EE 5531 Probability and Stochastic Processes
Fall Semester 2015

Monday November 16, 2015

Miniproject 2 — Due Wednesday December ~~14~~ 16, 2015 at 12:05pm (after class)

Part 1: Wiener Deconvolution

Background

Recall that in class we derived the Wiener Filter for estimating continuous-time WSS signals. Our setting was as follows. Suppose that we wish to estimate some (unobserved) random process V_t based on observing a related random process U_t. We seek a function $h(t), -\infty < t < \infty$ and corresponding estimator of the form

$$\widehat{V}_t = \int_{-\infty}^{\infty} h(\theta) U_{t-\theta} d\theta$$

such that the mean-square error:

$$\mathbb{E}\left[|V_t - \widehat{V}_t|^2\right] = \mathbb{E}\left[\left|V_t - \int_{-\infty}^{\infty} h(\theta) U_{t-\theta} d\theta\right|^2\right],$$

is minimum (i.e., the error of the estimator formed using $h(t)$ is smallest among all estimators of the form $\int_{-\infty}^{\infty} \widetilde{h}(\theta) U_{t-\theta} d\theta$ for real $\widetilde{h}(t)$). Under the assumption that V_t and U_t are zero-mean and J-WSS with known power spectral densities and known cross power spectral density, we showed that the optimal filter $h(t)$ can be expressed simply in the frequency domain as

$$H(f) = \frac{S_{VU}(f)}{S_U(f)},$$

where $H(f)$ denotes the Fourier Transform of $h(t)$.

Convolutional Observation Models

In the above derivation we didn't specify exactly how U_t and V_t are related, though that information is encoded implicitly in the (assumed known) power spectral densities. In other words, in practice the form of the Wiener filter depends (of course) on the actual observation model.

Suppose that U_t and V_t are related according to a noisy *convolutional* model. That is, suppose that the observed process U_t is given by

$$U_t = \int_{-\infty}^{\infty} g(\alpha) V_{t-\alpha} d\alpha + W_t,$$

where $g(t)$ is the (real) impulse response of a known deterministic system and W_t is a zero-mean WSS additive noise process with known power spectral density $S_W(f)$. We assume W_t and V_t are uncorrelated.

Tasks

Show that the Wiener filter in this case is given by

$$H(f) = \frac{G^*(f)S_V(f)}{|G(f)|^2 S_V(f) + S_W(f)},$$

where the superscript * denotes complex conjugate, by establishing the following results:

1. Show that $\cancel{R_{VU}(\tau) = g(-\tau) * R(\tau)}$ $R_{VU}(\tau) = g(-\tau) * R_V(\tau)$.
   (This implies $S_{VU}(f) = G^*(f)S_V(f)$.)

2. Show that $R_U(\tau) = g(-\tau) * g(\tau) * R_V(\tau) + R_W(\tau)$.
   (This implies $S_U(f) = |G(f)|^2 S_V(f) + S_W(f)$.)

Part 2: Discrete Time Wiener Filtering

Background

Consider a discrete time estimation setting analogous to that in part 1. Namely, let V_n be an unknown discrete-time process that we wish to estimate based on observing a related discrete-time process U_n. Suppose, further, that V_n and U_n are zero-mean and J-WSS with known power spectral densities and known cross power spectral density. In this setting, it should come as no surprise to you that the form of the Wiener filter is similar to that above, provided we use the discrete-time Fourier Transform to obtain the frequency domain representations of the quantities of interest.

