Boston University
Department of Electrical and Computer Engineering

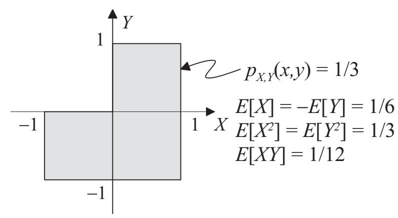
EC505 STOCHASTIC PROCESSES

Problem Set No. 9

Fall 2016

Issued: Monday, Nov. 28, 2016 **Due:** Wednesday, Dec. 7, 2016

Problem 9.1 The random variables X and Y are uniformly distributed over the region shown in the figure.



(a) Find $\widehat{x}_{BLS}(y)$ the Bayes least square estimate of X given Y and λ_{BLS}, the associated error variance.

(b) Find $\widehat{x}_{LLS}(y)$ the linear least square estimate of X based on Y and λ_{LLS} the associated error variance.

(c) Consider the following "modified least squares" cost function:

$$J(\widehat{x}, x) \quad = \quad \begin{cases} K(\widehat{x}-x)^2 & x > 0 \\ (\widehat{x}-x)^2 & x < 0 \end{cases}$$

where $K > 1$ is a constant. Determine $\widehat{x}_{MLS}(y)$, Bayes estimate of X corresponding to this cost criterion.

Hint: Examine the conditional form of the defining Bayes optimization problem and see if you can find a similar problem with the standard structure but a different density.

Solution:

(a) By inspection:

$$\widehat{x}_{BLS}(y) \quad = \quad E[X|Y] = \begin{cases} \frac{1}{2} & y > 0 \\ 0 & y < 0 \end{cases}$$

Now lets find the error variance. First we know that $\lambda_{BLS} = E[\lambda_{x|y}]$. Essentially by inspection we obtain:

$$\lambda_{x|y} \quad = \quad \begin{cases} \frac{1}{12} & y > 0 \\ \frac{1}{3} & y < 0 \end{cases}$$

Thus:

$$\begin{aligned} \lambda_{BLS} &= E[\lambda_{x|y}] = \lambda_{x|y>0}\Pr(Y>0) + \lambda_{x|y<0}\Pr(Y<0) = \frac{1}{12}\Pr(Y>0) + \frac{1}{3}\Pr(Y<0) \\ &= \frac{1}{12}\frac{1}{3} + \frac{1}{3}\frac{2}{3} = \frac{1}{36} + \frac{2}{9} = \frac{9}{36} \\ &= \frac{1}{4} \end{aligned}$$

(b) First we need to find σ_x^2, σ_y^2, σ_{xy}:

$$\sigma_x^2 = E[X^2] - E[X]^2 = \frac{1}{3} - \frac{1}{36} = \frac{11}{36} \qquad \sigma_y^2 = E[Y^2] - E[Y]^2 = \frac{1}{3} - \frac{1}{36} = \frac{11}{36}$$
$$\sigma_{xy} = E[XY] - E[X]E[Y] = \frac{1}{12} + \frac{1}{36} = \frac{1}{9}$$

Thus:

$$\begin{aligned}\widehat{x}_{LLS}(y) &= m_x + \frac{\sigma_{xy}}{\sigma_y^2}(y - m_y) = \frac{1}{6} + \left(\frac{4}{36}\right)\left(\frac{36}{11}\right)\left(y + \frac{1}{6}\right) \\ &= \frac{1}{6} + \frac{4}{11}\left(y + \frac{1}{6}\right) = \frac{4}{11}y + \frac{5}{22}\end{aligned}$$

$$\lambda_{LLS} = \sigma_x^2 - \frac{\sigma_{xy}^2}{\sigma_y^2} = \frac{11}{36} - \left(\frac{4}{36}\right)^2\frac{36}{11} = \frac{11}{36} - \frac{16}{36 \cdot 11} = \frac{11}{36} - \frac{4}{99} = \frac{105}{396} = 0.265$$

(c) We know that the optimal solution is to minimize the expected cost:

$$\widehat{x}_{MLS}(y) = \arg\min_x E[J(\widehat{x}, x)] = \arg\min_x E[J(\widehat{x}, x) \mid y]$$

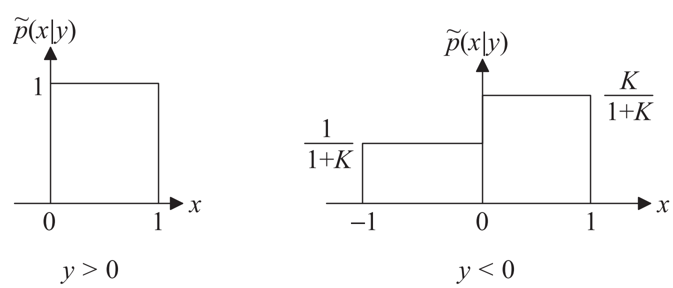
Now:

$$\begin{aligned}E[J(\widehat{x}, x) \mid y] &= \int J(\widehat{x}, x)\, p_{X|Y}(x|y)\, dx \\ &= \int_{x<0} (\widehat{x} - x)^2 p_{X|Y}(x|y)\, dx + \int_{x>0} (\widehat{x} - x)^2 K p_{X|Y}(x|y)\, dx\end{aligned}$$

Note that compared to the standard least squares problem, this cost function has the effect of scaling $p_{X|Y}(x|y)$ by a factor of K when $x > 0$. Thus we can simply find the standard BLS solution for the problem obtained by replacing $p_{X|Y}(x|y)$ with

$$\tilde{p}_{X|Y}(x|y) = \begin{cases} \frac{K p_{X|Y}(x|y)}{C} & x > 0 \\ \frac{p_{X|Y}(x|y)}{C} & x < 0 \end{cases}$$

where $C = \int_{x<0} p_{X|Y}(x|y)\, dx + \int_{x>0} K p_{X|Y}(x|y)\, dx$ is a normalization constant. In particular, the optimal estimate will be obtained as the conditional mean with respect to this new, scaled conditional density. The scaled conditional density $\tilde{p}_{X|Y}(x|y)$ is shown in the following figure.

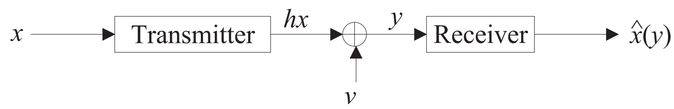


It is thus obvious that the new estimate is given by:

$$\widehat{x}_{MLS}(y) = \begin{cases} \frac{1}{2} & y > 0 \\ \frac{1}{2}\frac{(K-1)}{(K+1)} & y < 0 \end{cases}$$

Problem 9.2

Consider the communication system shown below. The message X is an $N(0, \sigma_x^2)$ random variable. The transmitter output is HX, and the receiver input is: $Y = HX + V$, where V is an $N(0, r)$ random variable that is statistically independent of X.



Suppose the transmitter is subject to intermittent failure, i.e., H is a random variable taking the values 0 and 1 with probabilities $1-p$ and p, respectively. Assume H is statistically independent of both X and V.

(a) Find $\hat{x}_{LLS}(y)$, the linear least-squares estimate of X based on observation of Y, and λ_{LLS}, its resulting error variance and mean-square estimation error.

(b) Show that: $E[X \mid Y = y] = \sum_{i=0}^{1} \Pr[H = i | Y = y]\, E[X \mid Y = y, H = i]$. Hint: Consider using iterated expectation.

(c) Find $\hat{x}_{BLS}(y)$, the Bayes least-squares estimate of X based on observation of Y.

Solution:

(a) Let's first calculate the pieces we will need:

$$
\begin{aligned}
m_X &= E(X) = 0 \\
m_Y &= E[Y] = E\left[HX + V\right] = E[H]E[X] + E[V] = 0 \\
\lambda_X &= \sigma_x^2 \\
\lambda_{XY} &= E[(X - m_X)(Y - m_Y)] = E[XY] = E[X(HX + V)] = E[H]E[X^2] + E[XV] = p\sigma_x^2 \\
\lambda_Y &= E[Y^2] - m_Y^2 = E[Y^2] = E[(HX + V)^2] = E[H^2X^2] + E[V^2] + 2E[HVX] = E[H^2]E[X^2] + r + 0 \\
&= p\sigma_x^2 + r \\
\Longrightarrow \\
\hat{x}_{LLS}(y) &= m_X + \frac{\lambda_{XY}}{\lambda_Y}(y - m_Y) \\
&= \left[\frac{p\sigma_x^2}{p\sigma_x^2 + r}\right] y \\
\lambda_{LLS} &= \lambda_X - \frac{\lambda_{XY}^2}{\lambda_Y} = \sigma_x^2 - \frac{(p\sigma_x^2)^2}{p\sigma_x^2 + r} = \frac{\sigma_x^4 p(1-p) + \sigma_x^2 r}{p\sigma_x^2 + r}
\end{aligned}
$$

Since our estimator is unbiased and this is a scalar problem MSE is equal to the variance.

