Lecture 16: Linear estimation theory: Noise

Recap

In Lecture 15, we started looking at linear estimation theory (also known as kriging or objective mapping, or Gauss-Markov estimation). We laid out key equations, starting with a linear model of the form:

$$\hat{y} = \mathbf{a}^T \mathbf{x}, \tag{1}$$

where $\mathbf{x}$ is measured data, $\hat{y}$ is an estimated mapped quantity, and $\mathbf{a}$ is a set of coefficients that allow us to map $\mathbf{x}$ to determine $\hat{y}$. We found:

$$\mathbf{a} = \langle \mathbf{x}\mathbf{x}^T \rangle^{-1} \langle \mathbf{x}y \rangle. \tag{2}$$

where $\langle \mathbf{x}\mathbf{x} \rangle$ is the covariance of the data with itself, and $\langle \mathbf{x}y \rangle$ is the covariance of the data with the mapped value $\hat{y}$.

As we noted, by convention, we assume that we know the covariance, but have no information about the true values y or the mapped approximation $\hat{y}$. In principle, to maintain the statistical accuracy of the solution, oceanographers have usually emphasized the importance of inferring the covariance matrices from independent sources, rather than relying on the data. It turns out that statisticians are less prescriptive about this, and given a large enough data set, they will infer covariances from the data that they are going to map. It also turns out that statisticians really enjoying working with Argo data, since the irregular sampling allows them to test different statistical methods.

Noise matters

As we noted last time, the symmetric data–data covariance matrix can run into trouble when data points are highly correlated, (e.g. if the decorrelation scales L_x and L_y are really long, or if two independent data points have the same coordinates). In a hypothetical two dimensional matrix, we could have:

$$\langle \mathbf{d}\mathbf{d}^T \rangle = A \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \tag{3}$$

which is singular.

So what do we do about noise? When we initially went through the derivation for linear estimation, we neglected the fact that measurements are intrinsically noisy. Data are noisy for a multitude of reasons:

1. The instrument used to measure a variable has intrinsic errors— instrumental error.

2. Our data may be sparse and not representative of the processes that we want to study—that is, our temperature profile might be in the middle of an eddy, when we think we're trying to map the mean state of the ocean. This is referred to as **representation error**. We need to take into account this intrinsic variability in the system we are measuring.

3. Beyond the challenges of representation error, we might simply have missing physics in our model that will show up as noise—for example, surface wave effects impact the the drag coefficient C_D that links wind speed to wind stress, but they aren't taken into account in many formulations of C_D, which can lead to spread when we try to infer wind stress.

While data are noisy for all the reasons that we mentioned above, in practice representation error is the biggest culprit. This means that even closely spaced measurements will not be as correlated as any given measurement is with itself. We need to build this measurement uncertainty into the matrix by adding noise. Assume that we measure $\hat{x}_1$ which is equivalent to the true x_1 plus noise:

$$\hat{x}_1 = x_1 + n_1 \tag{4}$$

The corresponding covariance is

$$\langle \hat{x}_1^2 \rangle = \langle x_1^2 \rangle + \langle n_1^2 \rangle, \tag{5}$$

where we assume no covariance between x_1 and the noise.

In essence, this means that the covariance $\langle x_1 x_1 \rangle$ needs to exceed the covariance of $x_1 x_2$ by our estimate of the noise variance. We should represnet this noise by adding the noise variance along the diagonal.

$$\langle \mathbf{x}\mathbf{x}^T \rangle = A \begin{bmatrix} 1+\sigma^2 & \exp\left[\frac{-4\delta^2}{T^2}\right] \\ \exp\left[\frac{-4\delta^2}{T^2}\right] & 1+\sigma^2 \end{bmatrix}. \tag{6}$$

Provided our noise is reasonably sized, this will protect us from having a singular matrix, and also provide a good representation of the uncertainty in the system.

In general, our data covariance should be:

$$\langle \mathbf{x}\mathbf{x}^T \rangle = \langle \tilde{\mathbf{x}}\tilde{\mathbf{x}}^T \rangle + \sigma^2 \mathbf{I}, \tag{7}$$

where σ is the prior uncertainty for the measurements. This assumes that the uncertainty is the same everywhere, but we are free to have σ depend on location.