Namely, suppose we wish to estimate V_n using an estimator of the form

$$\widehat{V}_n = \sum_{k=-\infty}^{\infty} h(k) U_{n-k},$$

where $h(n)$ is the (real) discrete time impulse response of a discrete-time LTI system. Again, the optimal (in a mean-square error sense) selection of $h(n)$ is easily expressed in the frequency domain as

$$H(f) = \frac{S_{VU}(f)}{S_U(f)},$$

where the frequency domain representation for $h(n)$ is given by the discrete-time[1] Fourier Transform

$$H(f) = \sum_{n=-\infty}^{\infty} h(n) e^{-j2\pi f n}.$$

Similarly, $S_{VU}(f)$ and $S_U(f)$ are the discrete time Fourier Transforms of the cross correlation $R_{VU}(m)$ and correlation function $R_U(m)$, respectively.

Now, (similar to part 1 above) let W_n be a zero-mean WSS process with power spectral density $S_W(f)$, and suppose that W_n and V_n are uncorrelated. Then, if the observed process U_n is of the form

$$U_n = \sum_{k=-\infty}^{\infty} g(k) V_{n-k} + W_n,$$

for some known real deterministic $g(n)$ we have again that the Wiener filter in this case is given by

$$H(f) = \frac{G^*(f)S_V(f)}{|G(f)|^2 S_V(f) + S_W(f)}.$$

[1] The discrete-time Fourier Transform is a *periodic* function of f, with period 1; see also problem 10.20.

Two-Dimensional Discrete Time Random Processes

In the next part of the project, we're going to examine some applications in digital image processing. But, so far we've only talked about one-dimensional random processes, while digital images are inherently two-dimensional objects. Not to worry...we can easily extend the notions inherent to our analysis of random processes to higher-dimensional settings.[2]

Let V_{n_1,n_2} be a real-valued discrete-time random process defined on integer pairs $(n_1, n_2) \in \mathbb{Z}\times\mathbb{Z}$, where $\mathbb{Z} = \{0, \pm 1, \pm 2, \dots\}$ is the set of all integers. We can define the mean function

$$m_V(n_1, n_2) = \mathbb{E}\left[V_{n_1,n_2}\right]$$

and correlation function

$$R_V((n_1+m_1, n_2+m_2), (n_1, n_2)) = R_V(n_1+m_1, n_2+m_2; n_1, n_2) = \mathbb{E}\left[V_{n_1+m_1,n_2+m_2}V_{n_1,n_2}\right]$$

in the natural way, extending the definitions we used in the one-dimensional cases. We say that V_{n_1,n_2} is wide-sense stationary when its mean function does not depend on n_1 or n_2, and when the correlation function depends only on the time shifts in each dimension, m_1 and m_2. That is, V_{n_1,n_2} is WSS if

$$m_V(n_1, n_2) = \mu$$

which does not depend on n_1 or n_2, and

$$R_V(n_1+m_1, n_2+m_2; n_1, n_2) = R_V(m_1, m_2; 0, 0) \triangleq R_V(m_1, m_2)$$

for all $(n_1, n_2) \in \mathbb{Z}\times\mathbb{Z}$. If U_{n_1,n_2} is another 2-D discrete time random process, we say that V_{n_1,n_2} and U_{n_1,n_2} are J-WSS if both U_{n_1,n_2} and V_{n_1,n_2} are WSS, and their cross correlation

$$R_{VU}((n_1+m_1, n_2+m_2), (n_1, n_2)) = R_{VU}(n_1+m_1, n_2+m_2; n_1, n_2) = \mathbb{E}\left[V_{n_1+m_1,n_2+m_2}U_{n_1,n_2}\right]$$

depends only on the time differences in each direction, m_1 and m_2, in which case

$$R_{VU}(n_1+m_1, n_2+m_2; n_1, n_2) = R_{VU}(m_1, m_2; 0, 0) \triangleq R_{VU}(m_1, m_2).$$

