

# Event and Pattern Detection at the Societal Scale

Daniel B. Neill  
H.J. Heinz III College  
Carnegie Mellon University  
E-mail: [neill@cs.cmu.edu](mailto:neill@cs.cmu.edu)

We gratefully acknowledge funding support from the National Science Foundation, grants IIS-0916345, IIS-0911032, and IIS-0953330, and the John D. and Catherine T. MacArthur Foundation.

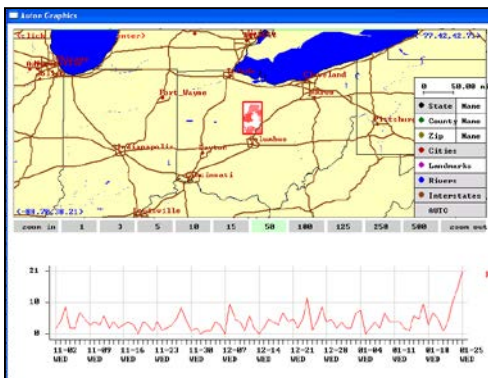
Carnegie Mellon University

EPD Lab

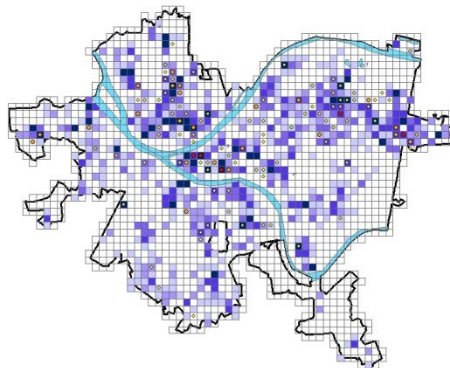
EVENT AND PATTERN DETECTION LABORATORY



Daniel B. Neill (neill@cs.cmu.edu)  
Associate Professor of Information Systems, Heinz College, CMU  
Director, Event and Pattern Detection Laboratory  
Courtesy Associate Professor of Machine Learning and Robotics



Disease Surveillance:  
Very early and accurate detection of emerging outbreaks.



Law Enforcement:  
Detection, prediction, and prevention of “hot-spots” of violent crime.



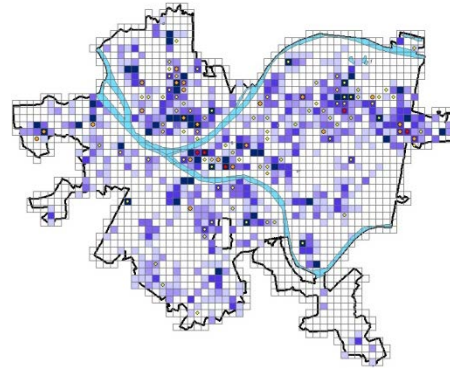
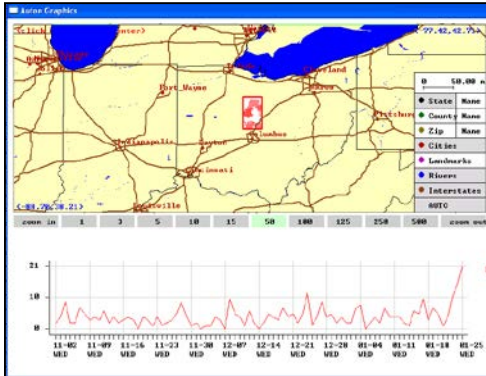
Medicine: Discovering new “best practices” of patient care, to improve outcomes and reduce costs.

My research is focused at the intersection of **machine learning** and **public policy**, with two main goals:

- 1) Develop new machine learning methods for better (more scalable and accurate) **detection** and **prediction** of events and other patterns in massive datasets.
- 2) Apply these methods to improve the quality of public health, safety, and security.



Daniel B. Neill (neill@cs.cmu.edu)  
 Associate Professor of Information Systems, Heinz College, CMU  
 Director, Event and Pattern Detection Laboratory  
 Courtesy Associate Professor of Machine Learning and Robotics



Disease Surveillance:  
 Very early and accurate detection of emerging outbreaks.

Law Enforcement:  
 Detection, prediction, and prevention of “hot-spots” of violent crime.

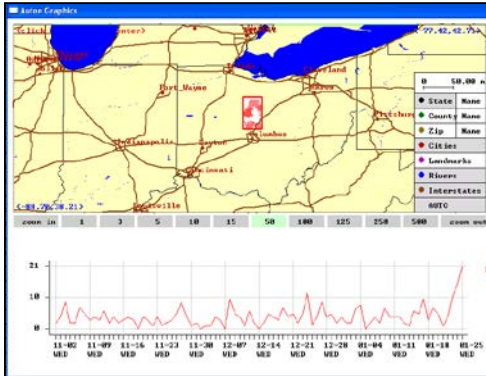
Medicine: Discovering new “best practices” of patient care, to improve outcomes and reduce costs.

Our disease surveillance methods are currently in use for deployed systems in the U.S., Canada, India, and Sri Lanka.

Our “CrimeScan” software has been in day-to-day operational use for predictive policing by the Chicago Police Dept. “CityScan” has been used by Chicago city leaders for prediction and prevention of rodent infestations using 311 call data.



Daniel B. Neill (neill@cs.cmu.edu)  
 Associate Professor of Information Systems, Heinz College, CMU  
 Director, Event and Pattern Detection Laboratory  
 Courtesy Associate Professor of Machine Learning and Robotics



Disease Surveillance:  
 Very early and accurate detection of emerging outbreaks.

*“CrimeScan was set up to run daily, completely autonomously. Predictions were sent to police analysts, and messages were compiled into detailed intelligence reports disseminated through the chain of command.*

*Based upon deployment suggestions indicated in the CrimeScan reports, **important arrests were effected, weapons were seized, and crimes were prevented.**”*

Our disease surveillance methods are currently in use for deployed systems in the U.S., Canada, India, and Sri Lanka.

Our “CrimeScan” software has been in day-to-day operational use for predictive policing by the Chicago Police Dept. “CityScan” has been used by Chicago city leaders for prediction and prevention of rodent infestations using 311 call data.

# Pattern detection by subset scan

One key insight that underlies much of my work is that pattern detection can be viewed as a **search** over subsets of the data.

## Statistical challenges:

Which subsets to search?  
Is a given subset anomalous?  
Which anomalies are relevant?

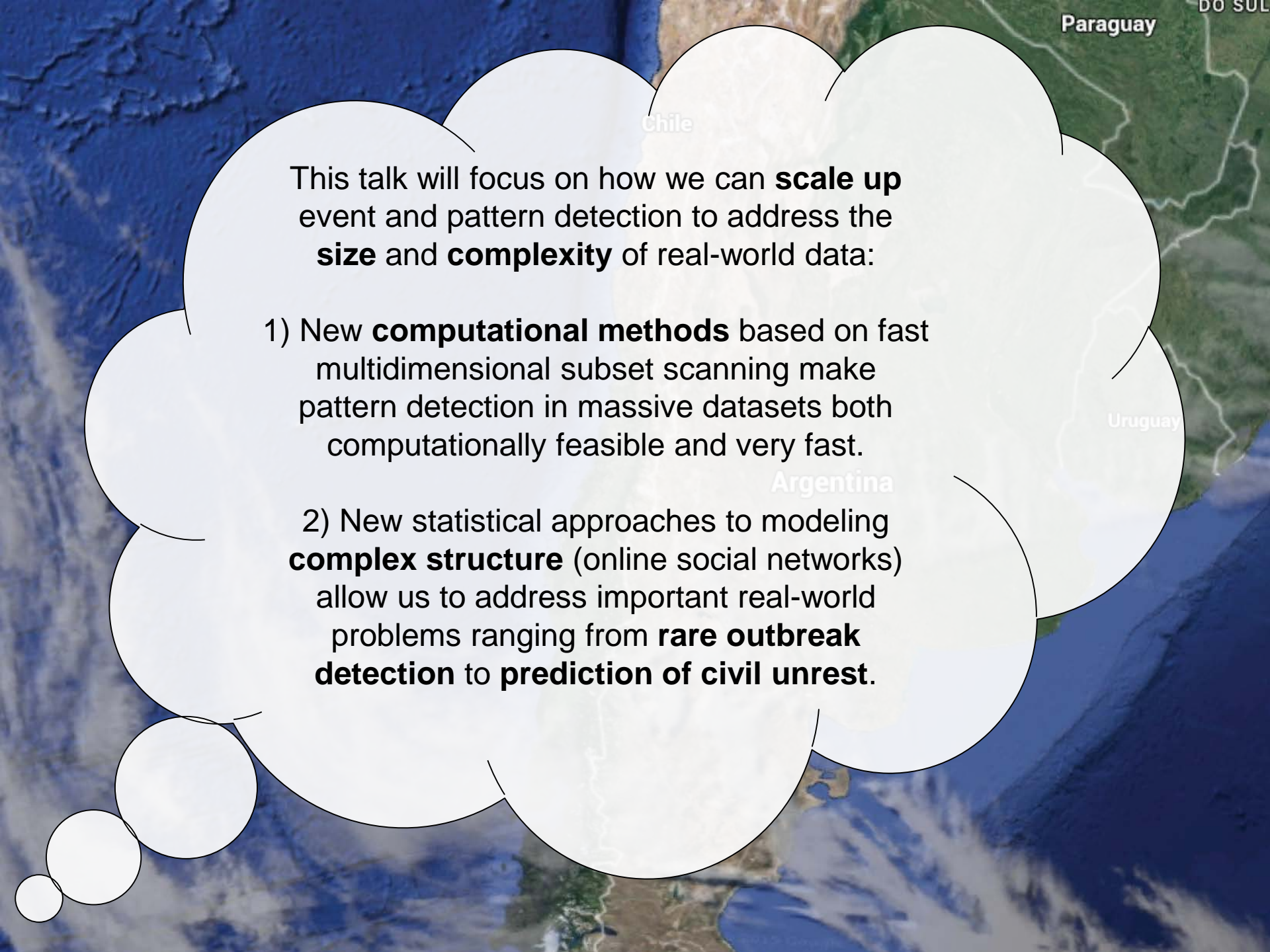
## Computational challenge:

How to make this search over subsets efficient for massive, complex, high-dimensional data?

New statistical methods enable more timely and more accurate detection by integrating **multiple data sources**, incorporating **spatial** and **temporal** information, and using **prior knowledge** of a domain.

New algorithms and data structures make previously impossible detection tasks computationally feasible and fast.

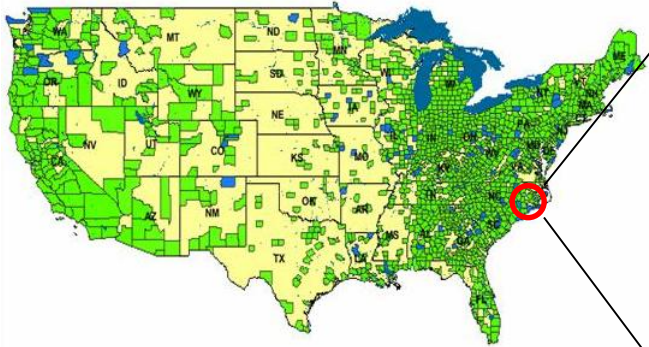
New machine learning methods enable our systems to learn from user feedback, modeling and distinguishing between relevant and irrelevant types of anomaly.



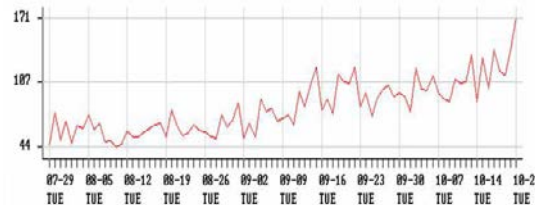
This talk will focus on how we can **scale up** event and pattern detection to address the **size** and **complexity** of real-world data:

- 1) New **computational methods** based on fast multidimensional subset scanning make pattern detection in massive datasets both computationally feasible and very fast.
- 2) New statistical approaches to modeling **complex structure** (online social networks) allow us to address important real-world problems ranging from **rare outbreak detection** to **prediction of civil unrest**.

# Multivariate event detection



Spatial time series data from spatial locations  $s_i$  (e.g. zip codes)



Time series of counts  $c_{i,m}^t$  for each zip code  $s_i$  for each data stream  $d_m$ .

## Outbreak detection

$d_1$  = respiratory ED

$d_2$  = constitutional ED

$d_3$  = OTC cough/cold

$d_4$  = OTC anti-fever

(etc.)

## Main goals:

**Detect** any emerging events.

**Pinpoint** the affected subset of locations and time duration.

**Characterize** the event, e.g., by identifying the affected streams.

## Compare hypotheses:

$H_1(D, S, W)$

$D$  = subset of streams

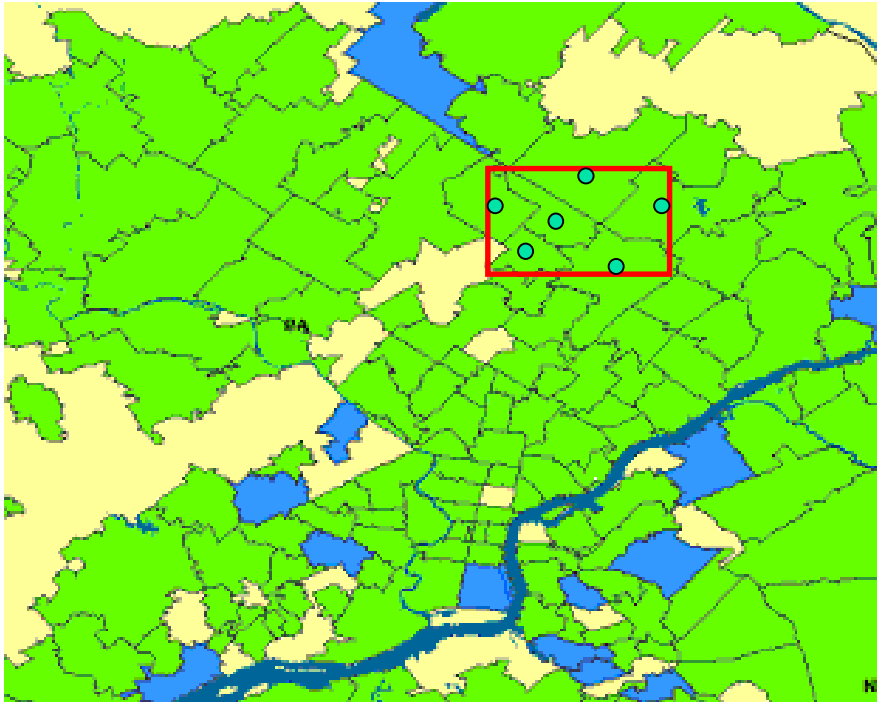
$S$  = subset of locations

$W$  = time duration

vs.  $H_0$ : no events occurring

# Expectation-based scan statistics

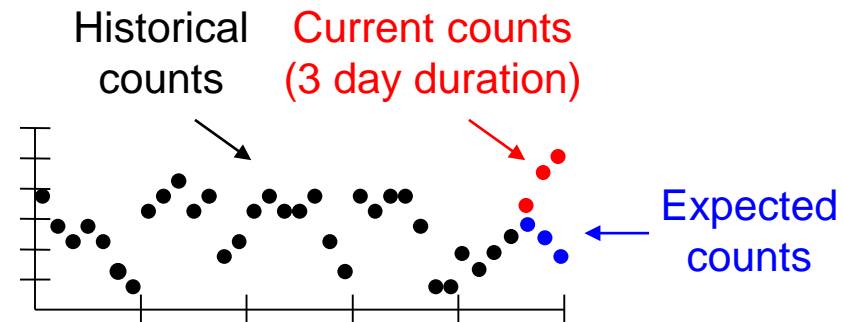
(Kulldorff, 1997; Neill and Moore, 2005)



We search for spatial regions (subsets of locations) where the recently observed counts for some subset of streams are significantly higher than expected.

We perform **time series analysis** to compute expected counts (“baselines”) for each location and stream for each recent day.

