Fast Graph Scan for Scalable Detection of Arbitrary Connected Clusters

Skyler Speakman & Daniel B. Neill
Carnegie Mellon University, Heinz College
ISDS Annual Conference

This work was partially supported by NSF grants IIS-0916345, IIS-0911032, and IIS-0325581
Daily health data from thousands of hospitals and pharmacies nationwide

Time series of counts $c_i^t$ for each zip code $s_i$

Use this data to detect anomalous patterns

Detect any emerging events (i.e. outbreaks of disease)
Pinpoint the affected areas

Biosurveillance
Expectation-based Scan Statistics

(Kulldorff, 1997; Neill and Moore, 2005)

Scan over multiple regions to detect where counts are higher than expected.

Aggregate the individual counts from each location within a region.

**Circles**

Choose a center location \( s_c \) and its \( k \) nearest neighbors.

Find the circle that maximizes a given score function of the aggregated counts and baselines.

**Expectation-based Scan Statistics**
Expectation-based Scan Statistics

(Kulldorff, 1997; Neill and Moore, 2005)

Scan over multiple regions to detect where counts are higher than expected.

Aggregate the individual counts from each location within a region.

Rectangles

Find the rectangle that maximizes a given score function of the aggregated counts and baselines.

Expectation-based Scan Statistics
Expectation-based Scan Statistics

(Kulldorff, 1997; Neill and Moore, 2005)

Power to Detect

Circles are useful for detecting tightly clustered outbreaks

However, they lose power to detect abnormally shaped clusters

- Affected locations
- Un-affected locations contributing to region score

Expectation-based Scan Statistics
Expectation-based Scan Statistics

(Kulldorff, 1997; Neill and Moore, 2005)

Power to Detect

There are similar issues with rectangles for some outbreaks

- Affected locations
- Un-affected locations contributing to region score

Expectation-based Scan Statistics
An alternative to scanning over shapes of regions is to find the *subset of locations* for a given region that has the highest score.

Affected locations

Un-affected locations contributing to region score

*Pattern Detection through Subset Scanning*

(Neill, 2008)
<table>
<thead>
<tr>
<th>PROBLEM:</th>
<th>The number of subsets grows exponentially with the size of the region (2^n)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This makes it computationally infeasible for regions with more than (~30) locations</td>
</tr>
<tr>
<td>SOLUTION:</td>
<td>Exploit a property of scoring functions to rule out subsets that cannot obtain the highest score</td>
</tr>
<tr>
<td></td>
<td>This reduction in the search space allows for exact and efficient calculation of the highest scoring subset</td>
</tr>
</tbody>
</table>

(Neill, 2008)
We wish to maximize a scoring function

\[ F \subseteq F \left( \sum_{s_i \in S} c_i, \sum_{s_i \in S} b_i \right) \]

over all possible subsets, \( S \)

We sort the locations according to a relevance criteria

For example,

\[ G(s_i) = \frac{c_i}{b_i} \]

works for Kulldorff’s Statistic and Expectation-based Poisson
Linear Time Subset Scanning

We wish to maximize a scoring function

$$F \subseteq F \left( \sum_{s_i \in S} c_i, \sum_{s_i \in S} b_i \right)$$

over all possible subsets, $S$

We sort the locations according to a relevance criteria

For example,

$$G(s_i) = \frac{c_i}{b_i}$$

works for Kulldorff’s Statistic and Expectation-based Poisson

This location has the highest count-to-baseline ratio

This location has the lowest count-to-baseline ratio

This ranking allows LTSS to take advantage of properties of a large number of scoring functions

(Neill, 2008)
The highest scoring subset is guaranteed to be one of the following subsets:

Decreases the search space from $2^N$ to $N$
Use adjacency of locations to form a *flexible* scan statistic (Tango & Takahashi, 2005)

Create an adjacency graph of the locations and score *every connected subset*

Increase power to detect non-circular clusters

Number of connected subsets is exponential in size of region. Infeasible for regions of >30 locations

**Connectivity Constraints**
Use property of LTSS to reduce the search space and rule out a large number of connected subsets.

