6.1 Solution for question 1, HW 2

$s_1, s_2, s_3$ are signals of finite energy, i.e. $\|s_i\|_2 < \infty$. Find scalars $a_1, a_2$ to minimize $\|a_1 s_1 + a_2 s_2 + s_3\|$. 

$s_1, s_2$ span a two dimensional linear subspace. We are looking for the point in this linear subspace that is closest to $-s_3$. As this is euclidean norm, the vector connecting $-s_3$ with the closest point in the subspace is orthogonal to the subspace, which means that it is orthogonal to $s_1$ and to $s_2$. Writing this in equation form we have 

$$s_1 \cdot (a_1 s_1 + a_2 s_2 - (-s_3)) = 0$$

and

$$s_2 \cdot (a_1 s_1 + a_2 s_2 - (-s_3)) = 0$$

which translates to two linear equations for $a_1, a_2$: 

$$a_1 s_1 \cdot s_1 + a_2 s_1 \cdot s_2 = -s_1 \cdot s_3$$

and 

$$a_1 s_2 \cdot s_1 + a_2 s_2 \cdot s_2 = -s_2 \cdot s_3$$

Solving this system of equations yields the values for $a_1$ and $a_2$.

6.2 Adaptive Noise cancellation

![Figure 6.1. Noise Cancellation](image-url) A block diagram describing the relations between signals in a noise cancellation system.
Consider the following problem. Suppose we want to record somebody speaking in a noisy environment. We use two microphones, one that is far from the speaker, and captures only the noise \( q(n) \), and one that is close to the speaker, and captures \( y(n) \), which is a mixture of the speaker’s voice \( x(n) \) and the noise \( q(n) \). Our goal is to remove the noise from \( y(n) \) to generate a signal close to the second microphone and generate a clean recording. See Figure 6.1.

It might seem that all we need to do is take the difference \( y(n) - q(n) \). However, that assumes that the amplitude in the noise is the same in the two microphones. Also, it assumes that the noise sound wave reaches both microphones at the same time. To see the problem with the second assumption, suppose that the two microphones are 1 meter apart and that the microphones are sampled 48,000 times per second. As the speed of sound is 300 m/sec the delay between the two microphones corresponds to 160 samples or to a full wavelength of a 300Hz wave - not a negligible amount.

Indoor setups bring additional complications, the sound does not necessarily travel between the two microphones along the straight line connecting them. In addition to the direct path the sound can travel along paths corresponding to reflections from walls, i.e., echoes. The attenuation along these different paths is likely to be different. Furthermore, the attenuation along different paths depends on the frequency (pitch) of the sound.

We model all of these complex interactions as a linear system which transforms the original noise signal \( q(n) \) to the noise signal at the second microphone \( u(n) \), we denote the transfer function of the room’s acoustics by \( H(z) = Q(z)/U(z) \).

In order to cancel the noise we therefore need to generate an estimate of \( H(Z) \). In general, this is an underdetermined problem. However, we can solve it if we assume that the speech signal and the noise signal are uncorrelated.

In the following we define the statistical properties of stationary random signals and show how the LMS algorithm can be used to solve the noise cancellation problem.

### 6.3 Randomness and Stationarity

When we talk about noise we think of a signal that has no structure, a signal that cannot be predicted. We formalize this by defining a distribution over \( \cdots, x(-3), x(-2), x(-1), x(0), x(1), x(2), x(3), \ldots \). This means that each event corresponds to a whole sequence. In this generality we can express any distribution.

In general, we tend to make the assumption that our distribution is stationary. This means that the distribution does not change if the time index is shifted. Specifically, it means that any set of conditions of the form \( a_i \leq x(n + k_i) \leq b_i \) defines a set whose probability is independent of \( n \). For example

\[
P(1 \leq x(n) \leq 1.1 \text{ and } x(n - 3) > 5 \text{ and } x(n + 100) < 7)
\]

is independent of the index \( n \). Essentially, this means that the probabilities of patterns along the sequence depends only on their relative locations and not on the absolute location in the sequence.

This definition of stationarity is too strict for practical applications. For our applications, which are based on measuring performance using mean squared error, it is enough to use a weaker definition of stationarity called broad sense stationarity. According to this definition a random signal \( x \) is stationary if \( E[x(n)] \) does not depend on \( n \) and \( E(x(n)x(n + k)) \) does not depend on \( n \).

If a stationary process is also ergodic then we can replace the expectations with respect to the distribution over the sequences (also called the ensamble average), with the expectation taken over all positions in a single random sequence (also called the time average). In other words, instead of taking the sum over different samples:

\[
E_{ensamble}(x(n)x(n-k)) = \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} (x_i(n)x_i(n-k))
\]
we can take the sum over all positions in a single sequence:

\[ E_{\text{time}}(x(n)x(n-k)) = \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} (x_1(i)x_1(i-k)) \]

and, with probability one, we’ll get the same answer from both.

### 6.4 Autocorrelation and Cross-Correlation

Suppose we have two random signals \( x \) and \( y \) that are both stationary and ergodic. Suppose further than their means are zero: \( E[x(n)] = E[y(n)] = 0 \).

We can characterize the correlations between \( x \) and \( y \) by the cross-correlation function:

\[ \phi_{xy}(k) = E(x(n)y(n-k)) = \sum_{n=-\infty}^{\infty} x(n)y(n-k) \]

the second equality follows from the ergodicity assumption.