Wiener Filtering of Two-Dimensional Discrete Time Random Processes

Suppose that we wish to estimate a two-dimensional random process V_{n_1,n_2} based on observing a related two-dimensional random process U_{n_1,n_2}. Let us restrict our attention to linear estimators of the form

$$\widehat{V}_{n_1,n_2} = \sum_{k=-\infty}^{\infty}\sum_{\ell=-\infty}^{\infty} h(k,l)U_{n_1-k,n_2-\ell},$$

which correspond to processing the observed U_{n_1,n_2} using a 2-D LTI system with impulse response $h(n_1, n_2)$. Now, when U_{n_1,n_2} and V_{n_1,n_2} are zero-mean J-WSS processes we have that the optimal choice of $h(n_1, n_2)$ (in a mean-square error sense) is given in the frequency domain as

$$H(f,\nu) = \frac{S_{VU}(f,\nu)}{S_U(f,\nu)},$$

where

$$H(f,\nu) = \sum_{n_1=-\infty}^{\infty}\sum_{n_2=-\infty}^{\infty} h(n_1,n_2)e^{-j2\pi(fn_1+\nu n_2)},$$

is the two-dimensional discrete time Fourier transform of $h(n_1, n_2)$, and similarly, $S_{VU}(f,\nu)$ and $S_U(f,\nu)$ are the two-dimensional discrete time Fourier transforms of $R_{VU}(m_1, m_2)$ and $R_U(m_1, m_2)$, respectively.

[2]Random processes that are defined over a multidimensional (vector) parameter are called *random fields*.

Wiener Filtering of Two-Dimensional Discrete Time Signals Under Convolutional Observation Models

Let W_{n_1,n_2} be a zero-mean WSS process with power spectral density $S_W(f,\nu)$, and suppose that W_{n_1,n_2} and V_{n_1,n_2} are uncorrelated. Then, if the observed process U_{n_1,n_2} is of the form

$$U_{n_1,n_2} = \sum_{k=-\infty}^{\infty} \sum_{\ell=-\infty}^{\infty} g(k,\ell) V_{n_1-k,n_2-\ell} + W_{n_1,n_2},$$

we have (not surprisingly, at this point) that the Wiener filter for estimating V_{n_1,n_2} in this case is given by

$$H(f,\nu) = \frac{G^*(f,\nu) S_V(f,\nu)}{|G(f,\nu)|^2 S_V(f,\nu) + S_W(f,\nu)}.$$

Tasks

Nothing for part 2!

Part 3: Applications in Digital Image Processing

Background

The noisy 2-D convolutional observation model above,

$$\begin{aligned} U_{n_1,n_2} &= \sum_{k=-\infty}^{\infty} \sum_{\ell=-\infty}^{\infty} g(k,\ell) V_{n_1-k,n_2-\ell} + W_{n_1,n_2} \\ &= g(n_1,n_2) * V_{n_1,n_2} + W_{n_1,n_2} \end{aligned}$$

is a widely used model in image processing applications. Simple denoising problems, for example, correspond to the case where $g(n_1,n_2) = \delta(n_1,n_2)$, the two-dimensional Kronecker delta function taking the value 1 when $(n_1,n_2) = (0,0)$, and zero otherwise. In this case, when V_{n_1,n_2} and W_{n_1,n_2} are uncorrelated the Wiener filter reduces to the simple form

$$H(f,\nu) = \frac{S_V(f,\nu)}{S_V(f,\nu) + S_W(f,\nu)}.$$

More interesting problems arise in cases where $g(n_1,n_2)$ may be non-trivial. Certain optical systems, for example, may have an inherent (and unintentional!) distortion, caused for example by lens aberrations. A famous example of this was the Hubble Telescope...more on this shortly. Another common problem is *blurring*, such as motion blur, which occurs when the imaging device undergoes translational motion during the time window over which the image is collected...more on this in part 4!

