Clarifications and Corrections for Project 4: Anomaly Detection in Time-evolving Graphs

There is no need to compare your paper with the other methods mentioned in your paper.

Every paper must do the following:

- Implement the algorithm to calculate the threshold value(s) for your time series. These values will be used in determining which time points are anomalous. The algorithm is described in more detail below.

- Generate a plot showing the time series for your algorithms output. If your paper uses a similarity score then plot the similarity value time series. If it uses a distance metric then plot the distance value time series. Indicate on the plot which time points are anomalous, as well as the threshold value. To indicate the threshold it is sufficient to plot a horizontal line at the calculated value.

Calculating the threshold: To obtain the threshold value you will need to calculate the moving range average, $\overline{MR}$, and the median value of the time series. The upper threshold will be given by $median + 3\overline{MR}$, and the lower threshold will be given by $median - 3\overline{MR}$. For a time series with n points, $\overline{MR}$ will be calculated as follows:

$$\overline{MR} = \frac{\sum_{i=2}^{n} MR_i}{n-1} \text{ where } MR_i = |x_i - x_{i-1}|$$

Clarifications for Paper 1: Implement **only** the algorithm described in Section 5.5 using Signature Similarity. Because of the way the similarity is calculated anomalous graphs are identified by *two* consecutive anomalous time points in the output. For example, similarity score 1 is between graphs 1 and 2 and similarity score two is between graphs 2 and 3. If both similarity score 1 and similarity score 2 are found to be anomalous, the anomalous *graph* is then graph 2, since it is the one in common.

Use the lower bound from the individual moving range threshold, and anything below it is an anomaly.

Clarifications for Paper 2: Implement **only** algorithm 2. You **do not** have to implement the graph clustering and classification uses, only the pairwise similarity. Test the algorithm using different values for g and report the results.

The equation used for fast belief propogation uses an undefined constant, ϵ. In Appendix A.1 they give a brief explanation about the constants used. For your implementation you are to use $\epsilon^2 = a$ and $\epsilon = c'$ as defined in the appendix. h_h is calculated using the following formula:

$$h_h = \sqrt{-c_1 + \frac{\sqrt{c_1^2 + 4c_2}}{8c_2}}, \text{ where } c_1 = 2 + \sum_i d_{ii}, \text{ and } c_2 = \sum_i d_{ii} - 1, \text{ and } d \text{ is the degree matrix}$$

Because of the way the similarity is calculated anomalous graphs are identified by *two* consecutive anomalous time points in the output. For example, similarity score 1 is between graphs 1 and 2 and similarity score two is between graphs 2 and 3. If both similarity score 1 and similarity score 2 are found to be anomalous, the anomalous *graph* is then graph 2, since it is the one in common.

Use the lower bound from the individual moving range threshold, and anything below it is an anomaly.

Clarifications for Paper 3: Because this paper uses directed, weighted graphs, you will have to implenting the algorithm using a different set of features. The features you should use are: *degree of the node*, *clustering coefficient of the node* and *number of edges in the egonet of the node.* Note that the egonet of a node is the subgraph induced by the node and its neighbors. A plot should be generated for each of the features.

You should use a window size W of 7, as in the paper.

Compute the "typical eigen-behavior" using **only** the average; using SVD is not required. Calculate the upper threshold of each feature using the moving range average. Any time point above the threshold is an anomaly.

Clarifications for Paper 4: It is not necessary to implement the clustering portion of the paper. Implement **only** the similarity scoring. The similarity score is the Canberra distance between the two graphs signiture vectors. To find the outliers do a pairwise comparison between consecutive time stamps. Compute the Canberra distance between graph G_t to G_{t+1}. Note, the egonet for a node is the subgraph induced by the vertex and it's neighbors.

Because of the way the similarity is calculated anomalous graphs are identified by *two* consecutive anomalous time points in the output. For example, similarity score 1 is between graphs 1 and 2 and similarity score two is between graphs 2 and 3. If both similarity score 1 and similarity score 2 are found to be anomalous, the anomalous *graph* is then graph 2, since it is the one in common.

Use the upper bound from the individual moving range threshold, and anything above it is an anomaly. Calculate the upper threshold of each feature using the moving range average. Any time point above the threshold is an anomaly.