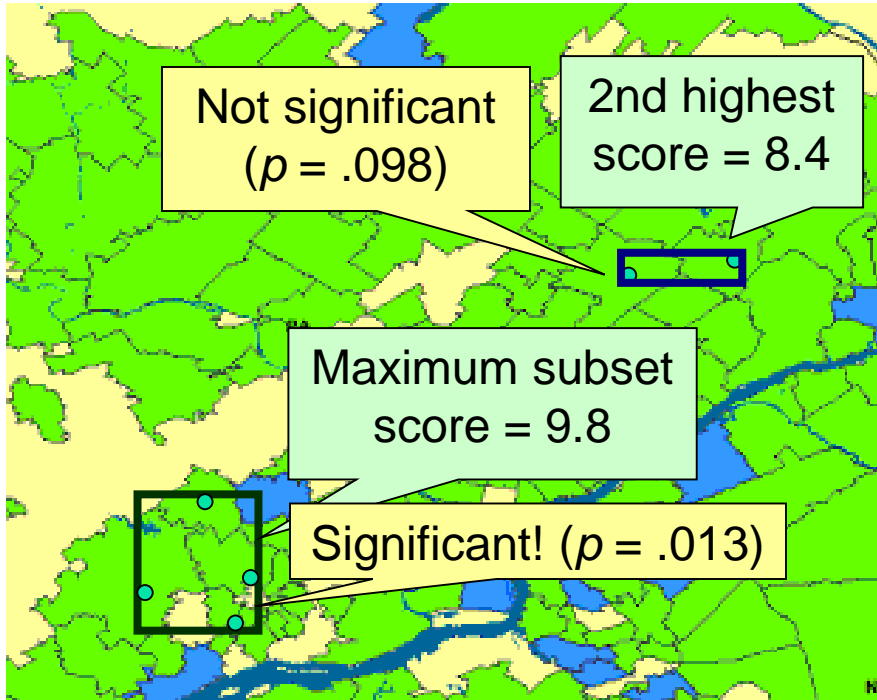
We then compare the actual and expected counts for each subset (D, S, W) under consideration.





# Expectation-based scan statistics

(Kulldorff, 1997; Neill and Moore, 2005)

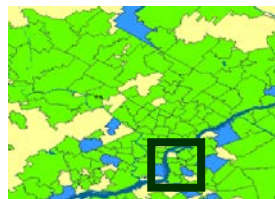


We find the subsets with highest values of a **likelihood ratio statistic**, and compute the  $p$ -value of each subset by randomization testing.

$$F(D, S, W) = \frac{\Pr(\text{Data} \mid H_1(D, S, W))}{\Pr(\text{Data} \mid H_0)}$$

To compute  $p$ -value  
Compare subset score to maximum subset scores of simulated datasets under  $H_0$ .

$F_1^* = 2.4$



$F_2^* = 9.1$



...

$F_{999}^* = 7.0$



# Likelihood ratio statistics

For our expectation-based scan statistics, the null hypothesis  $H_0$  assumes “business as usual”: each count  $c_{i,m}^t$  is drawn from some parametric distribution with mean  $b_{i,m}^t$ .  $H_1(S)$  assumes a multiplicative increase for the affected subset  $S$ .

## Expectation-based Poisson

$$H_0: c_{i,m}^t \sim \text{Poisson}(b_{i,m}^t)$$

$$H_1(S): c_{i,m}^t \sim \text{Poisson}(qb_{i,m}^t)$$

$$\text{Let } C = \sum_S c_{i,m}^t \text{ and } B = \sum_S b_{i,m}^t.$$

$$\text{Maximum likelihood: } q = C / B.$$

$$F(S) = C \log (C/B) + B - C$$

## Expectation-based Gaussian

$$H_0: c_{i,m}^t \sim \text{Gaussian}(b_{i,m}^t, \sigma_{i,m}^t)$$

$$H_1(S): c_{i,m}^t \sim \text{Gaussian}(qb_{i,m}^t, \sigma_{i,m}^t)$$

$$\text{Let } C' = \sum_S c_{i,m}^t b_{i,m}^t / (\sigma_{i,m}^t)^2 \\ \text{and } B' = \sum_S (b_{i,m}^t)^2 / (\sigma_{i,m}^t)^2.$$

$$\text{Maximum likelihood: } q = C' / B'.$$

$$F(S) = (C')^2 / 2B' + B'/2 - C'$$

Many possibilities: exponential family, nonparametric, Bayesian...

# Which regions to search?

Typical approach: “spatial scan” (Kulldorff, 1997)

Each search region  $S$  is a **sub-region** of space.

- Choose some region shape (e.g. circles, rectangles) and consider all regions of that shape and varying size.
- Low power for true events that do not correspond well to the chosen set of search regions (e.g. irregular shapes).

Our approach: “subset scan” (Neill, 2012)

Each search region  $S$  is a **subset** of locations.

- Find the highest scoring subset, subject to some constraints (e.g. spatial proximity, connectivity).
- For multivariate, also optimize over subsets of streams.
- Exponentially many possible subsets,  $O(2^N \times 2^M)$ : computationally infeasible for naïve search.

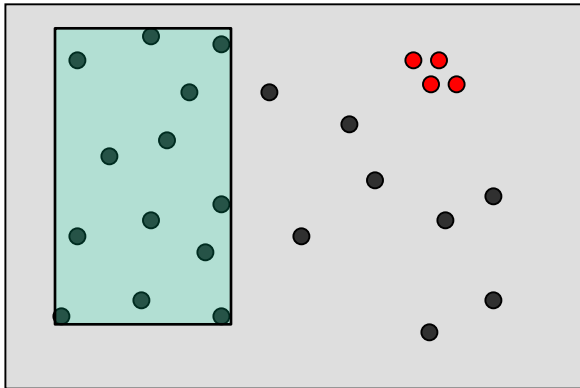
# Question: Why search over subsets?

## Answer: Simpler approaches can fail.

### Top-down detection approaches

Are there any globally interesting patterns? If so, recursively search the most interesting sub-partition.

Two examples: bump hunting;  
“cluster then detect”.

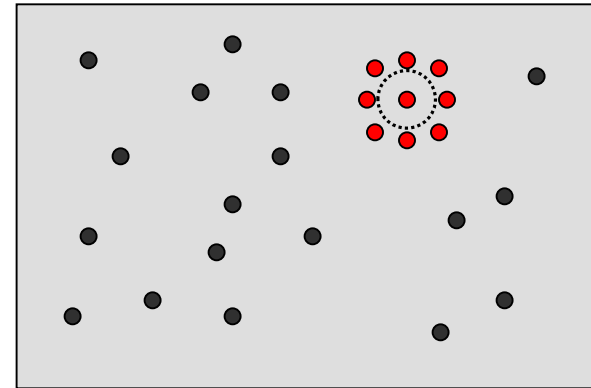


Top-down fails for **small-scale patterns** that are not evident from the global aggregates.

### Bottom-up detection approaches

Find individually (or locally) anomalous data points, and optionally, aggregate into clusters.

Two examples: anomaly/outlier detection;  
density-based clustering.



Bottom-up fails for **subtle patterns** that are only evident when a group of data records are considered collectively.

# Question: Why search over subsets? Answer: Simpler approaches can fail.

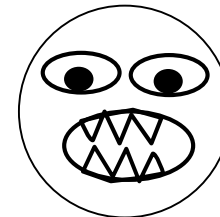
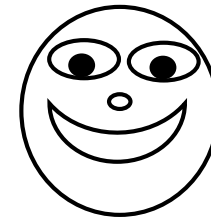
Top-down detection approaches

Are there any patterns?  
the most interesting

So here's where we are so far:

Treating pattern detection as a subset scan problem is statistically desirable for maximizing detection power...

but computationally infeasible (for exhaustive search at least).



Top-down fails to find **subtle patterns** that are not evident when a group of data records are considered collectively.

# Fast subset scan (Neill, 2012)

- In certain cases, we can optimize  $F(S)$  over the exponentially many subsets of the data, while evaluating only  $O(N)$  rather than  $O(2^N)$  subsets.
- Many commonly used scan statistics have the property of linear-time subset scanning:
  - Just sort the data records (or spatial locations, etc.) from highest to lowest priority according to some function...
  - ... then search over groups consisting of the top-k highest priority records, for  $k = 1..N$ .

The highest scoring subset is **guaranteed** to be one of these!

Sample result: we can find the **most anomalous** subset of Allegheny County zip codes in **0.03 sec** vs.  **$10^{24}$  years**.

# Linear-time subset scanning

- Example: Expectation-Based Poisson statistic
  - Sort data locations  $s_i$  by the ratio of observed to expected count,  $c_i / b_i$ .
  - Given the ordering  $s_{(1)} \dots s_{(N)}$ , we can **prove** that the top-scoring subset  $F(S)$  consists of the locations  $s_{(1)} \dots s_{(k)}$  for some  $k$ ,  $1 \leq k \leq N$ .
  - Key step: if there exists some location  $s_{\text{out}} \notin S$  with higher priority than some location  $s_{\text{in}} \in S$ , then we can show that  $F(S) \leq \max(F(S \cup \{s_{\text{out}}\}), F(S \setminus \{s_{\text{in}}\}))$ .
- Theorem: LTSS holds for expectation-based scan statistics in any exponential family. (Speakman et al., 2015)

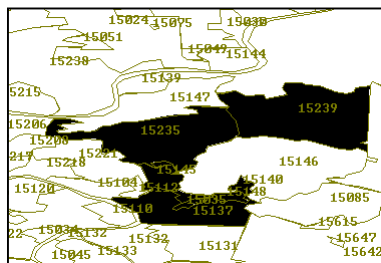
$$F(S) = \max_{q>1} \log \frac{P(\text{Data} \mid H_1(S))}{P(\text{Data} \mid H_0)} \quad \begin{array}{l} H_0 : x_i \sim \text{Dist}(\mu_i) \\ H_1 : x_i \sim \text{Dist}(q\mu_i) \end{array}$$

# Constrained fast subset scanning

LTSS is a new and powerful tool for **exact** combinatorial optimization (as opposed to approximate techniques such as submodular function optimization). But it only solves the “best unconstrained subset” problem, and cannot be used directly for constrained optimization.

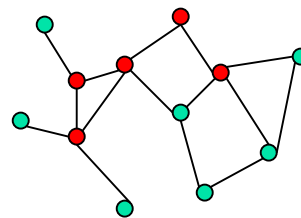
Many of our recent papers have focused on how LTSS can be extended to the many real-world problems with (hard or soft) constraints on our search.

- + Proximity constraints → Fast spatial scan (irregular regions)
- + Multiple data streams → Fast multivariate scan
- + Connectivity constraints → Fast graph scan
- + Group self-similarity → Fast generalized subset scan

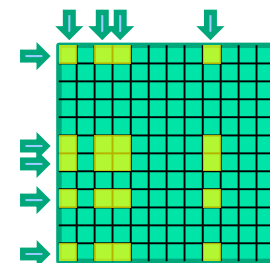


(Neill, *JRSS-B*, 2012)

(Neill et al., *Stat. Med.*, 2013)



(Speakman et al., *JCGS*, 2015)



(McFowland et al., *JMLR*, 2013)

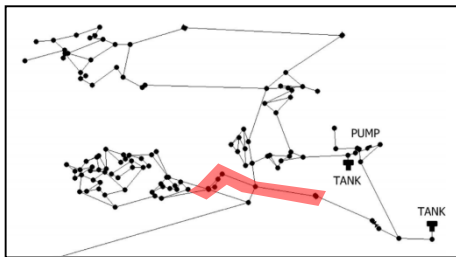


# Constrained fast subset scanning

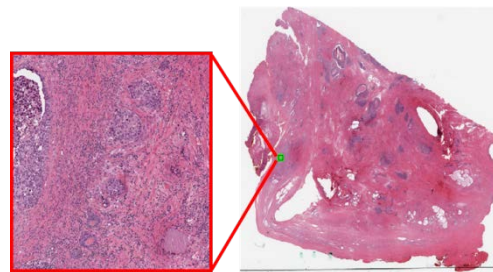
LTSS is a new and powerful tool for **exact** combinatorial optimization (as opposed to approximate techniques such as submodular function optimization). But it only solves the “best unconstrained subset” problem, and cannot be used directly for constrained optimization.

Many of our recent papers have focused on how LTSS can be extended to the many real-world problems with (hard or soft) constraints on our search.

- + Temporal dynamics → Spreading contamination in water supply
- + Hierarchical scanning → Prostate cancer in digital pathology slides
- + Scalable GP regression → Predicting and preventing rat infestations



(Speakman et al., ICDM 2013)



(Somanchi & Neill, DMHI 2013)



(Flaxman et al., 2015;  
Neill et al., in preparation)

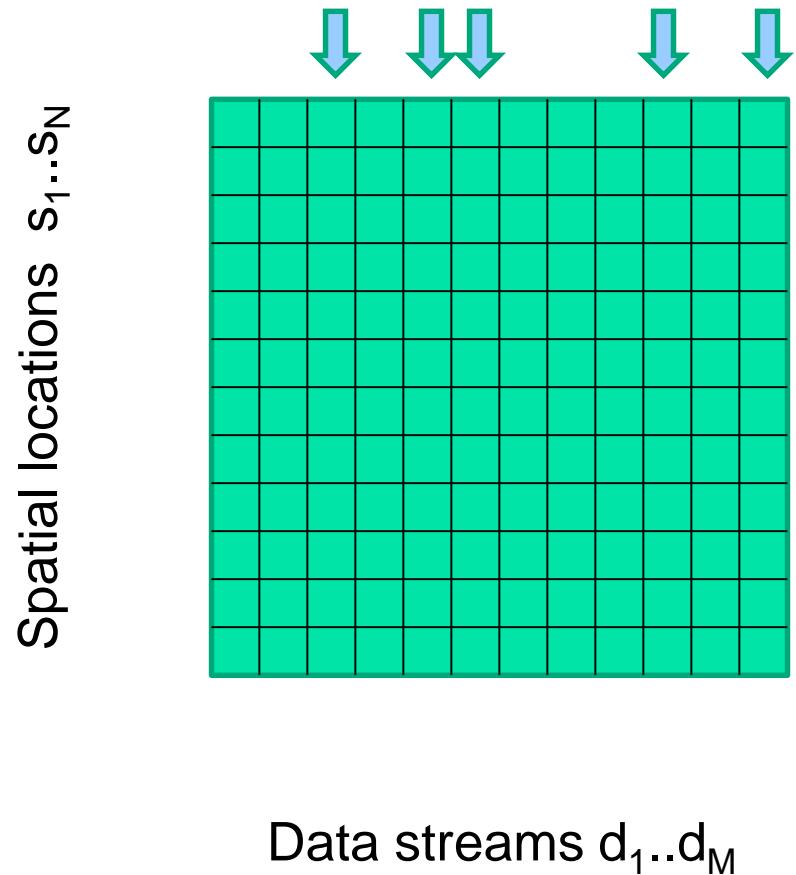
# Fast subset scan with spatial proximity constraints

- Maximize a likelihood ratio statistic over all subsets of the “local neighborhoods” consisting of a center location  $s_i$  and its  $k-1$  nearest neighbors, for a fixed neighborhood size  $k$ .
- Naïve search requires  $O(N \cdot 2^k)$  time and is computationally infeasible for  $k > 25$ .
- For each center, we can search over all subsets of its local neighborhood in  $O(k)$  time using LTSS, thus requiring a total time complexity of  $O(Nk) + O(N \log N)$  for sorting the locations.
- In Neill (2012), we show that this approach dramatically improves the timeliness and accuracy of outbreak detection for irregularly-shaped disease clusters.

