

Anomaly Detection using Adaptive Fusion of Graph Features on a Time Series of Graphs

Youngser Park* Carey E. Priebe† Abdou Youssef‡

Abstract

It is known that fusion of information from graph features, compared to individual features, can provide superior inference for anomaly detection [PPM⁺10]. However, selection of a fusion technique other than a naive equal weighting is not trivial. We present a multivariate methodology for fusion of features derived from time series of graphs, and investigate its inferential efficacy. The results demonstrate that our methodology has higher detection, estimation, and localization power than the standard linear weighting fusion techniques for a certain class of anomaly detection problems. Simulation results using a time series of random graphs, as well as illustrative results from a time series of Enron email data, are presented.

Key Words: feature representation, fusion, time series analysis, graph theory, anomaly detection, statistical method, hypothesis test

1. Introduction

In [PCP10], the authors consider an inference task on a graph to determine if it is homogeneous or not by comparing a graph feature. That is, the null hypothesis (H_0) is that all vertices have the same probability of connection. The alternative hypothesis (H_A) is that there exists a subset of vertices that are more highly likely to be inter-connected than the rest of the graph. For example, given a graph $G = (V, E)$, with the vertex set $V = \{1, \dots, n\}$ and edge set E , a subset of m vertices ($V_A \subset V$, $|V_A| = m$, $m \in \{2, \dots, n\}$) are connected with probability $p_{uv,A}$ where $p_{uv,A} > p_{uv,0}$. The remaining $n - m$ vertices are connected with probability $p_{uv,0}$, just like the entire graph under H_0 , to represent the portion of the population not “chatting”. An edge between a vertex in V_A and a vertex in $V \setminus V_A$ occur with probability $p_{uv,0}$. We will call this graph $\kappa(n, m, p, q)$ where $p = p_{uv,0}$ and $q = p_{uv,A}$, and it is illustrated in Figure 1.

In this paper, we extend this idea to a time series of graphs; let $G_1, G_2, \dots, G_{T=t_{max}}$ be a time series of graphs, with each graph $G_t = (V, E_t)$; that is, all graphs are on the same vertex set $V = [n]$ and the edges at time t , denoted E_t , are given pairs $(u, v) \in V \times V$. A statistical inference task is then to detect whether or not there has emerged a “chatter” group at time $t = t^*$ as shown in Figure 2.

Meanwhile, in [PPM⁺10], an idea of joint exploitation work was introduced by using the content and the externals (or activities) of communication graphs together and it was demonstrated that the technique could be superior to inference using either one. However, the way to combine both features in the paper was simply a linear weighting by adding both statistics together with equal weights.

*Center for Imaging Science, Johns Hopkins University, 3400 N. Charles St, Baltimore, MD 21212

†Dept. of Applied Mathematics & Statistics, Johns Hopkins University, 3400 N. Charles St, Baltimore, MD 21212

‡Dept. of Computer Science, George Washington Univeristy, 801 22nd St. NW, Washington, DC 20052

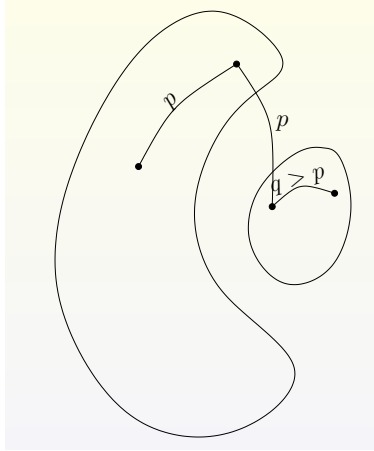


Figure 1: H_A : The “kidney-egg” random graph model, denoted as $\kappa(n, p, m, q)$. The small “egg” represents the m vertices (V_A) that exhibit chatter (each edge occurring with probability $q = p_{uv,A}$). The “kidney” is the population of $n - m$ vertices which are not exhibiting chatter (each edge occurring with probability $p = p_{uv,0} < q$). Edge between a vertex in the kidney and a vertex in the egg with probability $p = p_{uv,0}$. When $m = 0$, it is denoted as $H_0 : ER(n, p)$. **Redo this pic**

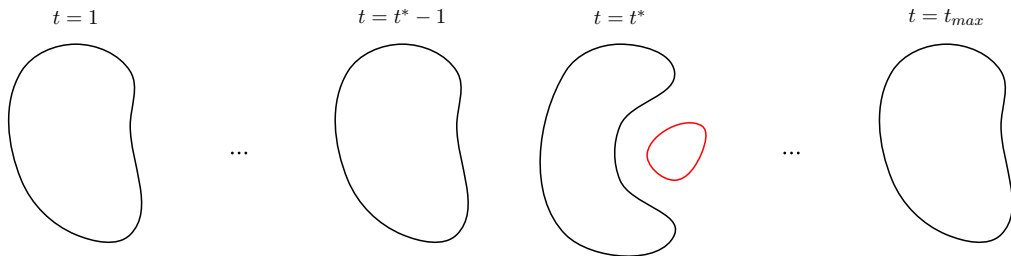


Figure 2: A time series of kidney-egg random graph model. All the graphs are $H_0 : G_t = ER(n, p)$ for all t except at $t = t^*$ where $H_A : G_{t^*} = \kappa(n, p, m, q)$.

