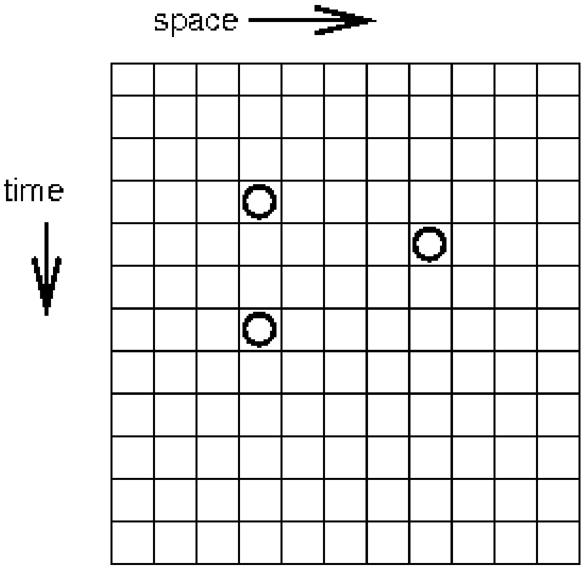

Figure 1: Sample data matrix, with missing data points marked with circles.

**Empirical Orthogonal Functions with Missing Data**

Empirical orthogonal functions are no problem to compute when you have a complete grid of data in time and space, but what happens when some data points are missing? The "EOFs with Missing Data" project is designed to address this concern. The written project summary can be a bit confusing, particularly because it's sometimes hard to distinguish between time and space in EOF calculations. Let's try to clarify what the project requires.

First, consider what happens when you have a complete $L \times N$ data matrix $\mathbf{D}$. Here $L$ represents time and $N$ represents space. One standard way to compute EOFs would be to determine a covariance matrix, $\mathbf{C}$:

$$\mathbf{C} = \frac{\mathbf{D}^T\mathbf{D}}{L}, \tag{1}$$

which is an $N \times N$ matrix, representing time time averages of the data values at each point in space, covaried with data at each other point in space. If the time series at position $x_i$ is identified as $\mathbf{d}_i$, then

$$C_{ij} = \langle \mathbf{d}_i \mathbf{d}_j^T \rangle = \frac{1}{L}\sum_{l=1}^{L} d_i(t_l) d_j(t_l). \tag{2}$$

$\mathbf{C}$ is a square-symmetric matrix, and its eigenvectors $\mathbf{e}_i$ represent the EOF spatial modes. Here $t_l$ is used to represent the $l$th element of the data time series. Temporal modes corresponding to $\mathbf{e}_i$ are determined by projecting the data onto the spatial modes. Thus the $i$th temporal mode is

$$\mathbf{a_i}^T = \mathbf{D}^T \mathbf{e}_i. \tag{3}$$

In other words, $\mathbf{a}_i$ is an $L$ element vector. In matrix we can write

$$\mathbf{A} = \mathbf{D}\mathbf{E}, \tag{4}$$

where the columns of $\mathbf{A}$ are temporal modes and columns of $\mathbf{E}$ are spatial modes.

Now what changes if we have missing data? Figure 1 depicts a sample data matrix with missing data points marked with circles. Even with a few missing data, we can still compute a covariance matrix $\mathbf{C}$. Instead of computing time-averaged covariances for $L$ data-data pairs, we'll need to compute them for $L - m$ data pairs, taking into account the number of data that are actually available. Thus

$$C_{ij} = \langle \mathbf{d}_i \mathbf{d}_j^T \rangle = \frac{1}{L-m}\sum_{l=1}^{L-m} \tilde{d}_i(t_l)\tilde{d}_j(t_l), \tag{5}$$

where $\tilde{d}_i$ represents a reduced data vector in which data elements aren't considered if they are missing. Despite the missing data, $\mathbf{C}$ can still be a good representation of the covariance matrix, and we can still compute eigenvectors $\mathbf{e}_i$.

Our challenge comes in determining $\mathbf{a}_i$ if some of the data are missing. If you simply try to compute $\mathbf{A} = \mathbf{DE}$, then what happens to the elements of $\mathbf{e}_i$ for which you have no corresponding data? As a first guess you could try leaving out the missing data elements. Thus to compute a best guess of $a_i(t_l)$, you could try

$$a_i(t_l) = \tilde{\mathbf{d}}(\mathbf{t_l})^T \tilde{\mathbf{e}}_i, \tag{6}$$

where both the data vector at time $t_l$ and the $i$'th eigenmode have elements removed to account for missing data. The problem in this case is that once we reduce the size of our data record our $\mathbf{e}_i$ are no longer orthogonal, so it's possible that we're getting a poor representation of our true temporal modes. A fix for this is to allow our temporal mode to be a linear combination of the projections into multiple different modes. Thus our best estimate of $a$ can be written

$$\hat{a}_i(t_l) = \sum_{i=1}^{N} (b_i \tilde{\mathbf{d}}(\mathbf{t_l})^T \tilde{\mathbf{e}}_i), \tag{7}$$

where $b_i$ is a coefficient that let's us determine how much of each modal solution to include.