As a special case, the cross correlation between a signal and itself is called the auto-correlation: Similarly, we characterize the relation between \( x \) and \( y \) using the cross-correlation function:

\[ \phi_{xx}(k) = E(x(n)x(n-k)) = \sum_{n=-\infty}^{\infty} x(n)x(n-k) \]

If \( x \) and \( y \) are independent signals then \( C_{xy}(k) = 0 \) for all \( k \).

White noise in discrete time is a distribution over signals where the value at each time step is chosen independently at random according to a gaussian distribution with mean zero and variance one. It is easy to see that the auto-correlation function of white noise is a delta function centered at 0. White noise in continuous time has the same properties, however, it’s formal mathematical definition is quite involved and requires advanced tools from measure theory.

### 6.5 Noise cancellation continued

Our assumption is that the noise \( q \) and the speech signal \( x \) are independent and therefor their cross correlation should be zero for all \( k \). Similarly, the cross correlation between \( u \) and \( y \) is close to zero for all \( k \).

On the other hand, the modified noise signal \( u \) is highly correlated with \( q \), after all, it a function of the original noise \( q \). As cross-correlation is a linear operator, the cross correlation between the combined signal \( y = x + u \) and the original noise \( q \) is equal to the cross correlation between \( u \) and \( q \):

\[ \phi_{yq}(k) = \phi_{xq}(k) + \phi_{uq}(k) = \phi_{uq}(k) \]

for all \( k \).

Intuitively, this means that any correlation between \( y \) and \( q \) is due to \( u \): the noise component of \( y \). If we can subtract out \( u \) from \( y \) then we will be left with a signal that is uncorrelated with \( q \), namely \( x \). Our goal is to find a weight vector \( w \) such that the signal \( d = y - \sum_{k=0}^{\infty} w(k)z^{-k}q \) has zero cross correlations with \( q \). Viewing these signals as vectors, we want to find the projection of \( y(n) \) onto the linear subspace spanned by the vectors \( q, z^{-1}q, z^{-2}q, \ldots \) This projection has the property that it is the point on the subspace that has the minimal distance to \( q \).
To create a realizable weight vector, we use a weight vector of finite dimension $L + 1$. The choice of $L$ will depend on the delays and frequencies that we expect to have in our physical setup. We assume that $\phi_{xx}(l)$ is negligibly small for $l > L$.

The problem we described is a special case of the LMS problem that we analyzed in the previous class. In this case the correlation matrix $\mathbf{R}$ is defined by the autocorrelations and the vector $\mathbf{P}$ is defined by the cross correlations:

\[
\mathbf{R} = E[\mathbf{Q Q}^T] = \begin{bmatrix}
\phi_{qq}(0) & \phi_{qq}(1) & \phi_{qq}(2) & \cdots & \phi_{qq}(L) \\
\phi_{qq}(1) & \phi_{qq}(0) & \phi_{qq}(1) & \cdots & \phi_{qq}(L-1) \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\phi_{qq}(L) & \phi_{qq}(L-1) & \phi_{qq}(L-2) & \cdots & \phi_{qq}(0)
\end{bmatrix}
\]

and

\[
\mathbf{P} = [\phi_{qy}(0), \ldots, \phi_{qy}(L)];
\]

And the Weiner optimal solution is

\[
\mathbf{W} = \mathbf{R}^{-1}\mathbf{P}
\]

We can use the LMS algorithm to find the optimal solution:

\[
\mathbf{W}(n+1) = \mathbf{W}(n) + 2\mu\epsilon(n)[q(n), q(n-1), \ldots, q(n-L)]^T
\]

where

\[
\epsilon(n) = y(n) - \mathbf{W}(n)^T[q(n), q(n-1), \ldots, q(n-L)]^T
\]

and $\epsilon(n)$ is also the recovered speech signal.

6.6 Homework 4

Implement and test an adaptive echo canceller.

- Record a speech signal (or get one from the web). This will be your $x$ signal.
- Record a noise signal (noise from the computer fan works fine).
- Normalize the two signals so that they have the same length and the same energy. This will be your $q$ signal.
- Create some linear filter by placing pairs of zeros and poles inside the unit sphere. Use at least two pairs of zeros and two pairs of poles. This is your model of the room transfer function.
- apply the filter to $q$ (use the command filter in matlab to create $u$.
- sum $u$ and $x$ to create $y$
- Compute and plot the auto-correlation functions for $u$ and for $q$.
- Compute the FFT of $u$ at all windows $[1 : N], [2, N+1], [3 : N+2], \ldots$, average the amplitude (abs(fft)) for all of these windows. Plot the result. Take the FFT of the auto-correlation function for $u$. Plot both, they should be very close to each other. Do the same for $q$.
- Implement the LMS algorithm to recover $x$ from $y$ and $q$. Apply the generated sequence of weight vectors to the signal. Note that the reconstruction should improve over time as the weights of the filter converge to their optimal value.
• Generate .wav files of the original speech \( x \), the original noise \( q \), the deformed noise \( u \), the combined speech and noise \( y \) and the reconstructed speech \( \epsilon(n) \).

• email me yfreund@ucsd.edu a zip or tar file with matlab files, the plots and the .wav files. Make the subject line be CSE291, HW4