We present here an experiment using a time series of simulated data as well as the Enron email corpus to demonstrate that a statistic which combines multiple features can be superior compared to a method that only uses an individual feature separately. One approach is to investigate a statistical power to detect abnormality by using a single graph feature, and another approach is to use multiple features jointly, either by equal weighting individual features or by non-equal (*adaptive*) weighting. Our results agrees to ones in [PPM⁺10] that an anomaly detection of time series of graphs using a collection of graph features is superior to a corresponding individual graph feature. Moreover, the results indicate that our new adaptive weighting fusion scheme has higher detection, estimation, and localization power than the previous linear equal weighting fusion technique for certain anomaly classes.

Section 2 of this paper presents a graph structure with multiple features we are using in this experiments, section 3 defines our detection from a hypothesis test, section 4 introduces a couple of ways of fusion techniques and their illustrated

examples, section 5 shows a few results with simulated data as well as Enron email data. We conclude the paper with discussion in section 6.

Is the flow of Introduction ok?

2. Graph Features

We use undirected simple graphs $G \in G_n$, the collection of all unlabeled graphs on the n vertices $V = \{1, \dots, n\}$. We denote the vertex set $V = V(G)$ and edge set $E = E(G)$; thus $G = (V, E)$. To denote edges in E , we use the notation e_{uv} for $u, v \in V$ (it is said that vertices u and v are adjacent). We will not consider loops, weighted, or parallel edges, thus if $e_{uv} \in E$ then $u \neq v$. Our graphs are undirected, so there is no distinction between e_{uv} and e_{vu} .

We consider some of the features that were used in [PCP10]; they are size, maximum degree, maximum average degree (greedy approximation), scan statistic, number of triangles, clustering coefficient, and average path length. In all features except the average path length (see below for detail), a large value of the feature is evidence in favor of H_A . We examine these seven graph statistics, $F_{t,i}$, where i is a feature index, as features to detect excessive local activity.

2.1 Size

The size of a graph is the number of edges in the graph, given by

$$\mathbf{size}(G) = |E(G)|.$$

Since we only consider an undirected graph without loops, the maximum possible size of a graph G is $\binom{n}{2} = \frac{n \times (n-1)}{2}$, where n is the order of a graph G .

2.2 Maximum Degree

The maximum degree $\Delta(G)$ is given by

$$\Delta(G) = \max_{v \in V} d(v)$$

where $d(v)$ is the degree of vertex v . This is the simplest local graph feature.

2.3 Maximum Average Degree

The maximum average degree over all subgraphs of G is denoted by $\mathbf{MAD}(G)$. If $d(v)$ is the degree of vertex v , then the average degree of a graph G is given by

$$\bar{d}(G) = \frac{1}{|V|} \sum_{v \in V} d(v) = \frac{2 \times \mathbf{size}(G)}{\mathbf{order}(G)}$$

where $\mathbf{order}(G) = |V|$, the number of vertices of graph G . Thus the maximum average degree is given by

$$\mathbf{MAD}(G) = \max_{\Omega \subset G} \bar{d}(\Omega)$$

where the maximum is over all induced subgraph of G , Ω .

Since this feature is difficult to compute exactly, we resort to consideration of greedy approximation, $\mathbf{MAD}_g(G)$ to estimate the maximum average degree of a graph. The algorithm iteratively removes a vertex with the smallest degree and calculate the average degree of the remaining induced subgraph. After removing all vertices, the largest average degree encountered is returned. This provides an approximation for the maximum average degree that is easy to implement [UE97].

2.4 Scan Statistic

Scan statistics [PCMP05] are graph features based on local neighborhoods of the graph. We will consider the scan statistics $S_k(G)$ to be the maximum number of edges over all k^{th} order neighborhoods. We will consider $k = \{1, 2, 3\}$, where $S_k(G)$ is given by

$$S_k(G) = \max_{v \in V} \text{size}(\Omega(N_k[v; G])).$$

2.5 Number of Triangles

We consider the total number of triangles in G . If A is the adjacency matrix for the graph G , then the number of triangles is given by

$$\tau(G) = \frac{\text{trace}(A^3)}{6}.$$

The trace is zero if and only if the graph is triangle-free.

2.6 Clustering Coefficient

We consider the global clustering coefficient (CC) in G :

$$CC = \frac{c}{o},$$

where c is the number of closed triplets (a subgraph with three vertices and three edges) and o is the number of open triplets (a subgraph with at least two edges). This measures the probability that the adjacent vertices of a vertex are connected. This is sometimes also called the *transitivity* of a graph.

2.7 Average Path Length

The average path length (APL) is defined by the inverse of the average length of the shortest paths to/from all the other vertices in the graph G :

$$APL = \frac{\sum_{u,v} \ell(u,v)}{n(n-1)},$$

where $\ell(u,v)$ is the shortest path between vertices u and v . This measures how many steps is required to access every other vertex from a given vertex on average. Unlike the others, this feature value gets smaller as a graph becomes more complete (or “abnormal”), so we use a negated value in this work.

