2011 11th IEEE International Conference on Data Mining

A Generalized Fast Subset Sums Framework for Bayesian Event Detection

Kan Shao
Machine Learning Department
Carnegie Mellon University
kshao@cmu.edu

Yandong Liu
Language Technologies Institute
Carnegie Mellon University
yandongl@cs.cmu.edu

Daniel B. Neill
H.J. Heinz III College
Carnegie Mellon University
neill@cs.cmu.edu

***Abstract*—We present Generalized Fast Subset Sums (GFSS), a new Bayesian framework for scalable and accurate detection of irregularly shaped spatial clusters using multiple data streams. GFSS extends the previously proposed Multivariate Bayesian Scan Statistic (MBSS) and Fast Subset Sums (FSS) approaches for detection of emerging events. The detection power of MBSS is primarily limited by computational considerations, which limit it to searching over circular spatial regions. GFSS enables more accurate and timely detection by defining a hierarchical prior over all subsets of the N locations, first selecting a local neighborhood consisting of a center location and its neighbors, and introducing a sparsity parameter p to describe how likely each location in the neighborhood is to be affected. This approach allows us to consider all possible subsets of locations (including irregularly-shaped regions) but also puts higher weight on more compact regions. We demonstrate that MBSS and FSS are both special cases of this general framework (assuming $p = 1$ and $p = 0.5$ respectively), but substantially higher detection power can be achieved by choosing an appropriate value of p. Thus we show that the distribution of the sparsity parameter p can be accurately *learned* from a small number of labeled events. Our evaluation results (on synthetic disease outbreaks injected into real-world hospital data) show that the GFSS method with learned sparsity parameter has higher detection power and spatial accuracy than MBSS and FSS, particularly when the affected region is irregular or elongated. We also show that the learned models can be used for event characterization, accurately distinguishing between two otherwise identical event types based on the sparsity of the affected spatial region.**

***Keywords*-event detection; biosurveillance; scan statistics**

I. Introduction

Event detection from multiple data streams is a ubiquitous problem with applications to public health (early detection of disease outbreaks), law enforcement (detecting emerging hot-spots of crime), and many other domains. In the event detection problem, we are given multivariate spatial time series data monitored at a set of spatial locations, with the essential goals of a) timely detection of emerging events (while maintaining a low false positive rate), b) correctly pinpointing the affected spatial areas, and c) accurately characterizing the event (e.g. distinguishing between multiple event types). Many previous event detection methods have been proposed based on the *spatial scan statistic* [4]: these methods search over a large number of spatial regions, identifying potential clusters which maximize a likelihood ratio statistic, and testing for statistical significance. While Kulldorff's original approach [4] searched over circular clusters for a single data stream, recent extensions of spatial scan can detect irregularly shaped clusters ([10], [3]) and integrate information from multiple streams [5].

Neill and Cooper [8] proposed the Multivariate Bayesian Scan Statistic (MBSS), and demonstrated several advantages of the Bayesian approach over frequentist spatial scan methods: MBSS is computationally efficient, can accurately differentiate between multiple event types, and its results (the posterior probability distribution of each event type) can be easily visualized and used for decision-making. However, MBSS is limited by computational considerations which only allow circular spatial regions to be searched. The recently proposed Fast Subset Sums (FSS) method [7] is an extension of MBSS which introduces a hierarchical prior distribution over regions, assigning non-zero prior probability to each of the 2^N subsets of locations. FSS can compute the total posterior probability of an event and its spatial distribution by efficiently computing the sum of the exponentially many region posterior probabilities, thus enabling detection of irregularly-shaped clusters.

Here we propose a Generalized Fast Subset Sums (GFSS) framework which improves the timeliness and accuracy of event detection, especially for irregularly shaped clusters. A new parameter p, representing the *sparsity* of the affected region, is introduced into the framework. This parameter can be viewed as the expected proportion of locations affected in the local neighborhood consisting of a center location and its nearest neighbors. Two specific values of p, $p = 1$ and $p = 0.5$, reduce to the previously proposed MBSS and FSS methods respectively, but detection performance can often be improved by considering a range of possible p values from 0 to 1. We show that the distribution of the p parameter can be accurately learned from labeled training data, and that the resulting learned distribution can be incorporated into the GFSS detection framework, resulting in substantially improved detection power and spatial accuracy.

A. Multivariate Bayesian Scan Statistics

The MBSS methodology aims at detecting emerging events (such as disease outbreaks), identifying the type of event and pinpointing the affected locations. MBSS

1550-4786/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDM.2011.11

compares a set of alternative hypotheses $H_1(S, E)$ with the null hypothesis H_0, where each hypothesis $H_1(S, E)$ represents the occurrence of some event type E in some subset of locations S, and the null hypothesis H_0 assumes that no events have occurred. These hypotheses are mutually exclusive. Therefore, according to Bayes' Theorem, the posterior probability of each hypothesis can be expressed as:

$$\Pr(H_1(S,E) \mid D) = \frac{\Pr(D \mid H_1(S,E)) \Pr(H_1(S,E))}{\Pr(D)}$$

$$Pr(H_0 \mid D) = \frac{\Pr(D \mid H_0) \Pr(H_0)}{\Pr(D)}$$

In this expression, D is the observed dataset, and its total probability $\Pr(D)$ is equal to $\Pr(D \mid H_0) \Pr(H_0) + \sum_{S,E} \Pr(D \mid H_1(S,E)) \Pr(H_1(S,E))$. MBSS assumes that the prior $\Pr(H_1(S,E))$ is uniformly distributed over all event types and all possible circular spatial regions S. Only circular regions are considered because this simplification reduces the computation time from exponential to quadratic in N; however, this assumption reduces the power of MBSS to detect non-circular clusters, especially if the affected region is highly elongated or irregular.