(b) For those comfortable with iterated expectation, the short proof is the following:

$$
E[X \mid Y = y] \;=\; E\left[E\left[X \mid H = i, Y = y\right]\right] = \sum_{i=0}^{1} \Pr(H = i \mid Y = y) E[X \mid H = i, Y = y]
$$

A more detailed way of showing this is the following:

$$
\begin{aligned}
E[X \mid Y = y] &= \int_{-\infty}^{\infty} X p_{X|Y}(x \mid Y = y)\, dx = \int_{-\infty}^{\infty} X \sum_{i=0}^{1} p_{X,H|Y}(x, H = i \mid Y = y)\, dx \\
&= \int_{-\infty}^{\infty} X \sum_{i=0}^{1} p_{X|H,Y}(x \mid H = i, Y = y) \Pr(H = i \mid Y = y)\, dx \\
&= \sum_{i=0}^{1} \Pr(H = i \mid Y = y) E[X \mid H = i, Y = y]
\end{aligned}
$$

(c) We want to compute $\hat{x}_{BLS}(y) = E[X \mid Y = y]$. But from part (b) we know that we can find this conditional expectation as a weighted sum of expectations conditioned on *both* Y and H, (i.e. in terms of $E[X \mid Y = y, H = i]$), where the weights are given by $\Pr[H = i \mid Y = y]$. So let's go find these quantities. Note that there are only 2 possible values of H so we can just enumerate them. When $H = 0$ we have

$$E[X \mid Y = y, H = 0] \quad = \quad E[X \mid Y = v] = E[X] = 0$$

When $H = 1$, then $Y = X + V$, so $E[X \mid Y = y, H = 1]$ is the Bayes least squares estimate for X given $Y = X + V$, where X and V are independent Gaussians hence the estimator is linear (this was the whole point of trying to write things in terms of the expectation conditioned on knowledge of H; it decouples things). Thus we immediately know that:

$$E[X \mid Y = y, H = 1] \quad = \quad m_{x|h=1} + \frac{\lambda_{xy|h=1}}{\lambda_{y|h=1}}(y - m_{y|h=1}) = 0 + \frac{\sigma_x^2}{\sigma_x^2 + r}(y - 0) = \frac{\sigma_x^2 y}{\sigma_x^2 + r}$$

Now we need to compute the quantities $\Pr(H = 0 \mid Y = y)$ and $\Pr(H = 1 \mid Y = y)$. Actually, noting that $E[X \mid Y = y, H = 0] = 0$, we can see that we really don't need $\Pr(H = 0 \mid Y = y)$, since it will be multiplied by 0 in the sum. So let's just find $\Pr(H = 1 \mid Y = y)$:

$$\begin{aligned}\Pr(H = 1 \mid Y = y) \quad &= \quad \frac{p_{Y|H}(y \mid H = 1)\Pr(H = 1)}{p_Y(y)} = \frac{p_{Y|H}(y \mid H = 1)\Pr(H = 1)}{\sum_{i=0}^{1} p_{Y|H}(y \mid H = i)\Pr(H = i)} \\ &= \quad \frac{p\, N(y; 0, \sigma_x^2 + r)}{p\, N(y; 0, \sigma_x^2 + r) + (1 - p)\, N(y; 0, r)}\end{aligned}$$

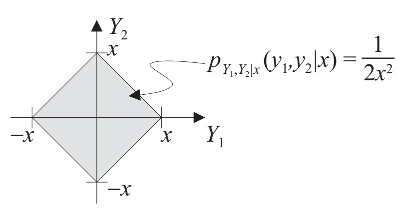
Now putting the pieces together we find:

$$\begin{aligned}\hat{x}_{BLS}(y) \quad &= \quad E[X \mid Y = y] = \Pr(H = 0 \mid Y = y)E[X \mid H = 0, Y = y] + \Pr(H = 1 \mid Y = y)E[X \mid H = 1, Y = y] \\ &= \quad \frac{p\, N(y; 0, \sigma_x^2 + r)}{p\, N(y; 0, \sigma_x^2 + r) + (1 - p)\, N(y; 0, r)}\ \frac{\sigma_x^2}{\sigma_x^2 + r}\, y\end{aligned}$$

Note that the answer is <u>not</u> linear.

Problem 9.3 (Old Exam Problem)

We wish to estimate an unknown, deterministic parameter x from two observations y_1 and y_2 that are uniformly distributed over the diamond shaped region parameterized by $x > 0$ shown in the figure. Hint: All parts can be answered with minimal calculation with some thought.



(a) Find $p_{Y_1|x}(y_1|x)$.

(b) Find a maximum-likelihood estimate of x based on y_1 alone.

(c) Find a maximum-likelihood estimate of x given both y_1 and y_2.

Solution: See Problem Set 8.6 Solutions.

Problem 9.4

The purpose of this problem is to determine the (unknown) probability of heads when flipping a particular coin.

(a) Suppose that the coin is flipped N times in succession, each toss statistically independent of all others, each with (unknown) probability p of heads. Let n be the number of heads that is observed. Find the maximum likelihood (ML) estimate $\widehat{p}_{ML}(n)$ of p based on knowledge of n.

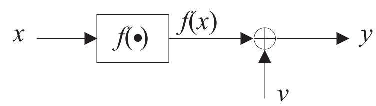
(b) Evaluate the bias $E[p-\widehat{p}_{ML}(n) \mid p]$ and the mean-square error $E[(p-\widehat{p}_{ML}(n))^2 \mid p]$ of the ML estimate.

(c) Is the ML estimate efficient (i.e. does the mean square error attain the Cramer-Rao bound)? An estimate is termed "consistent" if its mean square error goes to zero as $N \to \infty$. Is the ML estimate of p consistent. Briefly explain.

Solution: See Problem Set 8.7 Solutions.

Problem 9.5

A hypothetical communication system is shown the figure below:



The deterministic parameter x is unknown, and the receiver noise V is assumed to be a Gaussian random variable with mean 0 and variance σ^2. The function $f(\cdot)$ is given by

$$f(x) = (1 - e^{|x|})\,\mathrm{sgn}(x)$$

where $\mathrm{sgn}(x)$ denotes the sign of x.

(a) Find the maximum likelihood estimate of x based on observation of Y.

(b) Find an explicit lower bound on the mean-square estimation error of an arbitrary unbiased estimate of x based on observation of Y.

Solution:

(a) Our observation equation is:

$$Y = f(x) + V$$

The parameterized density for Y is a Gaussian $P(y|x) = N(y; f(x), \sigma^2)$. If we take the derivative of the natural log and set it equal to zero we get:

$$\frac{\partial}{\partial x} \ln P_{Y|x}(y|x) \quad = \quad \frac{(y - f(x))}{\sigma^2} \frac{\partial f(x)}{\partial x} = 0$$

Thus the ML estimate is given by the inverse function applied to y:

$$\widehat{x}_{ML}(y) \quad = \quad f^{-1}(y) = -\mathrm{sgn}(y) \ln(1 + |y|)$$

(b) The lower bound on the MSE for unbiased estimators is given by the CR bound. First find the Fisher information:

$$\begin{aligned} I_y(x) &= -E\left[\frac{\partial^2}{\partial x^2} \ln \Pr(y \mid x)\right] = -E\left[\frac{(y - f(x))}{\sigma^2} f''(x) - \frac{[f'(x)]^2}{\sigma^2}\right] \\ &= \frac{[f'(x)]^2}{\sigma^2} = \frac{e^{2|x|}}{\sigma^2} \end{aligned}$$

Thus a lower bound on MSE for unbiased estimators for this problem is given by:

$$\mathrm{MSE} \geq \frac{1}{I_y(x)} = \frac{\sigma^2}{e^{2|x|}}$$

Problem 9.6

Suppose we observe a random N-dimensional vector $\underline{Y}$, whose components are independent, identically-distributed Gaussian random variables, each with mean x_1 and variance x_2.

