Multi-scale time series analysis of phytoplankton data from lake observatories: A case study of 'bagging' from the the Global Lake Ecological Observatory Network (GLEON)

Paul Hanson, University of Wisconsin

Long-term data from lake observatories, and data from lake sensor networks share a common feature – they both have scale-dependent patterns that suggest complex interactions that are scale-dependent. Explaining pattern in a response variable from a suite of potential predictor variables requires searching both parameter and model space to find the best combination of predictors (including autocorrelation), and in the case of time series models, the best combination of time lags. For example, if we want to understand the best correlates of chlorophyll fluorescence through time in a lake, we might consider temperature, water column stability, dissolved oxygen, and irradiance as potential predictors. However, the relationship between a given predictor and the response variable may be time-scale dependent. For example, at seasonal scales, temperature may be a governing variable in determining phytoplankton species succession, leading to a long-term change in fluorescence characteristics of the community. At shorter time scales, temperature can alter the fluorescence characteristics of phytoplankton pigments, even though the underlying community does not change. Thus, temperature may be related with chlorophyll fluorescence differently, depending on the time scale, and those differences may interfere with each other when identifying predictive models. One solution to this problem is to partition the data by time scale, then identify and fit the best model for that scale, evaluate the model and visualize the result. This overall approach is termed bootstrap aggregating, or 'bagging'.

Prior to any analysis, data need to be assimilated from multiple sources (Fig. 1). This would be an excellent use of workflows! GLEON scientific working groups confront this problem often.

The GLEON phytoplankton working group is in the process of collecting data from multiple sites, assimilating those data into a common analytical framework, then executing a bagging analysis. In a meeting in Switzerland (Feb 2011), the group articulated the following pseudo code. Steps 1-10 are generic for most analyses, and steps 11-12 are specific to 'bagging':

Pseudo code

Generic data prep (assumes data from multiple sources already loaded into a structure):

1) Create vectors for analysis; select time period, select depth (integrated, discrete, sensor), identify where bad data occurs & select good data (remove NANs, negative values, outliers)
    a. What is threshold between structural vs. incidental missing values?
2) Fill in gaps of where the bad data were removed; technique depends on whether gab was incidental or structural
3) Synchronize vectors in time
    a. High-res data only- interpolation to synchronize data
    b. Lo-res data (or a combination of low and hi res)- aggregate in order to go to coarser time scale
4) (Optional) Downscaling step (see above for pruning description)
5) Standard normal transform all variables (to center the data around zero) and normalize variance

6) Pad the data (add leading & trailing zeros (add 25% of total length of dataset to the beginning and the end of dataset) to reduce edge effects in certain transforms
7) Pick wavelet (e.g., db, Mexican Hat, Morlet) to scale-partition the data
8) Pick levels (i.e., time scales; a multiple of the sampling interval)
9) Implement continuous wavelet transform (cwt in MatLab), which adds a new dimension to the data
10) Visualize all time scales with wavelet gui

<u>Bagging analysis:</u>

11) Choose model family, such as ARX or neural networks
12) Bagging (bootstrap aggregating)- (in bloody detail)- assumes no *a priori* knowledge: use this technique when there are many potential predictor variables and potential models but you don't know which combination of predictors or which model works best
    a. Find the best model, searching entire model space and scoring with AIC
    b. Bagging loop:
    c. Yp = Yhat + randomized residuals (Yp = pseudo-observations)
    d. Find the best model
    e. Using the best model, make nominal prediction and prediction from perturbed predictors and see whether they differ; keep track of the difference
    f. Repeat c-e 1000x
    g. From the 1000 differences calculated in step e, calculate a z score for each potential predictor; z score > ~2 is significant

