Efficient Threshold Monitoring for Distributed Probabilistic Data

Mingwang Tang, Feifei Li, Jeff M. Phillips, Jeffrey Jestes
Distributed threshold monitoring (DTM) problem:

\[ H \sum_{i=1}^{3} x_i > 240 \]

\[
\begin{array}{ccc}
    c_1 & c_2 & c_3 \\
    t_1 & 70 & 80 & 75 \\
    t_2 & 70 & 90 & 90 \\
    \vdots & \vdots & \vdots & \vdots \\
    t_T & 70 & 80 & 70 \\
\end{array}
\]
Distributed threshold monitoring (DTM) problem:

\[ H \]

225 > 240?

\[ c_1 \] \[ c_2 \] \[ c_3 \]

\begin{array}{ccc}
\text{\( t_1 \)} & 70 & 80 & 75 \\
\text{\( t_2 \)} & 70 & 90 & 90 \\
\vdots & \vdots & \vdots & \vdots \\
\text{\( t_T \)} & 70 & 80 & 70 \\
\end{array}
Distributed threshold monitoring (DTM) problem:

\[ H = 250 > 240? \]

\[
\begin{array}{ccc}
\hat{c}_1 & \hat{c}_2 & \hat{c}_3 \\
\end{array}
\]

\[
\begin{array}{ccc}
t_1 & 70 & 80 & 75 \\
t_2 & 70 & 90 & 90 \\
t_T & 70 & 80 & 70 \\
\end{array}
\]

Extensively studied: e.g., S. Jeyashanker et al. propose an adaptive technique dealing with DTM problem for deterministic data.
Distributed threshold monitoring (DTM) problem:

- $H > 240$?
- $t_1: 70, 80, 75$
- $t_2: 70, 90, 90$
- $t_T: 70, 80, 70$
- $c_1, c_2, c_3$
Distributed threshold monitoring (DTM) problem:

\[
H > 240? \\
\]

Extensively studied: e.g., S. Jeyashanker et al. propose an adaptive technique dealing with DTM problem \((\sum_{i=1}^{g} x_i \leq T)\) for deterministic data.

Distributed threshold monitoring (DTM) problem:

\[ H \sum_{i=1}^{3} x_i > 240 \]

Extensively studied: e.g., S. Jeyashanker et al. propose an adaptive technique dealing with DTM problem \((\sum_{i=1}^{g} x_i \leq T)\) for deterministic data.

Distributed threshold monitoring (DTM) problem:

\[ H \quad \sum_{i=1}^{3} x_i > 240 \]

Extensively studied: e.g., S. Jeyashanker et al. propose an adaptive technique dealing with DTM problem \((\sum_{i=1}^{g} x_i \leq T)\) for deterministic data.

Distributed threshold monitoring (DTM) problem:

\[ H \quad \sum_{i=1}^{3} x_i > 240 \]

Extensively studied: e.g., S. Jeyashanker et al. propose an adaptive technique dealing with DTM problem \((\sum_{i=1}^{g} x_i \leq T)\) for deterministic data.

Distributed threshold monitoring (DTM) problem:

\[ H \quad \sum_{i=1}^{3} x_i > 240 \]

\[ c_1 \quad 80 \quad c_2 \quad 80 \quad c_3 \quad 80 \]

\[ \land \quad ? \quad \land \quad ? \quad \land \quad ? \]

\[
\begin{array}{ccc}
t_1 & 70 & 80 & 75 \\
t_2 & 70 & 90 & 90 \\
\vdots & \vdots & \vdots & \vdots \\
t_T & 70 & 80 & 70 \\
\end{array}
\]

Extensively studied: e.g., S. Jeyashanker et al. propose an adaptive technique dealing with DTM problem (\( \sum_{i=1}^{g} x_i \leq T \)) for deterministic data.

Distributed threshold monitoring (DTM) problem:

\[ H \quad \sum_{i=1}^{3} x_i > 240 \]

Extensively studied: e.g., S. Jeyashanker et al. propose an adaptive technique dealing with DTM problem (\( \sum_{i=1}^{g} x_i \leq T \)) for deterministic data.