(a) Suppose the mean x_1 is unknown but the variance x_2 is known. Find the maximum likelihood estimate of the mean x_1 based on observation of $\underline{Y}$. Evaluate the bias and the mean-square error for this estimate. Is this estimate efficient?

(b) Suppose the mean x_1 is known but the variance x_2 is unknown. Find the maximum likelihood estimate of the variance x_2 based on observation of $\underline{Y}$. Evaluate the bias and the mean-square error for this estimate. Is this estimate efficient? Explain. Note: If $z \sim N(m, \lambda)$, then $(z-m)^2/\lambda$ is a Chi-squared random variable with one degree of freedom, which has variance 2.

(c) Suppose both the mean x_1 and the variance x_2 are unknown. Find $\widehat{\underline{x}}_{ML}$, the maximum likelihood estimate of:

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

based on observation of $\underline{Y}$. Evaluate the bias in both component estimates and $E[(x_1 - \widehat{x}_{1_{ML}})^2]$. Evaluate $E[(x_2 - \widehat{x}_{2_{ML}})^2]$ for $N = 2$. Compare with your results from parts (a) and (b). Note: If $z \sim N(0, \lambda)$, then $E[z^n] = \lambda^{n/2} \cdot 1 \cdot 3 \cdot 5 \cdots (n-1)$ if n is even and $E[z^n] = 0$ if n is odd.

Solution: First note that the pdf for $\underline{Y}$ in terms of x_1 and x_2 is given by:

$$P_{Y|\underline{x}}(y \mid \underline{x}) = \prod_{i=1}^{N} P_{Y_i|\underline{x}}(y \mid \underline{x}) = \frac{1}{(2\pi x_2)^{N/2}}\, e^{-\frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2}$$

(a) Now we assume that x_2 is known and fixed. So

$$\widehat{x}_{1,ML} = \arg\max_{x_1} P_{Y|x_1}(y \mid x_1) = \arg\max_{x_1} \ln P_{Y|x_1}(y \mid x_1) = \arg\max_{x_1} \left\{ \ln\left(\frac{1}{(2\pi x_2)^{N/2}}\right) - \frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2 \right\}$$

Now taking the derivative of the last expression with respect to x_1 and setting it equal to zero we find:

$$\begin{aligned}
\frac{\partial}{\partial x_1}\left[\ln\left(\frac{1}{(2\pi x_2)^{N/2}}\right) - \frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2\right] &= \frac{1}{x_2}\sum_{i=1}^{N}(y_i - x_1) = \frac{1}{x_2}\left(\sum_{i=1}^{N} y_i - N x_1\right) = 0 \\
\Longrightarrow \quad \widehat{x}_{1,ML} &= \frac{1}{N}\sum_{i=1}^{N} y_i
\end{aligned}$$

To check the bias we calculate:

$$E[x_1 - \widehat{x}_{1,ML} \mid x_1] = x_1 - E\left[\frac{1}{N}\sum_{i=1}^{N} y_i \mid x_1\right] = x_1 - \frac{1}{N}\sum_{i=1}^{N} E[y_i \mid x_1] = x_1 - \frac{1}{N}\sum_{i=1}^{N} x_1 = 0$$

Therefore it is unbiased. Now let's calculate the MSE:

$$\begin{aligned}
\text{MSE} &= E[(\widehat{x}_{1,ML} - x_1)^2 \mid x_1] = E[(\widehat{x}_{1,ML} - E(\widehat{x}_{1,ML} \mid x_1))^2 \mid x_1] = \text{Var}(\widehat{x}_{1,ML}|x_1) = \frac{1}{N^2}\sum_{i=1}^{N}\text{Var}\,(Y_i \mid x_1) \\
&= \frac{x_2}{N}
\end{aligned}$$

where the second to last equality follows from the fact that the Y_i are independent. To check efficiency we need to find the CR bound, or equivalently the Fisher information in $\underline{Y}$ about x_1:

$$I_{\underline{Y}}(x_1) = -E\left[\frac{\partial^2}{\partial x_1^2}\ln \Pr(\underline{Y} \mid x_1)\right] = -E\left[\frac{\partial^2}{\partial x_1^2}\left(\ln\left(\frac{1}{(2\pi x_2)^{N/2}}\right) - \frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2\right)\right] = \frac{N}{x_2}$$

Since $1/I_{\underline{Y}}(x_1) = E[(\widehat{x}_{1,ML} - x_1)^2 \mid x_1]$ the ML estimate of x_1 alone is indeed efficient.

(b) Now we assume that x_1 is known and fixed. The ML estimate of x_2 is given by:

$$\widehat{x}_{2,ML} = \arg\max_{x_2} P_{Y|x_2}(y \mid x_2) = \arg\max_{x_2} \ln P_{Y|x_2}(y \mid x_2) = \arg\max_{x_2}\left\{\ln\left(\frac{1}{(2\pi x_2)^{N/2}}\right) - \frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2\right\}$$

Now taking the derivative of the last expression with respect to x_2 and setting it equal to zero we find:

$$\begin{aligned}
\frac{\partial}{\partial x_2}\left[\ln(2\pi)^{-N/2} - \frac{N}{2}\ln x_2 - \frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2\right] &= -\frac{N}{2x_2} + \frac{1}{2x_2^2}\sum_{i=1}^{N}(y_i - x_1)^2 = 0 \\
\Longrightarrow \quad \widehat{x}_{2,ML} &= \frac{1}{N}\sum_{i=1}^{N}(y_i - x_1)^2
\end{aligned}$$

To check the bias we calculate:

$$E[x_2 - \widehat{x}_{2,ML} \mid x_2] = x_2 - E\left[\frac{1}{N}\sum_{i=1}^{N}(y_i - x_1)^2 \middle| x_2\right] = x_2 - \frac{1}{N}\sum_{i=1}^{N}\mathrm{Var}\,(Y_i \mid x_2) = x_2 - \frac{1}{N}\sum_{i=1}^{N} x_2 = 0$$

Therefore it is unbiased. Now let's calculate the MSE:

$$\begin{aligned}
\text{MSE} &= E[(\widehat{x}_{2,ML} - x_2)^2 \mid x_2] = E[(\widehat{x}_{2,ML} - E(\widehat{x}_{2,ML} \mid x_2))^2 \mid x_2] = \mathrm{Var}(\widehat{x}_{2,ML}|x_2) \\
&= \mathrm{Var}\left(\frac{1}{N}\sum_{i=1}^{N}(y_i - x_1)^2 \middle| x_2\right) = \frac{1}{N^2}\sum_{i=1}^{N}\mathrm{Var}\left((Y_i - x_1)^2 \mid x_2\right) = \frac{2x_2^2}{N}
\end{aligned}$$

where the second to last equality follows from the fact that the Y_i are independent and the last inequality follows from the fact that $(Y_i - x_1)^2/x_2$ is a Chi-squared random variable with one degree of freedom, and hence has variance 2. To check efficiency we need to find the CR bound, or equivalently the Fisher information in $\underline{Y}$ about x_2:

$$\begin{aligned}
I_{\underline{Y}}(x_2) &= -E\left[\frac{\partial^2}{\partial x_2^2}\ln \Pr(\underline{Y} \mid x_2)\right] = -E\left[\frac{\partial^2}{\partial x_2^2}\left(\ln(2\pi)^{-N/2} - \frac{N}{2}\ln x_2 - \frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2\right)\right] \\
&= -E\left[\frac{N}{2x_2^2} - \frac{1}{x_2^3}\sum_{i=1}^{N}(y_i - x_1)^2\right] = -\left(\frac{N}{2x_2^2} - \frac{Nx_2}{x_2^3}\right) = \frac{N}{2x_2^2}
\end{aligned}$$

Since $1/I_{\underline{Y}}(x_2) = E[(\widehat{x}_{2,ML} - x_2)^2 \mid x_2]$ the ML estimate of x_2 alone is also efficient.