The dataset D in the MBSS framework consists of multiple data streams D_m, for $m = 1 \ldots M$, and each stream contains spatial time series data collected from a set of locations s_i, for $i = 1 \ldots N$. For each location s_i and data stream D_m, we have a time series of observed counts $c_{i,m}^t$ and the corresponding expected counts (or baselines) $b_{i,m}^t$, where the baselines are estimated from time series analysis of the historical data for the given location and data stream. The subscript $t = 0$ represents the current time step, and $t = 1 \ldots T$ represent from 1 to T time steps ago respectively. For instance, a given count $c_{i,m}^t$ may represent the total number of Emergency Department visits for fever symptoms for a given zip code on a given day, and the corresponding baseline $b_{i,m}^t$ would represent the expected number of fever cases for that zip code on that day ([8], [7]).

Besides the prior $\Pr(H_1(S,E))$, another important quantity to consider is the likelihood function $\Pr(D \mid H_1(S,E))$, as shown in Figure 1. MBSS assumes that the observed count $c_{i,m}^t$ is modeled using the Poisson distribution: $c_{i,m}^t \sim \text{Poisson}(q_{i,m}^t b_{i,m}^t)$, where $q_{i,m}^t$ is the relative risk, or expected ratio of count to baseline. Further, the relative risk is modeled as $q_{i,m}^t \sim \text{Gamma}(\alpha_m, \beta_m)$ under the null hypothesis, and as $q_{i,m}^t \sim \text{Gamma}(x_{i,m}^t \alpha_m, \beta_m)$ under the alternative hypothesis, where α_m and β_m are parameter priors calculated from historical data, and $x_{i,m}^t$ is the impact of the event for the given data stream D_m, location s_i, and time step t. The distribution of $x_{i,m}^t$ is conditioned on the affected region S, the event type E, the temporal window W, and the severity parameter θ, which is assumed to be drawn from a discrete uniform distribution Θ. The temporal window W is drawn uniformly at random between 1 and W_{max}, the maximum temporal window size.

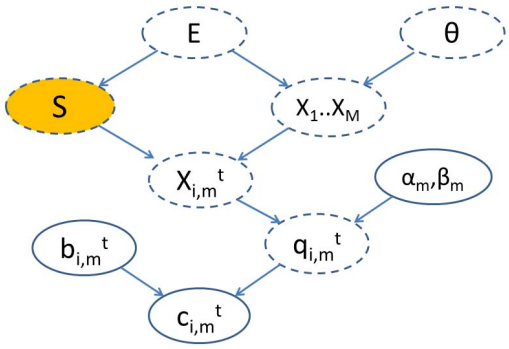

Figure 1. The structure of the Multivariate Bayesian Scan Statistic framework, from Neill and Cooper [8]. GFSS replaces the edge from E to S with the structure shown in Figure 2.

The total likelihood of the data given the alternative hypothesis $H_1(S,E)$ can be expressed as: $\Pr(D \mid H_1(S,E)) = \frac{1}{W_{max}|\Theta|} \sum_{\theta \in \Theta} \sum_{W \in 1 \ldots W_{max}} \Pr(D \mid H_1(S,E), \theta, W)$. The event type E and event severity θ define the effect x_m on each data stream D_m. Conditioned on W, θ, and E, the likelihood ratio for each location s_i can be computed as $LR_i = \prod_{m=1 \ldots M} \prod_{t=0 \ldots W-1} \frac{\Pr(c_{i,m}^t \mid b_{i,m}^t, x_m \alpha_m, \beta_m)}{\Pr(c_{i,m}^t \mid b_{i,m}^t, \alpha_m, \beta_m)}$, as described in [8]. The likelihood ratio for each spatial region S, again conditioned on W, θ, and E, is obtained by multiplying the likelihood ratios of all locations $s_i \in S$. We then marginalize over these parameters to obtain the total likelihood of region S, and combine the prior with the likelihood using Bayes' Theorem to obtain the posterior probability of each event type E in each region S.

II. Generalized Fast Subset Sums

As discussed in the previous section, the MBSS method is primarily restricted by its exhaustive computation over spatial regions S, which limits the search space to a small fraction of the 2^N possible subsets of locations. More precisely, all non-circular regions are assumed to have zero prior probability, thus reducing the method's computation time but also its detection power for irregular clusters. However, two important insights allow us to circumvent this limitation: first, the total posterior probability of an event type E is the sum of the region probabilities $\Pr(H_1(S,E) \mid D)$ over all spatial regions S, and second, the posterior probability that each spatial location s_i has been affected by event type E is the sum of the probabilities $\Pr(H_1(S,E) \mid D)$ over all spatial regions S which contain s_i. Thus we can efficiently search over all subsets S, including irregularly shaped regions, by defining a prior distribution $\Pr(H_1(S,E))$ which allows these sums to be efficiently computed *without* computing each of the individual region probabilities.

