

## 2 Quantization

In this question you will empirically study the effect of quantization on the accuracy of the digital representation of a signal. (20 pts)

Let

$$x[n] = 10\text{sinc}(t)$$

Create a vector $x$ which contains the samples of $x(t)$ at 100 times samples uniformly spaced between $-5$ to $5$.

a) Construct a function `quantize(x, mid_points)` which returns the element in the vector `mid_points` closest to $x$ in absolute value.

b) Use the function you created in (a) to quantize each sample in $x$ into $k = 8$ quantization levels (this corresponds to representing the signal with 3 bits) uniformly spaced between the maximum and the minimum of the signal. That is, the vector `mid_points` should contain $L$ numbers that represents the mid-points of the quantization bin boundaries. Plot on the same figure the original signal $x(t)$ and its quantized version $\hat{x}(t)$.

c) We define quantization error as $x[n] - \hat{x}[n]$. Plot a histogram of the quantization error you received in (b). What distribution does the error seem to follow?
d) Repeat part (b) with \( k = 4, k = 16, k = 32, \) and \( k = 64 \) quantization levels (2, 4, 5 and 6 bits, respectively). For each case, calculate the SNR (signal energy divided by noise energy) where we define noise energy as \( \sum_{n}(x[n] - \hat{x}[n])^2 \) and signal energy as \( \sum_{n}(x[n])^2 \). Plot SNR vs. log number of quantization levels \( L \).

- **Extra credit (extension of second matlab problem) (10 pts)**

Matlab has a built in function lloyds.m that quantizes a signal using Lloyd’s algorithm. The Lloyd algorithm (also known as the "K-means algorithm") takes as inputs a set of points on the real line \( x \) and the number \( L \) of “bins” or “clusters”. The algorithm divide the real line into \( L \) disjoints intervals, such that the distance from each point to the center of mass of all other points in its interval is minimal. The result is a quantization rule (or a partition) that is optimal with respect to the given data points in the sense that the quantization noise energy is minimal over all partitions of the real line into \( L \) regions.

Lloyd’s algorithm is also useful in partitioning more than one dimension. With more than one dimension, however, the algorithm is not guaranteed to converge to a global optimal partition.

Repeat part (a) - (c) above using this function to perform quantization. How do your results differ?
%Matlab Assignment Solutions for PSET 2

close all
%Problem 1
n = 0:124;
x1 = (sinc(0.4*(n-62))).^2;
x2 = (sinc(0.2*(n-62))).^2;

%part a
figure(1)
stem(n,x1)
xlabel('n')
ylabel('x1')
title('Stem plot of x1[n]')
figure(2)
stem(n,x2)
xlabel('n')
ylabel('x2')
title('Stem plot of x2[n]')

%part b
k = 0:2047;
omega = (2*pi*k)/2048;
X1 = fft(x1, 2048);
X1 = fftshift(X1);
X2 = fft(x2, 2048);
X2 = fftshift(X2);
figure(3)
hold on
plot(omega, abs(X1), 'b');
xlabel('omega');
ylabel('mag(DTFT)');
plot(omega, abs(X2), 'r');
legend('X1', 'X2');
title('DTFT of x1 and x2')
hold off

%part c
L= 3;
xe1 = zeros(1, L*length(x1));
xe1(1:L: length(xe1)) = x1;
xe2 = zeros(1, L*length(x2));
xe2(1:L: length(xe2)) = x2;

Xe1 =fft(xe1, 2048);
Xe1 = fftshift(Xe1);
Xe2 = fft(xe2, 2048);
Xe2 = fftshift(Xe2);
figure(4)
hold on
plot(omega, abs(Xe1), 'b');
xlabel("omega");
ylabel("mag(DTFT)");
plot(omega, abs(Xe2), 'r');
legend('Xe1','Xe2')
title('DTFT of xe1 and xe2')
hold off
%Problem 2 (extra credit portion is added to the solution)
\[ t = \text{linspace}(-5,5,100); \]
\[ x = 10\times \text{sinc}(t); \]

\[ \text{x\_max} = \text{max}(x); \]
\[ \text{x\_min} = \text{min}(x); \]
\[ \text{Amp} = \text{x\_max} - \text{x\_min}; \]
\[ \text{Ls} = 2.\times(2:6); \]
\[ \text{SNR} = []; \]
\[ \text{SNR\_L} = []; \]
\[ \text{for} \ L = \text{Ls} \]
\[ \quad \text{bin\_boundaries} = \text{x\_min}:\text{Amp}/L:\text{x\_max}; \]
\[ \quad \text{mid\_points} = \text{(bin\_boundaries}(1:\text{end}-1) + \text{bin\_boundaries}(2:\text{end}))/2; \]
\[ \quad \%\text{Lloyd's partition (extra credit)} \]
\[ \quad [~,\text{points\_Lloyd}] = \text{lloyds}(x,L); \]
\[ \text{x\_hat} = \text{zeros}(1,\text{length}(x)); \]
\[ \text{x\_hat\_L} = \text{zeros}(1,\text{length}(x)); \]
\[ \text{for} \ j = 1:\text{length}(x) \]
\[ \quad \text{x\_hat}(j) = \text{quantize}(\text{x}(j), \text{mid\_points}); \]
\[ \quad \text{x\_hat\_L}(j) = \text{quantize}(\text{x}(j), \text{points\_Lloyd}); \]
\[ \text{end} \]
\[ \text{e} = \text{x-\text{x\_hat}}; \]
\[ \text{e\_L} = \text{x-\text{x\_hat\_L}}; \]
\[ \text{if} \ (L==8) \]
\[ \quad \text{figure}(5) \]
clf
hold on
plot(x,'k')
plot(x_hat,'-ob')
plot(x_hat_L,'-or')
title('Original and Quantized Signals','fontsize',14);
legend('x','x_{hat}','x_{hat-Lloyd}')
hold off

figure(6)
clf
histogram(e)
title('Noise Histogram','fontsize',14);
end

sigE = sum(x.^2);
NE = sum(e.^2);
NE_L = sum(e_L.^2);
SNR = [SNR, sigE/NE];
%extra credit
SNR_L = [SNR_L sigE/NE_L];
end

figure(7)
clf
hold on
plot(2:6,SNR,'r')
plot(2:6,SNR_L,'b')
xlabel('bits','fontsize',14)
ylabel('SNR','fontsize',14)
title('SNR vs number of quantization bits','fontsize',14)
legend('Uniform','Lloyd')

function x_hat = quantize(x,mid_points)
%returns the closest element in mid_point to x
[~,i] = min(abs(mid_points - x));
x_hat = mid_points(i);
end
SNR vs number of quantization bits

Published with MATLAB® R2018a