(c) Now we assume that both x_1 ans x_2 are unknown. The ML estimate of x_1 and x_2 are given by:

$$\widehat{\underline{x}}_{ML} = \arg\max_{\underline{x}} P_{Y|\underline{x}}(y \mid \underline{x}) = \arg\max_{\underline{x}} \ln P_{Y|\underline{x}}(y \mid \underline{x}) = \arg\max_{\underline{x}}\left\{\ln\left(\frac{1}{(2\pi x_2)^{N/2}}\right) - \frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2\right\}$$

Now taking derivatives of the last expression with respect to x_1 and x_2 and setting them equal to zero we find:

$$
\begin{aligned}
\left.\frac{\partial}{\partial x_1}\left[\ln\left(\frac{1}{(2\pi x_2)^{N/2}}\right) - \frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2\right]\right|_{\substack{x_1 = \widehat{x}_{1,ML} \\ x_2 = \widehat{x}_{2,ML}}} &= \frac{1}{\widehat{x}_{2,ML}}\sum_{i=1}^{N}(y_i - \widehat{x}_{1,ML}) \\
&= \frac{1}{\widehat{x}_{2,ML}}\left(\sum_{i=1}^{N} y_i - N\widehat{x}_{1,ML}\right) = 0 \\
\Longrightarrow \qquad \widehat{x}_{1,ML} &= \frac{1}{N}\sum_{i=1}^{N} y_i
\end{aligned}
$$

Note that this is the <u>same</u> estimate we got before when we assumed that we knew x_2. Thus we see that the ML estimate of x_1 does not depend on knowledge of x_2! In words, the ML estimate of the mean of a Gaussian random variable is the same whether or not we know the variance. Now let's proceed to x_2:

$$
\begin{aligned}
\left.\frac{\partial}{\partial x_2}\left[\ln(2\pi)^{-N/2} - \frac{N}{2}\ln x_2 - \frac{1}{2x_2}\sum_{i=1}^{N}(y_i - x_1)^2\right]\right|_{\substack{x_1 = \widehat{x}_{1,ML} \\ x_2 = \widehat{x}_{2,ML}}} &= -\frac{N}{2\widehat{x}_{2,ML}} + \frac{1}{2\widehat{x}^2_{2,ML}}\sum_{i=1}^{N}(y_i - \widehat{x}_{1,ML})^2 = 0 \\
\Longrightarrow \qquad \widehat{x}_{2,ML} &= \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{x}_{1,ML})^2
\end{aligned}
$$

So we see that the ML estimate of x_2 <u>depends</u> on the ML estimate of x_1!

Now for the bias. Following the same argument as before we can see that $\widehat{x}_{1,ML}$ is unbiased and efficient. However for $\widehat{x}_{2,ML}$ we find

$$
\begin{aligned}
E[x_2 - \widehat{x}_{2,ML} \mid x_1, x_2] &= x_2 - E\left[\frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{x}_{1,ML})^2\right] \\
&= x_2 - \frac{1}{N}\sum_{i=1}^{N} E\left[\left(y_i - \frac{1}{N}\sum_{j=1}^{N} y_j\right)^2\right] \\
&= x_2 - \frac{1}{N}\sum_{i=1}^{N}\left(E(Y_i^2) - \frac{2}{N}\sum_{j=1}^{N} E(Y_i Y_j) + \frac{1}{N^2}\sum_{j=1}^{N}\sum_{\ell=1}^{N} E(Y_j Y_\ell)\right) \\
&= x_2 - \frac{1}{N^2}\sum_{i=1}^{N}\left(N(x_2 + x_1^2) - 2(Nx_1^2 + x_2) + \frac{1}{N}(N^2 x_1^2 + N x_2)\right) \\
&= x_2 - \frac{N-1}{N}x_2 = \frac{1}{N}x_2
\end{aligned}
$$

So $\widehat{x}_{2,ML}$ is biased when both x_1 and x_2 are unknown and thus can't be efficient.

Now for the MSE of $\widehat{x}_{1,ML}$:

$$
\begin{aligned}
\text{MSE} &= E[(\widehat{x}_{1,ML} - x_1)^2 \mid x_1] = E\left[(\widehat{x}_{1,ML} - E(\widehat{x}_{1,ML}))^2 \mid x_1\right] = \text{Var}(\widehat{x}_{1,ML} \mid x_1) = \frac{1}{N^2}\sum_{i=1}^{N}\text{Var}\left(Y_i \mid x_1\right) \\
&= \frac{x_2}{N}
\end{aligned}
$$

Finally for the MSE of $\widehat{x}_{2,ML}$ for $N = 2$:

$$MSE = E[(x_2 - \widehat{x}_{2,ML})^2 \mid x_1, x_2]$$

For $N = 2$

$$\begin{aligned} \widehat{x}_{1,ML} &= \frac{y_1 + y_2}{2} \\ \widehat{x}_{2,ML} &= \frac{(y_1 - y_2)^2}{4} \end{aligned}$$

This results in:

$$MSE = E\left[\left(x_2 - \frac{(y_1 - y_2)^2}{4}\right)^2\right]$$

For the zero mean Gaussian random variable $(Y_1 - Y_2)$ with variance $2x_2$, we have the following identity

$$E\left[(y_1 - y_2)^n\right] = \begin{cases} (2x_2)^{n/2} 1 \cdot 3 \cdot 5 \cdots (n-1) & \text{n even} \\ 0 & \text{n odd} \end{cases}$$

Using this property, we have

$$\begin{aligned} MSE &= E\left[x_2^2 - 2x_2 \frac{(y_1 - y_2)^2}{4} + \frac{(y_1 - y_2)^4}{16}\right] \\ &= x_2^2 - x_2^2 + \frac{3}{4} x_2^2 \\ &= \frac{3}{4} x_2^2 \end{aligned}$$

At a first glance this result may seem counter intuitive when compared to the MSE expression obtained in part (b). For $N = 2$ the MSE for the variance estimate with known mean would be x_2^2. Now, with both parameters unknown we get a lower MSE! But we can argue that this is possible by noting that our variance estimate in the latter case is biased.

So the performance of $\widehat{x}_{1,ML}$ is the same as in (a) if the value of x_2 is the same. In contrast, the performance of $\widehat{x}_{2,ML}$ is not the same if x_1 is not known. In particular, this estimate is not efficient.

Problem 9.7

Suppose we observe the process:

$$y(t) = x(t) + v(t)$$

where $x(t)$ and $v(t)$ are uncorrelated, zero-mean processes, with

$$S_{XX}(s) = \frac{3}{1 - s^2} \qquad S_{VV}(s) = \frac{5}{9 - s^2}$$

Find the noncausal Wiener filter for estimating $x(t)$ based on $y(t)$. Also find the corresponding mean-square estimation error.

Solution: This is a straightforward noncausal Wiener filtering problem. Just find the pieces and plug in:

$$\begin{aligned} S_{YY}(s) &= S_{XX}(s) + S_{VV}(s) = \frac{3}{1 - s^2} + \frac{5}{9 - s^2} = \frac{32 - 8s^2}{(1 - s^2)(9 - s^2)} = \frac{8(4 - s^2)}{(1 - s^2)(9 - s^2)} \\ S_{YX}(s) &= S_{XX}(s) = \frac{3}{1 - s^2} \end{aligned}$$

Therefore, recall that the optimal noncausal Wiener filter is given by:

$$H(s) = \frac{S_{YX}(s)}{S_{YY}(s)} = \frac{3(9-s^2)}{8(4-s^2)}$$

Recall that the corresponding estimation error covariance is given by:

$$S_{EE}(s) = S_{XX}(s) - \frac{S_{YX}(s)S_{YX}(-s)}{S_{YY}(s)} = \frac{3}{1-s^2} - \frac{9(9-s^2)}{8(1-s^2)(4-s^2)} = \frac{15}{8(4-s^2)} = \frac{15/32}{2+s} + \frac{15/32}{2-s}$$

Thus:

$$\begin{aligned} K_{EE}(\tau) &= \frac{15}{32}e^{-2|\tau|} \\ \Longrightarrow \qquad \Lambda_{\text{LSE}} = E[e^2(t)] &= K_{EE}(0) = \frac{15}{32} \end{aligned}$$

Problem 9.8

Let $x(t)$ be a zero-mean wide-sense stationary stochastic process with covariance function $K_{XX}(t)$, and suppose we observe $x(0)$ and $x(T)$.

(a) What is the linear least-squares estimate of $x(t)$ for any $0 < t < T$, based on $x(0)$ and $x(T)$? Hint: Given its a linear estimate, start with the form of the estimator and then use orthogonality conditions to obtain a set of equations for the coefficients.

(b) What is the resulting mean-square estimation error?

(c) Evaluate parts (a) and (b) for $t = T/2$.

(d) Let $d = \int_0^T x(u)\,du$ Find the linear least-squares estimate of d based on $x(0)$ and $x(T)$, and compute the resulting mean-square estimation error. Hint: Again follow the procedure described in part (a).