# Multivariate fast subset scan

(Neill, McFowland, and Zheng, 2013)

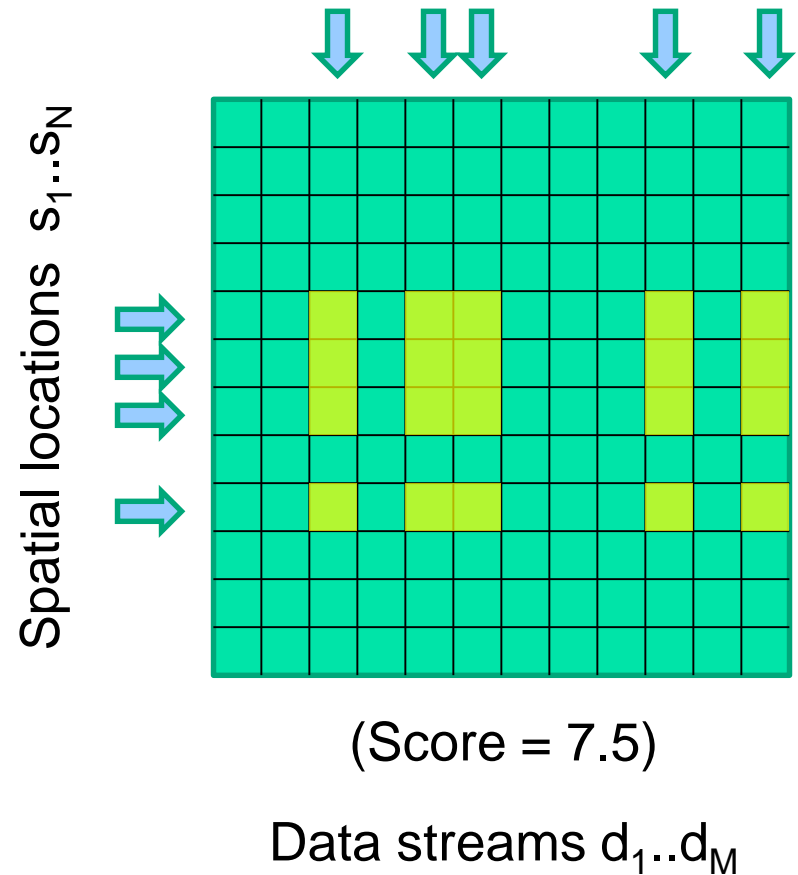
- The LTSS property allows us to efficiently optimize over subsets of spatial locations for a given subset of streams.



# Multivariate fast subset scan

(Neill, McFowland, and Zheng, 2013)

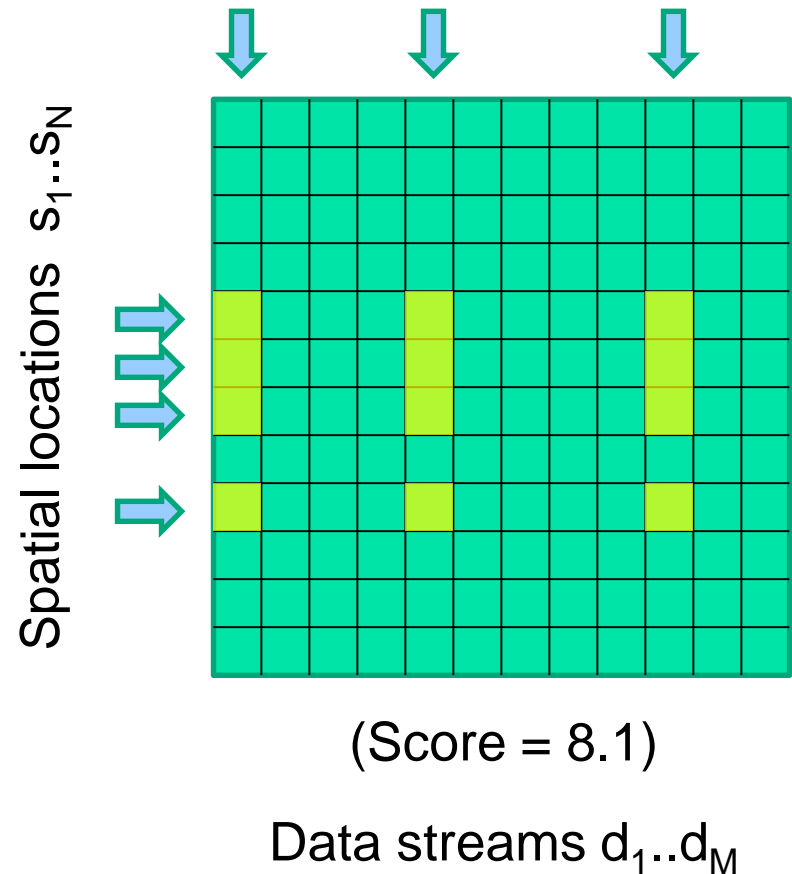
- The LTSS property allows us to efficiently optimize over subsets of spatial locations for a given subset of streams.



# Multivariate fast subset scan

(Neill, McFowland, and Zheng, 2013)

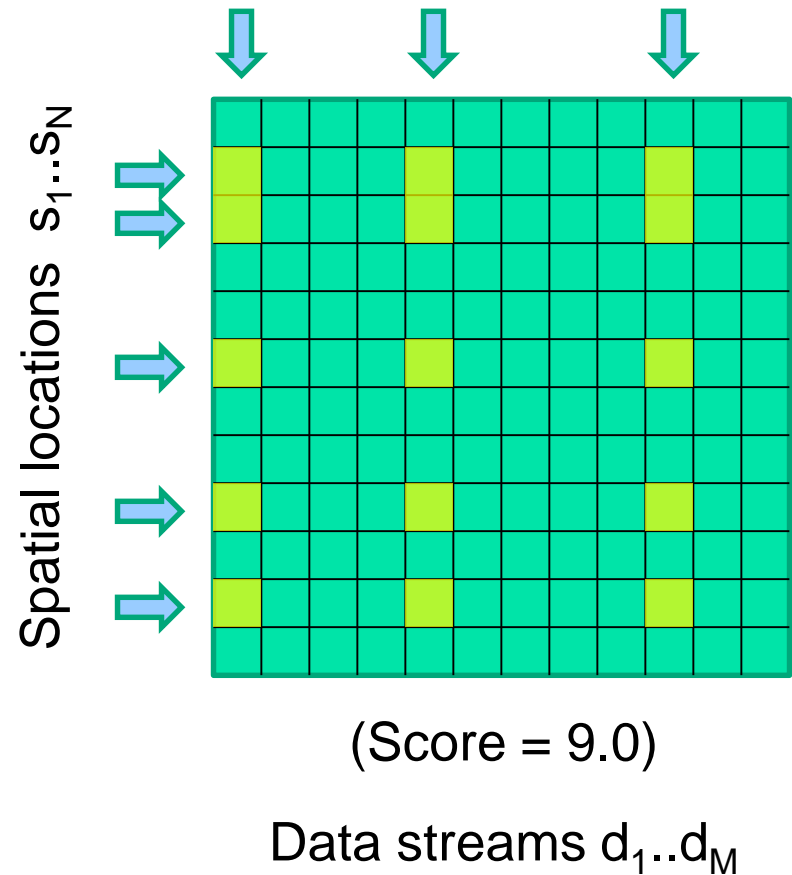
- The LTSS property allows us to efficiently optimize over subsets of spatial locations for a given subset of streams.
- But it also allows us to efficiently optimize over subsets of **streams** for a given subset of **locations**...



# Multivariate fast subset scan

(Neill, McFowland, and Zheng, 2013)

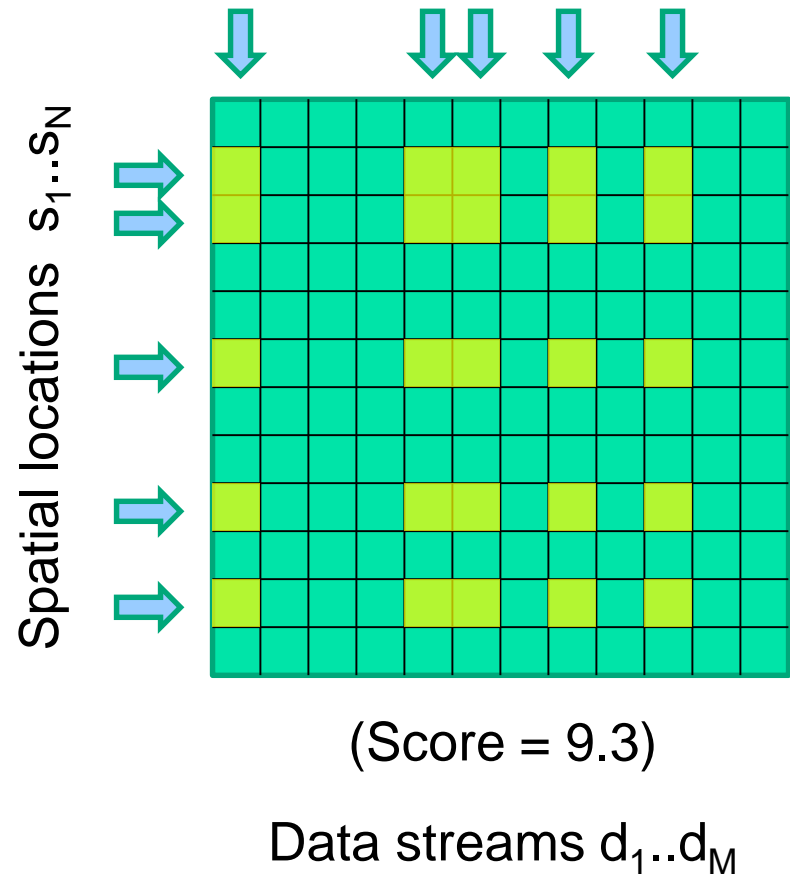
- The LTSS property allows us to efficiently optimize over subsets of spatial locations for a given subset of streams.
- But it also allows us to efficiently optimize over subsets of **streams** for a given subset of **locations**...
- So we can jointly optimize over subsets of streams **and** locations by iterating between these steps!



# Multivariate fast subset scan

(Neill, McFowland, and Zheng, 2013)

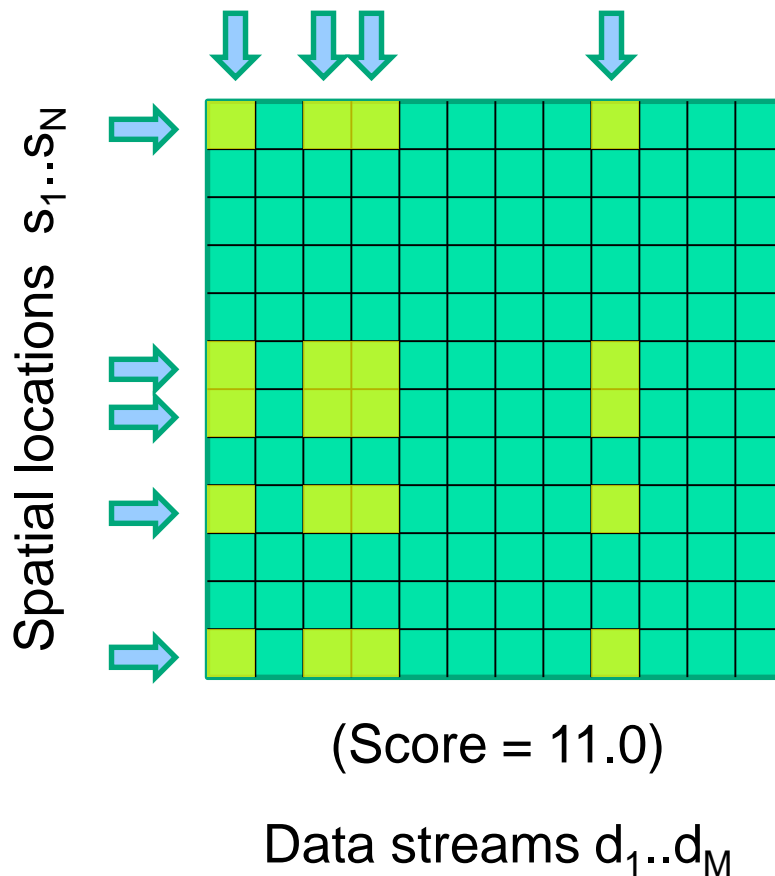
- The LTSS property allows us to efficiently optimize over subsets of spatial locations for a given subset of streams.
- But it also allows us to efficiently optimize over subsets of **streams** for a given subset of **locations**...
- So we can jointly optimize over subsets of streams **and** locations by iterating between these steps!



# Multivariate fast subset scan

(Neill, McFowland, and Zheng, 2013)

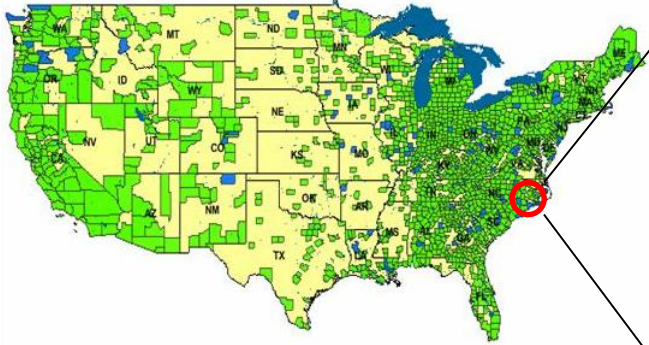
- The LTSS property allows us to efficiently optimize over subsets of spatial locations for a given subset of streams.
- But it also allows us to efficiently optimize over subsets of **streams** for a given subset of **locations**...
- So we can jointly optimize over subsets of streams **and** locations by iterating between these steps!
- Converges to local maximum: we do multiple random restarts to approach the global maximum.
- For general datasets, a similar approach\* can be used to jointly optimize over subsets of data records and attributes.



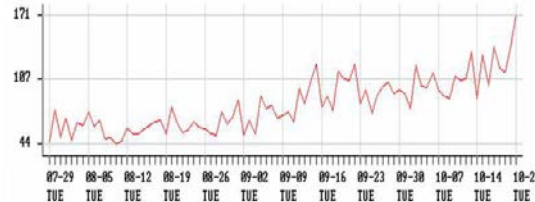
\*McFowland, Speakman, and Neill, *JMLR*, 2013



# Multivariate event detection



Spatial time series data from spatial locations  $s_i$  (e.g. zip codes)



Time series of counts  $c_{i,m}^t$  for each zip code  $s_i$  for each data stream  $d_m$ .

## Outbreak detection

- $d_1$  = respiratory ED
- $d_2$  = constitutional ED
- $d_3$  = OTC cough/cold
- $d_4$  = OTC anti-fever  
(etc.)

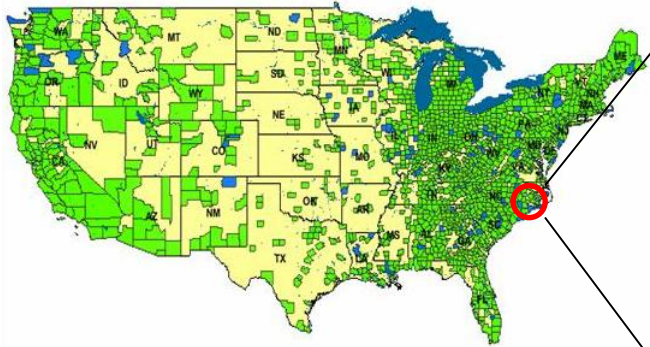
## Main goals:

- Detect** any emerging events.
- Pinpoint** the affected subset of locations and time duration.
- Characterize** the event, e.g., by identifying the affected streams.

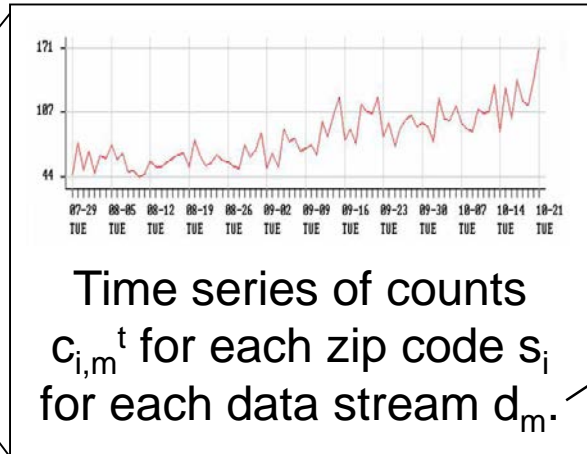
## Compare hypotheses:

- $H_1(D, S, W)$
- $D$  = subset of streams
- $S$  = subset of locations
- $W$  = time duration
- vs.  $H_0$ : no events occurring

# Multidimensional event detection



Spatial time series data from spatial locations  $s_i$  (e.g. zip codes)



## Outbreak detection

- $d_1$  = respiratory ED
- $d_2$  = constitutional ED
- $d_3$  = OTC cough/cold
- $d_4$  = OTC anti-fever (etc.)

Additional goal: identify any differentially affected **subpopulations**  $P$  of the monitored population.

- Gender (male, female, both)
- Age groups (children, adults, elderly)
- Ethnic or socio-economic groups
- Risk behaviors: e.g. intravenous drug use, multiple sexual partners

More generally, assume that we have a set of additional discrete-valued attributes  $A_1..A_J$  observed for each individual case. We identify not only the affected streams, locations, and time window, but also a **subset** of values for each attribute.