Histograms of all nine features F_i (seven different features with three in scan statistics) for H_0 and H_A are depicted in Figure 3. Pink refers to H_0 and cyan refers to H_A . As we can see from this figure, some features (*e.g.*, size, maximum degree, and scan statistics) have relatively larger variances than the rest. It is our goal to measure the performance of each individual graph feature and then compare them with the effectiveness of combining features on a statistical inference task introduced in the next section. **Does this make sense?**

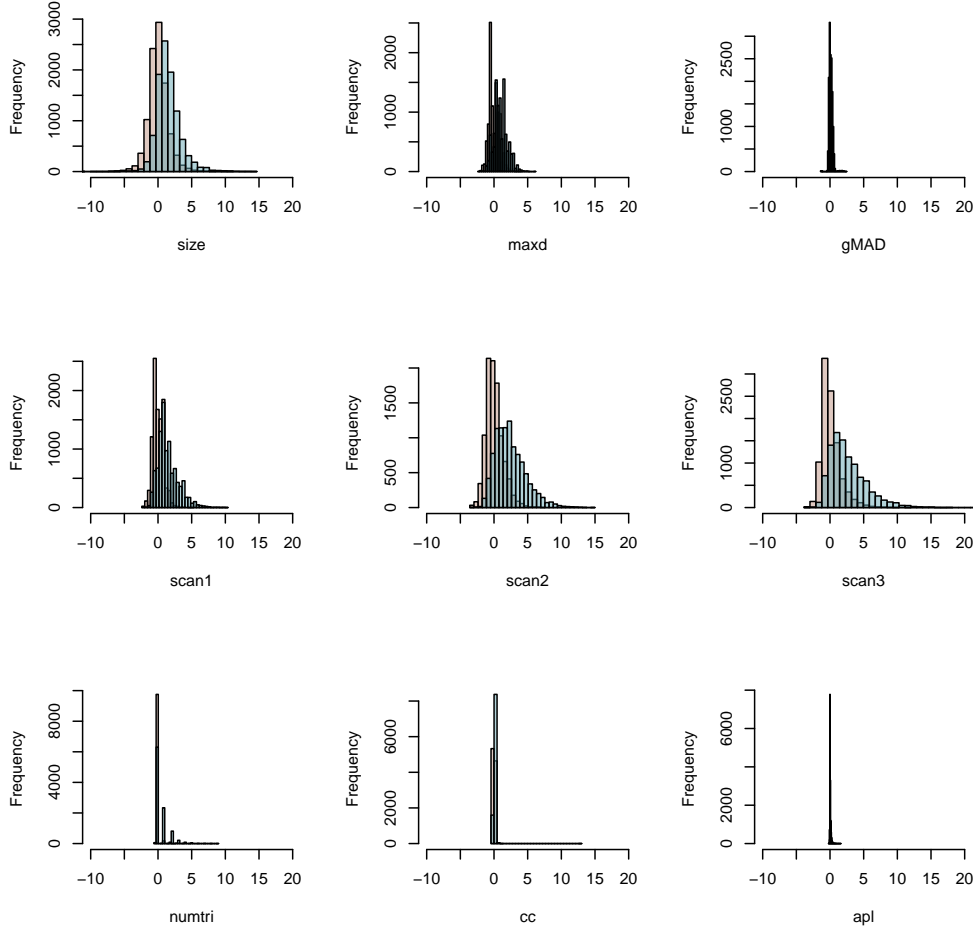


Figure 3: Histograms of 10,000 Monte Carlo replicates of F_i for H_0 and H_A with $q = 0.3$.

3. Inference Statistics

The purpose of our inference is to detect a local (temporal) behavior change in the time series of graphs. In particular, we wish to consider as our alternative hypothesis that a small (unspecified) collection of vertices (“egg”) increase their within-group activity at some (unspecified) time t^* as compared to recent past (an anomaly based on graph feature information), while the majority of vertices (“kidney”) continue with their normal behavior. The null hypothesis, then, is a form of temporal homogeneity – no probabilistic behavior changes in terms of graph features. See Figure 4.

As mentioned in [PCMP05], the raw features $F_{t,i}$ are standardized using a quantity computed from the recent past observations:

$$S_{t,i} = (F_{t,i} - \tilde{\mu}_{t,i,\ell}) / \max(\tilde{\sigma}_{t,i,\ell}, 1),$$

where $\tilde{\mu}_{t,i,\ell}$ and $\tilde{\sigma}_{t,i,\ell}$ are the running mean and standard deviation estimates of F_t based on the most recent ℓ time steps. Then, detections are defined here at time t for which F_t achieves a large value (*e.g.*, five standard deviations above its mean).

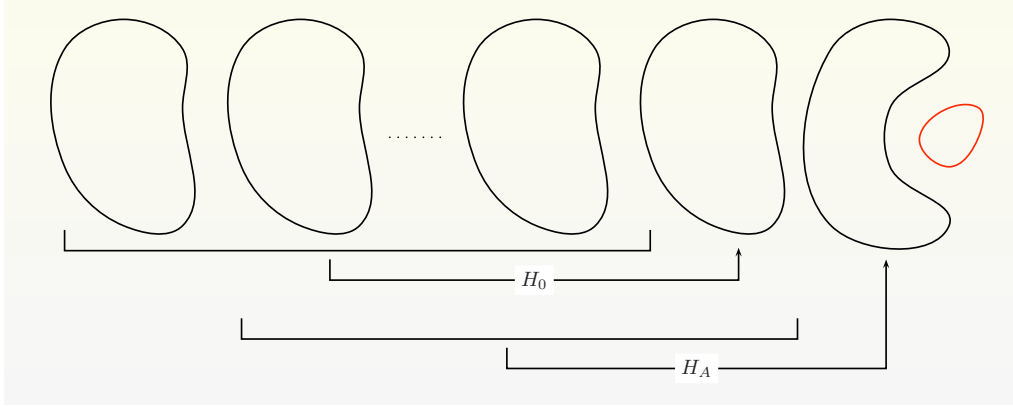


Figure 4: H_0 at $t = t^* - 1$ and H_A at $t = t^*$. Redo this pic

4. Fusion

4.1 Equal Weighting

In [PPM⁺10], we introduced a simple fusion technique by combining a pair of statistics with equal weights:

$$S_t^j = \sum_{i=1}^d w_{t,i} S_{t,i}, \quad w_{t,i} = 1/d,$$

where d is the number of graph features, which is nine in this case. With this simple “fusion” method, we demonstrated that an anomaly at time t^* can be detectable even though neither of individual statistics detect. However, the selection of fusion weights for each feature is too trivial for this case, so more sophisticated weight selection scheme is desirable.