New challenge in the DTM problem: uncertainty naturally exist in distributed data

- data integration produces fuzzy matches
- noisy sensor readings
New challenge in the DTM problem: uncertainty naturally exist in distributed data
- data integration produces fuzzy matches
- noisy sensor readings

The Shipboard Automated Meteorological and Oceanographic System (SAMOS)
- Attribute-level uncertain model (with a single attribute score)

<table>
<thead>
<tr>
<th>tuples</th>
<th>attribute score</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_1$</td>
<td>$X_1 = {(v_{1,1}, p_{1,1}), (v_{1,2}, p_{1,2})... (v_{1,b_1}, p_{1,b_1})}$</td>
</tr>
<tr>
<td>$d_2$</td>
<td>$X_2 = {(v_{2,1}, p_{2,1}), (v_{2,2}, p_{2,2})... (v_{2,b_2}, p_{2,b_2})}$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>$d_t$</td>
<td>$X_t = {(v_{t,1}, p_{t,1}), (v_{t,2}, p_{t,2})... (v_{t,b_t}, p_{t,b_t})}$</td>
</tr>
</tbody>
</table>
Distributed probabilistic threshold monitoring (DPTM):

\[ H \prod \Pr[Y = \sum_{i=1}^{g} X_i > \gamma] > \delta? \]

\[
\begin{align*}
   &c_1 \quad c_2 \quad \ldots \quad c_g \\
   &t_1 \quad X_{1,1} \quad X_{2,1} \quad \ldots \quad X_{g,1} \\
   &t_2 \quad X_{1,2} \quad X_{2,2} \quad \ldots \quad X_{g,2} \\
   &\vdots \quad \vdots \quad \vdots \quad \ldots \quad \vdots \\
   &t_T \quad X_{1,T} \quad X_{2,T} \quad \ldots \quad X_{g,T}
\end{align*}
\]
Distributed probabilistic threshold monitoring (DPTM):

\[ H \quad \Pr[Y = \sum_{i=1}^{g} X_i > \gamma] > \delta ? \]

\begin{align*}
& t_1 \quad X_{1,1} \quad X_{2,1} \quad X_{g,1} \\
& t_2 \quad X_{1,2} \quad X_{2,2} \quad X_{g,2} \\
& \vdots \quad \vdots \quad \vdots \quad \vdots \\
& t_T \quad X_{1,T} \quad X_{2,T} \quad X_{g,T}
\end{align*}
Distributed probabilistic threshold monitoring (DPTM):

\[ H \quad \Pr[Y = \sum_{i=1}^{g} X_i > \gamma] > \delta ? \]

\[ t_1 \quad X_{1,1} \quad X_{2,1} \quad X_{g,1} \]
\[ t_2 \quad X_{1,2} \quad X_{2,2} \quad X_{g,2} \]
\[ \vdots \quad \vdots \quad \vdots \quad \vdots \]
\[ t_T \quad X_{1,T} \quad X_{2,T} \quad X_{g,T} \]
Distributed probabilistic threshold monitoring (DPTM):

\[ H | \Pr[Y = \sum_{i=1}^{g} X_i > \gamma] > \delta ? \]

\[
\begin{array}{ccc}
  c_1 & 
  c_2 & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
  t_1 & X_{1,1} & X_{2,1} & X_{g,1} \\
  t_2 & X_{1,2} & X_{2,2} & X_{g,2} \\
  \vdots & \vdots & \vdots & \vdots \\
  t_T & X_{1,T} & X_{2,T} & X_{g,T} \\
\end{array}
\]
Our Approach

Exact Methods:

- Computing \( \Pr[Y > \gamma] \) exactly is expensive in terms of both communication \( (O(gT)) \) and computation \( (O(n^g T)) \).
Our Approach

- **Exact Methods:**
  - Computing $\Pr[Y > \gamma]$ exactly is expensive in terms of both communication ($O(gT)$) and computation ($O(n^g T)$).
  - Incorporates pruning techniques.
**Our Approach**

- **Exact Methods:**
  - Computing $\Pr[Y > \gamma]$ exactly is expensive in terms of both communication ($O(gT)$) and computation ($O(n^g T)$).
  - Incorporates pruning techniques.