Implementation Issues

In the general case of estimating a 2-D discrete time WSS process observed under the noisy convolutional model above, the Wiener filter is given by

$$H(f,\nu) = \frac{G^*(f,\nu) S_V(f,\nu)}{|G(f,\nu)|^2 S_V(f,\nu) + S_W(f,\nu)}.$$

In order to implement this filter, we need to know the impulse response $g(n_1, n_2)$ of the system that acts on V_{n_1,n_2} (or, equivalently, its Fourier Transform) as well as the power spectral densities of the signal and the noise process. In practice we may know, or be able to estimate, $g(n_1, n_2)$ quite well (using, for example, extensive empirical calibration or physical modeling). We may even know (or assume) the statistics of the noise process. However, estimating the power spectral density of the object that we are trying to estimate seems to be a circular problem...if we knew the object well enough to estimate its power spectral density then we don't need to worry about the signal estimation problem in the first place!

Despite this apparent "flaw," there are straight-forward techniques that can be used to approximate the power spectral density of the unknown signal in practice. Suppose that we don't know V_{n_1,n_2} itself, but we do have access to other realizations generated from a random process with the same statistics as the image we want to acquire. One popular approach for estimating the unknown power spectral density is the so-called *periodogram* method. Let $\widetilde{v}_{n_1,n_2}$ denote a size-$N_1 \times N_2$ realization of a random process having the same correlation function as the unknown V_{n_1,n_2}. The periodogram estimate $\widehat{S}_V(f, \nu)$ is given by

$$\widehat{S}_V(f,\nu) = \frac{1}{N_1 N_2}\left|\sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1}\widetilde{v}_{n_1,n_2}e^{-j2\pi(n_1 f + n_2 \nu)}\right|^2$$

Notice that if we restrict our attention to a "grid" of frequencies given by $f = k/N_1$ and $\nu = \ell/N_2$ for $k \in \{0, 1, 2, \ldots, N_1 - 1\}$ and $\ell \in \{0, 1, 2, \ldots, N_2 - 1\}$, then

$$\sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1}\widetilde{v}_{n_1,n_2}e^{-j2\pi(n_1 f + n_2 \nu)} = \sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1}\widetilde{v}_{n_1,n_2}e^{-j2\pi(n_1 \frac{k}{N_1} + n_2 \frac{\ell}{N_2})}.$$

In this case, the periodogram estimate restricted to the grid of values given by $f = k/N_1$ and $\nu = \ell/N_2$ is

$$\widehat{S}_V(k,\ell) = \frac{1}{N_1 N_2}\left|\sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1}\widetilde{v}_{n_1,n_2}e^{-j2\pi(n_1 \frac{k}{N_1} + n_2 \frac{\ell}{N_2})}\right|^2$$

The term inside the absolute value operation is nothing but the 2-D discrete Fourier Transform (DFT) of the 2-D signal $\widetilde{v}_{n_1,n_2}$, which can easily be computed using Fast Fourier Transforms (FFT's).

Tasks

Download the MATLAB file `hubble.mat` from the course website. When you load this file you'll find three data structures:

- `blurred_galaxy.mat` – a simulated grayscale image of a blurred and noisy galaxy, such as would have been acquired by the Hubble space telescope before its lens was fixed
- `estimated_g.mat` – the two-dimensional impulse response of the lens aberration that caused the distortion
- `clean_galaxy.mat` – another (clean) image of a similar galaxy, to be used for approximating the power spectral density of the unknown galaxy image

(Images from http://science.nationalgeographic.com/science/photos/galaxies-gallery/, and the lens impulse response is from Wikipedia.)

The noise in the blurred and noisy image is additive white (i.e., uncorrelated) zero-mean Gaussian noise with variance σ^2. For this task you will use the approach outlined above to estimate the blurred and noisy image. Note that I've already made the relevant quantities (the signal and the additive noise process) zero-mean, so we can apply the theory above directly!