# Multidimensional LTSS

- Our **MD-Scan** approach (Neill and Kumar, 2013) extends LTSS to the multidimensional case:
  - For each time window and spatial neighborhood (center + k-nearest neighbors), we do the following:
    1. Start with randomly chosen subsets of **locations**  $S$ , **streams**  $D$ , and **values**  $V_j$  for each attribute  $A_j$  ( $j=1..J$ ).
    2. Choose an attribute (randomly or sequentially) and use LTSS to find the highest scoring subset of values, locations, or streams, conditioned on all other attributes.
    3. Iterate step 2 until convergence to a local maximum of the score function  $F(D, S, W, \{V_j\})$ , and use multiple restarts to approach the global maximum.

# MD-Scan challenges and solutions

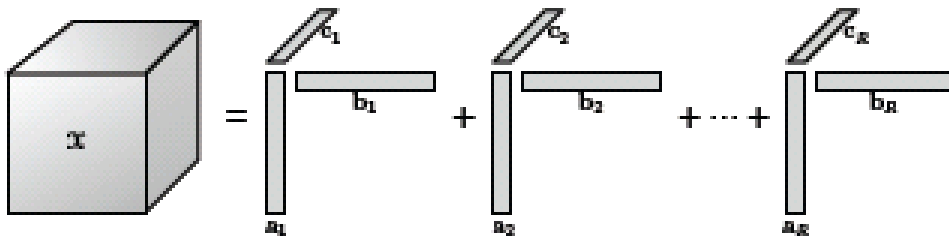
- Original approach: compute separate baselines for each tensor cell (e.g., by 28-day moving average).
  - Statistical challenge: data sparsity leads to increasingly poor baseline estimates.
  - Computational challenge: very large tensor, often with dozens of modes, so need sparse representation.
  - We don't really believe that any baselines are zero!
- Solution: tensor decomposition!
  - 1) How to efficiently decompose?
  - 2) How to efficiently compute baselines?

# Efficient factorization

- PARAFAC decomposition: approximate tensor by sum of outer products,

$$X = \sum_{r=1..R} (a^{(r)} \circ b^{(r)} \circ c^{(r)} \circ \dots)$$

or equivalently,  $x_{ijk\dots} = \sum_{r=1..R} (a_i^{(r)} b_j^{(r)} c_k^{(r)} \dots)$



# vectors =  $R * \#$  modes

Each vector is of length = arity of that mode (or # of values of that attribute).

- Very large, sparse, high-order tensors: we want to run in time proportional to # of non-zero elements and independent of tensor size (product of arities).

# Computing baselines

- Given PARAFAC representation, the aggregate baseline of subset  $S = S_1 \times S_2 \times \dots \times S_M$  is:

$$B = \sum_{r=1..R} \prod_{m=1..M} \sum_{i \in S_m} u_{i,m}^{(r)},$$

where  $u_{i,m}^{(r)}$  is the  $i^{\text{th}}$  value of the  $m^{\text{th}}$ -mode vector of the  $r^{\text{th}}$  PARAFAC component.

- Example of why this works, for three modes:

$$\begin{aligned} B &= \sum_{i \in S_1} \sum_{j \in S_2} \sum_{k \in S_3} b_{ijk} \\ &= \sum_{i \in S_1} \sum_{j \in S_2} \sum_{k \in S_3} \sum_{r=1..R} u_i^{(r)} v_j^{(r)} w_k^{(r)} \\ &= \sum_{r=1..R} \left( \sum_{i \in S_1} u_i^{(r)} \right) \left( \sum_{j \in S_2} v_j^{(r)} \right) \left( \sum_{k \in S_3} w_k^{(r)} \right) \end{aligned}$$

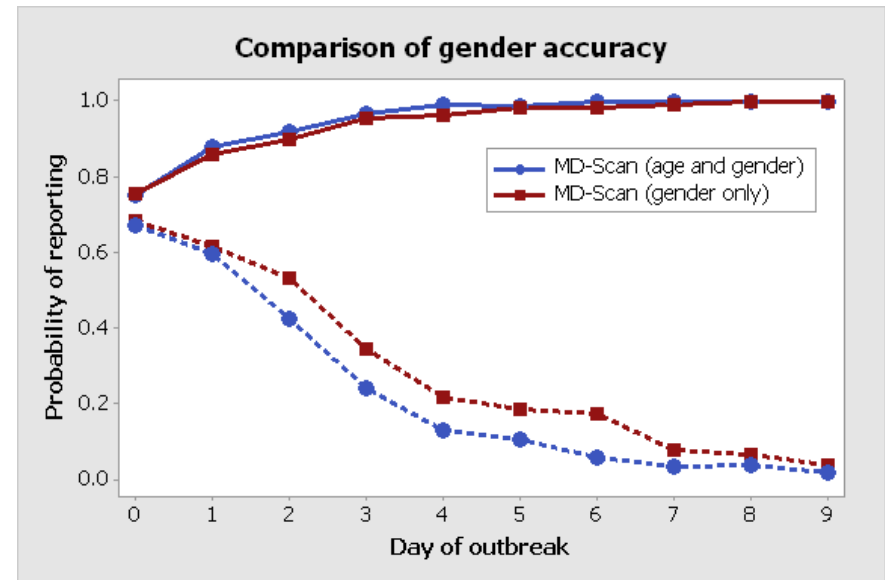
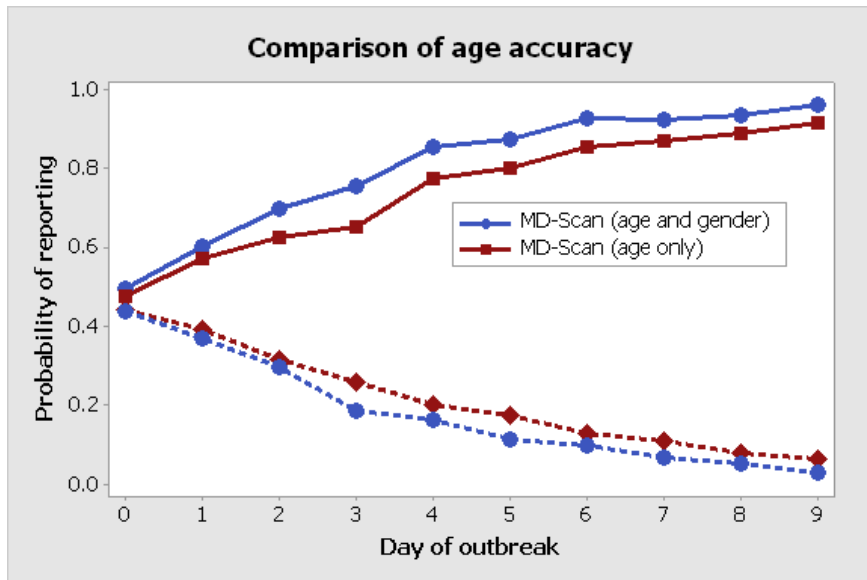
- By writing the sum of products as a product of sums, we can compute in time proportional to  $|S_1| + |S_2| + \dots + |S_M|$  rather than  $|S_1| \times |S_2| \times \dots \times |S_M|$ .

# Empirical evaluation

- We evaluated the detection performance of MD-Scan for detecting disease outbreaks injected into real-world Emergency Department data from Allegheny County, PA.
- We considered outbreaks with various types and amounts of age and gender bias.
- Shown here: preliminary eval with comparisons to multivariate linear-time subset scan.
- Additional comparisons, and application to detecting patterns of near-repeat crimes in data from the Cambridge PD, are in progress.

# 1) Identifying affected subpopulations

By the midpoint of the outbreak, MD-Scan is able to correctly identify the affected gender and age deciles with high probability, without reporting unaffected subpopulations.



**Proportions of correct and incorrect groups reported vs. time since start of outbreak.**

Solid lines: affected gender and/or age deciles. Dashed lines: unaffected.

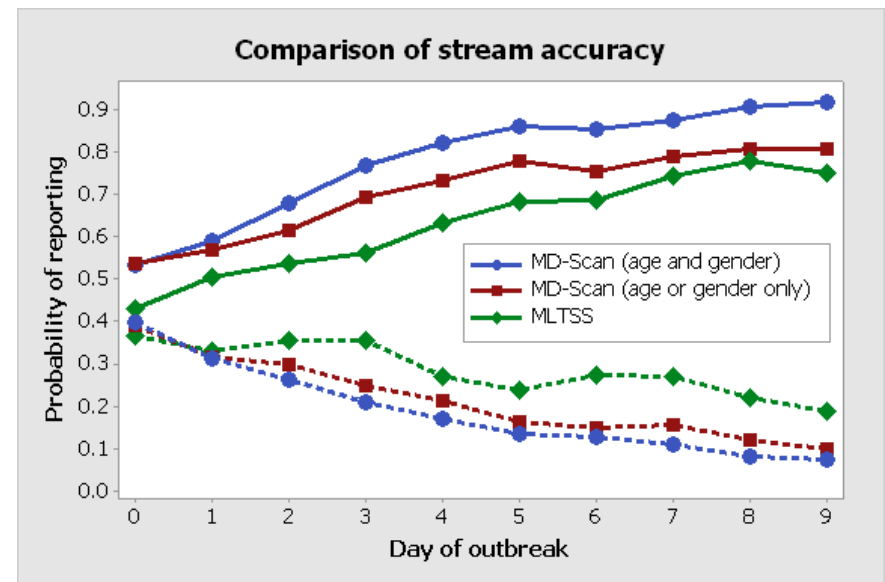
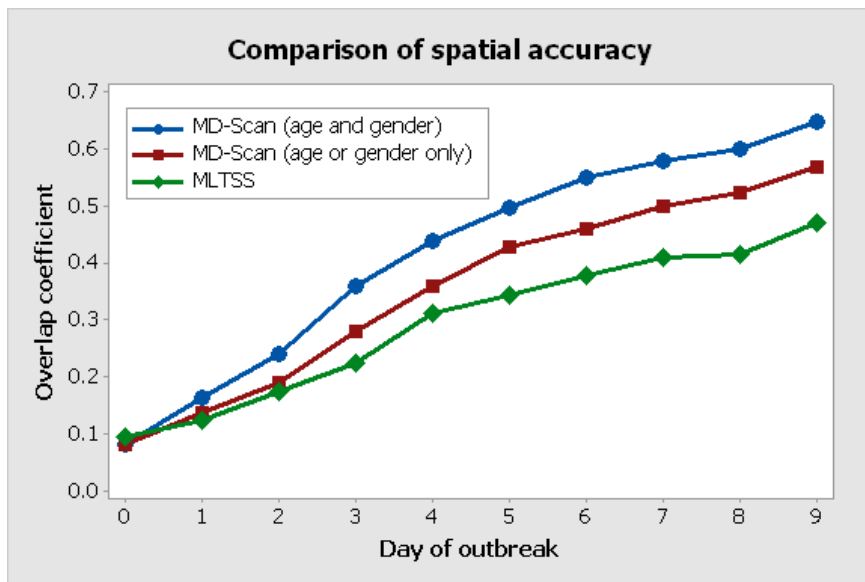
Blue lines: outbreaks with differential effects by both age and gender (easier).

Red lines: outbreaks with differential effects by age or gender only (harder).



## 2) Characterizing affected streams

As compared to the previous state of the art (multivariate linear-time subset scanning), MD-Scan is better able to characterize the affected spatial locations and subset of the monitored streams.



**Left: overlap coefficient between true and detected subsets of spatial locations.**  
**Right: Proportions of correct and incorrect streams reported vs. day of outbreak.**

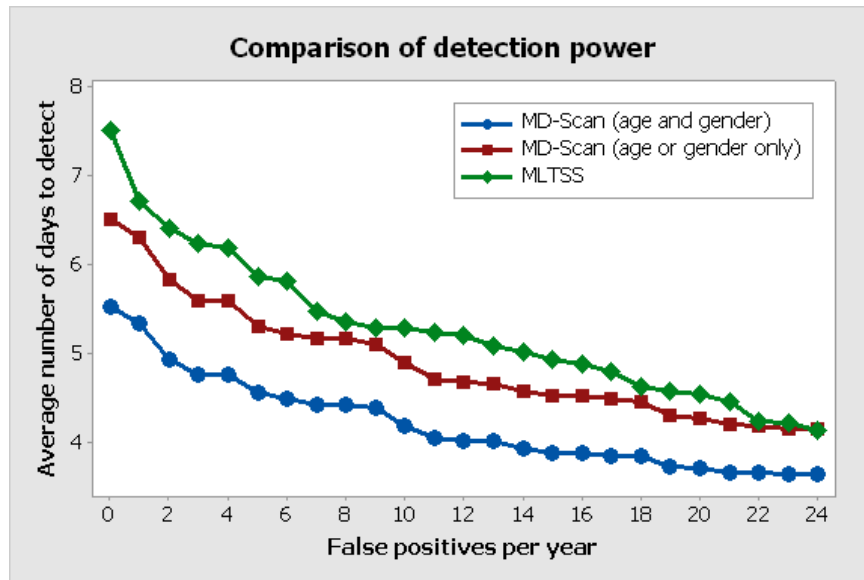
Blue lines: outbreaks with differential effects by both age and gender (easier).

Red lines: outbreaks with differential effects by age or gender only (harder).

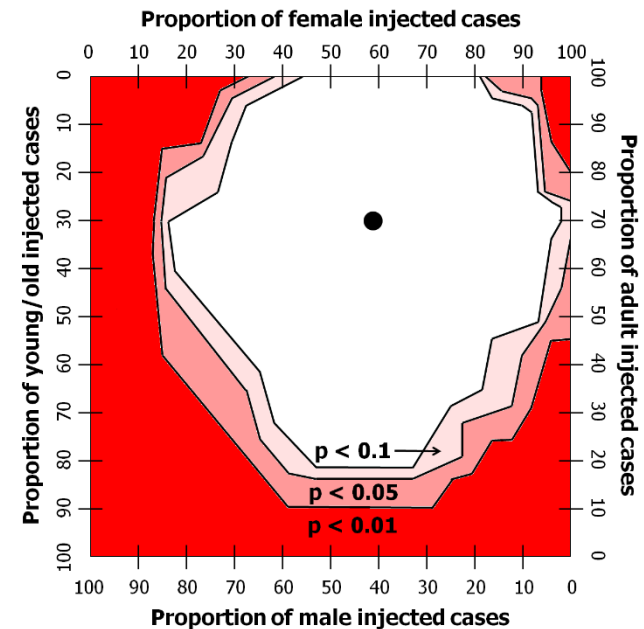
Green lines: MLTSS, ignoring age and gender information

# 3) Timeliness of outbreak detection

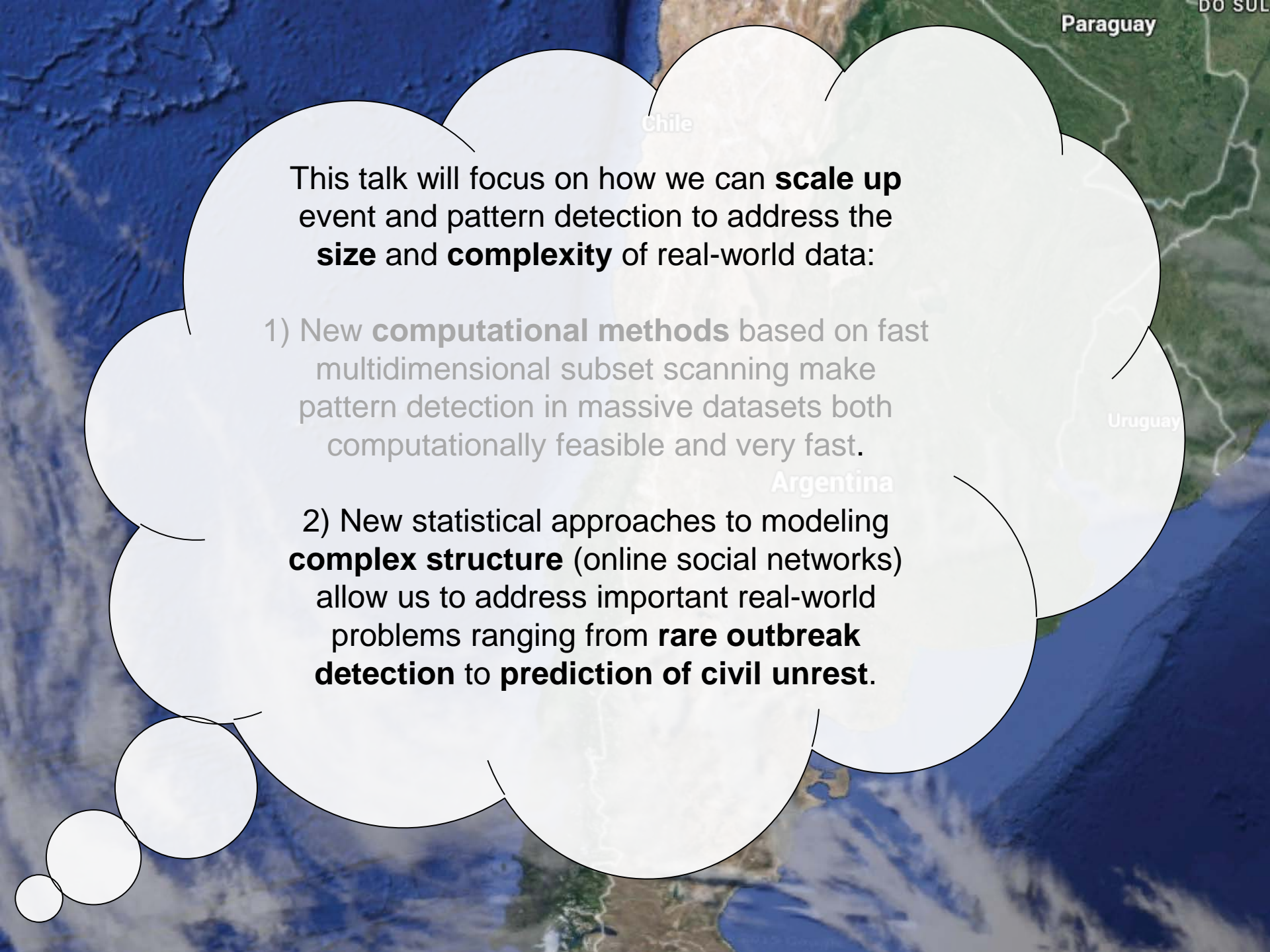
MD-Scan achieved significantly more timely detection for outbreaks that were sufficiently biased by age and/or gender.