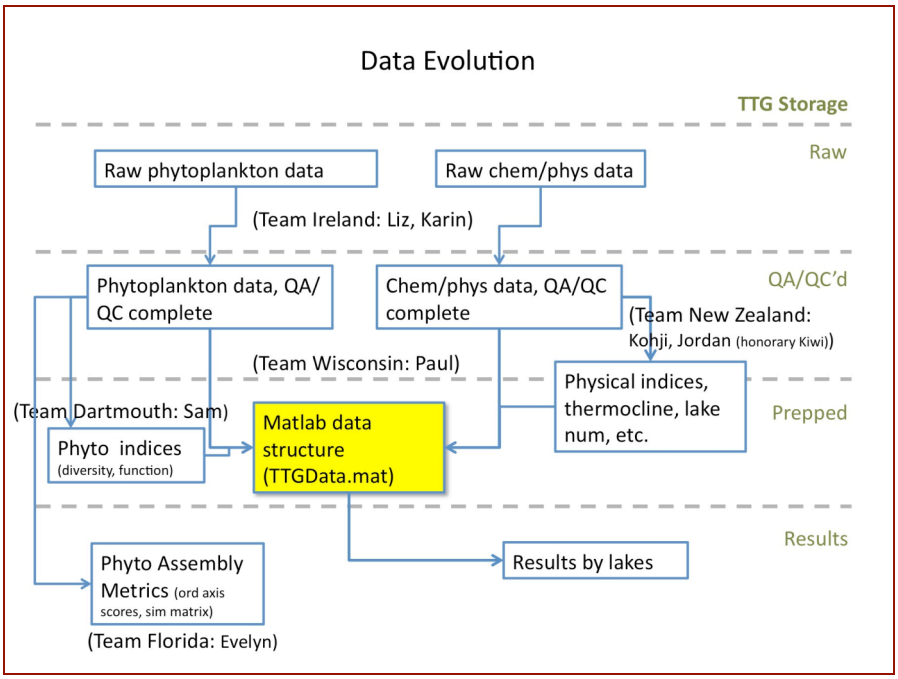

Figure 1. Raw data are submitted by site according to an agreed upon format and controlled vocabulary for variables. Ultimately, data are loaded into a Matlab data structure for analysis.

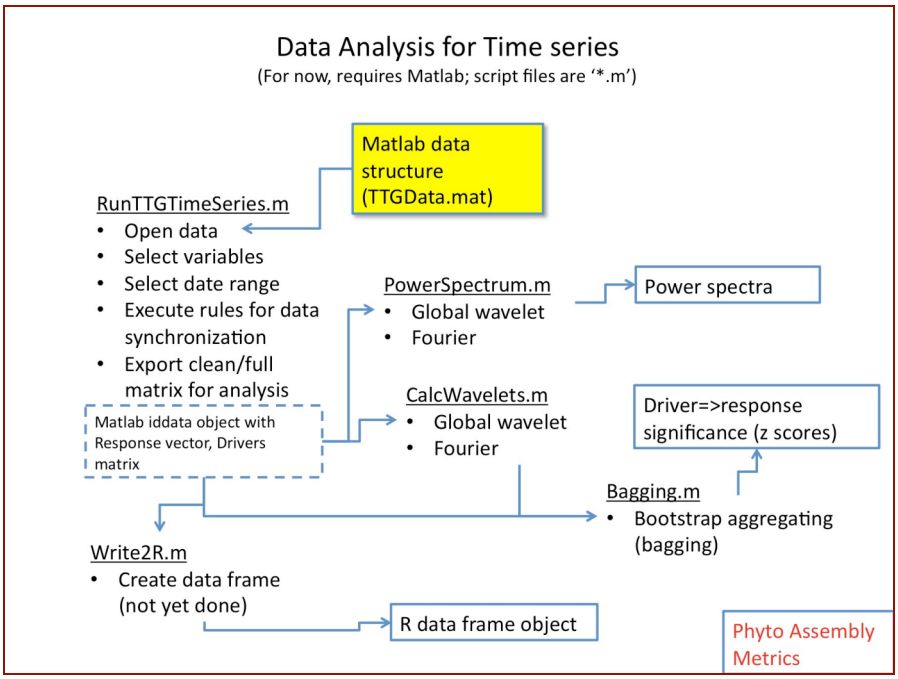

Figure 2. Data from the Matlab data structure are selected for analysis, including spectral analysis and 'bagging'.