Rank the locations according to relevance criteria.

Only scan connected subsets that have potential for highest score.

Graphscan:
If location $s_i$ is contained in the optimal subset $S^*$ and if a neighbor of $s_i$ with higher relevance does not disconnect the subgraph, then $s_i$ can also be contained in $S^*$.
The Graphscan algorithm would end up evaluating the sets:

1 2 3

Why not the sets 3 or 1 3 or 2 3 ?

Because these sets could include a higher ranked neighbor that would increase the set’s score.

Brief Example
The GraphScan method was evaluated using Emergency Department data from 91 Allegheny County zip codes.

**Runtimes**

We can use LTSS to quickly determine the unconstrained bound of a given subset.

If the subset’s bound is less than the current high score, we do not have to include it.

...for a single day of data.
We compared the detection power and accuracy of GraphScan to the original Kulldorff scan statistic (circular regions) on multiple semi-synthetic outbreaks injected into the data.

<table>
<thead>
<tr>
<th>Average over all types of injects</th>
<th>% of Injects Detected</th>
<th>Days to detect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circles</td>
<td>83.6%</td>
<td>8.6</td>
</tr>
<tr>
<td>GraphScan K=25</td>
<td>88.2%</td>
<td>8.2</td>
</tr>
<tr>
<td>GraphScan K=50</td>
<td>89.4%</td>
<td>8.1</td>
</tr>
<tr>
<td>GraphScan Single Region</td>
<td>88.6%</td>
<td>8.1</td>
</tr>
</tbody>
</table>
We compared the detection power and accuracy of GraphScan to the original Kulldorff scan statistic (circular regions) on multiple semi-synthetic outbreaks injected into the data.

<table>
<thead>
<tr>
<th>Compact Cluster</th>
<th>% Detected</th>
<th>Days to Detect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circles</td>
<td>68%</td>
<td>10.4</td>
</tr>
<tr>
<td>Graphscan K=25</td>
<td>84%</td>
<td>9.3</td>
</tr>
<tr>
<td>Graphscan K=50</td>
<td>88%</td>
<td>8.3</td>
</tr>
<tr>
<td>Graphscan Single Region</td>
<td>88%</td>
<td>8.6</td>
</tr>
</tbody>
</table>
We compared the detection power and accuracy of GraphScan to the original Kulldorff scan statistic (circular regions) on multiple semi-synthetic outbreaks injected into the data.

<table>
<thead>
<tr>
<th>Elongated Cluster</th>
<th>% Detected</th>
<th>Days to Detect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circles</td>
<td>66%</td>
<td>10.4</td>
</tr>
<tr>
<td>Graphscan K=25</td>
<td>87%</td>
<td>8.5</td>
</tr>
<tr>
<td>Graphscan K=50</td>
<td>92%</td>
<td>8.0</td>
</tr>
<tr>
<td>Graphscan Single Region</td>
<td>92%</td>
<td>8.2</td>
</tr>
</tbody>
</table>
We compared the detection power and accuracy of GraphScan to the original Kulldorff scan statistic (circular regions) on multiple semi-synthetic outbreaks injected into the data.

<table>
<thead>
<tr>
<th>Irregular Cluster</th>
<th>% Detected</th>
<th>Days to Detect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circles</td>
<td>90%</td>
<td>8.7</td>
</tr>
<tr>
<td>Graphscan K=25</td>
<td>97%</td>
<td>7.6</td>
</tr>
<tr>
<td>Graphscan K=50</td>
<td>98%</td>
<td>7.5</td>
</tr>
<tr>
<td>Graphscan Single Region</td>
<td>96%</td>
<td>7.4</td>
</tr>
</tbody>
</table>
Thanks!