For outbreaks with strong age and gender biases, time to detection improved from 5.2 to 4.0 days at a fixed false positive rate of 1/month.



Smaller biases in age or gender were sufficient for significant improvements; even when no age/gender signal is present, MD-Scan performs comparably to MLTSS.



This talk will focus on how we can **scale up** event and pattern detection to address the **size** and **complexity** of real-world data:

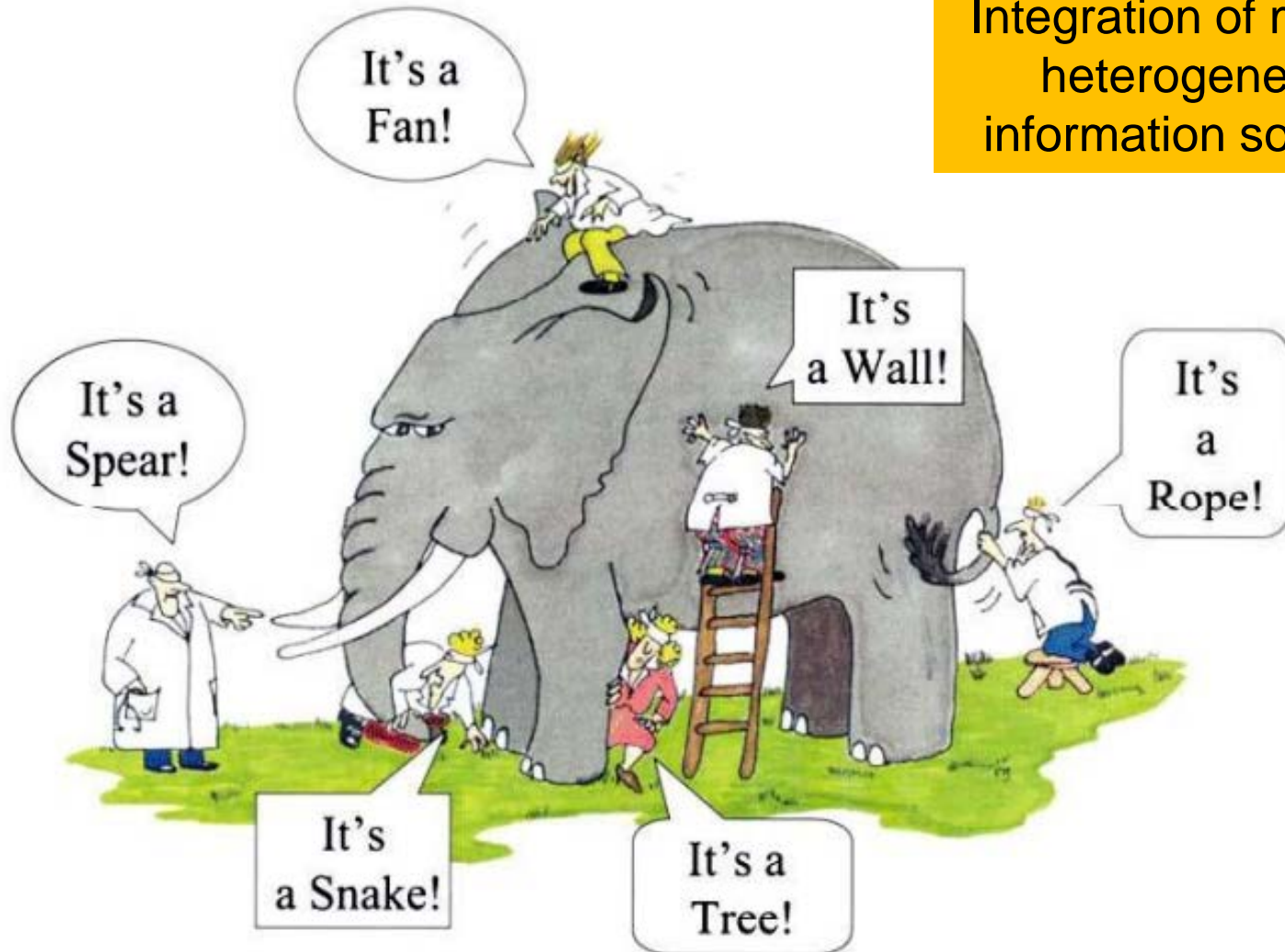
1) New **computational methods** based on fast multidimensional subset scanning make pattern detection in massive datasets both computationally feasible and very fast.

2) New statistical approaches to modeling **complex structure** (online social networks) allow us to address important real-world problems ranging from **rare outbreak detection** to **prediction of civil unrest**.



# Technical Challenges

Integration of multiple heterogeneous information sources!



# Technical Challenges

One week before Mexico's 2012 presidential election:

Hashtag "#Megamarch"  
mentioned 1,000 times



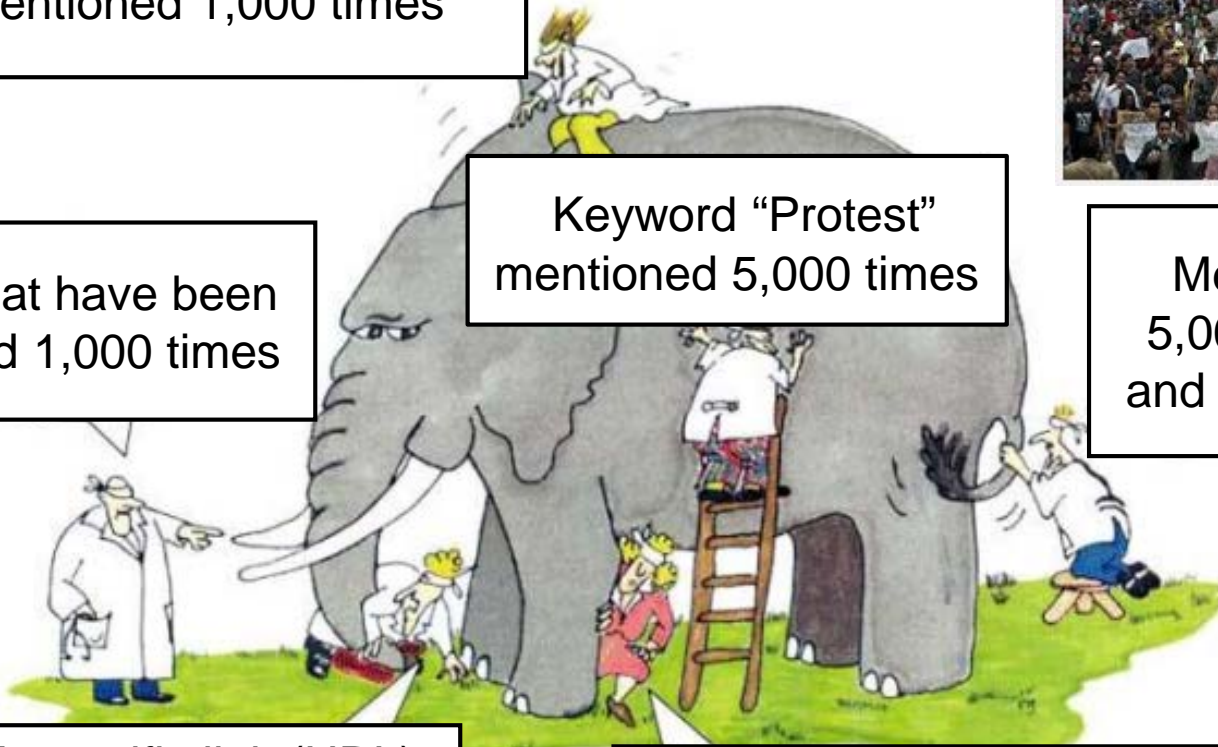
Tweets that have been  
re-tweeted 1,000 times

Keyword "Protest"  
mentioned 5,000 times

Mexico City has  
5,000 active users  
and 100,000 tweets

A specific link (URL)  
was mentioned  
866 times

Influential user "Zeka"  
posted 10 tweets



# Technical Challenges

One week before Mexico's 2012 presidential election:

Hashtag "#Megamarch"  
mentioned 1,000 times



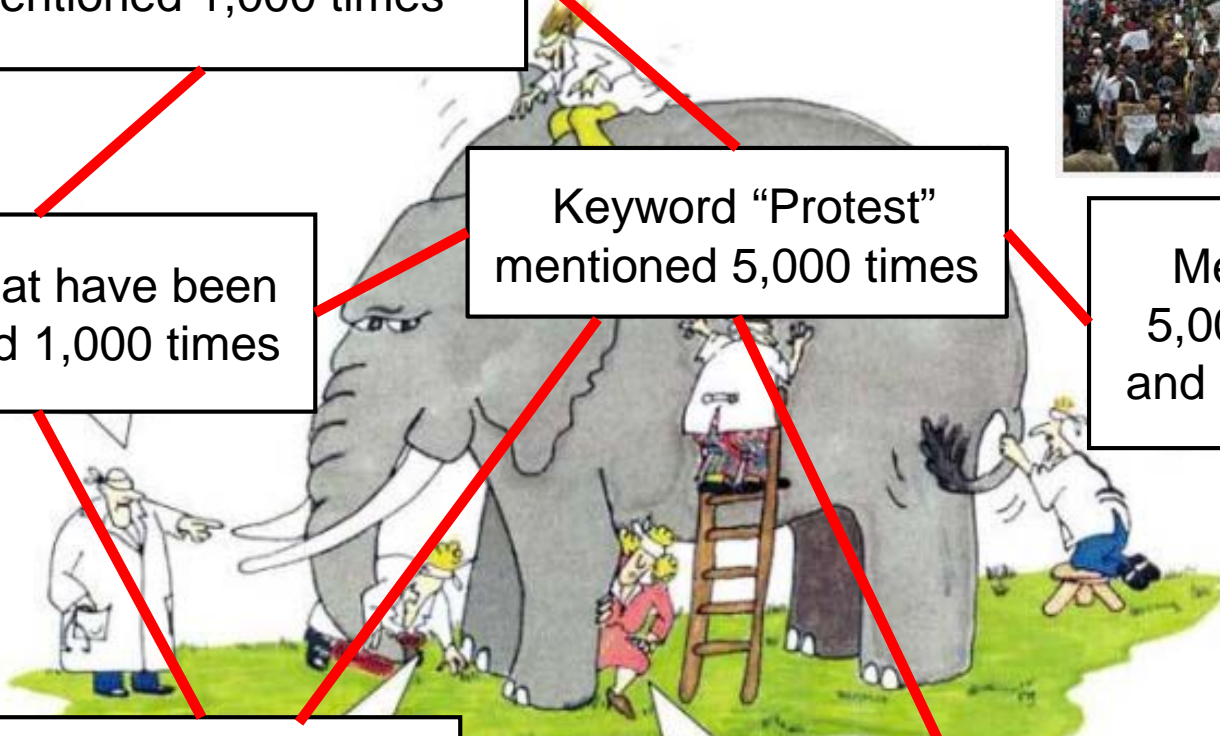
Tweets that have been  
re-tweeted 1,000 times

Keyword "Protest"  
mentioned 5,000 times

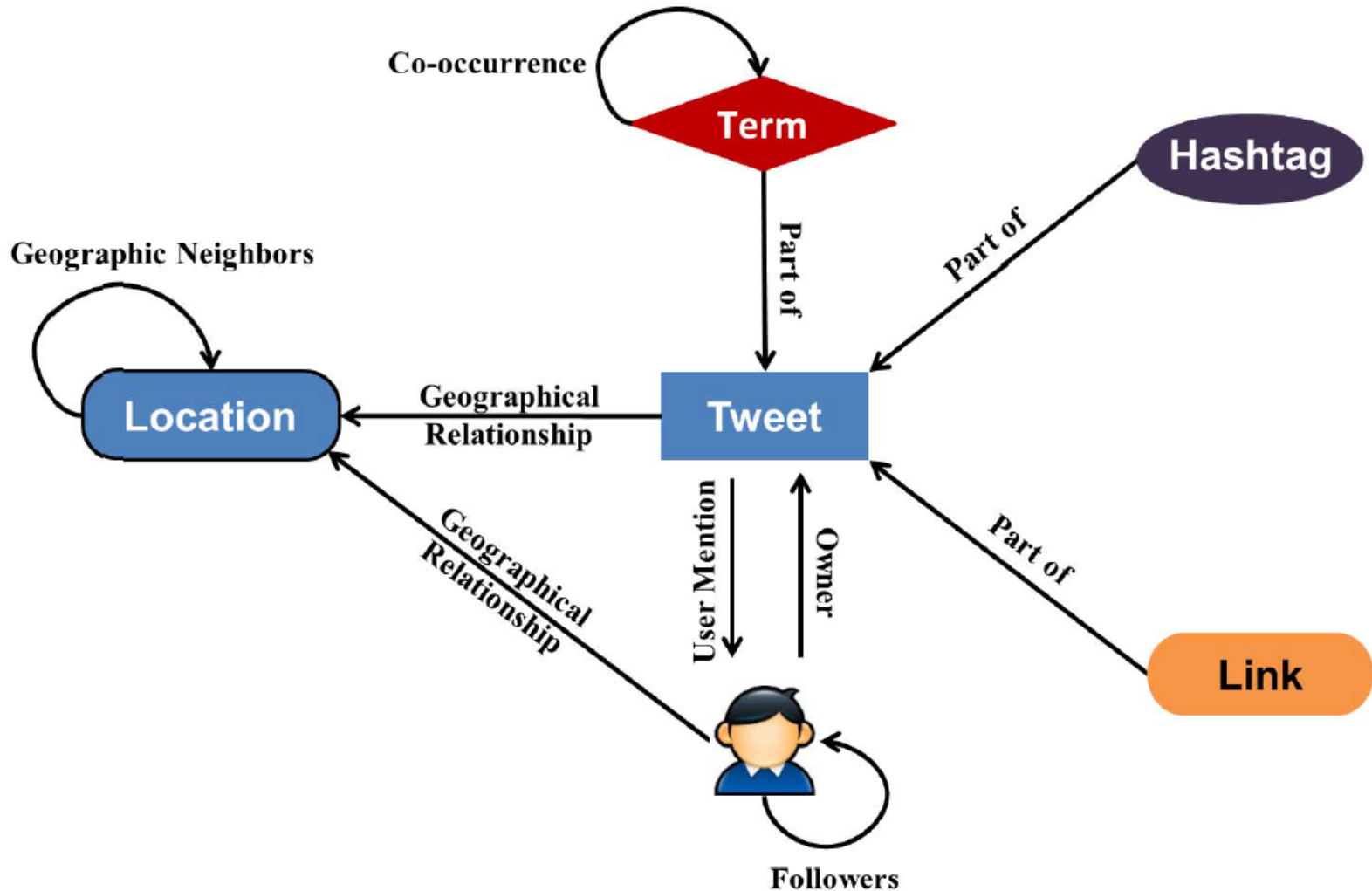
Mexico City has  
5,000 active users  
and 100,000 tweets