The project description advises you to define a time-varying cost function of the form

$$\epsilon(t) = \langle [\hat{a}_i(t) - a_i(t)]^2 \rangle \tag{8}$$

and to minimize it to obtain

$$\mathbf{b} = \left( \tilde{\mathbf{E}}^T \langle \tilde{\mathbf{d}} \tilde{\mathbf{d}}^T \rangle \tilde{\mathbf{E}} \right)^{-1} \tilde{\mathbf{E}}^T \langle \tilde{\mathbf{d}} \mathbf{d^T} \rangle \mathbf{e_i}. \tag{9}$$

Sorting this out is left as an exercise.

The major challenge is to decide how to define the reduced covariance matrices $\langle \tilde{\mathbf{d}} \tilde{\mathbf{d}}^T \rangle$ and $\langle \tilde{\mathbf{d}} \mathbf{d}^T \rangle$. The project description explains that the first covariance has deleted rows and columns corresponding to missing data (so it should by $N - n \times N - n$ in size). The second has deleted rows but not columns (so should be $N - n \times N$. How do you compute these?

One option would be to remove all rows and columns with missing data from the original data matrix $\mathbf{D}$. In examples where very few data were missing, that might be a viable approach, but in typical oceanographic examples, where nearly every point in space has missing data at some point in time (and vice versa), that would result in an empty covariance matrix. Instead, we need to compute the covariance matrix more selectively.

Let's think about what we're trying to derive. Our problem came about because at time $t_l$, our data vector had missing values and was incomplete. At time $t_{l+1}$, our data sampling problems could be completely different. So let's compute a different covariance matrix, and a different solution for $\mathbf{b}$ for each time $t_l$. Figure 2 shows a blue arrow highlighting the first point in time. At time $t_1$, data are available at every point in the space domain, so we have no problem computing a temporal mode amplitude from the available information. In this case we remove no data from our data matrix, $\tilde{\mathbf{d}} = \mathbf{d}$, and $\tilde{\mathbf{E}} = \mathbf{E}$, and $\mathbf{b}$ is equivalent to the identity matrix.

Now consider the case shown in Figure 3. At time $t_4$, no observations are available from the 4th spatial position. Thus, we need to remove the 4th column from our data matrix before computing the reduced size data-data covariance matrix, $\langle \tilde{\mathbf{d}} \tilde{\mathbf{d}}^T \rangle$. Or perhaps more efficiently, we could simply remove the 4th row and column from the original covariance matrix $\mathbf{C}$. More generally, this means that our reduced state $\tilde{\mathbf{C}}$ should be dimensioned $N - n \times N - n$. For the covariance matrix defined as $\langle \tilde{\mathbf{d}} \mathbf{d}^T \rangle$, we remove the 4th column from the data matrix, but multiply it by the original data matrix to determine a covariance matrix. This is equivalent to removing the 4th row from $\mathbf{C}$ without changing any of the columns. Ultimately, despite confusing notation, the calculation should be straightforward.

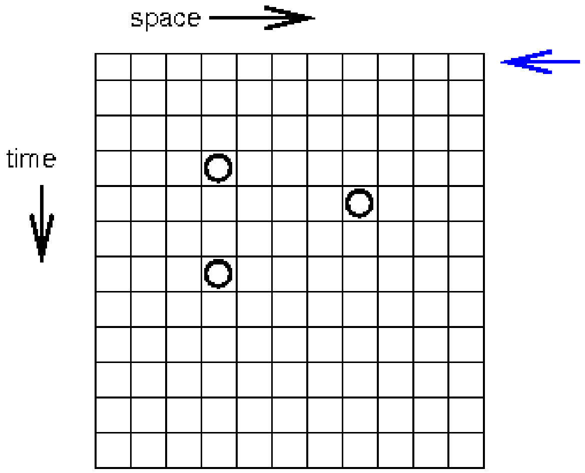

Figure 2: Sample data set with time $t_1$ highlighted. No data are missing at time $t_1$, so the corresponding covariance matrix is the full covariance matrix.

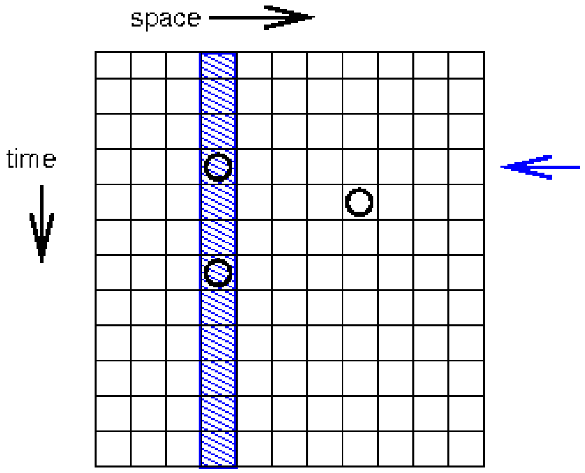

Figure 3: Sample data set with time $t_4$ highlighted. Data are missing at time $t_4$, and the corresponding covariance matrix will need to leave out the spatial positions corresponding to these missing data.