4.2 Adaptive Weighting

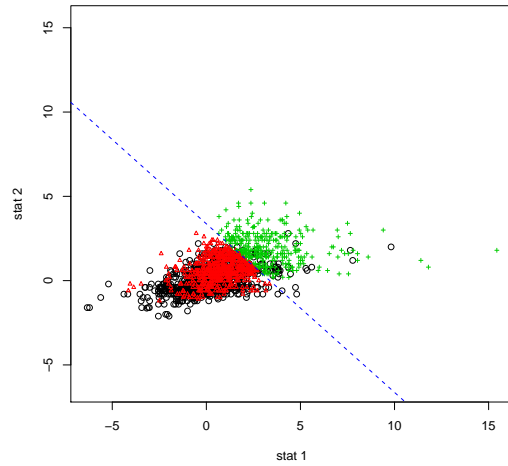
An alternative weighting scheme introduced in this paper is an *adaptive* weighting:

$$w_{t,i} = (S_{t,i} - \mu_i) / \sigma_i,$$

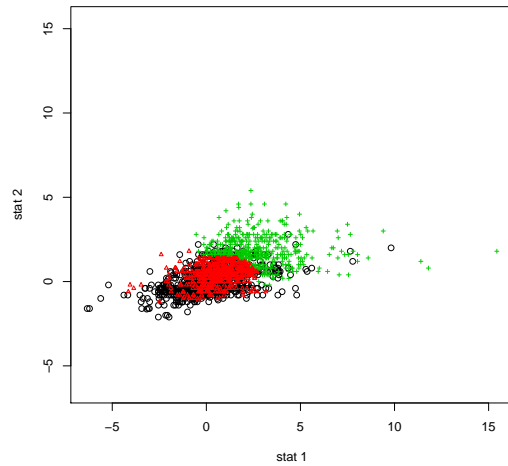
where μ_i and σ_i are the mean and the standard deviation of feature i . This is known as the *standardization* or the *1-dimensional Mahalanobis distance* from $S_{t,i}$ to μ_i , which is also called the *z-score* [DHS00]. This means that features with larger distances (or variances) get higher weights and contribute more on the inference task with fusion.

A graphical example is illustrated in Figures 5 and 6. In Figure 5, each point represents a Monte Carlo replicate of graph in two-dimensional Euclidean space using the first two features (size and maximum degree). The black points are S_{H_0} , and the color points are S_{H_A} ; the points above the detection boundaries (critical values on statistical power analysis) are colored in green and represent the power of the test. Notice that this boundary is linear for the equal weighting while it is not for the adaptive weighting. The former is because the boundary is calculated based on equal weighting for all S_{H_0} points; the slope of the line is always -1 and the intercept can be calculated with a given significance level of the test (*e.g.*,

$\alpha = 0.5$). For the adaptive weighting case, meanwhile, the color of the S_{H_A} points are determined by the distance from each point to μ_i . This means that every point gets a different weight and therefore the detection boundary is not linear. Figure 6 shows the adaptive weighting case for various values for q . As q increases, there are more green points, which implies for higher power.



(a) Equal Weighting



(b) Adaptive Weighting

Figure 5: Scatter plots for each fusion technique. Each point represents a Monte Carlo replicate of graph in two-dimensional Euclidean space. The black points are S_{H_0} , and the color points are S_{H_A} ; the points above the detection boundaries (critical values) are colored in green.