A specific link (URL)  
was mentioned  
866 times

Influential user "Zeka"  
posted 10 tweets

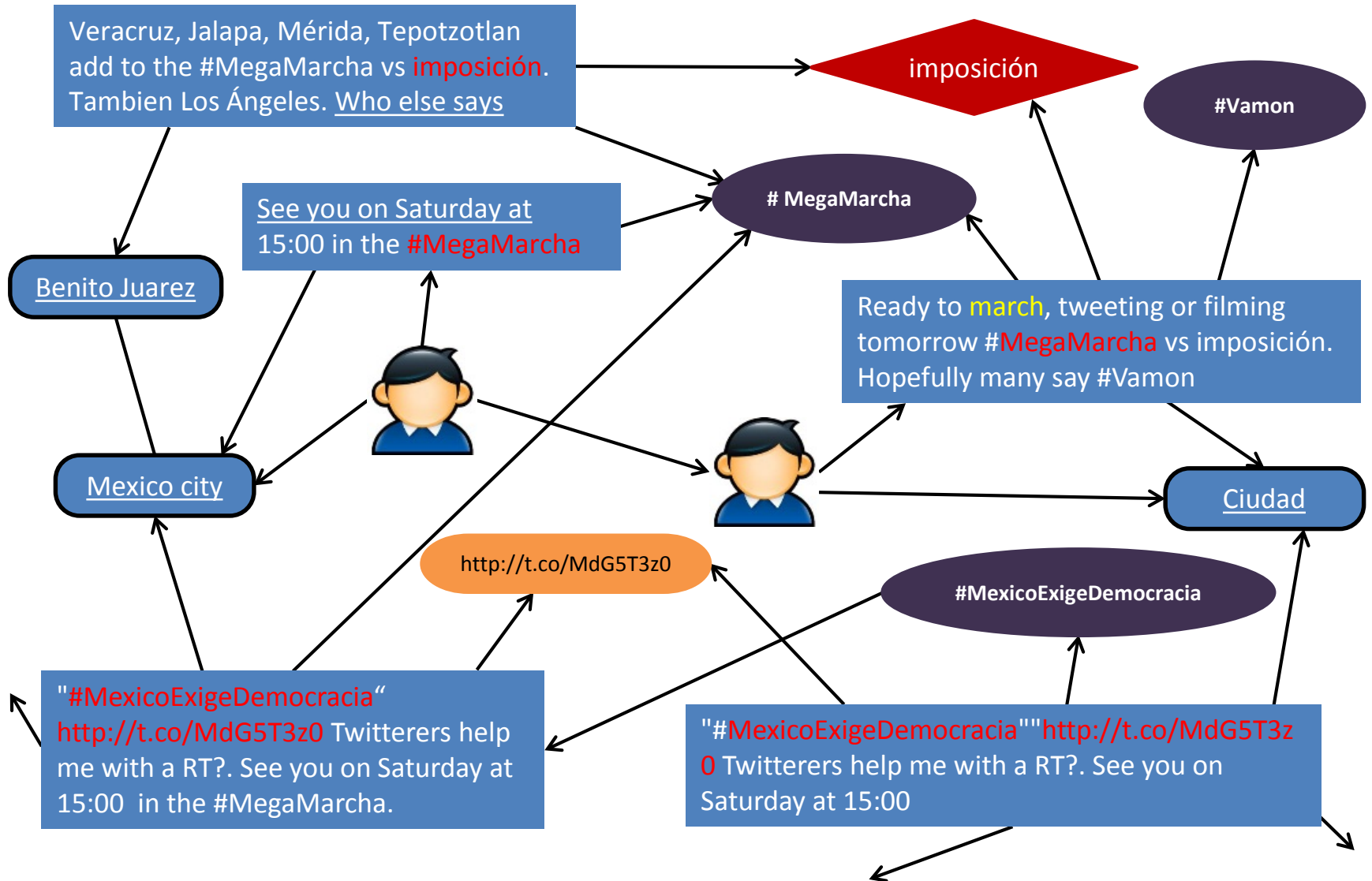


# Twitter Heterogeneous Network

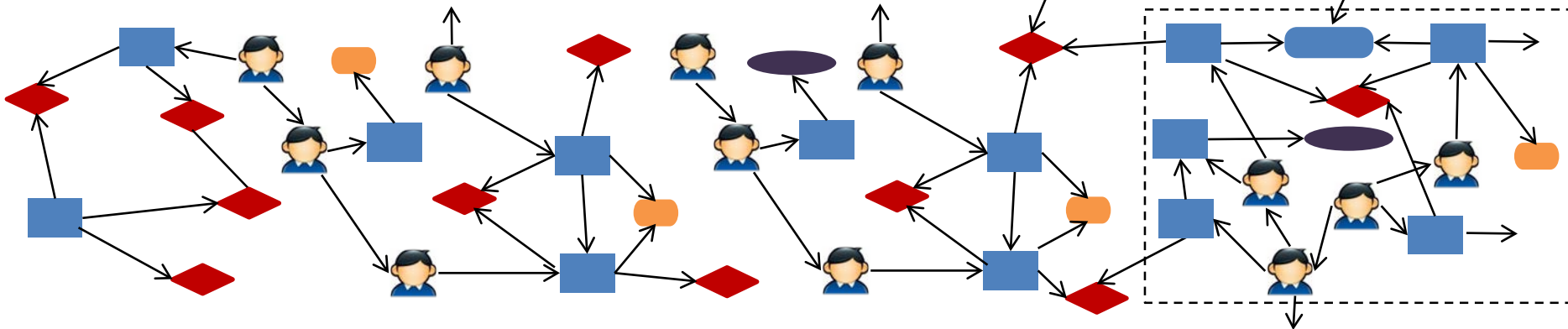
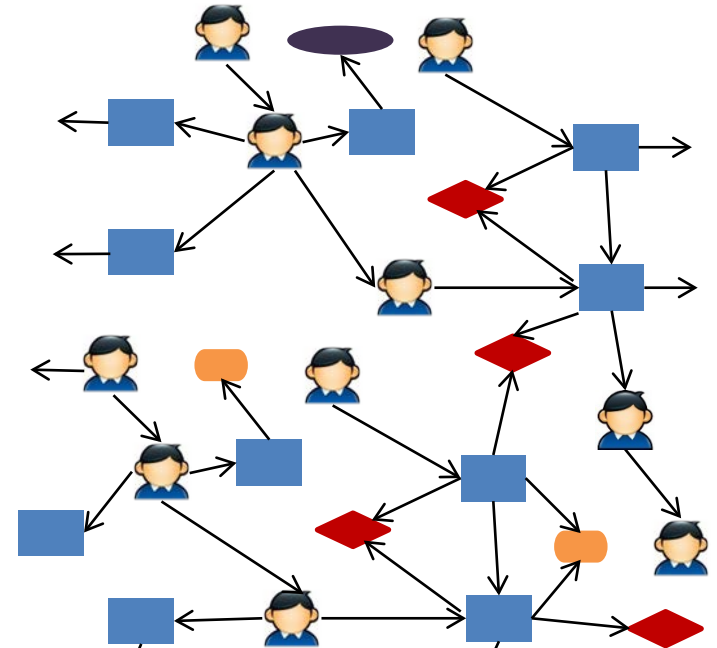
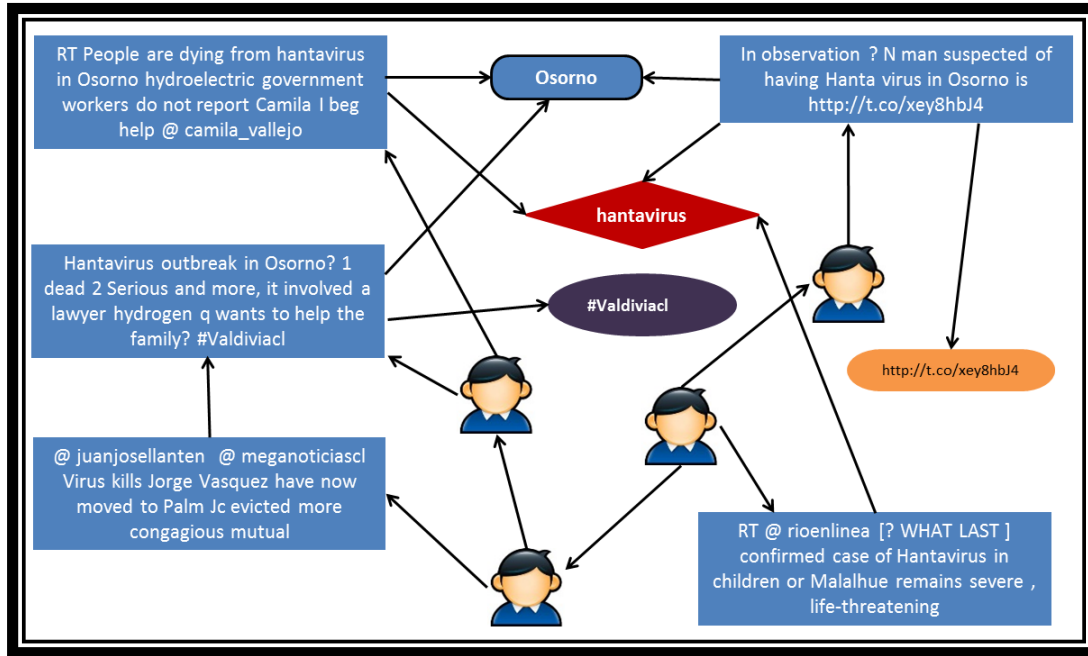




# Twitter Heterogeneous Network



# Twitter Heterogeneous Network



# Nonparametric Heterogeneous Graph Scan

(Chen and Neill, KDD 2014)

1) We model the heterogeneous social network as a **sensor network**.

Each node senses its local neighborhood, computes multiple features, and reports the overall degree of anomalousness.

2) We compute an **empirical p-value** for each node:

- Uniform on  $[0,1]$  under the null hypothesis of no events.
- We search for subgraphs of the network with a higher than expected number of low (significant) empirical p-values.

3) We can scale up to very large heterogeneous networks:

- Heuristic approach: **iterative subgraph expansion** (“greedy growth” to subset of neighbors on each iteration).
- LTSS can efficiently find the best subset of neighbors, ensuring that the subset remains connected, at each step.

# Sensor network modeling

Each node reports an empirical p-value measuring the current level of anomalousness for each time interval (hour or day).

Object Type	Features
User	# tweets, # retweets, # followers, #followees, #mentioned_by, #replied_by, diffusion graph depth, diffusion graph size
Tweet	Klout, sentiment, replied_by_graph_size, reply_graph_size, retweet_graph_size, retweet_graph_depth
City, State, Country	# tweets, # active users
Term	# tweets
Link	# tweets
Hashtag	# tweets

Features

empirical  
calibration

Individual p-value  
for each feature

min

Minimum  
empirical p-  
value for  
each node

empirical  
calibration

Overall p-value  
for each node

# Nonparametric scan statistics

Number of nodes in  $S$  with p-values  $\leq \alpha$ .

Subgraph

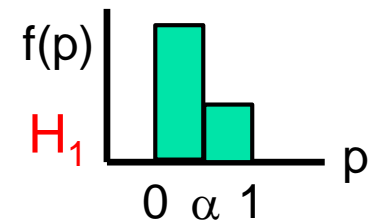
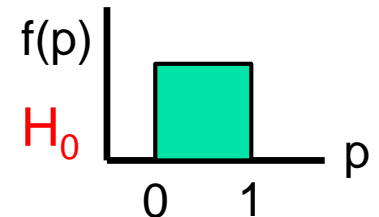
$$F(S) = \max_{\alpha \leq \alpha_{max}} F_{\alpha}(S) = \max_{\alpha \leq \alpha_{max}} \phi(\alpha, N_{\alpha}(S), N(S))$$

Significance level

Number of nodes in  $S$

Berk-Jones (BJ) statistic:

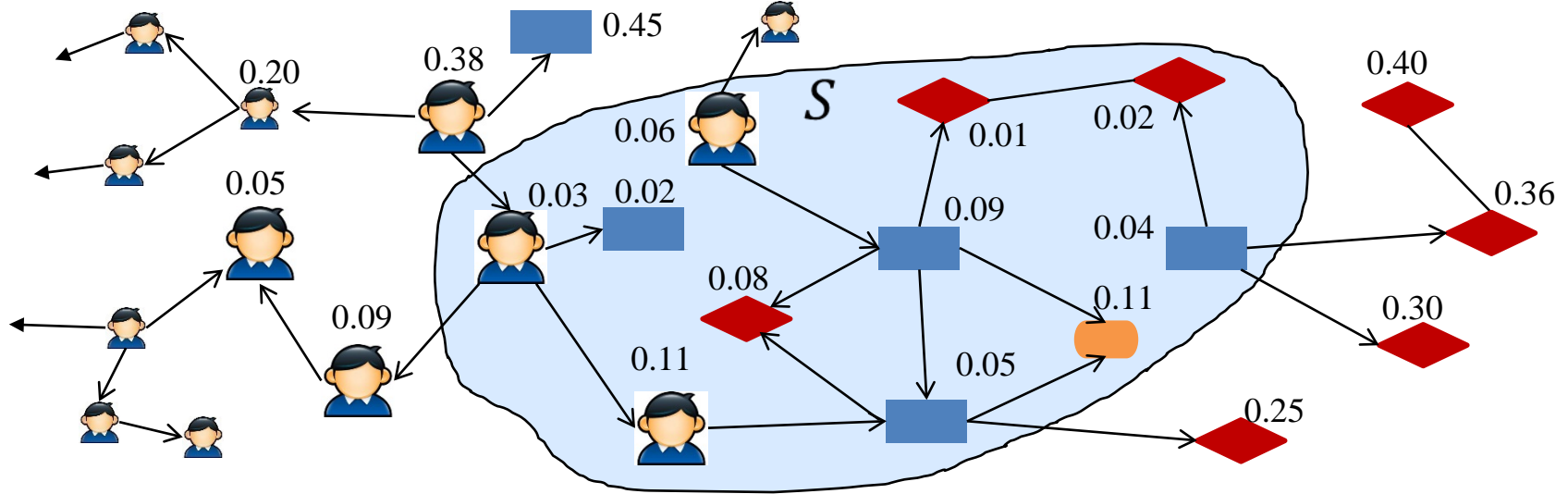
$$\phi_{BJ}(\alpha, N_{\alpha}(S), N(S)) = N(S)K\left(\frac{N_{\alpha}}{N}, \alpha\right)$$



Kullback-Liebler divergence:

$$K(x, y) = x \log\left(\frac{x}{y}\right) + (1 - x) \log\left(\frac{1 - x}{1 - y}\right)$$

# Nonparametric graph scanning



$$S^* = \operatorname{argmax}_{S \in V: S \text{ is connected}} F(S)$$

We propose an approximate algorithm with time cost  $O(|V| \log |V|)$ .

# NPHGS evaluation- civil unrest

Country	# of tweets	News source*
Argentina	29,000,000	Clarín; La Nación; Infobae
Chile	14,000,000	La Tercera; Las Últimas Noticias; El Mercurio
Colombia	22,000,000	El Espectador; El Tiempo; El Colombiano
Ecuador	6,900,000	El Universo; El Comercio; Hoy

**Gold standard dataset:** 918 civil unrest events between July and December 2012.

Example of a gold standard event label:

PROVINCE = “El Loa”

COUNTRY = “Chile”

DATE = “2012-05-18”

LINK = “<http://www.pressenza.com/2012/05/...>”

DESCRIPTION = “A large-scale march was staged by inhabitants of the northern city of Calama, considered the mining capital of Chile, who demanded the allocation of more resources to copper mining cities”

We compared the detection performance of our NPHGS approach to homogeneous graph scan methods and to a variety of state-of-the-art methods previously proposed for Twitter event detection.