The diagram illustrates the decision-making process based on the value of $H$. The deterministic data derived from $X_i$'s is indicated, with specific variables $t_1, t_2, \ldots, t_T$ for different time points.
Our Approach

- **Exact Methods:**
  - Computing $\Pr[Y > \gamma]$ exactly is expensive in terms of both communication ($O(gT)$) and computation ($O(n^g T)$).
  - Incorporates pruning techniques.
  - Combine the adaptive threshold algorithm for deterministic data when it's applicable.

```
Pr[Y > \gamma] < upperbound < \delta ?
Pr[Y > \gamma] > lowerbound > \delta ?
```

deterministic data derived based $X_i$'s

t_1 \quad X_{1,1} \quad X_{2,1} \quad X_{g,1}
t_2 \quad X_{1,2} \quad X_{2,2} \quad X_{g,2}
\vdots \quad \vdots \quad \vdots \quad \vdots
t_T \quad X_{1,T} \quad X_{2,T} \quad X_{g,T}
```
Our Approach

- **Exact Methods:**
  - Computing $\Pr[Y > \gamma]$ exactly is expensive in terms of both communication ($O(gT)$) and computation ($O(n^g T)$).
  - Incorporates pruning techniques.
  - Combine the adaptive threshold algorithm for deterministic data when it’s applicable.

- **Approximate Methods:**
  - Replace the exact computation of $\Pr[Y > \gamma]$ using sampling method (but with the same monitoring instance).
Outline

1. Introduction and Motivation
2. Exact Methods
3. Approximate Methods
4. Experiments
5. Conclusion
Markov’s inequality: $\Pr[Y > \gamma] \leq \frac{\mathbb{E}(Y)}{\gamma}$
Baseline method (Madaptive)

- Markov’s inequality: \( \Pr[Y > \gamma] \leq \frac{\mathbb{E}(Y)}{\gamma} \)
Baseline method (Madaptive)

- Markov’s inequality: $\Pr[Y > \gamma] \leq \frac{E(Y)}{\gamma} < \delta$?

\[
H \quad \Pr[Y = \sum_{i=1}^{g} X_i > \gamma] > \delta?
\]

$$t \quad X_1 \quad X_2 \cdots X_g$$
Markov’s inequality: $\Pr[Y > \gamma] \leq \frac{E(Y)}{\gamma} < \delta \? \rightarrow ture (no \ alarm)$
Markov’s inequality: $\Pr[Y > \gamma] \leq \frac{E(Y)}{\gamma} < \delta \quad \rightarrow \text{ture (no alarm)}$
Markov’s inequality: $\Pr[Y > \gamma] \leq \frac{E(Y)}{\gamma} < \delta \Rightarrow ture$ (no alarm)
Baseline method (Madaptive)

- Markov's inequality: $\Pr[Y > \gamma] \leq \frac{E(Y)}{\gamma} < \delta \implies \text{true (no alarm)}$

- $E(Y) = \sum_{i=1}^{g} E(X_i) < (\gamma \delta) \text{? constant}$

- $E(X_1) \quad E(X_2) \quad E(X_g)$

- $t \quad X_1 \quad X_2 \quad \ldots \quad X_g$

- Leverage on the adaptive thresholds algorithm for deterministic data
Improved method

- Combine the Chebyshev bound and Chernoff bound pruning.
Improved method

- Combine the Chebyshev bound and Chernoff bound pruning.
- Chebyshev gives one-sided bound using $E(X_i)$ and $\text{Var}(X_i)$
Improved method

- Combine the Chebyshev bound and Chernoff bound pruning.
- Chebyshev gives one-sided bound using $\mathbb{E}(X_i)$ and $\text{Var}(X_i)$

$$H \quad \Pr[Y = \sum_{i=1}^{g} X_i > \gamma] > \delta ?$$

$c_1 \quad c_2 \quad \ldots \quad c_g$

$t \quad X_1 \quad X_2 \quad \ldots \quad X_g$
Improved method

- Combine the Chebyshev bound and Chernoff bound pruning.
- Chebyshev gives one-sided bound using $E(X_i)$ and $\text{Var}(X_i)$

\[ H \quad \Pr[Y > \gamma] \leq \frac{\text{Var}(Y)}{\text{Var}(Y) + (\gamma - E(Y))^2} \leq \delta ? \]

\[ E(X_1) \quad E(X_2) \quad \cdots \quad E(X_g) \]
\[ \text{Var}(X_1) \quad \text{Var}(X_2) \cdots \quad \text{Var}(X_g) \]