Solution:

(a) Clearly we seek the LLSE estimate of $x(t)$ based on observation of $x(0)$ and $x(T)$ (i.e. just two observations). Since we seek a linear estimate we know that the estimate will be a linear combination these two points:

$$\widehat{x}(t) = A_0 x(0) + A_T x(T) \tag{1}$$

where the constants A_0 and A_T will in general depend on t. This assumed form is correct for the case where the processes are zero mean. How would this assumed form change if the processes were *not* zero mean? (note in this case we can consider estimating $x(t) - m_x(t)$ based on $x(0) - m_x(0)$ and $x(T) - m_x(T)$).

Now following the hint, in order to find A_0 and A_T we apply the orthogonality conditions for optimality. In particular, we know that the error in the optimal estimate must be orthogonal to (i.e. uncorrelated with) both $x(0)$ and $x(T)$, yielding the following two conditions:

$$\begin{aligned} E[(x(t) - \widehat{x}(t))x(0)] &= 0 \\ E[(x(t) - \widehat{x}(t))x(T)] &= 0 \end{aligned}$$

Substituting (1) into these conditions we obtain:

$$\begin{aligned} K_{XX}(t) - A_0 K_{XX}(0) - A_T K_{XX}(T) &= 0 \\ K_{XX}(T-t) - A_0 K_{XX}(T) - A_T K_{XX}(0) &= 0 \end{aligned}$$

Solving these equations for A_0 and A_T we obtain:

$$\begin{aligned}
A_0 &= \frac{K_{XX}(t)K_{XX}(0) - K_{XX}(T-t)K_{XX}(T)}{[K_{XX}(0)]^2 - [K_{XX}(T)]^2} \\
A_T &= \frac{K_{XX}(0)K_{XX}(T-t) - K_{XX}(t)K_{XX}(T)}{[K_{XX}(0)]^2 - [K_{XX}(T)]^2}
\end{aligned}$$

Notice that indeed the optimal coefficients depend on time t.

(b) Here we just go and compute it, as in class:

$$\begin{aligned}
E\left([x(t) - \widehat{x}(t)]^2\right) &= E\left[(x(t) - \widehat{x}(t))\, x(t)\right] - \underbrace{E\left[(x(t) - \widehat{x}(t))\, \widehat{x}(t)\right]}_{= 0 \text{ since } e(t) \perp \widehat{x}(t)} \\
&= E\left[(x(t) - A_0 x(0) + A_T x(T))\, x(t)\right] \\
&= K_{XX}(0) - A_0 K_{XX}(t) - A_T K_{XX}(T-t)
\end{aligned}$$

where the second term on the first line is zero since linear functions of the observations are uncorrelated with the estimation error and the estimate is certainly a linear function of the data.

(c) When $t = \frac{T}{2}$ we obtain for the optimal estimate we get:

$$\begin{aligned}
A_0 = A_T &= \frac{K_{XX}\left(\frac{T}{2}\right)[K_{XX}(0) - K_{XX}(T)]}{[K_{XX}(0)]^2 - [K_{XX}(T)]^2} \\
&= \frac{K_{XX}\left(\frac{T}{2}\right)}{K_{XX}(0) + K_{XX}(T)}
\end{aligned}$$

Notice that the estimate equally weights the two points. For the error covariance we get:

$$\begin{aligned}
E\left([x(t) - \widehat{x}(t)]^2\right) &= K_{XX}(0) - 2A_0 K_{XX}\left(\frac{T}{2}\right) \\
&= K_{XX}(0) - \frac{2K_{XX}^2\left(\frac{T}{2}\right)}{K_{XX}(0) + K_{XX}(T)}
\end{aligned}$$

(d) We again follow the procedure we used in part (a). Note that d, $x(0)$, and $x(T)$ are all zero mean and we seek the LLSE estimate of d based on observation of $x(0)$ and $x(T)$, so we know the estimate will be of the form:

$$\widehat{d} = B_0 x(0) + B_T x(T) \tag{2}$$

We now apply the orthogonality conditions for optimality. In particular, we know that the error in the optimal estimate must be orthogonal to (i.e. uncorrelated with) both $x(0)$ and $x(T)$, yielding the following two conditions:

$$\begin{aligned}
E[(d - \widehat{d})x(0)] &= 0 \\
E[(d - \widehat{d})x(T)] &= 0
\end{aligned}$$

Substituting (2) into the first of these conditions we obtain:

$$\begin{aligned}
E\left[\int_0^T x(u)x(0)\, du\right] - B_0 K_{XX}(0) - B_T K_{XX}(T) &= 0 \\
\Longrightarrow \qquad \int_0^T K_{XX}(u)\, du &= B_0 K_{XX}(0) + B_T K_{XX}(T)
\end{aligned}$$

Substituting (2) into the second of these conditions we obtain a second equation for the coefficients:

$$\begin{aligned}
E\left[\int_0^T x(u)x(T)\,du\right] - B_0K_{XX}(T) - B_TK_{XX}(0) &= 0 \\
\Longrightarrow \qquad \int_0^T K_{XX}(T-u)\,du &= B_0K_{XX}(T) + B_TK_{XX}(0) \\
\Longrightarrow \qquad \int_0^T K_{XX}(u)\,du &= B_0K_{XX}(T) + B_TK_{XX}(0)
\end{aligned}$$

Making a change of variables to obtain the last equation. Now this is a system of two equations in two unknowns:

$$\left(\int_0^T K_{XX}(u)\,du\right)\begin{bmatrix}1\\1\end{bmatrix} = \begin{bmatrix}K_{XX}(0) & K_{XX}(T)\\ K_{XX}(T) & K_{XX}(0)\end{bmatrix}\begin{bmatrix}B_0\\ B_T\end{bmatrix}$$

Suppose $K_{XX}(0) \neq K_{XX}(T)$. Then there is a unique solution to this set of equations, given by:

$$B_0 = B_T = \frac{\int_0^T K_{XX}(u)\,du}{K_{XX}(0) + K_{XX}(T)}$$

The resulting optimal estimate is given by:

$$\widehat{d} = [x(0) + x(T)]\,\frac{\int_0^T K_{XX}(u)\,du}{K_{XX}(0) + K_{XX}(T)}$$

Now if $K_{XX}(0) = K_{XX}(T)$ the matrix defining the solution drops rank and so there is no unique solution. All the equations do is constrain the *sum* of B_0 and B_1. In particular, in this case all we can say is:

$$B_0 + B_1 = 2\frac{\int_0^T K_{XX}(u)\,du}{K_{XX}(0) + K_{XX}(T)}$$

Now let us find the mean-square estimation error.

$$E\left[(d-\widehat{d})^2\right] = E\left[(d-\widehat{d})d\right] - E\left[(d-\widehat{d})\widehat{d}\right] = E[d^2] - E[d\widehat{d}]$$

where we have used the fact that $(d-\widehat{d})$ is uncorrelated with $\widehat{d}$. Now, the first term is given by:

$$\begin{aligned}
E[d^2] &= E\left[\left(\int_0^T x(t)\,dt\right)\left(\int_0^T x(u)\,du\right)\right] \\
&= \int_0^T\int_0^T K_{XX}(t-u)\,du\,dt
\end{aligned}$$

With some more algebra this can be simplifed a bit to a single integral. The second term in the estimation error covariance is given by:

$$\begin{aligned}
E[d\widehat{d}] &= \frac{\int_0^T K_{XX}(u)\,du}{K_{XX}(0) + K_{XX}(T)}\left(E\left[dx(0)\right] + E\left[dx(T)\right]\right) \\
&= 2\frac{\left[\int_0^T K_{XX}(u)\,du\right]^2}{K_{XX}(0) + K_{XX}(T)}
\end{aligned}$$

Thus the estimation error covariance is given by:

$$E\left[(d-\widehat{d})^2\right] = \int_0^T\int_0^T K_{XX}(t-u)\,du\,dt - 2\frac{\left[\int_0^T K_{XX}(u)\,du\right]^2}{K_{XX}(0)+K_{XX}(T)}$$

Problem 9.9

Let $x(n)$ be a signal that you want to estimate from noisy observations $y(n) = x(n) + v(n)$. The noise is white ($S_{VV}(z) = 1$) and independent of $x(n)$ and the signal is a first-order auto-regressive process with power spectral density in the $\mathcal{Z}$ domain:

$$S_{XX}(z) = \frac{1}{(1-0.9z)(1-0.9z^{-1})}$$

Find the causal Wiener filter for estimating $x(n)$, and its associated mean-squared error. Note that the causal Wiener filter is the same as the discrete-time Kalman filter. You will have to first represent the system in state-space form. Then compute the Kalman Filter for the resulting state-space system.