# NPHGS results- civil unrest

Method	FPR (FP/Day)	TPR (Forecasting)	TPR (Forecasting & Detection)	Lead Time (Days)	Lag Time (Days)	Run Time (Hours)
ST Burst Detection	0.65	0.07	0.42	1.10	4.57	30.1
Graph Partition	0.29	0.03	0.15	0.59	6.13	18.9
Earthquake	0.04	0.06	0.17	0.49	5.95	18.9
RW Event	0.10	0.22	0.25	0.93	5.83	16.3
Geo Topic Modeling	0.09	0.06	0.08	0.01	6.94	9.7
NPHGS (FPR=.05)	0.05	0.15	0.23	0.65	5.65	38.4
NPHGS (FPR=.10)	0.10	0.31	0.38	1.94	4.49	38.4
NPHGS (FPR= .15)	0.15	0.37	0.42	2.28	4.17	38.4
NPHGS (FPR=.20)	0.20	0.39	0.46	2.36	3.98	38.4

Table 3: Comparison between NPHGS and Existing Methods on the civil unrest datasets

NPHGS outperforms existing representative techniques for both event detection and forecasting, increasing **detection power**, **forecasting accuracy**, and **forecasting lead time** while reducing **time to detection**.

Similar improvements in performance were observed on a second task:

Early detection of rare disease outbreaks, using gold standard data about 17 hantavirus outbreaks from the Chilean Ministry of Health.





# Conclusions

Real-world problems at the societal scale require new computational methods to deal with both the **size** and the **complexity** of data.



**Fast subset scanning** (with constraints) can serve as a fundamental building block for efficient, scalable pattern detection in massive data.

Practical solutions to societal challenges also require an understanding of complex data (text, networks, images, streams, ...), leading to **new statistical and algorithmic tools** for extracting relevant patterns.

# Future work

Three broad areas, one **application-driven** and two **methods-driven**:

1. Addressing **critical real-world problems** in collaboration with public sector organizations (public health, police, city leaders, ...)
2. Expanding the **scope** of problems that detection can address.
3. Expanding the **scale** of problems that detection can address.

# Future work

Three broad areas, one **application-driven** and two **methods-driven**:

1. Addressing **critical real-world problems** in collaboration with **public sector organizations** (public health, police, city leaders, ...)
2. Expanding the **scope** of problems that detection can address.
3. Expanding the **scale** of problems that detection can address.



**Safer  
Cities**



**Cleaner  
Cities**



**Healthier  
Cities**



RK Mellon Foundation funded project on crime prediction and prevention:

Integrating geographic, subgroup, and individual-level crime prediction.

Incorporating many data sources: 911 and 311 calls, incident reports, criminal justice, human services...

Analyzing social media to identify causal mechanisms leading to outbreaks of violence.

Integrating precisely targeted policing with non-punitive interventions by city and county.

# Future work

Three broad areas, one **application-driven** and two **methods-driven**:

1. Addressing **critical real-world problems** in collaboration with **public sector organizations** (public health, police, city leaders, ...)
2. Expanding the **scope** of problems that detection can address.
3. Expanding the **scale** of problems that detection can address.



**Safer  
Cities**



**Cleaner  
Cities**



**Healthier  
Cities**



Computational public health and epidemiology (NSF Expeditions, CDC, ...)

Asyndromic and pre-syndromic surveillance

Combining detection and simulation approaches

Sensing and monitoring individual health:

“Your cell phone should know whether you’re sick”

Sensing and monitoring population health:

“Distributed, privacy-preserving outbreak detection”

# Future work

Three broad areas, one **application-driven** and two **methods-driven**:

1. Addressing **critical real-world problems** in collaboration with public sector organizations (public health, police, city leaders, ...)
2. Expanding the **scope** of problems that detection can address.
3. Expanding the **scale** of problems that detection can address.

DETECT4: Using **detection** as a building block **for** other problems

Detection for prediction (key component of CrimeScan and CityScan)

Causal inference (estimation of heterogeneous treatment effects)

Classifier model validation and refinement (“boosting systematic errors”)

Active learning for subsets (noisy oracle, crowdsourcing/citizen science)

Graph structure learning from unlabeled data

Incorporating more complex data types and more flexible constraints

Tensors, text, massive images (e.g., satellite data), social media, ...

Irregularly shaped spatial regions (StarScan, Support Vector Subset Scan)

Soft constraints (from element-wise to pairwise and subset-based penalties)

# Future work

Three broad areas, one **application-driven** and two **methods-driven**:

1. Addressing **critical real-world problems** in collaboration with public sector organizations (public health, police, city leaders, ...)
2. Expanding the **scope** of problems that detection can address.
3. Expanding the **scale** of problems that detection can address, via parallelization, sampling, hierarchy, and real-world graph structure.

# Scaling up to even **bigger** data...

Currently the fast subset scan scales to datasets with **millions** of records.

But enforcing certain hard constraints (e.g., graph connectivity) dramatically impacts scalability.

Spatial constraints (FSS)  
Similarity constraints (FGSS)  
Soft constraints (PFSS)

GraphScan: 250 nodes  
Additive Graphscan : 25K nodes

How to scale up to larger graphs with millions of nodes?

← ongoing EPD Lab research →

How to scale up to datasets with billions or trillions of records?

Many possible answers!

Sampling  
Parallelization

Problem Partitioning  
Randomization

Locality-Sensitive Hashing

Sublinear-Time Algorithms  
Summarization  
Hierarchy



# Idea #1: Massive parallelization

For example, what if we have a trillion records but a million processors?

Certain aspects of fast subset scan are **trivially parallelizable**:

- Randomization testing, to determine statistical significance.
- Scanning over many local neighborhoods (with proximity constraints).
- Scoring many subsets (but not exponentially many!).

For **unconstrained subset scan**, we have the necessary pieces:

- Parallel sorting (merge sort, sample sort):  $O(\log N)$  with  $N$  processors.
- “Scan” (accumulate sums of top- $k$  elements by priority):  $O(\log N)$ .

To incorporate **spatial proximity** or more general **similarity** constraints:

- **Locality-sensitive hashing** → neighborhoods of similar elements.

With more general constraints (e.g., graphs), we must develop new ways to partition the search space and merge solutions to sub-problems.

# Idea #2: Incorporate hierarchy

**Subsampling** the raw data can miss a arbitrarily strong signal that affects a small enough proportion of the dataset.

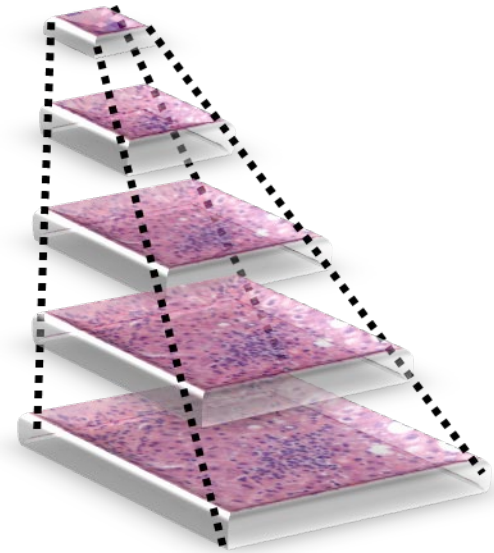
Possible solution: **summarization**.

Represent the data **hierarchically**, maintain summary statistics at each level of hierarchy, and search over coarse and fine resolutions.

Goal: find the most interesting subsets while only looking at a small fraction of the raw data.

Challenge 1: building the hierarchy may be expensive (though parallelizable).

Challenge 2: how to search the hierarchy, so that we are unlikely to miss small areas?

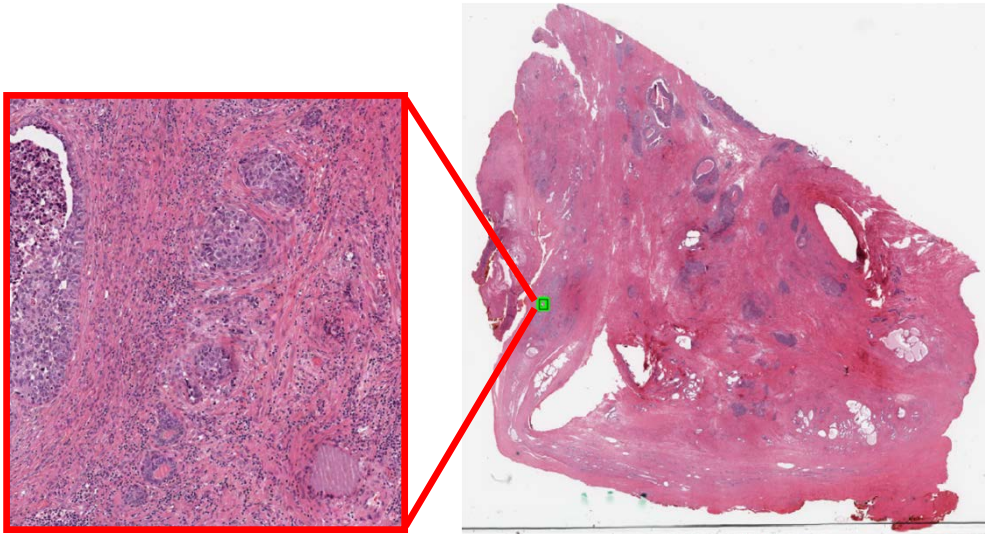


Example: image data  
digital pathology slides,  
satellite images, etc.

Hierarchical Linear-  
Time Subset Scanning

(Somanchi & Neill, DMHI 2013)

# Idea #2: Incorporate hierarchy



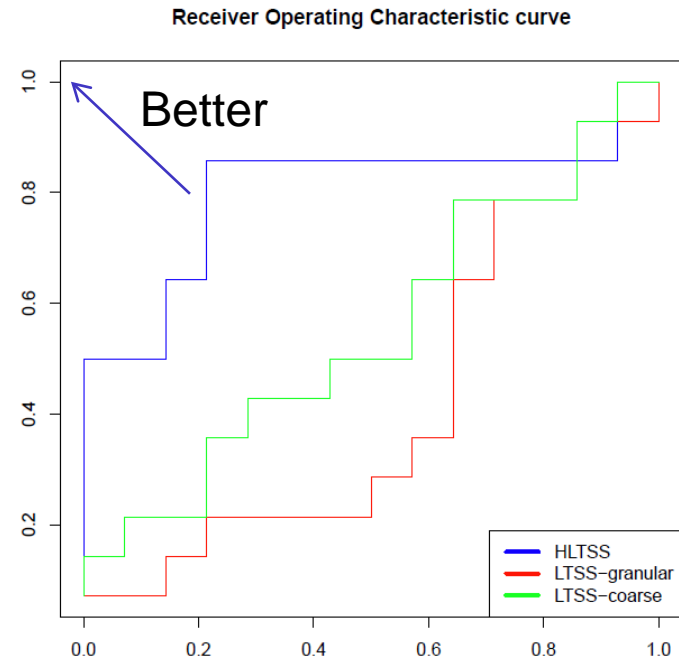
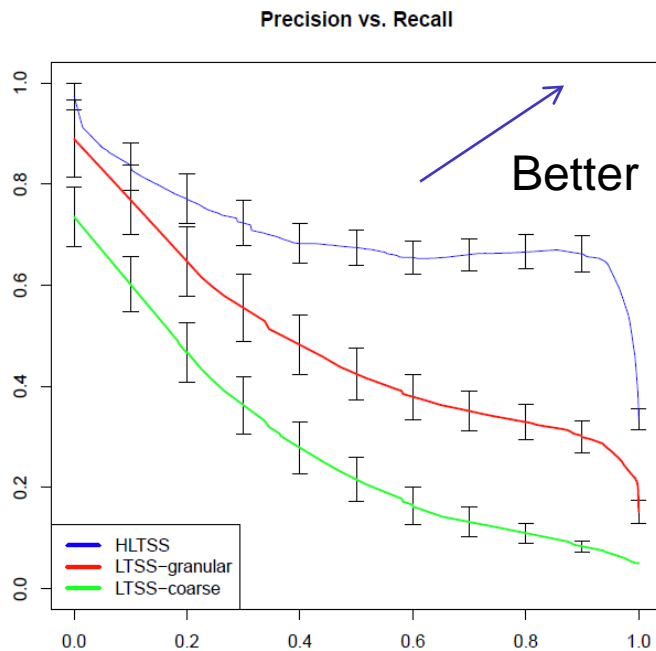
HLTSS has been successfully applied to detect regions of interest in digital pathology slides, and works surprisingly well to detect prostate cancer!

Example: image data  
digital pathology slides,  
satellite images, etc.

Hierarchical Linear-  
Time Subset Scanning

(Somanchi & Neill, DMHI 2013)

# Idea #2: Incorporate hierarchy



HLTSS improves both the accuracy of detecting which pixels within a slide are cancerous (left panel) and the ability to differentiate cancerous from non-cancerous slides (right panel).

# Acknowledgements

- Event and Pattern Detection Laboratory (current members and alumni):  
<http://epdlab.heinz.cmu.edu/people>
- Students, postdocs, and collaborators:  
Feng Chen, Skyler Speakman, Ed McFowland, Sriram Somanchi, Tarun Kumar, Kenton Murray, Yandong Liu, Chris Dyer, Jay Aronson.
- Funding support: NSF and MacArthur Foundation.

# References

- D.B. Neill. Fast subset scan for spatial pattern detection. *Journal of the Royal Statistical Society (Series B: Statistical Methodology)* 74(2): 337-360, 2012.
- D.B. Neill, E. McFowland III, and H. Zheng. Fast subset scan for multivariate event detection. *Statistics in Medicine* 32: 2185-2208, 2013.
- E. McFowland III, S. Speakman, and D.B. Neill. Fast generalized subset scan for anomalous pattern detection. *Journal of Machine Learning Research*, 14: 1533-1561, 2013.
- F. Chen and D.B. Neill. Non-parametric scan statistics for event detection and forecasting in heterogeneous social media graphs. *Proc. 20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2014.
- F. Chen and D.B. Neill. Human rights event detection from heterogeneous social media graphs. *Big Data* 3(1): 34-40, 2015.
- S. Speakman, S. Somanchi, E. McFowland III, D.B. Neill. Penalized fast subset scanning. *Journal of Computational and Graphical Statistics*, 2016, in press.
- T. Kumar and D.B. Neill. Fast tensor scan for event detection and characterization. Revised version in preparation.
- S. Flaxman, D.B. Neill, et al., Fast Kronecker inference in Gaussian processes with non-Gaussian likelihoods. *Proc. Intl. Conf. on Machine Learning*, 2015.



Thanks for listening!

More details on our web site:

<http://epdlab.heinz.cmu.edu>

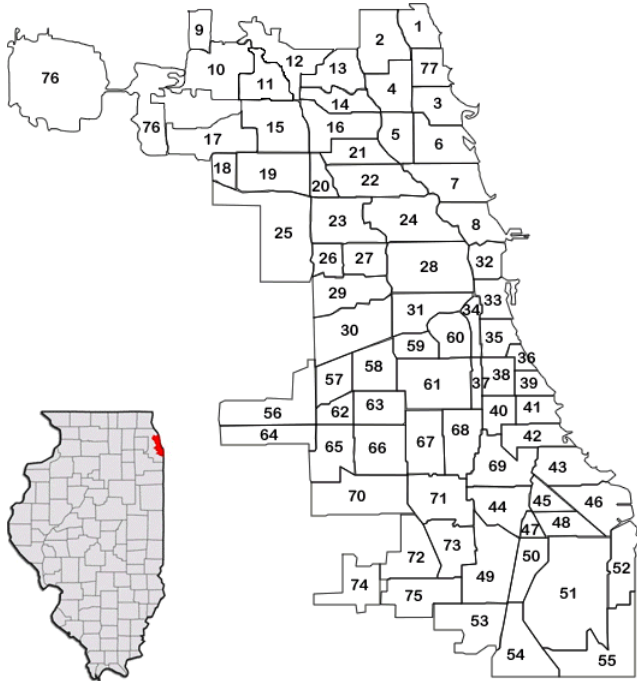
Or e-mail me at:

[neill@cs.cmu.edu](mailto:neill@cs.cmu.edu)

**Extra slides:  
Crime Prediction  
and Urban Analytics**



# Case study: Crime prediction in Chicago



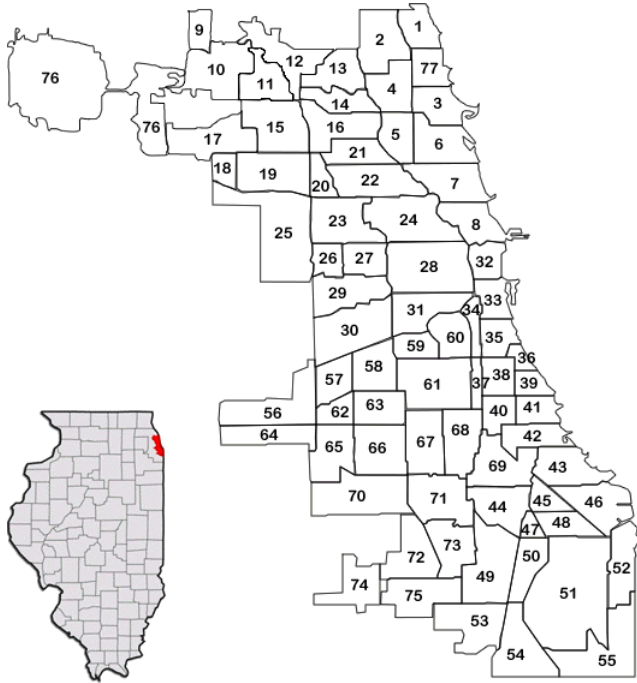
Since 2009, we have been working with the Chicago Police Department (CPD) to predict and prevent emerging clusters of violent crime.