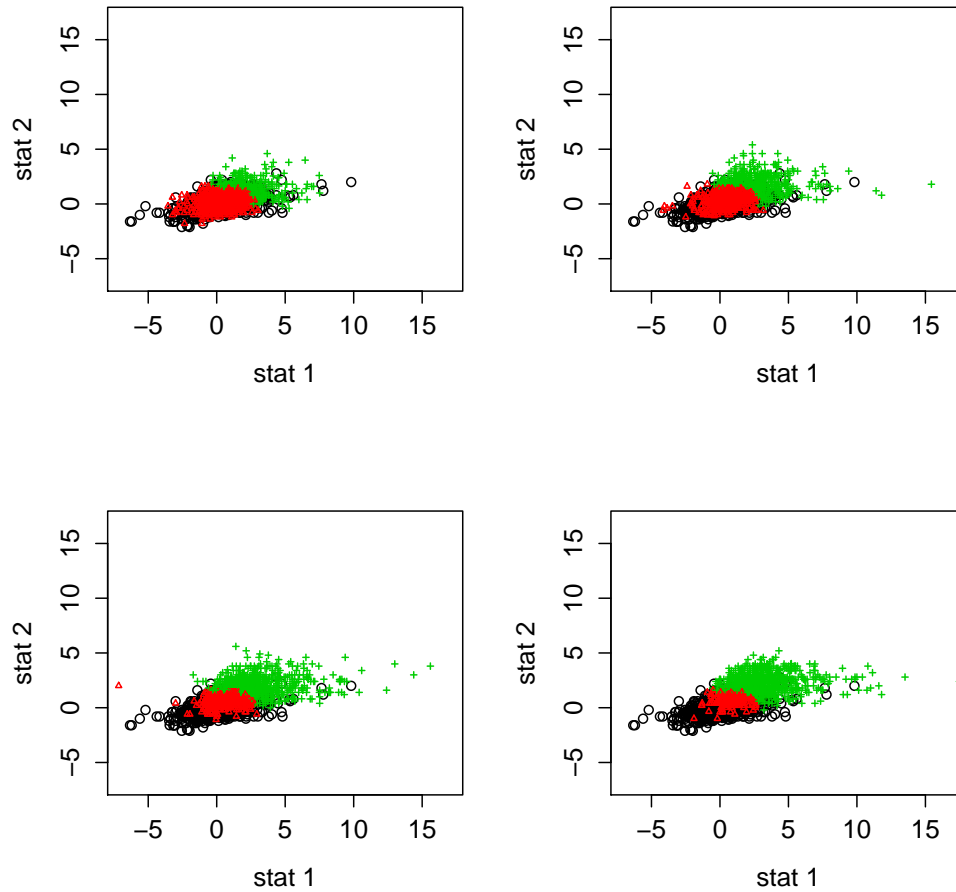


Figure 6: Scatter plots for adaptive weighting case for various q values. They are $q = \{0.2, 0.3, 0.4, 0.5\}$ from top left in clockwise. As q increases, there are more green points, which implies for higher power of the test. **add labels?**

5. Experiments

5.1 Synthetic Data

The graph model in this experiment only considers a simple scenario in which all edges are mutually independent [PCP10]. For example, if the vertices represent the actors, *i.e.*, the senders and the recipients of emails and each edge represents an email between two actors, then H_0 states that each actor communicates with each other actor with equal probability while H_A states that there is a subset of actors which may exchange an email with higher probability amongst each other – there is more “chatter” among these vertices. In addition to this spatial conditional independence, we also adapt a temporal independence, *i.e.*, these probabilities remain the same throughout time. In this sense, we call this *the first approximation model*.

In this paper, we also consider more sophisticated models including what we call *the second approximation model* and *the exact model*. For the second approximation model, the probability of each edge is computed from the dot product of random probability vectors. The dependence between two time intervals come from the latent Markov processes [LP09]. **What about the exact?** The general algorithm for implementing the time series of random dot product graph is listed in Algorithm 5.1. The only difference among these three models occurs in line 3 where the probability vectors for vertices are obtained; the first approximation uses identical vectors based on (p, q) pair throughout time while the second approximation and the exact models use random probability vectors. A detail of how to generate these vectors is beyond the scope of this paper and readers are encouraged to refer [LP09].

Algorithm 5.1 Time Series of Random Dot Product Graph

Require: $n, p, m, q, tmax$

```
1: for all time  $t$  such that  $0 < t \leq tmax$  do
2:   initialize  $A_t$ 
3:    $vp \leftarrow$  calculate probability vectors for all vertices using  $(n, p, m, q)$ 
4:   for all vertex  $u$  such that  $1 \leq u \leq n$  do
5:     for all vertex  $v$  such that  $1 \leq v \leq n$  do
6:       if  $i > j$  then
7:          $e \leftarrow \langle vp_u, vp_v \rangle$  {vector dot product}
8:         if  $e > 0.5$  then
9:            $A_t[u, v] \leftarrow A_t[v, u] \leftarrow 1$  {draw an edge  $uv$ }
10:        end if
11:       end if
12:     end for
13:   end for
14:    $A[t] \leftarrow A_t$ 
15: end for
16: return  $A$ , time series of graph
```

Our Monte Carlo experiment with synthetic data is performed with a time series of graphs G_1, \dots, G_{tmax} , where $tmax = 100$, $t^* = 51$, and $G_{t^*} = \kappa(n = 50, p = 0.01, m = 6, q)$ with $q \in \{0.2, 0.3, 0.4, 0.5\}$. For each q , we perform $R = 10,000$ Monte Carlo simulation, yielding statistical power estimates for each graph feature or fusion of features. The result for individual comparative power analysis is depicted in Figure 7 with a color bar for each feature. For a difficult case (when q is small, *e.g.*, $q = 0.2$ in blue), the performance for all features are equally poor,

while the difference is apparent as q increases. This results agree with the ones in [PCP10], that is, maximum average degree, scan statistics, and number of triangles outperforms the rest. The performance of fusion with all nine features is depicted as horizontal lines on top of bars in Figure 8. In all cases, the fusion lines are above the corresponding individual bars, where the adaptive weighting fusion lines are above the equal weighting fusion lines.