Linear estimation theory: Adding noise to the problem

While the process implied in the model $\hat{y} = \alpha x$ may be the object of our study, other processes affecting the measured variables are certainly also occurring. Suppose that linear relationship truly exists between two variables in the form

$$\tilde{y} = \tilde{\alpha}\tilde{x} \tag{8}$$

where the $\sim$ indicates that this is true relationship between the variables. The reality is that our measurements are noisy so that we have access to the variables as follows

$$y = \tilde{y} + y_e = \tilde{\alpha}\tilde{x} + y_e \tag{9}$$

$$x = \tilde{x} + x_e \tag{10}$$

where the e subscript indicates noise. Statistics we calculate from our observations will include contributions from noise. So the gain calculated from this data would be

$$\alpha = \frac{\langle xy \rangle}{\langle x^2 \rangle} \tag{11}$$

$$= \frac{\langle (\tilde{x} + x_e)(\tilde{\alpha}\tilde{x} + y_e) \rangle}{\langle (\tilde{x} + x_e)^2 \rangle} \tag{12}$$

$$= \frac{\tilde{\alpha}\langle \tilde{x}^2 \rangle + \tilde{\alpha}\langle \tilde{x}x_e \rangle + \langle \tilde{x}y_e \rangle + \langle x_e y_e \rangle}{\langle \tilde{x}^2 \rangle + 2\langle \tilde{x}x_e \rangle + \langle x_e^2 \rangle} \tag{13}$$

Assume that the noise on x and y is uncorrelated with the true variables (the signal) and with each other

$$\langle \tilde{x} x_e \rangle = \langle \tilde{x} y_e \rangle = \langle x_e y_e \rangle = 0 \tag{14}$$

so that

$$\alpha = \tilde{\alpha} \frac{\langle \tilde{x}^2 \rangle}{\langle \tilde{x}^2 \rangle + \langle x_e^2 \rangle} \tag{15}$$

In the limit that the noise variance $\langle x_e^2 \rangle$ is zero, the true gain is recovered. With larger noise, the estimated gain α will always be closer to zero than the true gain $\tilde{\alpha}$. As the noise variance approaches infinity, the gain approaches zero. The reason the optimum gain is reduced in the presence of noise is that amplifying noise degrades the MSE. As we are considering variables with zero mean from here on, the essential notion is that minimum MSE estimates tend to fade toward the mean in the presence of noise.

Since our estimate always will be closer to zero than the true value, this is a reminder that we always want to start from our best prior guess, so that our estimate will be as close as possible to what we already know.

Choosing a decorrelation function

Objective mapping problems are classically laid out using Gaussian covariance functions. However, there's no obligation to specify any given analytic form for the covariance. We could assume a different analytic form (e.g. a double exponential), an empirical form based on observations, or if y was a complex function of x, we could build the functional relationships between y and x into to covariance—this is done to map dynamic topography from velocity information, for example.

One obvious choice is the double exponential:

$$\rho(\tau) = A \exp\left(-\frac{|\tau|}{T}\right) \tag{16}$$

Dependence on time or space

We set up the covariance to depend only on the separation between t_1 and t_2, but not on the actual values of t_1 and t_2. This is a computationally convenient decision that is appropriate much of the time, but it's not required. It is however, essential that your covariance matrix be symmetric. In other words, I need to require that

$$\langle x(t_1) x(t_2) \rangle = \langle x(t_2) x(t_1) \rangle \tag{17}$$

In the case above, we'd run into trouble if we used $t_1 - t_2$ with an exponential and without an absolute value sign, or if we varied the decorrelation scale T, but had it depend only on the first index. Consider the challenges in having a covariance matrix of the form:

$$\langle \mathbf{x}\mathbf{x}^T \rangle = A \begin{bmatrix} 1+\sigma^2 & \exp\left[\frac{-\tau}{T}\right] \\ \exp\left[\frac{+\tau}{T}\right] & 1+\sigma^2 \end{bmatrix} \tag{18}$$

or

$$\langle \mathbf{x}\mathbf{x}^T \rangle = A \begin{bmatrix} 1+\sigma^2 & \exp\left[\frac{-|\tau|}{T_1}\right] \\ \exp\left[\frac{-|\tau|}{T_2}\right] & 1+\sigma^2 \end{bmatrix}. \tag{19}$$

Neither of these matrices meets the fundamental requirement that the data–data covariance be symmetric, that is that the covariance between $x(t_1)$ and $x(t_2)$ has to be the same as the covariance between $x(t_2)$ and $x(t_1)$.