Our new crime prediction methods have been incorporated into our **CrimeScan** software, run twice a day by CPD and used operationally for deployment of patrols.

From the Chicago Sun-Times, February 22, 2011:

“It was a bit like “Minority Report,” the 2002 movie that featured genetically altered humans with special powers to predict crime. The CPD’s new crime-forecasting unit was analyzing 911 calls and produced an intelligence report predicting a shooting would happen soon on a particular block on the South Side. Three minutes later, it did...”

# Case study: Crime prediction in Chicago



Since 2009, we have been working with the Chicago Police Department (CPD) to predict and prevent emerging clusters of violent crime.

Our new crime prediction methods have been incorporated into our **CrimeScan** software, run twice a day by CPD and used operationally for deployment of patrols.

*“CrimeScan was set up to run daily, completely autonomously. Predictions were sent to police analysts, and messages were compiled into detailed intelligence reports disseminated through the chain of command. Based upon deployment suggestions indicated in the reports, **important arrests were affected, weapons were seized, and crimes were prevented.**”*

# CrimeScan

The key insight of our method is to **use detection for prediction**:

We can **detect emerging clusters** of various leading indicators (minor crimes, 911 calls, etc.) and use these to **predict** that a cluster of violent crime is likely to occur nearby.

Some advantages of the CrimeScan approach:

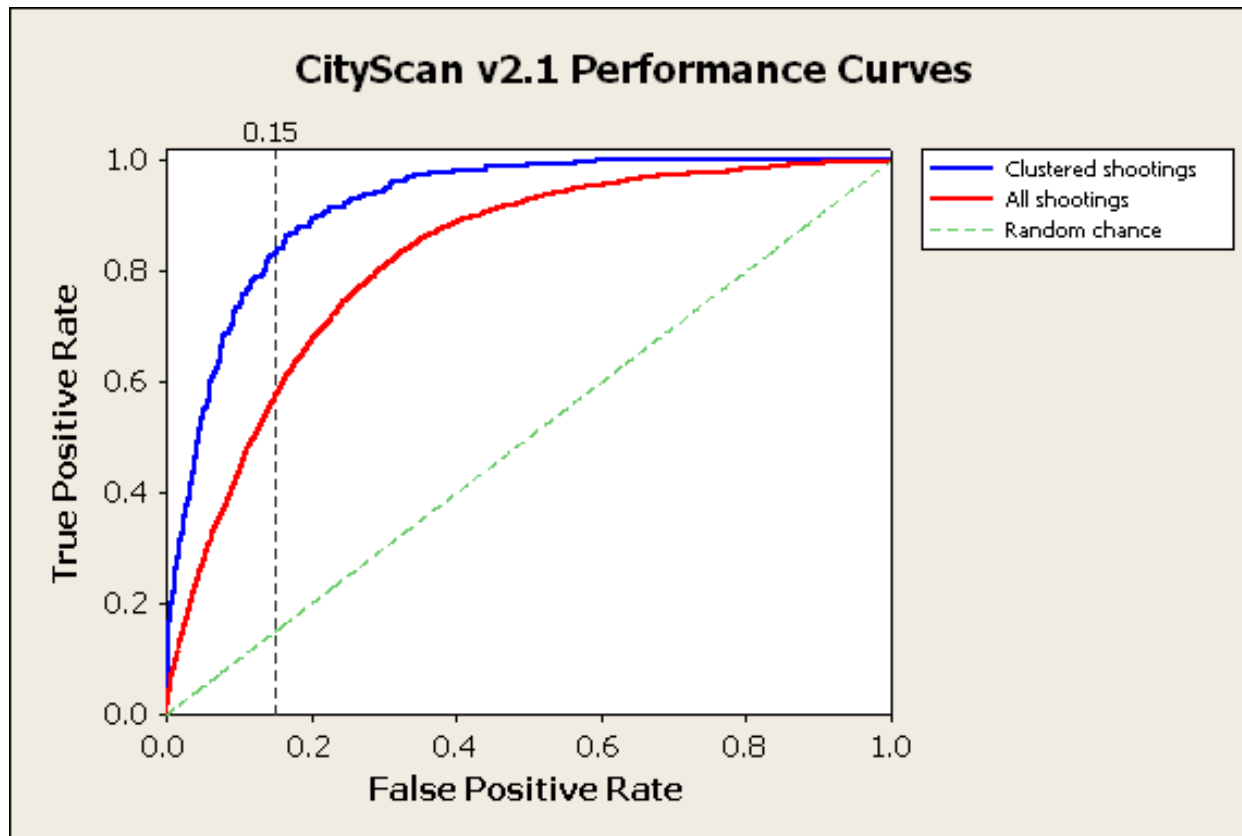
- Advance prediction (up to 1 week) with high accuracy.
- High spatial and temporal resolution (block x day).
- Predicting **emerging hot spots** of violence, as opposed to just identifying bad neighborhoods.

How to detect leading indicator clusters?

How to use these for prediction?

Which leading indicators to use?

# Model-based prediction results



In our preliminary evaluation on 2011-2013 data, the latest version of CityScan predicts **83%** of clustered shootings/homicides and **57%** of all shootings/homicides at a 15% false positive rate.

(Keep in mind that only 15% would be predicted by chance at this false positive rate!)

# CrimeScan

The key insight of our method is to **use detection for prediction**:

We can **detect emerging clusters** of various leading indicators (minor crimes, 911 calls, etc.) and use these to **predict** that a cluster of violent crime is likely to occur nearby.

The fast subset scanning approaches described above enable **early and accurate detection** of emerging clusters.

How to detect leading indicator clusters?

How to use these for prediction?

Which leading indicators to use?

# CrimeScan

The key insight of our method is to **use detection for prediction**:

We can **detect emerging clusters** of various leading indicators (minor crimes, 911 calls, etc.) and use these to **predict** that a cluster of violent crime is likely to occur nearby.

The fast subset scanning approaches described above enable **early and accurate detection** of emerging clusters.

Proximity to detected clusters → features in a predictive model.

We use **scalable Gaussian process regression** to model spatial correlation and improve prediction accuracy.

How to detect leading indicator clusters?

How to use these for prediction?

Which leading indicators to use?

# CrimeScan

The key insight of our method is to **use detection for prediction**:

We can **detect emerging clusters** of various leading indicators (minor crimes, 911 calls, etc.) and use these to **predict** that a cluster of violent crime is likely to occur nearby.

“Kitchen sink” penalized regression does not work so well.

Correlation-based LI selection is confounded by purely spatial and purely temporal correlations.

Our solution is a new bivariate “kernel space-time independence” test that identifies space-time interactions between LI types while controlling for space and time.

How to detect leading indicator clusters?

How to use these for prediction?

**Which leading indicators to use?**

# From CrimeScan to CityScan...

We have been working with city leaders in Chicago, Pittsburgh, and Baltimore to predict emerging spatial patterns of **311 calls** (non-emergency service requests). By providing support for precisely targeted interventions, we will enable cities to respond **proactively** and **effectively** to emerging challenges and citizen needs.



Indicators of neighborhood decay (graffiti, abandoned buildings, etc.)



Health and sanitation issues, particularly focusing on rodent prevention.



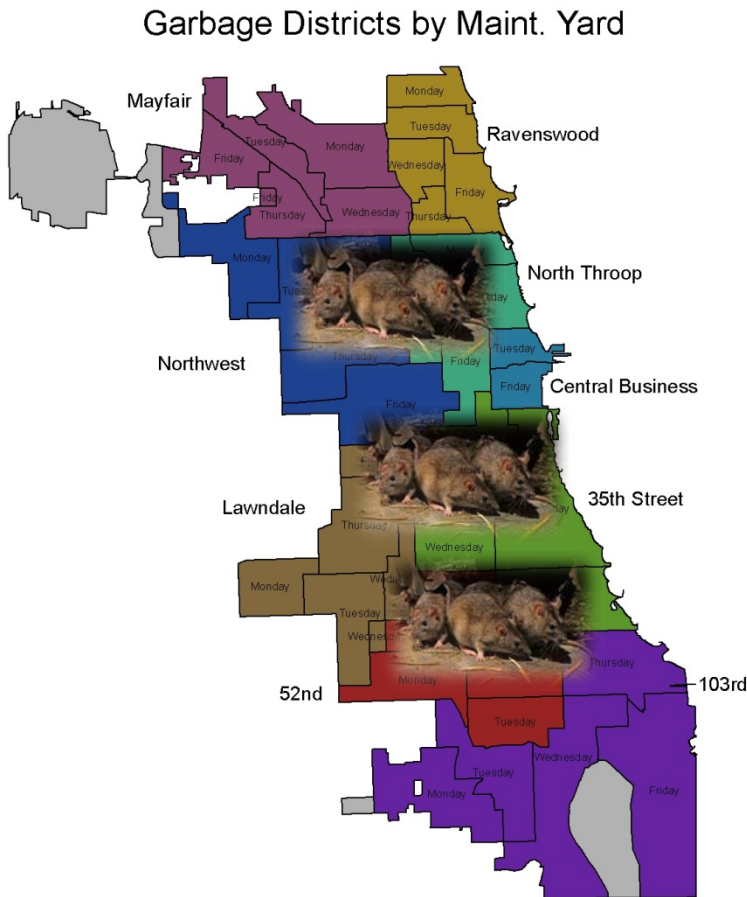
# Top 30 Call Types- Pittsburgh

Potholes	38,893	Dumping ( <i>public &amp; private</i> )	4,901
Weeds/Debris	31,697	ES / Missed Pickup	4,686
Snow & Ice Control	13,042	Tree – Pruning	4,407
ABDV ( <i>on public property</i> )	9,093	Patrol (suspicious persons)	4,383
Building Violation – Res.	8,488	Clean ( <i>low income, weeds/debris</i> )	4,257
Overgrowth ( <i>public property</i> )	7,635	Sign Request	4,238
Miscellaneous	7,117	Sewers	4,001
Signs – Replacement	6,477	Street Cleaning	3,575
Vacant and Open ( <i>squatters</i> )	5,557	Drug Enforcement ( <i>sent to police</i> )	3,260
ES / Violations	5,508	Debris ( <i>public property</i> )	3,233
Street Light	5,377	Rodent Control ( <i>private property</i> )	3,055
Tree – Removal	5,193	Sidewalk - Snow covered only	2,987
Parking	5,092	Graffiti ( <i>sent to police</i> )	2,956
Street Resurfacing	5,062	Traffic	2,831
		County Property Reassessment	2,492

# CityScan: Preventing rat infestations

We are currently performing a controlled experiment with Chicago's Dept. of Streets and Sanitation, with the goal of predicting and preventing rodent infestations.

- Measured by "rodent complaint" 311 calls.
- Other 311 call types as leading indicators.



## "Treatment" garbage districts:

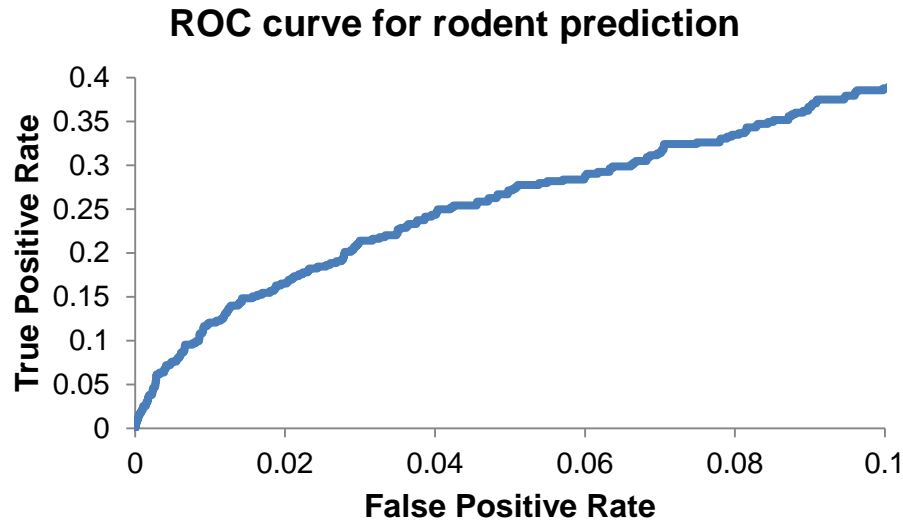
We predict rodent complaints using CityScan and use predictions to direct the city's preventative rat baiting crews.

## "Control" garbage districts:

Preventative baiting performed as usual.

**Featured in Chicago Business Journal and Baltimore Sun-Times:  
"Carnegie Mellon smells a rat, and Chicago is grateful"**

# Rodent Prediction- Pittsburgh



Out-of-sample prediction results (7 days in advance):

40% TPR at 10% FPR

27% TPR at 5% FPR

15% TPR at 1.4% FPR

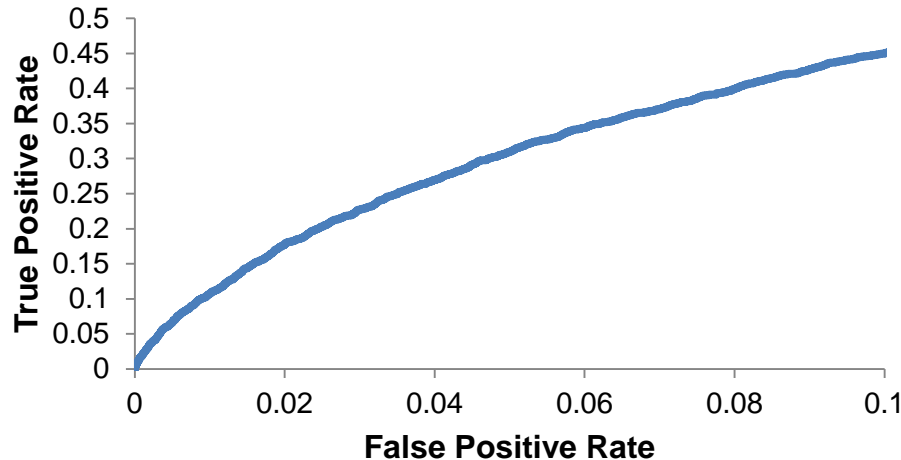
These results suggests large potential returns to a small but precisely targeted preventative rodent abatement program.

Predictors used in model: spatial (KDE), temporal (month, weekend), lagged weekly rodent counts, lagged weekly leading indicator counts.

Leading indicators (10): Sanitation (Partial Pickup, Early Set Out), Dead Animal, Building Violation (Commercial and Residential), Tree Removal, **Weeds/Debris**, Vacant and Open, Overgrowth, Litter Can

# Rodent Prediction- Baltimore

ROC curve for rodent prediction



Out-of-sample prediction results (7 days in advance):

45% TPR at 10% FPR

31% TPR at 5% FPR

15% TPR at 1.5% FPR

These results suggests large potential returns to a small but precisely targeted preventative rodent abatement program.

Predictors used in model: spatial (KDE), temporal (month, weekend), lagged weekly rodent counts, lagged weekly leading indicator counts.

Leading indicators (7): Animals, Sanitation Property, Dumping, Food Facility Complaint, Vacant Building, Dirty Alley, Dirty Street