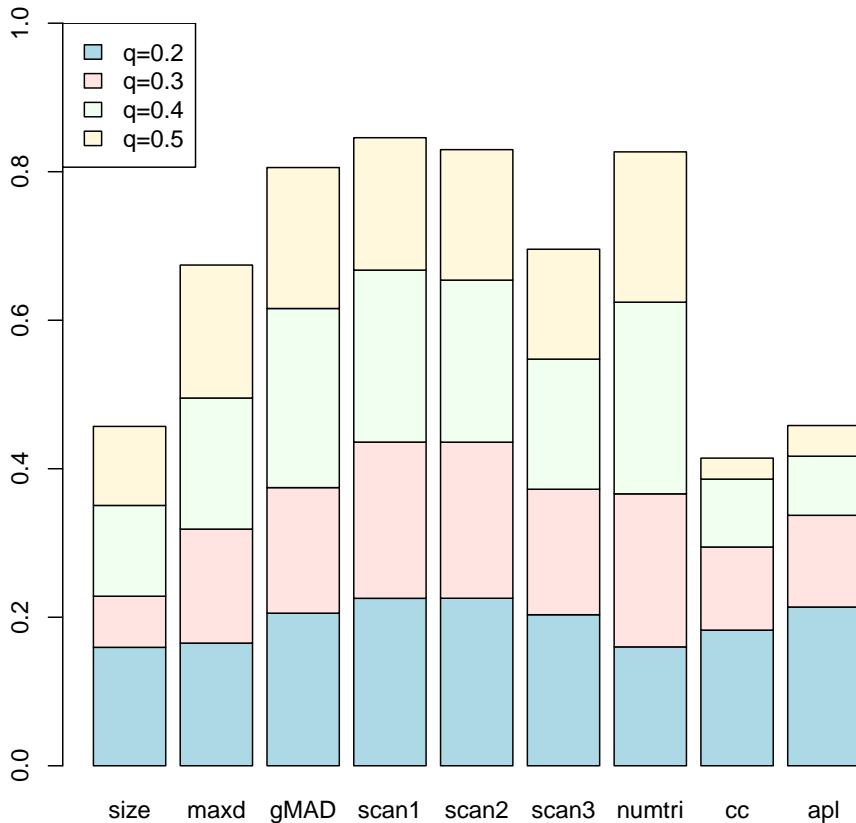


Figure 7: A statistical power plot as a function of various $q \in \{0.2, 0.3, 0.4, 0.5\}$ values with $\alpha = 0.05$, $R = 10,000$ Monte Carlo replicates each. All color bars start from 0.

Figure 9 depicts power as a function of feature dimension for different fusion techniques for three different models respectively. The difference among these models are minimal as expected while the superiority for the adaptive weighting scheme (labeled as “fus3” in green) is apparent.

5.2 Enron Email Data

We use the same Enron email data used in [PCMP05] for this experiment. The nine features, denoted as S_t where $1 \leq t \leq 189$, from a graph built with email messages during one week period among 184 executives are calculated. Figure 10 depicts histograms of S .

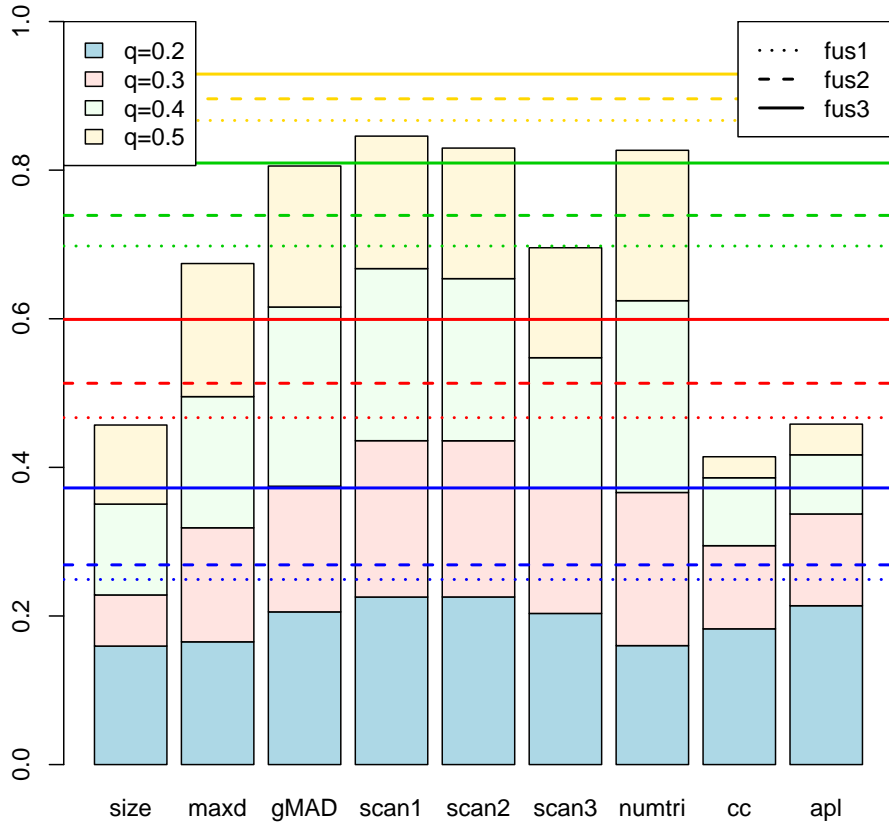


Figure 8: A statistical power plot as a function of various $q \in \{0.2, 0.3, 0.4, 0.5\}$ values with $\alpha = 0.05$, $R = 10,000$ Monte Carlo replicates each. The horizontal lines show the power using joint statistics $S_t^j = \sum_{i=1}^9 w_{t,i} S_{t,i}$. ~~remove fus2~~

Figure 11 depicts a scatter plot of S in the first two principal component space. Each point represents S_t , where $0 \leq t \leq 189$. Some of outliers are marked with week numbers, and three detections appeared in [PCMP05] are shown in red.

Our interest is the “alias” detection at week 132 in [PCMP05], when an employee changes his/her email address, therefore we choose $t^* = 132$, the third week of May 2001. Figure 12 depicts scatter plots of G_t for $t = \{120, \dots, 132\}$ with various pairs of dimensions, where the $G_{t^*=132}$ is shown in red. It shows that the combination of (size,maxd) feature pairs detects G_{t^*} with both weighting fusions, depicted in (a), while only the adaptive weighting fusion can detect G_{t^*} with other combinations of feature pairs.