Now, *implementing* the Wiener filter is straightforward in the frequency domain. We want to form the 2-D convolution

$$\widehat{V}_{n_1,n_2} = \sum_{k=-\infty}^{\infty} \sum_{\ell=-\infty}^{\infty} h(k,l) U_{n_1-k,n_2-\ell},$$

but convolution in time is just multiplication in frequency. Given an expression for $H(f,\nu)$, the Wiener Filter in the frequency domain, implementing the filter in the frequency domain amounts to multiplying $H(f,\nu)$ (element-wise) by the 2-D Fourier Transform of the blurred, noisy image. Then, taking the inverse transform gives the estimate of the signal.

Now, we're interested in performing the computations in the computer, so storing and processing infinite length data is not a good idea. Instead, we can use finite length-data and the (2-D) discrete Fourier transform (DFT, also called the discrete time Fourier series, DTFS), implemented via the MATLAB `fft2` command, as a surrogate for the full discrete time Fourier Transform. The procedure, then, is as follows:

- Use the function `zero_pad.m` on the course website to zero-pad (i.e., resize) the impulse response to the size of the image. This function also shifts the impulse response (a minor technicality) so that the filtering operation doesn't induce an artificial phase shift.
- Compute the DFT of the zero-padded impulse response using the MATLAB `fft2` command.
- Compute the (finite size) frequency-domain representation of the Wiener filter using the DFT to compute $G(f,\nu)$ (on the appropriate grid of frequencies), and the periodogram method applied to the `clean_galaxy` image to approximate the power spectral density of the unknown image (on the same grid of frequencies). Both of these are easily implemented using MATLAB's `fft2` command (and a little extra arithmetic, when estimating the power spectral density!).
- Compute the DFT of the blurred, noisy image (on the appropriate grid of frequencies).
- Multiply the frequency-domain representations of the blurred, noisy image with the frequency-domain representation of the Wiener filter. (Be careful here! Note that the Wiener Filter is described element-wise in frequency...so in MATLAB you want your arithmetic to be element-wise! Element-wise multiplication in MATLAB is performed using the `.*` command rather than the `*` command, the latter of which is "standard" matrix multiplication. Likewise, the filtering operation is again element-wise multiplication in frequency.)
- Compute the inverse DFT of this product (using MATLAB's `ifft2` command). There may be some small imaginary residual on the estimate arising from machine-precision roundoff errors, so use MATLAB's `real` command to extract just the real part. This is your estimate, which depends explicitly on your choice/guess of the noise level σ^2 (as well as your estimated $g(n_1,n_2)$ and estimated power spectral density of the unknown signal).

For this task, do the following

1. Use the Wiener filter formulation to estimate the signal, assuming that $\sigma^2 = 0$. Submit a printout of the estimated image.

2. What happened in this case? Examine the form of the Wiener filter (in the frequency domain) when $\sigma^2 = 0$. What happens, in particular, at frequencies that are severely attenuated (i.e., are near zero) by the aberration $g(n_1, n_2)$? Can we hope to recover frequency components that have been set to zero? Submit a written explanation of your answer to this question.

3. Use the Wiener filter formulation to estimate the signal, assuming that $\sigma^2 = 0.00001, 0.0001, 0.001, 0.01, 0.1$. Submit a printout of the estimated image in each of these 5 cases.

4. Which of the estimates looks the cleanest? Submit a written explanation of your answer to this question.

For each of the three estimates above, use MATLAB's `imagesc` command, and the command `'colormap gray'` to render the estimate as a grayscale image.

Part 4: Extensions!

Download the MATLAB file `fireworks.mat` from the course website. There you'll find an image, `blurry_fireworks.mat`, that I took at a July 4th fireworks show. As you can see, I'm not very steady with my camera...the image has noticeable motion blur!

Tasks

Try to use what you learned in Part 3 above to "deblur" my picture. You'll have to be clever here to estimate the impulse response of the motion blur operator. (Maybe you can estimate this from the image itself...do you notice any parts that look like the motion caused by a single bright spot? If so, you could crop this as an estimate of the blur kernel.) You'll also have to estimate the power spectral density of the original (unknown) image. Feel free to be creative for that...