For each event type E, we define a non-uniform, hierar-

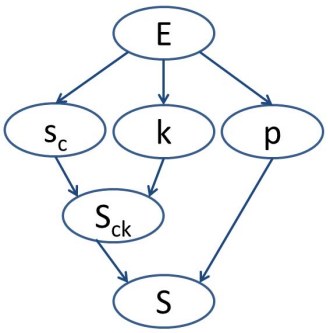

Figure 2. The structure of the Generalized Fast Subset Sums framework.

chical prior distribution over regions $\Pr(H_1(S,E) \mid E)$ such that all 2^N subsets have non-zero prior probability, but more compact regions have a larger prior. Our hierarchical prior consists of four steps:

1) Choose the center location s_c from $\{s_1 \dots s_N\}$, given a multinomial distribution $\Pr(s_c \mid E)$.
2) Choose the neighborhood size k from $\{1 \dots K\}$, given a multinomial distribution $\Pr(k \mid E)$.
3) Define the local neighborhood S_{ck} to consist of s_c and its $k-1$ nearest neighbors, based on the Euclidean distance between the zip code centroids.
4) For each spatial location $s_i \in S_{ck}$, include s_i in S with probability p, for a fixed constant $0 < p \leq 1$.

The additional structure of GFSS is shown in Figure 2, replacing the direct edge between the event type E and affected region S in Figure 1. Here we assume uniform distributions over the center s_c and neighborhood size k, but future work will learn these distributions from data. For a given local neighborhood S_{ck}, we independently choose whether to include or exclude each location $s_i \in S_{ck}$ in the affected region S. Each location is included with probability p or excluded with probability $1-p$, where p is a constant ($0 < p \leq 1$) which we term the *sparsity parameter*. We note that the assumption of conditional independence of locations, given the neighborhood S_{ck}, is necessary for efficient computation, as discussed below. Dependence between nearby locations is introduced by the neighborhood structure, since locations which are closer together are more frequently either both included or both excluded in a given local neighborhood.

The sparsity parameter p can also be viewed as the expected proportion of locations affected within a given (circular) local neighborhood. Hence, the previously proposed MBSS method (searching over circular regions) corresponds to the special case of $p = 1$. Additionally, the previously proposed FSS method assumes a similar hierarchical prior but without including the sparsity parameter p. Instead, FSS assumes that the affected subset of locations S is drawn uniformly at random from the neighborhood S_{ck}, i.e. all 2^k subsets of S_{ck} are equally likely. This is equivalent to independently including each location with probability 0.5, and hence FSS is also a special case of GFSS with $p = 0.5$. We demonstrate below that the additional flexibility provided by including the sparsity parameter in our GFSS framework can substantially improve the timeliness and accuracy of event detection: higher values of p result in improved detection of compact clusters, while lower values of p enhance detection of elongated or irregular clusters.

This hierarchical prior, and particularly the assumption that each location is drawn independently given the neighborhood, enables us to calculate the posterior probabilities much more efficiently. For a given center location s_c and neighborhood size k, and conditioning on the event type E, event severity θ, and temporal window W, we can compute the total posterior probability of the 2^k spatial regions $S \subseteq S_{ck}$ in $O(k)$ time. Since we consider $O(N)$ center locations and $O(K)$ neighborhood sizes, this enables us to compute the total posterior probability in time $O(NK^2)$.

To do so, we first compute the average likelihood ratio over all 2^k subsets of S_{ck}. We know that $\sum_{S \subseteq S_{ck}} \Pr(S \mid D) \propto \sum_{S \subseteq S_{ck}} \Pr(S) \prod_{s_i \in S} LR_i$, where $\Pr(S) = p^{|S|}(1-p)^{(k-|S|)}$ is the prior probability of region S and LR_i is the likelihood ratio of location s_i. Then $\sum_{S \subseteq S_{ck}} \Pr(S) \prod_{s_i \in S} LR_i = (1-p)^k \sum_{S \subseteq S_{ck}} \prod_{s_i \in S} \left(\frac{p}{1-p}\right) LR_i$. Since we are summing over all 2^k subsets of S_{ck}, we can write the sum of 2^k products as a product of k sums: $\sum_{S \subseteq S_{ck}} \prod_{s_i \in S} \left(\frac{p}{1-p}\right) LR_i = \prod_{s_i \in S_{ck}} \left(1 + \left(\frac{p}{1-p}\right) LR_i\right)$. Multiplying by $(1-p)^k$, we obtain the expression for the average likelihood ratio, $\prod_{s_i \in S_{ck}} ((1-p) + p \times LR_i)$. Thus the posterior probability of event E, conditioned on the temporal window W, event severity θ, center location s_c, and neighborhood size k, is proportional to the product of the smoothed likelihood ratios $LR'_i = (1-p) + p \times LR_i$ for all locations $s_i \in S_{ck}$. We can then compute the total posterior probability of event E by marginalizing over all W, θ, s_c, and k.

The posterior probability that event E affects each location s_j can be computed using a procedure very similar to the above, but in this case we only consider the neighborhoods S_{ck} that contain s_j, and sum over the 2^{k-1} subsets $S \subseteq S_{ck}$ with $s_j \in S$. Conditioning on the temporal window W, event severity θ, center location s_c, and neighborhood size k, we write the sum of 2^{k-1} products as the product of $k-1$ sums, obtaining an average likelihood ratio of $(pLR_j) \prod_{s_i \in S_{ck} - \{s_j\}} ((1-p) + p \times LR_i)$. Again, we can compute the total posterior probability of event E in spatial regions containing s_j by marginalizing over all W, θ, s_c, and k such that s_j is contained in S_{ck}.