\[ t \quad X_1 \quad X_2 \quad \cdots \quad X_g \]
Improved method

- Combine the Chebyshev bound and Chernoff bound pruning.
- Chebyshev gives one-sided bound using $E(X_i)$ and $\text{Var}(X_i)$

$$H \quad \Pr[Y > \gamma] > 1 - \frac{\text{Var}(Y)}{\text{Var}(Y) + (E(Y) - \gamma)^2} \geq \delta$$

![Diagram](image-url)
Improved method

- Combine the Chebyshev bound and Chernoff bound pruning.
- Chebyshev gives one-sided bound using $E(X_i)$ and $\text{Var}(X_i)$
- Chernoff bound using the moment generating function
  - $M(\beta) = E(e^{\beta Y})$, $M_i(\beta) = E(e^{\beta X_i})$ for any $\beta \in \mathbb{R}$
  - $M(\beta) = \prod_{i=1}^{g} M_i(\beta)$
Improved method

- Combine the Chebyshev bound and Chernoff bound pruning.
- Chebyshev gives one-sided bound using $\mathbb{E}(X_i)$ and $\text{Var}(X_i)$
- Chernoff bound using the moment generating function
  - $M(\beta) = \mathbb{E}(e^{\beta Y})$, $M_i(\beta) = \mathbb{E}(e^{\beta X_i})$ for any $\beta \in \mathbb{R}$
  - $M(\beta) = \prod_{i=1}^{g} M_i(\beta)$
- $\beta_1 > 0$, Chernoff gives an upper bound
- $\beta_2 < 0$, Chernoff gives a lower bound

$$
Pr[Y > \gamma] \leq e^{-\beta_1 \gamma M(\beta_1)} \leq \delta \ ?
$$

$$
Pr[Y > \gamma] > 1 - e^{-\beta_2 \gamma M(\beta_2)} \geq \delta \ ?
$$

$$
\begin{array}{cccc}
  H & Pr[Y > \gamma] \leq e^{-\beta_1 \gamma M(\beta_1)} \leq \delta \ ? \\
  & Pr[Y > \gamma] > 1 - e^{-\beta_2 \gamma M(\beta_2)} \geq \delta \ ? \\
\end{array}
$$

\[c_1 \xrightarrow{M_1(\beta_1)} c_2 \xrightarrow{M_2(\beta_1)} \cdots \xrightarrow{M_g(\beta_1)} c_g\]

\[c_1 \xrightarrow{M_1(\beta_2)} c_2 \xrightarrow{M_2(\beta_2)} \cdots \xrightarrow{M_g(\beta_2)} c_g\]

$t \ X_1 \ X_2 \ \cdots \ X_g$
Improved Adaptive Method (Iadaptive)

- \[ \sum_{i=1}^{g} \ln M_i(\beta_1) \leq \ln \delta + \beta_1 \gamma, \text{ (monitoring instance } J_1) \].
- \[ \sum_{i=1}^{g} \ln M_i(\beta_2) \leq \ln(1 - \delta) + \beta_2 \gamma, \text{ (monitoring instance } J_2) \].

Practical considerations

- Use the adaptive thresholds algorithm.
- Get a tight upper bound (lower bound).

Approaches

- Fix the values of \( \beta_1 \) and \( \beta_2 \) in each period of \( k \) time instance.
- Reset the optimal values of \( \beta_1 \) and \( \beta_2 \) periodically.
- Periodically decide which monitoring instance to run.
\[ \sum_{i=1}^{g} \ln M_i(\beta_1) \leq \ln \delta + \beta_1 \gamma, \text{ (monitoring instance } J_1) \].

\[ \sum_{i=1}^{g} \ln M_i(\beta_2) \leq \ln(1 - \delta) + \beta_2 \gamma, \text{ (monitoring instance } J_2) \].

Practical considerations
  - Use the adaptive thresholds algorithm
  - Get a tight upper bound (lower bound)
Improved Adaptive Method (Iadaptive)

\[ \sum_{i=1}^{g} \ln M_i(\beta_1) \leq \ln \delta + \beta_1 \gamma, \text{ (monitoring instance } J_1 \text{).} \]

\[ \sum_{i=1}^{g} \ln M_i(\beta_2) \leq \ln(1 - \delta) + \beta_2 \gamma, \text{ (monitoring instance } J_2 \text{).} \]

Practical considerations
- Use the adaptive thresholds algorithm
- Get a tight upper bound (lower bound)

Approaches
- Fix the values of \( \beta_1 \) and \( \beta_2 \) in each period of \( k \) time instance.
- Reset the optimal values of \( \beta_1 \) and \( \beta_2 \) periodically.
- $\sum_{i=1}^{g} \ln M_i(\beta_1) \leq \ln \delta + \beta_1 \gamma$, (monitoring instance $J_1$).
