Each student is expected to complete the exam individually using only course notes, the book, and technical literature, and without aid from other outside sources.

Aside from the most general conversation of the exam material, I assert that I have neither provided help nor accepted help from another student in completing this exam. As such, the work herein is mine and mine alone.

__________________________________________  __________
Signature                                      Date

__________________________________________  __________
Name (printed)                                 Student ID #

* Attach as cover page to completed exam.
EECS 844 Exam 3 (Due November 12)

Data sets can be found at http://www.ittc.ku.edu/~sdblunt/844/EECS844_Exam3

Provide:
- Complete and concise answers to all questions
- Matlab code with solutions as appropriate
- All solution material (including discussion and figures) for a given problem together (i.e. don’t put all the plots or code at the end)
- Email final Matlab code to me in a zip file (all together in 1 email)

** All data time sequences are column vectors with increasing time index as one traverses down the vector. (you will need to properly orient the data into “snapshots”)
** Provide figures (magnitude and phase) of all solutions.

1. In P1.mat the output of a hypothesized unknown linear system is the observed signal \(d(n)\), to which we shall apply a whitening filter via linear prediction (these are the solid lines in the figure). In contrast, the dashed lines represent the inverse filtering formulation that we could perform if the hypothesized “driving signal” \(x(n)\) were actually known (in practice it is not).
   a) Determine \(w\) as the linear prediction error (LPE) filter using only \(d(n)\) when the linear prediction component is \(M = 29\) (so LPE filter is length 30).
   b) Determine \(w\) as the inverse system identification Wiener filter using both \(d(n)\) and \(x(n)\) for \(M = 30\).
   c) Plot the time-domain magnitude/phase and frequency response of each filter. Comment on how they are related.

2. Derive and implement the steepest-descent algorithm for the cost function

\[
J(w) = w^H R w + |w^H s(\omega_1) - 1|^2 + |w^H s(\omega_2) - 0.5|^2
\]

where \(w\) is a time-domain filter, \(R\) is a correlation matrix, and \(s(\omega_1)\) and \(s(\omega_2)\) are steering vectors for particular frequencies. For \(N = 40\), use the \(N\times N\) correlation matrix and \(N\times 1\) steering vectors provided in P2.mat and initializing the \(N\times 1\) filter \(w(0)\) with all zeros, generate 2000 steepest-descent iterations using step-sizes of \(\mu = 1/(10N)\), \(\mu = 1/(50N)\), and \(\mu = 1/(300N)\). Plot the convergence curve \((J(w(n))\) vs. \(n)\) for each case. For each of these three cases, plot the magnitude frequency responses (in dB) of \(w(n=20)\), \(w(n=200)\), and \(w(n=2000)\). Discuss what you observe.
3. Using the same notation developed in class, analytically show that the GSC, given the constraint matrix \( C = s(\omega_0) \) with associated constraint gain \( g = 1 \), is equivalent to the MVDR beamformer.

*Hints:* 
- a) See Section II of the Breed & Strauss paper and Section II (the two paragraphs above equation 2) of the Apolinario, et al paper.
- b) For matrix \( A \) full-rank and square, pseudo-inverse(\( A \)) = inverse(\( A \)) where pseudo-inverse(\( A \)) = \((A^H A)^{-1} A^H\) by Least-Squares.
- c) Use the identity for the inverse of a partitioned matrix.

4. Data set P4.mat contains a known signal \( x \) of length \( M = 100 \) that we wish to deconvolve from other unknown signals.

a) Generate the normalized matched filter as \( h_{NMF} = x^H / (x^H x) \) and plot the convolution of this filter with the signal \( x \). Note that a filter output is an amplitude when plotting in dB.

b) Implement the Least-Squares mismatched filter \( h_{MMF} \) with a length of \( 3M \) and place the ‘1’ in the elementary vector in element \( m = 2M \). Once determined, normalize the MMF as

\[
h_{NMMF} = \frac{h_{MMF}}{(h_{MMF}^H h_{MMF})^{1/2} (x^H x)^{1/2}}
\]

to allow for determination of the loss in SNR. Plot (in dB) the convolution of the normalized MMF with the signal \( x \). Comment on what you observe. *(Hint: the ‘toeplitz’ command is useful for constructing the matrix \( A \) comprised of delay shifted versions of \( x \))*

c) Repeat part b) except modify the matrix \( A \) by replacing the values in the \((m-1)th\) and \((m+1)th\) rows with zeros. Comment on what you observe.

d) Repeat parts b) and c) except incorporate a diagonal load term that is 2% of the largest eigenvalue of the matrix \( A^H A \).

e) For each of the four normalized mismatched filters obtained in the above steps, compute

\[
mismatch \ loss = -20 \log_{10} \left( \frac{\max |h_{NMMF} * x|}{\max |h_{NMF} * x|} \right) dB,
\]

for * representing convolution. Comment on what you observe.

5. Data set P4.mat also contains the received signal \( y(n) \) that is the result of convolving the known signal \( x \) from Prob. 4 with some unknown system. Apply each of the 5 filters (MF and 4 MMF) from Prob. 4 to perform deconvolution to estimate the unknown system. Plot the results (in dB) and comment on what you observe.
6. Data set P6.m contains time samples for an $M = 30$ element uniform linear array (ULA) with half-wavelength element spacing. Plot in dB and according to electrical angle:
   a) the non-adaptive spatial power spectrum (see Appendix A; same as previous exam),
   b) the MVDR spatial power spectrum using $\mathbf{R} = (1/L)\mathbf{X}\mathbf{X}^H$,
   c) the MVDR spatial power spectrum using $\mathbf{R}$ obtained from forward/backward averaging via Appendix B,
   d) and each of the above two MVDR cases diagonally loaded with $(10)\mathbf{I}$.

What do you observe from the various results (there are 5 cases)?

**Appendix A: Determining a non-adaptive power spectrum**
Define the matrix $\mathbf{S}$ comprised of steering vectors “over-sampled” in angle as described in Exam 1. Given the set of snapshots $\mathbf{X} = [\mathbf{x}_1 \mathbf{x}_2 \cdots \mathbf{x}_L]$, for each spatial angle $\phi$ compute the non-adaptive power spectrum as

$$p(\phi) = \frac{1}{L} \sum_{k=1}^{L} |s^H(\phi) \mathbf{x}_k|^2.$$ 

**Appendix B – Forward-Backward Averaging**
One way to form the forward-backward averaged covariance matrix given the $M \times L$ data matrix $\mathbf{X}$ for a uniform linear array is

$$\mathbf{R}_{FB} = \frac{1}{2L} (\mathbf{X}\mathbf{X}^H + \mathbf{J}\mathbf{X}\mathbf{X}^T\mathbf{J})$$

where $\mathbf{J}$ is the $M \times M$ reflection matrix (looks like a reversed identity matrix) defined as

$$\mathbf{J} = \begin{bmatrix}
0 & \cdots & 0 & 1 \\
\vdots & \ddots & \vdots & \vdots \\
0 & \cdots & 1 & 0 \\
1 & 0 & \cdots & 0 
\end{bmatrix}.$$