Problem 9.10

(a) (Shanmugan and Breipohl 7.30) Suppose the discrete-time random process $x(t)$ obeys the following AR difference equation:

$$\begin{aligned} x(t+1) &= 0.9x(t) + w(t) \\ y(t) &= x(t) + v(t) \\ E[w(t)w(s)] &= 1\delta(t-s) \\ E[v(t)v(s)] &= 1\delta(t-s) \\ m_x(1) = 0, &\quad P_x(1) = 10 \end{aligned}$$

where $v(t)$ and $w(t)$ are independent of each other. The following data is observed:

$$y(1) = 1, \quad y(2) = 1.1, \quad y(3) = 1.2, \quad y(4) = .9, \quad y(5) = 1.2$$

Use the Kalman filtering formulas to find $\widehat{x}(t|t-1)$, $P(t|t-1)$, $\widehat{x}(t|t)$, $P(t|t)$, and the Kalman gain $K(t)$ for $t = 1, \ldots, 6$. Use Matlab or a spreadsheet to do the calculation and make a table like that below to keep your answers organized.

t	$\widehat{x}(t\|t-1)$	$y(t)$	$P(t\|t-1)$	$K(t)$	$\widehat{x}(t\|t)$	$P(t\|t)$	$\widehat{x}(t+1\|t)$	$P(t+1\|t)$
1								
2								
3								
4								
5								
6								

(b) (Shanmugan and Breipohl 7.31) What is the steady state Kalman gain $\lim_{t\to\infty} K(t)$ for part (a)? What is the steady state error covariance $\lim_{t\to\infty} P(t|t) = P$.

(c) Consider the input/output relationship between $\widehat{x}(n|n)$ and $y(n)$ corresponding to your Kalman filter solution in the steady state and show that the Kalman filter is the same as the causal Wiener filter by finding the Kalman filter system function $H_k(z)$. Look up the notes and verify that the solution you get here is the same as what you would get if you solved for a causal wiener filter.

Solution:

(a) This is just a Kalman filtering problem. Let's translate to the notation of the notes:

$$\begin{aligned}
x(t+1) &= .9x(t) + w(t) \\
y(t) &= x(t) + v(t) \\
A(t) &= .9 \\
G(t) = C(t) &= 1 \\
B(t) &= 0 \\
E[w(t)w(s)] &= Q(t)\delta(t-s) = 1\delta(t-s) \\
E[v(t)v(s)] &= R(t)\delta(t-s) = 1\delta(t-s) \\
m_x(1) &= 0 \\
P_x(1) &= 10 \\
\text{Observations :} & \quad y(1) = 1,\ y(2) = 1.1,\ y(3) = 1.2,\ y(4) = 0.9,\ y(5) = 1.2
\end{aligned}$$

Now we apply the Kalman filtering equations. Substituting in the values for this problem into the formulas in the notes we obtain:

Initialization:

$$\begin{aligned}
\widehat{x}(1|0) &= m_x(1) = 0 \\
P(1|0) &= P_x(1) = 10
\end{aligned}$$

Update Step:

$$\widehat{x}(t|t) = \widehat{x}(t|t-1) + \frac{P(t|t-1)}{P(t|t-1)+1}\left[y(t) - \widehat{x}(t|t-1)\right] \tag{3}$$

$$P(t|t) = P(t|t-1) - \frac{P^2(t|t-1)}{P(t|t-1)+1} = \frac{P(t|t-1)}{P(t|t-1)+1} \tag{4}$$

Prediction Step:

$$\widehat{x}(t+1|t) = .9\,\widehat{x}(t|t) \tag{5}$$

$$P(t+1|t) = (.9)^2\,P(t|t) + 1 \tag{6}$$

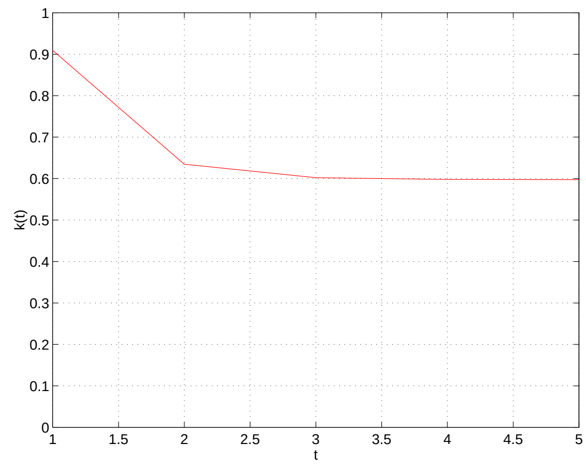
Let us define the Kalman gain $K(t)$ as the value multiplying the innovations term in (3):

$$K(t) = \frac{P(t|t-1)}{P(t|t-1)+1} \tag{7}$$

Now we can make a table of all these values as time progresses:

t	$\widehat{x}(t\|t-1)$	$y(t)$	$P(t\|t-1)$	$K(t)$	$\widehat{x}(t\|t)$	$P(t\|t)$	$\widehat{x}(t+1\|t)$	$P(t+1\|t)$
1	0	1	10	0.9091	0.9091	0.9091	0.8182	1.7364
2	0.8182	1.1	1.7364	0.6346	0.9970	0.6346	0.8973	1.5140
3	0.8973	1.2	1.5140	0.6022	1.0796	0.6022	0.9716	1.4878
4	0.9716	0.9	1.4878	0.5980	0.9288	0.5980	0.8359	1.4844
5	0.8359	1.2	1.4844	0.5975	1.0535	0.5975	0.9481	1.4840
6	0.9481		1.4840					

I did mine as a spread sheet. Note that the value of the gain appears to be approaching a constant. This is the focus of the next problem. A plot of $K(t)$ is shown in the following figure



(b) Note from (4) and (7) that $K(t) = P(t|t)$. Thus for the gain to reach a steady state as $t \to \infty$ we must have $P(t|t)$ reach a steady state, i.e. $\lim_{t\to\infty} P(t|t) = P$ for some P. Now suppose this is true. Then from (4) and (6) we have that:

$$\begin{aligned} P = P(t|t) &= \frac{P(t|t-1)}{P(t|t-1)+1} \\ P(t|t-1) &= (.9)^2\, P(t-1|t-1) + 1 = (.9)^2\, P + 1 \end{aligned}$$

Combining these two equations we obtain:

$$\begin{aligned} P &= \frac{.9^2 P + 1}{.9^2 P + 2} \\ .9^2 P^2 + (2 - .9^2)P - 1 &= 0 \\ \implies \qquad P &= \frac{-1.19 + \sqrt{1.19^2 + 4(0.81)}}{2(0.81)} \\ \implies \qquad P &= 0.5974 \end{aligned}$$

where we have taken the positive root of the quadratic, as that is the only one that yields a positive value for the steady state covariance. Thus since $\lim_{t\to\infty} K(t) = P$ we have:

$$\lim_{t\to\infty} K(t) = 0.5974$$

This value appears to agree with our numerical results from part (a).

(c) In part (b) we showed that the error covariance approached a steady state value of:

$$P = \Lambda_{\mathrm{LSE}} = 0.5974 \tag{8}$$

and as a result the Kalman gain approached a steady state value of

$$K = P = 0.5974$$

In the steady state (i.e. K a constant) for LTI systems the Kalman filtering equations become:

$$\begin{aligned} \widehat{x}(t+1|t) &= A\widehat{x}(t|t) \qquad (9) \\ \widehat{x}(t|t) &= \widehat{x}(t|t-1) + K\left[y(t) - C\widehat{x}(t|t-1)\right] \qquad (10) \end{aligned}$$

Using (9) shifted back a time step together with (10) we have:

$$\begin{aligned}\widehat{x}(t|t) &= \widehat{x}(t|t-1) + K\left[y(t) - C\widehat{x}(t|t-1)\right] = (1-KC)\widehat{x}(t|t-1) + Ky(t)\\ &= (1-KC)A\widehat{x}(t-1|t-1) + Ky(t)\end{aligned}$$

$$\begin{aligned}\Longrightarrow \qquad \widehat{x}(t|t) - (1-KC)A\widehat{x}(t-1|t-1) &= Ky(t)\\ \Longrightarrow \qquad \left[1-(1-KC)Az^{-1}\right]\widehat{X}(z) &= KY(z)\\ \Longrightarrow \qquad H(z) &= \frac{\widehat{X}(z)}{Y(z)} = \frac{K}{\left[1-(1-KC)Az^{-1}\right]}\end{aligned}$$

Now from part (b) we had $K = 0.5974$ and $A = 0.9$ and $C = 1$, thus, we find that the steady state Kalman filter is given by:

$$H_k(z) = \frac{0.5974}{\left[1-(1-0.5974)0.9z^{-1}\right]} = \frac{0.5974}{\left[1-0.3623z^{-1}\right]}$$

and, of course the estimation error covariance is given in (8). Notice that these values do indeed result in the same filter as we found in Problem 11.5 for the causal Wiener filter, showing that the causal Wiener filter and the steady state Kalman filter yield the same results, as they must.