A. Learning the Sparsity Parameter

Our detection results, shown below, demonstrate that optimizing the sparsity parameter p can substantially improve detection power. However, since the value of p must be

supplied as a parameter to the GFSS detection framework, we must consider how an appropriate value can be chosen. Here we propose to *learn* the distribution of the sparsity parameter from labeled data to improve the timeliness and accuracy of event detection. The assumption of labeled data means that we are given the affected subset of locations S for each training example; however, we are not given the values of the three latent variables (center location s_c, neighborhood size k, and sparsity p). Let $S_1 \dots S_J$ represent a set of J labeled training examples. For each training example S_j, we can calculate the likelihood of the affected region given the sparsity parameter p by marginalizing over the center location s_c and neighborhood size k: $\Pr(S_j \mid p) = \sum_{s_c} \sum_k \Pr(S_j \mid p, s_c, k) \Pr(s_c) \Pr(k)$. Then the conditional likelihood $\Pr(S_j \mid p, s_c, k)$ can be further expressed as $p^{|S_j|}(1-p)^{k-|S_j|}$ if all of the locations in S_j are contained in the local neighborhood S_{ck}, and $\Pr(S_j \mid p, s_c, k) = 0$ otherwise. Hence we can write $\Pr(S_j \mid p) = \left(\frac{p}{1-p}\right)^{|S_j|} \sum_{s_c} \Pr(s_c) \sum_{k=k_c \dots K} \Pr(k)(1-p)^k$, where k_c is the smallest neighborhood size such that S_{ck} contains all locations in S_j.

For simplicity, we assume a discrete distribution for p, where each training example S_j has sparsity parameter $p_j \in P$. In our experiments, we use ten components: $P = \{0.1, 0.2, \dots, 1.0\}$. Assuming that each value p_j is drawn independently from a discrete distribution θ, we compute the posterior distribution of θ given $S_1 \dots S_j$, representing the probability that p will take on each value in P. Additionally, we assume a Dirichlet prior on θ. Let x_k denote the kth component of P, and θ_k denote the posterior probability that p will take on value x_k. If the value of p_j for each training example S_j was observed, we could easily obtain the resulting posterior distribution of θ, by computing $\theta_k = \frac{\frac{1}{|P|} + \sum_{j=1\dots J} 1\{p_j = x_k\}}{1+J}$ for each $x_k \in P$. However, since the value of p_j for each training example S_j is not observed, we must first compute the posterior probabilities $\Pr(p_j = x_k \mid S_j) = \frac{\Pr(S_j \mid p=x_k)}{\sum_{x_k \in P} \Pr(S_j \mid p=x_k)}$ for each value $x_k \in P$ and each training example S_j. We then compute $\theta_k = \frac{\frac{1}{|P|} + \sum_{j=1\dots J} \Pr(p_j = x_k \mid S_j)}{1+J}$ for each $x_k \in P$.

B. Related Work

The present study proposes the Generalized Fast Subset Sums (GFSS) framework, which generalizes the previously proposed Multivariate Bayesian Scan Statistic [8] and Fast Subset Sums [7] methods for multivariate Bayesian event detection. The Bayesian spatial scan framework is a variant of the traditional frequentist, hypothesis test-based spatial scan methods [4]. Two recently proposed frequentist spatial scan methods, Kulldorff's multivariate scan [5] and the nonparametric scan statistic [9], also allow integration of multiple data streams for detection. However, unlike the Bayesian spatial scan approaches, these methods cannot differentiate between multiple event types. The recently proposed "linear-time subset scanning" approach enables an efficient search over the 2^N subsets of locations while only evaluating $O(N)$ subsets. However, the LTSS method simply finds the most anomalous (highest scoring) subset, and cannot be used to compute the total posterior probability of an event or its posterior distribution in space and time, which require summing over all subsets of locations. Previously, learning approaches to improve detection power by using non-uniform priors on each search region were explored in ([8], [6]). However, these methods are still constrained by the computational limitations inherent in the MBSS framework, preventing them from being used to learn priors over all subsets of the data rather than just circular regions. Finally, several other Bayesian event detection methods have been proposed, such as WSARE [11] and PANDA ([1], [2]). Unlike the present work, these purely temporal event detection methods do not take spatial information into account.

III. Evaluation

In this section, we evaluate the learning performance, as well as compare the detection power and spatial accuracy of the GFSS method with the MBSS and FSS approaches. Our experiments focus on detection of simulated disease outbreaks injected into real-world hospital Emergency Department (ED) data. The original dataset contains de-identified ED visit records collected from ten hospitals in Allegheny County, Pennsylvania, from January 1, 2004 to December 31, 2005. The records have been classified into various data streams according to the patient's chief complaint: for this study we focused on two data streams, patients with cough symptoms and nausea symptoms respectively. For each data stream, we have the count of ED visits of that type on each day for each of the 97 Allegheny County zip codes.

For each of the experiments described below, simulated outbreaks were generated by first choosing a set of affected zip codes, then injecting a number of simulated disease cases that grows linearly over the duration of the outbreak. Each outbreak was assumed to be 10 days in duration. For each affected zip code s_i and data stream D_m, and for each day of the outbreak $t = 1 \dots 10$, $\delta^t_{i,m} \sim \text{Poisson}(t \times w_{i,m})$ additional cases are injected, incrementing the value of $c^t_{i,m}$. Here we assume that each zip code's weight is proportional to its total count for the entire dataset: $w_{i,m} = \frac{\sum_t c^t_{i,m}}{\sum_i \sum_t c^t_{i,m}}$.