Mapping in multiple dimensions

So far we've been looking at linear estimation in one dimension only, essentially considering variations on interpolating in time to between $x(t_1)$ and $x(t_2)$ to find $x(t)$. However, our most interesting and challenging problems involve mapping in two-dimensional space, or three-dimensional space, or some combination of space and time. This requires a little thought for the covariance.

Now that we've gotten started, we're going to examine what happens when we implement this with real data, with noise.

Mapping in practice

Suppose we have an ocean full of Argo data that we want to map. How do we set up the problem? First, we'll have our data, for example temperature at 10 m depth:

$$\mathbf{d} = \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_N \end{bmatrix} \tag{20}$$

For this, we'll need a data–data covariance matrix, which depends not at all on temperature, but only on the geographic separation between measurement points (here identified as i and j, since we used x and y previously as variables:

$$\mathbf{R} = \begin{bmatrix} \rho(0,0) & \rho(i_1-i_2, j_1-j_2) & \rho(i_1-i_3, j_1-j_3) & \dots \\ \rho(i_2-i_1, j_2-j_1) & \rho(0,0) & \rho(i_2-i_3, j_2-j_3) & \dots \\ \rho(i_3-i_1, j_3-j_1) & \rho(i_3-i_2, j_3-j_2) & \rho(0,0) & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{21}$$

We also have a location or set of locations to which we want to map our data, so we can define the data–model covariance:

$$\mathbf{Q} = \begin{bmatrix} \rho(i_1-i_{m1}, j_1-j_{m1}) & \rho(i_1-i_{m2}, j_1-j_{m2}) & \dots \\ \rho(i_2-i_{m1}, j_2-j_{m1}) & \rho(i_2-i_{m2}, j_2-j_{m2}) & \dots \\ \vdots & \vdots & \vdots \end{bmatrix}. \tag{22}$$

So if we want to map our data, our formula for **a** says that we should compute:

$$\hat{\mathbf{y}} = \mathbf{Q}\mathbf{R}^{-1}\mathbf{d}. \tag{23}$$

The measure of the quality of the fit is the fractional mean squared error:

$$\frac{\langle(\hat{\mathbf{y}} - \mathbf{y})^2\rangle}{\langle \mathbf{y}^2 \rangle} = 1 - \frac{\mathbf{Q}^T\mathbf{R}\mathbf{Q}}{\mathbf{P}} \tag{24}$$

where $\mathbf{P}$ is the covariance of the mapped values, and is usually used as a diagonal matrix.

Now let's incorporate noise into the covariance matrix formulation. As above, each individual measurement is far more correlated with itself than with any measurements collected nearby in time or space. To address this, we add noise (e.g. σ^2) along the diagonal of our data–data covariance matrix.

$$\mathbf{R} = \begin{bmatrix} \rho(0,0)+\sigma^2 & \rho(i_1-i_2, j_1-j_2) & \rho(i_1-i_3, j_1-j_3) & \dots \\ \rho(i_2-i_1, j_2-j_1) & \rho(0,0)+\sigma^2 & \rho(i_2-i_3, j_2-j_3) & \dots \\ \rho(i_3-i_1, j_3-j_1) & \rho(i_3-i_2, j_3-j_2) & \rho(0,0)+\sigma^2 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{25}$$

In addition to being physical, this has the added benefit of ensuring that $\mathbf{R}$ will be invertible.

Examples

Finally, in class, we examined some specific examples of objectively mapped results and discussed a series of questions:

1. What is the mapped quantity?
2. What data were used?
3. What goes in the data–data covariance matrix?
4. What goes in the data–model covariance matrix?
5. What challenges do you see?

One key point that emerges when we look at real examples is the fact that in classic objective mapping, we impose the covariance matrices without knowing the specific data values. The data–data covariance matrix and the data–model covariance matrix depend only on the spatial and temporal separation between data but not on the actual data values.