- $\sum_{i=1}^{g} \ln M_i(\beta_2) \leq \ln(1 - \delta) + \beta_2 \gamma$, (monitoring instance $J_2$).

Practical considerations
- Use the adaptive thresholds algorithm
- Get a tight upper bound (lower bound)
- Running $J_1$ and $J_2$ together is communication expensive.

Approaches
- Fix the values of $\beta_1$ and $\beta_2$ in each period of $k$ time instance.
- Reset the optimal values of $\beta_1$ and $\beta_2$ periodically.
Improved Adaptive Method (Iadaptive)

\[ \sum_{i=1}^{g} \ln M_i(\beta_1) \leq \ln \delta + \beta_1 \gamma, \text{ (monitoring instance } J_1) \].

\[ \sum_{i=1}^{g} \ln M_i(\beta_2) \leq \ln(1 - \delta) + \beta_2 \gamma, \text{ (monitoring instance } J_2) \].

Practical considerations
- Use the adaptive thresholds algorithm
- Get a tight upper bound (lower bound)
- Running \( J_1 \) and \( J_2 \) together is communication expensive.

Approaches
- Fix the values of \( \beta_1 \) and \( \beta_2 \) in each period of \( k \) time instance.
- Reset the optimal values of \( \beta_1 \) and \( \beta_2 \) periodically.
- Periodically decides which monitoring instance to run
1 Introduction and Motivation

2 Exact Methods

3 Approximate Methods

4 Experiments

5 Conclusion
Our Approach

- **Exact Methods:**
  - Computing $\Pr[Y > \gamma]$ exactly is expensive in terms of both communication ($O(gT)$) and computation ($O(n^g T)$).
  - Incorporates pruning techniques.
  - Combine the adaptive threshold algorithm for deterministic data when it’s applicable.

- **Approximate Methods:**
Our Approach

- **Exact Methods:**
  - Computing $\Pr[Y > \gamma]$ exactly is expensive in terms of both communication ($O(gT)$) and computation ($O(n^g T)$).
  - Incorporates pruning techniques.
  - Combine the adaptive threshold algorithm for deterministic data when it’s applicable.

- **Approximate Methods:**
  - We use $\epsilon$-Sampling methods to estimate the condition when monitoring instances fail to make a decision.
Our Approach

**Exact Methods:**
- Computing $\Pr[Y > \gamma]$ exactly is expensive in terms of both communication ($O(gT)$) and computation ($O(n^g T)$).
- Incorporates pruning techniques.
- Combine the adaptive threshold algorithm for deterministic data when it’s applicable.

**Approximate Methods:**
- We use $\epsilon$-Sampling methods to estimate the condition when monitoring instances fail to make a decision.
- Replace the exact computation using sampling based method (but with the same monitoring instance): we get MadaptiveS, ImprovedS, IadaptiveS.
Random Distributed $\varepsilon$-Sample (RD$\varepsilon$S)

- $H$ asks for a random sample $x_i$ from each client according to the distribution of $X_i$
- $\Pr[\tilde{Y} = \sum_{i=1}^{g} x_i > \gamma]$ is an unbiased estimate of $\Pr[Y > \gamma]$
- Repeating this sampling $\kappa = O\left(\frac{1}{\varepsilon^2} \ln \frac{1}{\phi}\right)$ times.
- $\Pr[|\Pr[\tilde{Y} > \gamma] - \Pr[Y > \gamma]| \leq \varepsilon] \geq 1 - \phi$ using $O\left(\frac{g}{\varepsilon^2} \ln \frac{1}{\phi}\right)$ bytes.
Using $\kappa = O\left(\frac{g}{\epsilon}\right)$ evenly spaced sample points from each $X_i$. 
Deterministic Distributed $\varepsilon$-Sample (DD$\varepsilon$S)

- Using $\kappa = O(\frac{g}{\varepsilon})$ evenly spaced sample points from each $X_i$.
- $\int_{x=x_j}^{x_{j+1}} \Pr[X_i = x] dx = \frac{\varepsilon}{g}$