To evaluate detection power, we measured average time to detection at a fixed false positive rate of 1/month. To do so, for a given method and a given set of simulated outbreaks, we first compute the total posterior probability of an outbreak, $\Pr(H_1|D) = \sum_{S,E} \Pr(H_1(S,E)|D)$, for each day of the original dataset with no outbreaks injected. Then for each simulated outbreak, we compute $\Pr(H_1|D)$ for each outbreak day. For a given false positive rate r, the detection time d for a given outbreak is computed as the first outbreak

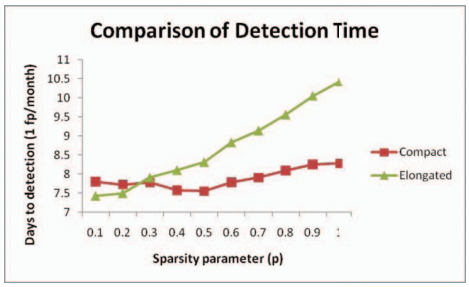

Figure 3. The average time to detection at 1 false positive/month for GFSS variants with different values of the sparsity parameter p, for compact and elongated outbreak regions.

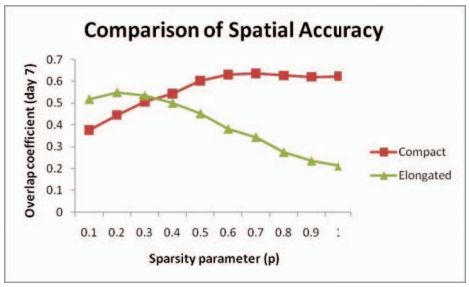

Figure 4. The average spatial accuracy (overlap coefficient) for GFSS variants with different values of the sparsity parameter p, for compact and elongated outbreak regions.

day ($t = 1 \ldots 10$) with posterior outbreak probability higher than the $100(1-r)$ percentile of the posterior probabilities for the original dataset. For a fixed false positive rate of 1/month, this corresponds to the 96.7th percentile. If no day of the outbreak has probability higher than this threshold, the method has failed to detect that outbreak, and we set $d = 10$. To evaluate spatial accuracy, we computed the average overlap coefficient between the true and detected clusters at day 7 of the outbreak. Given the set of locations S^* identified by the detection method (all s_i with posterior probabilities greater than half the total posterior probability of an outbreak) and the true set of affected locations S_{true}, the overlap coefficient is defined as $\frac{|S^* \cap S_{true}|}{|S^* \cup S_{true}|}$.

A. Preliminary Results

We first performed a simple evaluation of ten variants of the GFSS method with fixed values of the sparsity parameter $p = 0.1, 0.2, \ldots, 1.0$. As noted above, $p = 0.5$ and $p = 1.0$ correspond to the previously proposed FSS and MBSS methods respectively. We compared the detection power and spatial accuracy of these ten methods for two different outbreak types, one affecting a compact spatial region and one affecting an elongated region. As can be seen from Figures 3 and 4, substantial differences in the timeliness and accuracy of detection were observed with varying p: in particular, higher values of p tended to result in improved detection performance for more compact clusters, and lower values of p enhanced detection performance for more elongated clusters. For compact clusters, GFSS with $p = 0.5$ achieved the most timely detection, while $p = 0.7$ had slightly higher spatial accuracy. For elongated clusters, however, GFSS with $p = 0.2$ improved the timeliness of detection by nearly one day, and had a 10% higher overlap coefficient, than $p = 0.5$. These preliminary results demonstrate the importance of choosing an appropriate value for p; the experiments below demonstrate that the distribution of p can be learned accurately from labeled training data.

B. Outbreak Simulations for Learning Results

We now evaluate the detection performance of the learned GFSS model, as compared to the previously proposed MBSS and FSS methods, and also as compared to the GFSS approach assuming a uniform prior distribution of p. For these experiments, simulated outbreaks were generated using the same hierarchical generative model as assumed in the GFSS framework: given a value of the sparsity parameter p, the set of zip codes was selected by first choosing the center location and neighborhood size uniformly at random, and then independently choosing whether to include (with probability p) or exclude (with probability $1-p$) each location in that local neighborhood. We considered six different outbreak types: outbreaks generated using five different values of the sparsity parameter p ($p = 0.2, 0.4, \ldots, 1.0$), and a sixth outbreak type which consisted of an equal mixture of $p = 0.2$ and $p = 0.8$. For each combination of the value of sparsity parameter p and data stream, 100 outbreaks were injected to form a set of training data and another 100 outbreaks for forming the testing data. For each of the two data streams, this gives us a total of six datasets for training and another six corresponding datasets for testing, with each pair of datasets assuming a different value or mixture of the sparsity parameter p.

C. Learning Performance

In the present study, we assume that the value of p is drawn from a discrete distribution with ten components from 0.1 to 1.0, and thus we wish to learn the probability of each of the ten possible values of p from the training data (100 simulated injects for a given outbreak type). For each of the 12 training datasets (six for each data stream), the posterior distribution of the sparsity parameter p was learned and shown in Figures 5 and 6. For each of the first five experiments ($p = 0.2, 0.4, \ldots, 1.0$), we observe that the learned distribution of p correctly peaks at the true value of p for that set of simulated injects, for both cough and nausea cases. For the last two experiments, with half of the training examples assuming $p = 0.2$ and half assuming $p = 0.8$, we observe that the learned distribution is again able to recover the true bimodal distribution of p.

D. Detection Power and Spatial Accuracy Results

In this section, we compare the detection power and spatial accuracy of four different methods: (1) the previ-

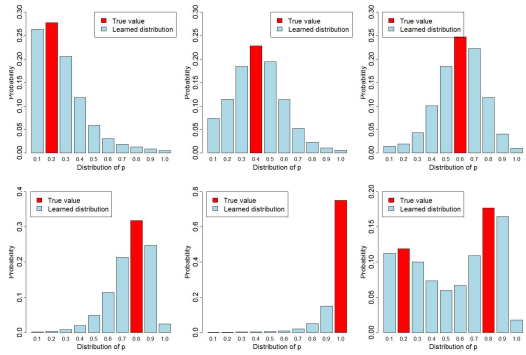

Figure 5. True value and learned distribution of sparsity parameter p, for six different simulated outbreak types injected into cough data from Allegheny County, PA.

