Data analysis assignment: Principal Components.

For this assignment, you will use microarray time-course data from the yeast cell cycle contained in the file **yeast_cycle.txt**. This is a preprocessed subset of the data first presented and analyzed in Spellman et al. 1998.

• T=15 equi-spaced time points cover 2+ cycles (cells were synchronized with the “cdc” method to allow observation of cycling behaviors).
• The data is from spotted arrays, and represents normalized log-ratios (red to green; the sample on the green channel came from un-synchronized cells) . ~800 genes were originally selected as showing periodic expression behavior, and hence being cell-cycle related. Of these, you have N=679 for which no entries were missing. Thus, you do not have to perform normalization, imputation of missing values, or preliminary filtering (identification of relevant genes) for this assignment.
• The file also contains row-standardized data for the first 12 columns/time points.

THESE ARE THE SAME DATA ANALYZED IN THE LECTURE ON PCA.

Part 1: Repeat the analysis performed in class. Summarize results with tables and plots of PCA output, making sure you address the following:

1. What is the “complexity” (dimensionality) of the data.
2. What are the basic expression patterns underlying the data, and how can they be interpreted.
3. How do we visualize the data in low-dimension through projections.
4. How do we eliminate noise or artifacts through low-dimensional reconstructions of the data.
5. How do we identify genes “close” to basic expression patterns, whose interpretation is of particular interest.

<u>Part 2</u>: Use resampling or random permutation procedures, as computational means to provide a statistical assessment of the results in Part 1. Concentrate on a few pieces of output, such as:

- Eigenvalues
- First two or three eigenvectors
- Ranked proximities of genes to an eigen-direction or plane

and chose one among the following options:

1. Evaluate sampling variability. Bootstrap the data (i.e. <u>resample rows with replacement</u>), and recompute the output of interest on each bootstrap data set (N=size for each; produce B of them).

2. Evaluate stability. Implement perturbations by deletion (i.e. delete rows, or equivalently <u>resample rows without replacement</u>), and recompute the output of interest on each perturbed data set (0.8N=size for each, produce B of them).

3. Construct a "chance background" (null scenario). Implement <u>random permutations</u> (to scramble away certain features of the data while preserving others), and recompute the output of interest on each permuted data set (N=size of each, produce B of them). Pay special attention to what you are scrambling and what you are preserving. Hints: randomly permute cells within each column; randomly permute cells across the whole matrix.

Results from these analyses can be presented through tables containing relevant intervals, or superimposing bands on plots representing output from the original data. For example:

Eigen value	Actual	Simul center (mean; med)	Simul Low (mean-aSD; q quantile)	Simul High (mean+aSD; (100-q) quantile)
1				
. . .				
T