Deterministic Distributed $\varepsilon$-Sample (DD$\varepsilon$S)

- Using $\kappa = O\left(\frac{g}{\varepsilon}\right)$ evenly spaced sample points from each $X_i$.
- \[
\int_{x=x_j}^{x_{j+1}} \Pr[X_i = x] dx = \frac{\varepsilon}{g}
\]
- The evaluation space $\Pr[\tilde{Y} > \gamma]$ is in $O(\kappa g)$

\[
\begin{align*}
  c_1 : & \quad S_1 \{x_{1,1}, x_{1,2}, \ldots, x_{1,\kappa}\} \\
  c_2 : & \quad S_2 \{x_{2,1}, x_{2,2}, \ldots, x_{2,\kappa}\} \\
  & \quad \vdots \quad \vdots \\
  c_g : & \quad S_g \{x_{g,1}, x_{g,2}, \ldots, x_{g,\kappa}\}
\end{align*}
\]
Deterministic Distributed $\varepsilon$-Sample (DD$\varepsilon$S)

- Using $\kappa = O\left(\frac{g}{\varepsilon}\right)$ evenly spaced sample points from each $X_i$.
- $\int_{x=x_j}^{x_{j+1}} \Pr[Xi = x] dx = \frac{\varepsilon}{g}$
- The evaluation space $\Pr[\tilde{Y} > \gamma]$ is in $O(\kappa^g)$

$c_1 : S_1\{x_{1,1}, x_{1,2}, \ldots, x_{1,\kappa}\}$
$c_2 : S_2\{x_{2,1}, x_{2,2}, \ldots, x_{2,\kappa}\}$
\[ \cdots \]
$c_g : S_g\{x_{g,1}, x_{g,2}, \ldots, x_{g,\kappa}\}$
Deterministic Distributed $\varepsilon$-Sample (DD$\varepsilon$S)

- Using $\kappa = O(\frac{g}{\varepsilon})$ evenly spaced sample points from each $X_i$.
- $\int_{x=x_j}^{x_{j+1}} \Pr[X_i = x] dx = \frac{\varepsilon}{g}$
- The evaluation space $\Pr[\tilde{Y} > \gamma]$ is in $O(\kappa g)$
- In practice, $O(\kappa^m)$ (e.g., $m = 2$) random selected evaluations.

$$c_1 : S_1 \{x_{1,1}, x_{1,2}, \ldots, x_{1,\kappa}\}$$
$$c_2 : S_2 \{x_{2,1}, x_{2,2}, \ldots, x_{2,\kappa}\}$$
$$\vdots$$
$$\vdots$$
$$\vdots$$
$$c_g : S_g \{x_{g,1}, x_{g,2}, \ldots, x_{g,\kappa}\}$$
Deterministic Distributed \(\varepsilon\)-Sample (\(\text{DD}\varepsilon\text{S}\))

- Using \(\kappa = O\left(\frac{g}{\varepsilon}\right)\) evenly spaced sample points from each \(X_i\).
- \(\int_{x=x_j}^{x_{j+1}} \Pr[X_i = x]dx = \frac{\varepsilon}{g}\)

- The evaluation space \(\Pr[\tilde{Y} > \gamma]\) is in \(O(\kappa^g)\)

- In practice, \(O(\kappa^m)\) (e.g., \(m = 2\)) random selected evaluations.

\[
\begin{align*}
&c_1 : S_1\{x_{1,1}, x_{1,2}, \ldots, x_{1,\kappa}\} \\
&c_2 : S_2\{x_{2,1}, x_{2,2}, \ldots, x_{2,\kappa}\} \\
&\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \frac{55}{76}
\end{align*}
\]

- \(\text{DD}\varepsilon\text{S}\) gives \(|\Pr[\tilde{Y} > \gamma] - \Pr[Y > \gamma]| \leq \varepsilon\) with probability 1 in \(O(g^2/\varepsilon)\) bytes.
A randomized improvement of \textit{DDES (\(\alpha\text{DDES}\))}

\[
\int_{x=x_{i,j}}^{x_{i,j+1}} Pr[X_i = x] dx = \alpha
\]
A randomized improvement of $\text{DD}\varepsilon S$ ($\alpha\text{DD}\varepsilon S$)

- $\int_{x=x_{i,j}}^{x_{i,j+1}} Pr[X_i = x] dx = \alpha$
- Computes $x_\alpha$ where the integral of $pdf$ first reaches $\alpha$
A randomized improvement of DD$\varepsilon$S ($\alpha$DD$\varepsilon$S)

- $\int_{x=x_i,j}^{x_{i,j+1}} Pr[X_i = x]dx = \alpha$
- Computes $x_\alpha$ where the integral of $pdf$ first reaches $\alpha$
- Chooses the smallest sample point at random (within $x_\alpha$).

\[\int_{x=x_i,j}^{x_{i,j+1}} Pr[X_i = x]dx = \alpha\]
Computes $x_\alpha$ where the integral of $pdf$ first reaches $\alpha$
Chooses the smallest sample point at random (within $x_\alpha$).
A randomized improvement of $\text{DD}\varepsilon S (\alpha \text{DD}\varepsilon S)$