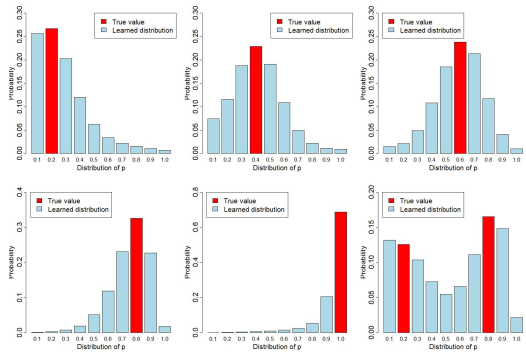

Figure 6. True value and learned distribution of sparsity parameter p, for six different simulated outbreak types injected into nausea data from Allegheny County, PA.

ously proposed MBSS method (special case of GFSS with $p = 1.0$), (2) the previously proposed FSS method (special case of GFSS with $p = 0.5$), (3) GFSS assuming a uniform distribution of sparsity parameter p (each value of p from $p = 0.1$ to $p = 1.0$ has an equal probability of 0.1), and (4) GFSS with a distribution of p learned from 100 labeled training examples.

The comparison of detection times for each of the two data streams, for simulated injects with $p = 0.2, 0.4, \ldots, 1.0$, is shown in Figure 7. The average detection time of each method, assuming a fixed false positive rate of 1 fp/month, is displayed on the graphs. When the value of p is small, corresponding to an elongated or irregular outbreak region, GFSS with learned p is able to detect the outbreaks substantially earlier than the other methods. The FSS method (equivalent to putting of all of the probability mass at $p = 0.5$) performs well for values of p near 0.5, and the MBSS method (equivalent to putting all of the probability mass at $p = 1.0$) performs well for values of p near 1.0, as expected, but both methods lose detection power when the assumed value of p is incorrect.

Next we evaluated the spatial accuracy of each method by computing the average overlap coefficient between the

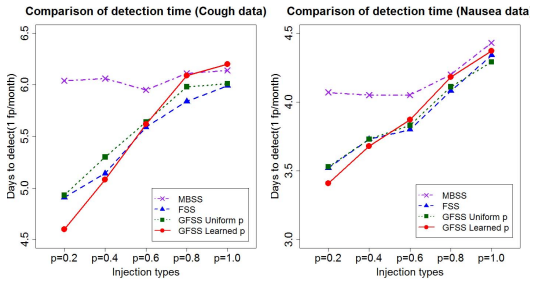

Figure 7. The detection time of four competing methods, for five different simulated outbreak types injected into cough and nausea data from Allegheny County, PA.

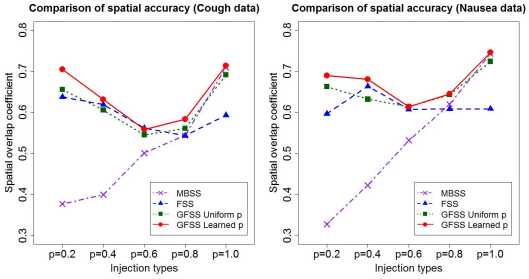

Figure 8. The spatial accuracy (overlap coefficient) of four competing methods, for five different simulated outbreak types injected into cough and nausea data from Allegheny County, PA.

true and detected clusters at day 7 of the outbreak. The comparison of overlap coefficients for each of the two data streams, for simulated injects with $p = 0.2, 0.4, \ldots, 1.0$, is shown in Figure 8. From these results, it can be observed that the learned GFSS method achieves similar spatial accuracy to the *best* of the other three methods for each value of p, and achieves significantly higher spatial accuracy when the outbreak region is elongated or irregular (i.e. for low values of the sparsity parameter p).

E. Detection Ability for Mixture Outbreak Type

In this section, we examine the detection time and spatial accuracy of these different methods for the mixed outbreak type (half of outbreaks generated with $p = 0.2$ and half of outbreaks generated with $p = 0.8$). We first consider a single distribution of p learned from the mixed outbreak type, as compared to MBSS, FSS, and GFSS with a uniform distribution of p. The last graphs of Figures 5 and 6 demonstrate that the single model can accurately capture the bimodal distribution of p. The results of detection time and spatial accuracy for the mixed outbreaks by using four different methods are listed in Table 1. The best results and those not significantly different from the best results are shown in bold. The GFSS with learned p slightly outperforms FSS and GFSS with uniform p for detecting mixture outbreaks, with all three methods outperforming MBSS by a large margin.

Next we assumed that the two values of p in the mixed outbreak type corresponded to two different outbreaks, and

Table I
COMPARISON OF DETECTION TIME AND SPATIAL ACCURACY FOR THE MIXED OUTBREAK TYPE

Data	Evaluation	MBSS	FSS	GFSS-uniform p	GFSS-learned p
Cough	Days to detect (1 fp/month)	6.16	**5.36**	**5.51**	**5.51**
Cough	Spatial overlap coefficient	0.478	**0.586**	**0.641**	**0.641**
Nausea	Days to detect (1 fp/month)	4.00	**3.67**	**3.59**	**3.61**
Nausea	Spatial overlap coefficient	0.517	0.623	**0.688**	**0.694**