Computer Problems

Problem 9.11 2-D MAP Estimation Based on Brownian Motion Priors

In this problem we extend the results of the 1-D Problem 9.6 to 2-D through the use of a 2-D Brownian motion-type prior model to estimate an image from noisy observations. This 2-D Brownian motion prior model will be a simple example of a *Markov Random Field* model. Note that we will be processing large arrays of numbers in MATLAB in this project, so you will need a computer with a reasonable amount of RAM or you will run into trouble. In addition, we will need to be careful about creating the large covariance matrices or their inverses needed to find the MAP estimate or we will run out of memory. We will need to use MATLAB's sparse matrix capabilities (see `help sparfun` in MATLAB).

2-D Prior Model: To proceed, we need to define what we mean by a 2-D Brownian motion prior model. A natural approach, and the one we will take, is to develop a model of the form of equation (7) of Problem Set 9. Our first issue is to define what we mean by the vector $\underline{x}$. For our 1-D problems, this vector was naturally defined to be the ordered time points of the process. For an image there is no similar, obvious ordering of the points, so we will simply choose a convention and use it throughout this problem. The most common mapping of an $N_1 \times N_2$ array of image pixels x_{n_1,n_2} to an $N = N_1 N_2$ vector $\underline{x}$, and the one we will use, is obtained by *stacking* one column on top of another, as follows:

$$[x_{n_1,n_2}] = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,N_2}\\ x_{2,1} & x_{2,2} & \cdots & x_{2,N_2}\\ \vdots & \vdots & \vdots & \vdots\\ x_{N_1,1} & x_{N_1,2} & \cdots & x_{N_1,N_2}\end{bmatrix} \quad \Longrightarrow \quad \underline{x} = \begin{bmatrix} x_{1,1}\\ \vdots\\ x_{N_1,1}\\ \hline x_{1,2}\\ \vdots\\ x_{N_1,2}\\ \hline \vdots \end{bmatrix} \tag{11}$$

In MATLAB this operation taking a 2-D image `X` (i.e. a matrix) to its vector representation `x` is performed using the colon operator "`:`", so that `x = X(:);`. The opposite operation taking a vector representation back to an $N_1 \times N_2$ image matrix is done using the MATLAB function `reshape.m`, as follows: `X = reshape(x,N1,N2);`.

Our next issue is how to actually define an appropriate 2-D Brownian motion prior model that we can use for estimation. Using equation (7) of Problem Set 9 as motivation, we will define our 2-D Brownian motion prior model explicitly as:

$$\underline{x} \quad \sim \quad N\left(0, q\left(D_t^T D_t\right)^{-1}\right) \tag{12}$$

where q represents the overall prior variance and we define the "total" derivative operator D_t for the 2-D case as:

$$D_t \quad \equiv \quad \left[\frac{D_1}{D_2}\right] \tag{13}$$

where the $N \times N$ matrices D_1 and D_2 are discrete first partial derivative operators along the n_1 (i.e. the "x") and n_2 (i.e. the "y") directions, respectively, so that the total derivative operator D_t is formed by *stacking* D_1 over D_2. Thus, the vector $D_1\underline{x}$ is the image gradient in the n_1 direction while the vector $D_2\underline{x}$ is the image gradient in the n_2 direction. This prior model extends the 1-D prior model by simply using the 2-D gradient in place of the 1-D derivative.

The model (12) gives a prior covariance defined by $Q = q\left(D_t^T D_t\right)^{-1}$. We will use this model as the prior in our estimate. Given the above definitions we can see that:

$$\underline{x} \quad \sim \quad N\left(0, q\left(D_1^T D_1 + D_2^T D_2\right)^{-1}\right) \tag{14}$$

While it is not necessary for our purposes, note that an associated *implicit* prior model for $\underline{x}$ which is consistent with (12) can be defined by:

$$D_t^T D_t \underline{x} \quad = \quad \underline{v}', \qquad \underline{v}' \sim N\left(0, q D_t^T D_t\right) \tag{15}$$

This implicit model is fundamentally different than in the 1-D case. In particular, note that a *correlated* driving noise $\underline{v}'$ is now needed. Further, the covariance of this noise is the *inverse* of the prior covariance and the implicit model matrix on the left hand side is also essentially the inverse of the prior covariance. In the 1-D case the driving noise was white and the model matrix on the left hand side was the square root of the inverse of the prior covariance. These differences in the 1-D and 2-D models reflect the differences between 1-D and 2-D AR models. Basically, while causality of an AR model in 1-D can always be obtained through appropriate redefinition of the time axis, in 2-D this may not be possible. It is the constraints imposed by our inability to find a total ordering of the points in 2-D that require a correlated driving noise.

One way of understanding what is happening is to consider that in 2-D there are many paths connecting any two pixels. To relate the values between these two pixels we may add up the increments along any such path. Since the paths must end up in the same place, the increments cannot be independent. In short, life is much more complicated in 2-D! A detailed treatment of such issues is beyond our present scope.

Observation Model: Lastly, we will need an observation model. This observation model will be given by:

$$y_{n_1,n_2} \quad = \quad x_{n_1,n_2} + w_{n_1,n_2}, \qquad w_{n_1,n_2} \sim N(0, r_{n_1,n_2}) \tag{16}$$

For this problem we will again use a stationary observation noise model with $r_{n_1,n_2} = 400$. In terms of the vector representation of the image fields this observation model is given by:

$$\underline{y} \quad = \quad \underline{x} + \underline{w}, \qquad \underline{w} \sim N(0, R) \tag{17}$$

where $R = 400I$.

(a) We will use the "trees" image in MATLAB as our truth image `X` and create data by corrupting this image with additive Gaussian noise according to (18) or (17). To this end, load the trees image `X` (e.g. `load trees`), create the associated noisy observed image `Y` as described in (18) and then find the corresponding vector representations x and y. Also define the corresponding noise covariance matrix R of the observation process for later use in estimation. You will want to create R as a sparse matrix using the MATLAB function speye.m (which only saves the nonzero entries) to save space.

Display the original image X and the noisy data image Y together using the MATLAB function imagesc.m . Make sure you use the same grayscale colormap for all your images so you can make a fair comparison. You can do this in MATLAB as follows (which we demonstrate for X):

```
imagesc(X,[1 128]);
axis image
colormap gray
```

The second argument to imagesc.m sets the intensity limits of the image. What is the mean square error of the noisy data? Save your data X and Y for use in Problem Set 11.

(b) To perform MAP estimation of x based on y we need to define a prior. We will use the model in (12), but need to find the discrete partial derivative matrices D_1 and D_2 as well as a reasonable value for the overall driving noise power q. On the web site you will find a MATLAB function dgrad.m which will produce the matrices D1 and D2. To choose a value for q we will use the interpretation of q as the variance of the increments or jumps in the noise free process per discrete coordinate step. This suggests a reasonable choice for q should be related to the variance of the derivative of the noise free image $D_t\underline{x}$. We will (somewhat arbitrarily) set $q = 3 * \mathrm{Var}\left[D_t\underline{x}\right]$

Use the supplied function dgrad.m to find the total derivative matrix Dt and set q as described above. The function dgrad.m creates the derivative operators D1 and D2 as sparse matrices to save space. Note that in this 2-D case we *will not* find the explicit prior process covariance Q, as the computation and memory requirements necessary to find and store Q are prohibitive. To find the MAP estimate we do not need Q, but rather only its *inverse*, which may be written directly in terms of Dt.