[More text here!](#)

6. Discussion

We demonstrated that our adaptive weighting fusion methodology has higher detection, estimation, and localization power than the standard equal weighting fusion

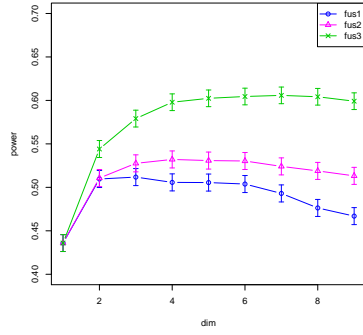
scheme as well as individual feature statistics. Not only is the performance of our new fusion technique better than the existing methods, but also the superiority is statistically significant.

A possible extension of this work can be to apply this technique into communication networks where the message contents or topics are also considered. In this way, not only an excessive chatter group but also the topic changes of a certain group throughout time can be detected.

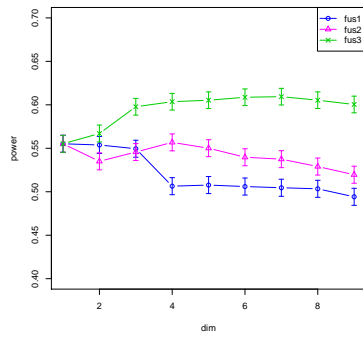
[More text here?](#)

References

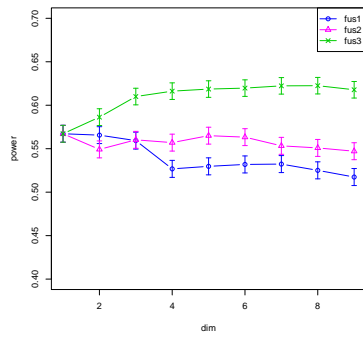
- [DHS00] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2000.
- [LP09] Nam H Lee and Carey E Priebe. A latent process model for time series of attributed random graphs. *submitted for publication*, 2009.
- [PCMP05] C. E. Priebe, J. M. Conroy, D. J. Marchette, and Y. Park. Scan statistics on enron graphs. In *Computational and Mathematical Organization Theory*, volume 11, pages 229–247. Springer Science+Business Media B.V., October 2005.
- [PCP10] H. Pao, G.A. Coppersmith, and C.E. Priebe. Statistical inference on random graphs: Comparative power analyses via monte carlo. In *Journal of Computational and Graphical Statistics*. accepted for publication, 2010.
- [PPM⁺10] C.E. Priebe, Y. Park, D.J. Marchette, J.M. Conroy, J. Grothendieck, and A.L. Gorin. Statistical inference on attributed random graphs: Fusion of graph features and content: An experiment on time series of enron graphs. In *Computational Statistics and Data Analysis*, volume 54, pages 1766–1776, 2010.
- [UE97] D. Ullman and E.R.Scheinerman. *Fractional Graph Theory*. Wiley, 1997.



(a) first approximation



(b) second approximation



(c) exact

Figure 9: Statistical power plots of fusion statistics for three different models as a function of feature dimension when $q = 0.3$. remove red lines

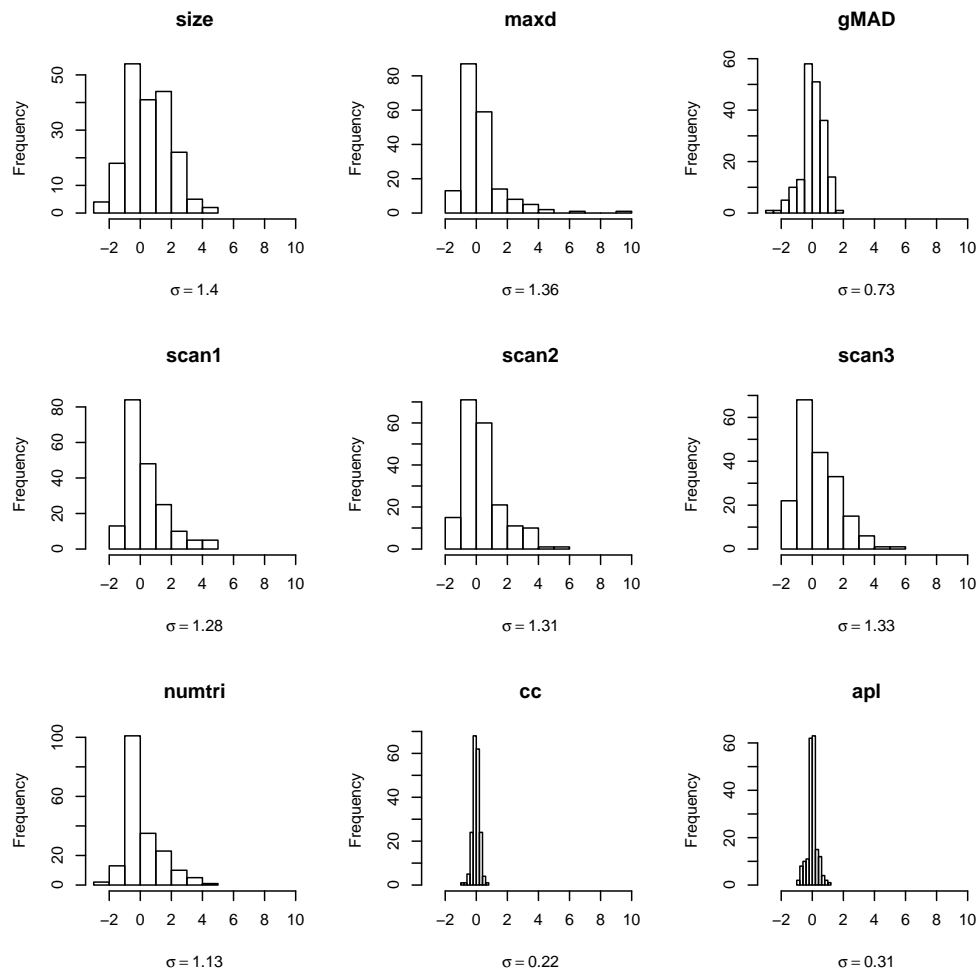


Figure 10: Histograms of S for Enron email data.