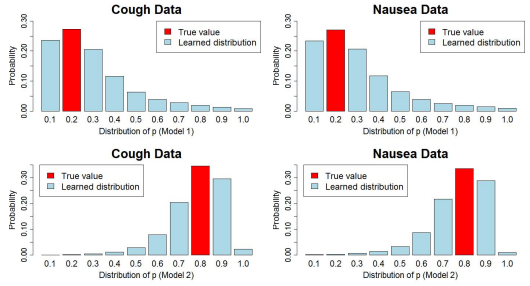

Figure 9. True value and learned distribution of the sparsity parameter p for the mixed outbreak type (equal mixture of $p = 0.2$ and $p = 0.8$) assuming two different outbreak models

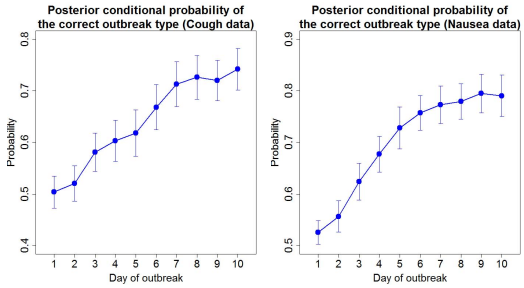

Figure 10. Posterior probability of the correct outbreak type as a function of day of outbreak, for cough and nausea data.

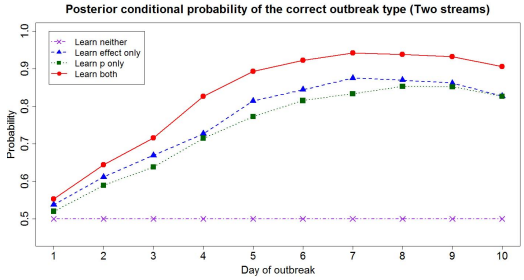

Figure 11. Posterior probability of the correct outbreak type as a function of day of outbreak, assuming two outbreak models and monitoring two data streams.

evaluated the ability of the GFSS framework to distinguish between these two outbreak types. Figure 9 is the result of learning the mixed type of outbreaks by using two GFSS models. We note that each model can capture each outbreak type quite well for both data streams. Additionally, using two models to learn the mixed outbreak type can also help us improve the ability to discriminate between the two different outbreak types. Figure 10 shows the average posterior conditional probability of the correct outbreak type, $\Pr(correct\ type \mid Data)/(\Pr(correct\ type \mid Data) + \Pr(incorrect\ type \mid Data))$, as a function of the outbreak day. As we can see, near the start of the outbreak, the posterior probability of an outbreak is divided nearly 50/50 between the correct and incorrect outbreak type, but by the end of the outbreak, posterior conditional probability of the correct outbreak type has risen to 76% for a cough outbreak or 79% for a nausea outbreak.

Finally, we note that, in addition to learning the distribution of the sparsity parameter, we can also learn the distribution of each outbreak type's relative effects on the two data streams from the same labeled training data, as in [8]. We considered two outbreak types which had both different values of p for the injected outbreaks ($p = 0.2$ and $p = 0.8$, as above) and also different relative effects on the two data streams: one outbreak type affected the cough stream twice as much as the nausea stream, and one type affected nausea twice as much as cough. As can be seen from Figure 11, either learning the sparsity parameter p or learning the relative effects of the outbreak on the two labeled data streams enabled accurate differentiation of the two outbreak types, with average posterior conditional probability of the correct outbreak type increasing to approximately 85% over the course of the outbreak. However, simultaneously learning *both* the sparsity and the effects enabled even higher accuracy, with average posterior probability of the correct outbreak type increasing to approximately 95%.

F. Robustness of the Learned GFSS Model

We performed three sets of follow-up experiments to evaluate the robustness of the learned Generalized Fast Subset Sums model to variation in a) the number of training examples, b) the number of discrete components used to learn the distribution of the sparsity parameter p, and c) the method used to generate simulated disease outbreaks.

First, while the experiments above assumed 100 training examples, we also evaluated the effects of using a smaller or larger training dataset, re-running the experiments using 25, 50, and 200 outbreaks as training data. The learned distribution of the sparsity parameter p in each case was very similar, and there were no significant differences in detection performance. Hence we conclude that the distribution of p can be learned accurately, and used to enhance the timeliness and accuracy of event detection, even when learning from only 25 labeled training examples.

Second, while the experiments above assumed that the sparsity parameter p for each training example was drawn from a discrete distribution $P = \{0.1, 0.2, \ldots, 1.0\}$ with ten components, we also evaluated the effects of using a

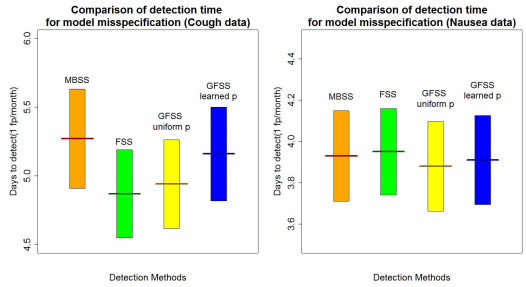

Figure 12. The average time to detection at 1 false positive/month, with 95% confidence intervals, of four competing methods, for simulated disease outbreaks generated based on spatial spread.

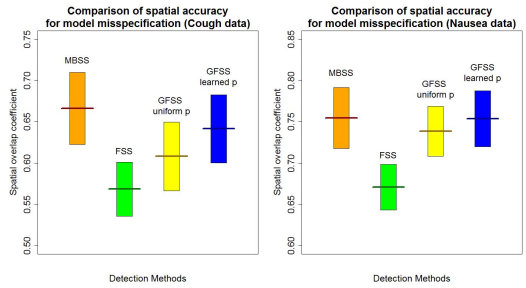

Figure 13. The average spatial accuracy (overlap coefficient), with 95% confidence intervals, of four competing methods, for simulated disease outbreaks generated based on spatial spread.

larger number of components, using the 100-component distribution $P = \{0.01, 0.02, \ldots, 1.0\}$. The learned distribution again converged around the true value of p for each experiment. However, there were no significant differences in detection power or spatial accuracy, and computation time for the 100-component distribution was approximately ten times as long as the 10-component distribution. These results suggest that our original choice of ten components was sufficient for learning the sparsity parameter p.