(c) Use the problem elements found in parts (a) and (b) to find the vector representing the MAP estimate xmap of the image x from the noisy data in y. Find the corresponding image estimate Xmap by reshaping xmap. This can be done in MATLAB as follows:

```
xmap = ( inv(R) + inv(q)*Dt'*Dt )\(inv(R)*y);
Xmap = reshape(xmap,N1,N2);
```

Make sure you are representing R and Dt as sparse matrices, or you will run out of memory! Also, do not try to explicitly invert the MAP normal equation, but rather make use of MATLAB's "backslash" operator as shown above. This computation will take a significant amount of time even on a fast computer (e.g. 90 seconds on an Ultrasparc).

Display the original image X, the noisy data image Y, and the MAP estimate image Xmap together. Again, be careful to use the same grayscale colormap as discussed in (a). What is the mean square error of the MAP estimate? How does the MAP estimate compare with the original image and the noisy data in terms of MSE and appearence? What are some advantages to this approach to denoising images? What are some problems with this approach? Later, we will compare this "time domain" approach with the frequency domain approach corresponding to the Wiener filter.

Solution:

(a) The original 'trees' image is shown in Figure 3 along with the noisy data image. The mean square error of the noisy image is 400 units per pixel.

(c) The MAP estimate of the original image computed using the Brownian Motion prior is shown in Figure 2. In this case the MSE is computed to be 88 units per pixel. Visually too, the reconstructed image looks less grainy than the observed data image.

(a)

(b)

Figure 1: Original and noisy version of the "trees" image

Problem 9.12 The Two-Dimensional Discrete-Space Wiener Filter

In this problem we will extend the Wiener filter theory to 2-D and implement a discrete-space 2-D noncausal Wiener filter. We will apply this filter to the restoration of the same noisy image we have examined and compare the results. As you know, the Wiener filter is the linear minimum square estimator of a stationary process, and thus for stationary Gaussian processes can be viewed as a frequency domain implementation of the MAP estimator we have examined in these earlier problem sets.

(a) As we did in Problem Set 10, we will use the "trees" image in MATLAB as our truth image `X` and create data by corrupting this image with additive Gaussian noise according to the model:

$$y_{n_1,n_2} \quad = \quad x_{n_1,n_2} + w_{n_1,n_2}, \qquad w_{n_1,n_2} \sim N(0, r_{n_1,n_2}) \tag{18}$$

For this problem we will again use a stationary observation noise model with $r_{n_1,n_2} = 400$. Either use the same noisy image you created in Problem Set 10 or load the trees image `X` and recreate the associated noisy observed image `Y` as described in (18). What is the mean square error of the noisy data? In this problem we will not need to represent the images by vectors as we did in Problem Set 10.

Figure 2: The MAP estimate of the "trees" image

You may wish to display the original image `X` and noisy data image `Y` as described in Problem Set 10. Remember to use a common grayscale colormap.

(b) From class we know that the 1-D noncausal Wiener filter $H(f)$ of the 1-D process x_n based on observation of the 1-D process $y_n = x_n + w_n$, where w_n is white noise is given by:

$$H(f) \quad = \quad \frac{S_{YX}(f)}{S_{YY}(f)} = \frac{S_{XX}(f)}{S_{XX}(f) + S_{WW}(f)} \tag{19}$$

For the 2-D case given in (18), the expression for the Wiener filter is exactly the same, only 2-D spectral densities are used to define an equivalent 2-D filter:

$$H(f_1, f_2) \quad = \quad \frac{S_{YX}(f_1, f_2)}{S_{YY}(f_1, f_2)} = \frac{S_{XX}(f_1, f_2)}{S_{XX}(f_1, f_2) + S_{WW}(f_1, f_2)} = \frac{S_{XX}(f_1, f_2)}{S_{XX}(f_1, f_2) + 400} \tag{20}$$

where we have used the fact that $S_{WW}(f) = 400$ for the problem defined in (18).

Thus to apply the Wiener filter we only need to find the 2-D power spectral density $S_{XX}(f_1, f_2)$ of the image X. We will use a simple estimate of $S_{XX}(f_1, f_2)$ based on our previous work on spectral estimation. In particular, we will use the "periodogram" estimate of the power spectral density based on the noise-free image X. Recall that the periodogram estimate of $S_{XX}(f_1, f_2)$ is nothing more than the magnitude squared of the 2-D Fourier transform of X scaled by the number of points in the field: $S_{XX}(f_1, f_2) = |\mathcal{F}[X(n_1, n_2)]|^2/N$, where $\mathcal{F}[\cdot]$ represents the 2-D Fourier transform. Estimate $S_{XX}(f_1, f_2)$ based on this formula in MATLAB as follows:

```
Sxx = (abs(fft2(X)).^2)/N;
```

The PSD you have estimated provides an estimate of the correlation function $R_{XX}(n_1, n_2)$ of the process. Does the estimated $R_{XX}(n_1, n_2)$ that you have computed seem reasonable?

(c) Use the $S_{XX}(f_1, f_2)$ you found in part (b) together with the formulas in (20) to calculate the frequency response $H(f_1, f_2)$ of the 2-D non-causal Wiener filter for this problem. You can do a point-by-point division in MATLAB using the dot-divide operator "./". Display the magnitude of this frequency response as an image using `imagesc.m`. You may wish to use the matlab function fftshift.m to shift the zero frequency part of the spectrum to the center of the image. What type of filter is this (high pass, low pass, or band pass)?

(d) Apply the Wiener filter `H` you found in part (c) to the noisy data in `Y` to obtain the corresponding Wiener filter estimated signal `Xwf`. This simply involves a point by point product of the two signals in the frequency domain followed by 2-D inverse Fourier transformation. In MATLAB this is done as follows:

```
Xwf = real(ifft2(fft2(Y).*H));
```

where we take the real part of the solution because of numerical round-off. What is the mean square error of the estimate `Xwf`?

Using a common gray scale colormap, display the true image `X`, the noisy signal `Y`, and the Wiener filtered image `Xwf`. Compare the Wiener filtered estimate to the MAP estimate you found in Problem Set 10. Which approach is more computationally efficient? The Wiener filtered or MAP estimated images certainly appear to have reduced noise and lower mean square error. Are there any problems with these estimated images? What degradation exists in these estimated images relative to the original images?

Solution:

(a) The original 'trees' image is shown in Figure 3 along with the noisy data image. As in the previous problem set the mean square error of the noisy image is 400 units per pixel.

(a) (b)

Figure 3: Original and noisy version of the "trees" image

(b) The power spectral density of the image X is estimated by finding the periodogram of the image. This gives us an estimate of the correlation function $R_{XX}(n_1, n_2)$ for the *particular* image to be restored unlike the estimate given by the matrix Q (corresponding to the 2-D Brownian motion model) whose structure is independent of the actual image to be restored.

(c) The magnitude frequency response of the 2-D non-causal Wiener filter for this problem is shown in Figure 4. Note that the zero frequency part of the spectrum has been shifted to the center of the image. As most of the larger coefficients for this filter correspond to "low" frequencies, the 2-D Wiener filter for this problem is low-pass.

(d) The result of applying the Wiener filter found to the noisy image Y is shown in Figure 5. The figure also shows, for reference, the MAP estimate of the same image. The MSE in the image recovered by the Wiener filter is computed to be 72 units per pixel.

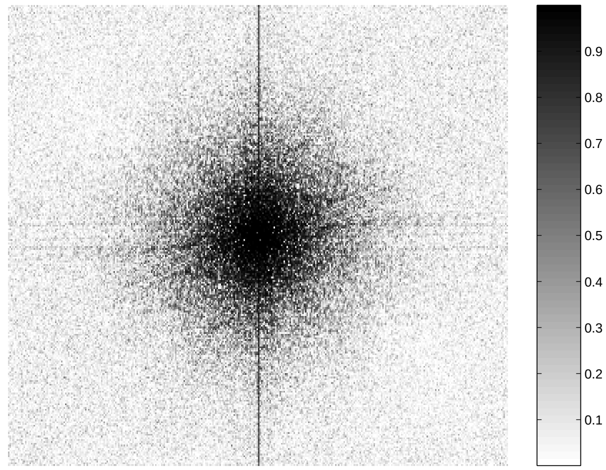


Figure 4: The magnitude response of the 2-D non-causal Wiener filter

Both the approaches we have studied have been successful in reducing the noise and lowering the MSE error, at the expense of blurring of some sharp features in the original image. The Wiener filter however seems to be computationally more efficient requiring about one-fourth the number of floating point computations compared to the MAP estimator.

(a) (b)

Figure 5: (a) The Weiner filter estimate of the “trees” image, (b) The MAP estimate