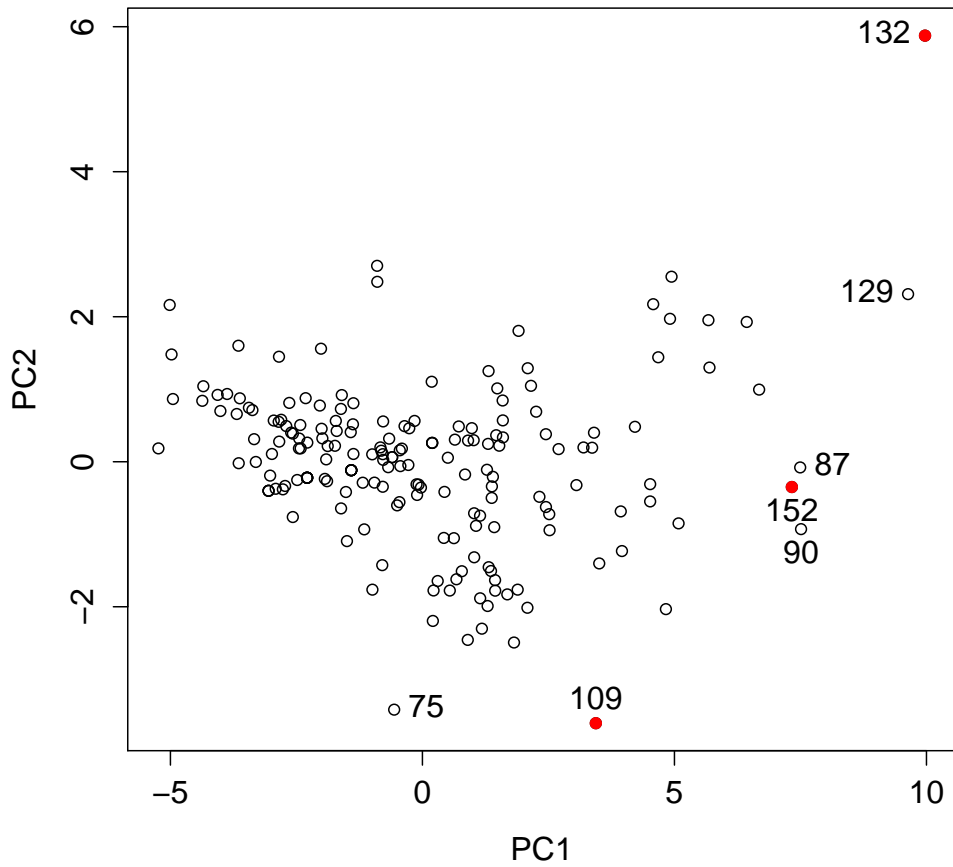
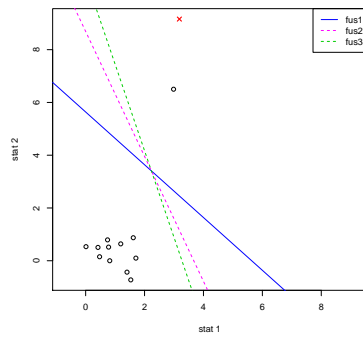
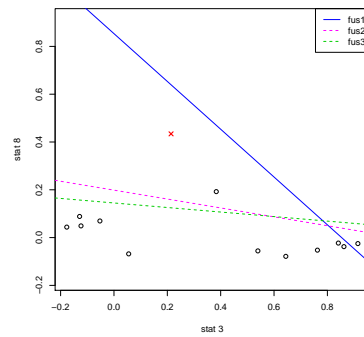


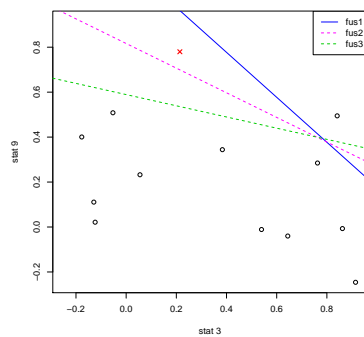
Figure 11: A scatter plot of S for Enron email data in the first two principal component space. Each point represents a week of communication graph. Some of outliers are marked with week numbers, and three detections appeared in [PCMP05] are shown in red.



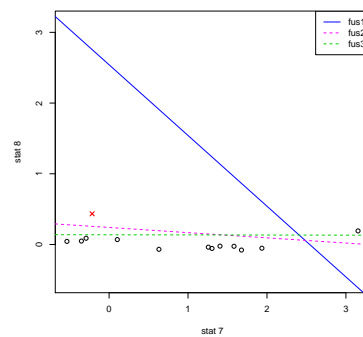
(a) size vs. maxd



(b) gMAD vs. cc



(c) gMAD vs. apl



(d) cc vs. apl

Figure 12: Scatter plots of G_t for $t = \{120, \dots, 132\}$ in various pairs of dimensions. The $G_{t^*=132}$ is shown in red. ~~remove red lines~~