Third, while the experiments above assumed a correctly specified generative model for the simulated injects (i.e. each inject was generated using the hierarchical prior distribution assumed by the GFSS model), we also tested the robustness of GFSS to model misspecification, by evaluating the performance of the learned GFSS model (as compared to the uniform GFSS model, MBSS, and FSS) on a separate set of 100 training and 100 test outbreaks which were not generated using the GFSS model. Instead, these injects assume a spatial model of disease spread: the outbreak starts at a randomly selected zip code, and on each outbreak day it affects that zip code and its $k-1$ nearest neighbors, based on Euclidean distance between the affected zip codes. The number of zip codes affected, and the expected number of injected cases, both grow linearly over the course of the outbreak, with zip codes near the center of the outbreak receiving a proportionately greater number of cases.

The detection time and spatial accuracy results for this set of experiments are shown in Figures 12 and 13 respectively. We observe that performance of the GFSS method with learned distribution of p is not significantly different from the best performing method in terms of detection time or spatial accuracy, suggesting that a useful distribution for p can still be learned even when the model is misspecified.

IV. Conclusions and Future Work

The Generalized Fast Subset Sums (GFSS) framework is an generalization of the previously proposed Multivariate Bayesian Scan Statistic (MBSS) and Fast Subset Sums (FSS) methods, and includes both MBSS and FSS as special cases. A novel hierarchical prior over the 2^N subsets of the data is proposed, parameterized by the center location s_c, neighborhood size k, and sparsity p. The new sparsity parameter p in the GFSS framework describes the expected proportion of locations affected within a given circular neighborhood, and thus can be varied to emphasize detection of more compact or more dispersed clusters. We demonstrate that the posterior distribution of the sparsity parameter can be learned accurately based on labeled training data, even when the size of the training sample is small. With the learned sparsity parameter, the GFSS method has higher detection power and higher spatial accuracy than the previously proposed FSS and MBSS methods, especially for elongated or irregular outbreaks. Additionally, learning two different models for outbreak types which have different sparsities (but are otherwise identical) allows us to precisely distinguish between the two outbreak types. Finally, we demonstrate that the GFSS method with learned distribution of p also performs very well even for outbreaks which are not generated using the GFSS framework.

In future work, we will extend the GFSS framework by also learning the distributions of the center location s_c and the neighborhood size k from labeled training data. We will also consider the case of partially labeled data, when only a subset of the affected locations is identified. Finally, we will examine the effects of allowing the probability that a location is affected given the neighborhood to vary spatially rather than assuming that p is constant. For example, each location s_i could be affected with a probability p_i that decreases with its distance from the center location s_c. If the value of p_i is only dependent on the location s_i and local neighborhood S_{ck} under consideration, then efficient computation of posterior probabilities is still possible in this more general setting.

Acknowledgment

This work was partially supported by the National Science Foundation under grants IIS-0916345, IIS-0911032, and IIS-0953330.

REFERENCES

[1] G. F. Cooper, D. H. Dash, J. D. Levander, W.-K. Wong, W. R. Hogan, and M. M. Wagner. Bayesian biosurveillance of disease outbreaks. In *Proc. Conference on Uncertainty in Artificial Intelligence*, 2004.

[2] G. F. Cooper, J. N. Dowling, J. D. Levander, and P. Sutovsky. A Bayesian algorithm for detecting CDC Category A outbreak diseases from emergency department chief complaints. *Advances in Disease Surveillance*, 2:45, 2007.

[3] L. Duczmal and R. Assuncao. A simulated annealing strategy for the detection of arbitrary shaped spatial clusters. *Comp. Stat. and Data Analysis*, 45:269–286, 2004.

[4] M. Kulldorff. A spatial scan statistic. *Communications in Statistics: Theory and Methods*, 26(6):1481–1496, 1997.

[5] M. Kulldorff, F. Mostashari, L. Duczmal, W. K. Yih, K. Kleinman, and R. Platt. Multivariate scan statistics for disease surveillance. *Statistics in Medicine*, 26:1824–1833, 2007.

[6] M. Makatchev and D. B. Neill. Learning outbreak regions in Bayesian spatial scan statistics. In *Proc. ICML/UAI/COLT Workshop on Machine Learning for Health Care Applications*, 2008.

[7] D. B. Neill. Fast Bayesian scan statistics for multivariate event detection and visualization. *Statistics in Medicine*, 30(5):455–469, 2011.

[8] D. B. Neill and G. F. Cooper. A multivariate Bayesian scan statistic for early event detection and characterization. *Machine Learning*, 79(3):261–282, 2010.

[9] D. B. Neill and J. Lingwall. A nonparametric scan statistic for multivariate disease surveillance. *Advances in Disease Surveillance*, 4:106, 2007.

[10] G. P. Patil and C. Taillie. Upper level set scan statistic for detecting arbitrarily shaped hotspots. *Envir. Ecol. Stat.*, 11:183–197, 2004.

[11] W.-K. Wong, A. W. Moore, G. F. Cooper, and M. M. Wagner. Bayesian network anomaly pattern detection for disease outbreaks. In *Proc. 20th International Conference on Machine Learning*, 2003.