- $\int_{x=x_{i,j}}^{x_{i,j+1}} Pr[X_i = x] dx = \alpha$
- Computes $x_\alpha$ where the integral of pdf first reaches $\alpha$
- Chooses the smallest sample point at random (within $x_\alpha$).
A randomized improvement of \( \text{DD}\varepsilon S (\alpha\text{DD}\varepsilon S) \)

- \( \int_{x=x_{i,j}}^{x_{i,j+1}} \Pr[X_i = x]dx = \alpha \)
- Computes \( x_\alpha \) where the integral of pdf first reaches \( \alpha \)
- Chooses the smallest sample point at random (within \( x_\alpha \)).

\[ \Pr[| \Pr[\tilde{Y} > \gamma] - \Pr[Y > \gamma]| \leq \varepsilon] > 1 - \phi \text{ in } O\left(\frac{g}{\varepsilon} \sqrt{2g \ln \frac{2}{\phi}}\right) \text{ bytes.} \]
Experiment setup

- A Linux machine with an Intel Xeon CPU at 2.13GHz and 6GB of memory. GMP library are used in calculating $M_i(\beta)$.
- Server-to-client using broadcast and client-to-server using unicast.
- Data sets:
  - Real datasets (11.8 million records in the Wecoma research vessels) from the SAMOS project.
  - Each record contain four measurements: wind direction (WD), wind speed (WS), sound speed (SS), and temperature (TEM), which leads to four single probabilistic attribute datasets.
  - Group the records every $\tau$ consecutive seconds and represent it using a pdf.
The default experimental parameters:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau$</td>
<td>grouping interval</td>
<td>300</td>
</tr>
<tr>
<td>$T$</td>
<td>number of time instances</td>
<td>3932</td>
</tr>
<tr>
<td>$g$</td>
<td>number of clients</td>
<td>10</td>
</tr>
<tr>
<td>$\delta$</td>
<td>probability threshold</td>
<td>0.7</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>score threshold</td>
<td>30% alarms (230.g for WD)</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>sample size per client</td>
<td>30</td>
</tr>
</tbody>
</table>
Response time:

- $\gamma$: score threshold

![Graph showing response time vs $\gamma$ for Madaptive, Improved, and Iadaptive models]
\( \gamma \): score threshold

![Graph showing communication metrics: number of messages and number of bytes vs. score threshold (\( \gamma \)).](image)
\( \kappa \): number of samples
$\kappa$: number of samples
Performance of all methods

- Madaptive, MadaptiveS
- Improved, ImprovedS
- Iadaptive, IadaptiveS

Number of messages

WD  WS  SS  TEM

4 / 6 / 8 / 10
Performance of all methods

<table>
<thead>
<tr>
<th>Method</th>
<th>WD</th>
<th>WS</th>
<th>SS</th>
<th>TEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Madaptive</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>MadaptiveS</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Improved</td>
<td>2</td>
<td>10</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>ImprovedS</td>
<td>10</td>
<td>10</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>Iadaptive</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>IadaptiveS</td>
<td>10</td>
<td>10</td>
<td>3</td>
<td>10</td>
</tr>
</tbody>
</table>
Performance of all methods

- WD
- WS
- SS
- TEM

Madaptive, MadaptiveS
Improved, ImprovedS
Iadaptive, IadaptiveS
Future work:

- Other aggregation constraints (e.g., max) beside sum constraint.
- Extend our study to the hierarchical model that is often used in a sensor network.
- Handle the case when data from different sites are correlated.
Thank You

Q and A