Pervasive and Mobile Computing 9 (2013) 372–391

Contents lists available at SciVerse ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc

Application and analysis of multidimensional negative surveys in participatory sensing applications

Michael M. Groat [a,*], Benjamin Edwards [a], James Horey [b], Wenbo He [c], Stephanie Forrest [a,d]

[a] University of New Mexico, Albuquerque, NM 87131, United States
[b] Oak Ridge National Laboratory, Oak Ridge, TN 37831, United States
[c] McGill University, Montreal, Quebec H3A 2T5, Canada
[d] Santa Fe Institute, Santa Fe, NM 87501, United States

ARTICLE INFO

Article history:
Available online 9 January 2013

Keywords:
Multidimensional data
Negative surveys
Privacy
Participatory sensing applications

ABSTRACT

Participatory sensing applications rely on individuals to share personal data to produce aggregated models and knowledge. In this setting, privacy concerns can discourage widespread adoption of new applications. We present a privacy-preserving participatory sensing scheme based on *negative surveys* for both continuous and multivariate categorical data. Without relying on encryption, our algorithms enhance the privacy of sensed data in an energy and computation efficient manner. Simulations and implementation on Android smart phones illustrate how multidimensional data can be aggregated in a useful and privacy-enhancing manner.

1. Introduction

Participatory sensing applications [1] sense, collect, analyze, and share data collected locally from a large population. They have a wide range of applications such as urban planning [2], public health [3], and vehicular transportation monitoring [4,5]. Protecting the privacy of the participants and their data is important in these applications, especially when information travels across open wireless networks. Trade-offs exist between the goal of protecting privacy and the usefulness the collected data. Energy efficiency of sensing devices is also a concern.

Existing approaches for protecting the privacy of multidimensional data [6–8] are designed for database applications, where large numbers of records from different users are available to a centralized server that summarizes statistics about the records [6,7,9,10]. However, in participatory sensing applications, individual nodes typically have access to only their own sensed values. Participants might not be willing to share sensitive information with other participants or trust a central collection server.

Our method applies negative surveys to multidimensional categorical data, where the dimensions represent different variables (e.g., temperature, time) and the categories might be symbolic values (e.g., hair color, race) or a coarse-graining of numerical data. In each dimension, the set of categories forms a proper partition of the data. Individual participants disguise their data by reporting a category chosen from the set complement of the sensed value. This operation is conducted independently for each dimension. We present algorithms that allow a base station to reconstruct the original distribution of sensed values from the disguised data [4]. This approach avoids complicated and energy intensive encryption and key management schemes.

Participatory sensing applications on wireless sensor networks (WSNs) often collect multiple observations for each sensor, for example, several different environmental values, plus time and location data. We seek to preserve the privacy of

[*] Corresponding author. Tel.: +1 505 277 3112; fax: +1 505 277 6927.
E-mail addresses: mgroat@cs.unm.edu (M.M. Groat), bedwards@cs.unm.edu (B. Edwards), horeyjl@ornl.gov (J. Horey), wenbohe@cs.mgill.ca (W. He), forrest@cs.unm.edu (S. Forrest).

1574-1192/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
doi:10.1016/j.pmcj.2012.12.004

multidimensional data such as these. For example, we describe a radiation detection scenario in Section 6.1 that estimates the distribution of radiation levels at various locations. Participants disguise both dimensions: their geographic location and their local radiation level. In some cases, data from every dimension may be considered private, and in others, values from one (non-private) dimension might reveal information about another through correlation analysis.

Our threat model assumes that the base station is *honest but curious* [11,12]. That is, we assume that it faithfully follows the network protocols but could mischievously try to collect information to use against the nodes. Additional threats come from eavesdroppers intercepting packets in transit to the base station. We do not address the problem of protecting the individual senors from adversaries.

Previous work on negative surveys required an unreasonably large number of participants to reconstruct the original data accurately [4,13]. Even small increases in the number of categories within a single dimension were problematic, and there was no attempt to address the problem of multidimensional data. This paper presents a method called dimensional adjustment, which mitigates these problems by greatly reducing the number of required data samples. Dimensional adjustment sacrifices a small amount of privacy to gain a much larger amount of utility.

When applied to continuous data, negative surveys can reconstruct probability density functions. We compare our technique to random data perturbation (RDP), which has been used in privacy-preserving data mining. We show that with sufficient samples negative surveys outperform RDP, especially when the underlying probability density functions have discontinuities.

Our protocols are implemented on Android smart phones. As a demonstration, we gridded the University of New Mexico campus into 24 different location categories and sampled the ambient sound using the phone's microphone. The campus is surrounded by streets with heavy traffic, while the interior regions are relatively quiet. The goal of the experiment was to distinguish these noisier boundary locations from quieter locations on campus.

The main contributions of this paper include the following.[1] (1) Algorithms and implementations for protecting the privacy of multidimensional sensor data, in particular, focusing on participatory sensing applications that report to a base station physical location together with sensor values. (2) Algorithms that guarantee efficient and accurate reconstruction of disguised data at the base station. (3) Simulations and a prototype implementation on Android phones that demonstrate the practicality of the method under various application scenarios. (4) The dimensional adjustment technique, which reduces (from earlier work) the number of participants required to maintain a given level of utility. This technique can also be used for single-dimensional data to improve accuracy.

Roadmap: In the remainder of the paper, we first review related work, giving background information on negative surveys and randomized response techniques (Section 2). Our protocols are presented in Section 3, and Section 4 describes the privacy and utility metrics used in the analysis. Dimensional adjustment is introduced and analyzed in Section 5. Section 6 introduces a cell phone radiation detection simulation and a simulation of MDNSs on continuous data, which is compared to random data perturbation. This section also describes the implementation of MDNS on Android smart phones. We discuss the benefits and limitations of our algorithms in Section 7, speculating about how to improve the performance and security of the simulations and implementation in future work. Section 8 concludes the paper.

2. Related work

Privacy-preserving algorithms have been developed for data mining [14–17], data aggregation [4,18–22], and other applications [23,24]. There are four main classes of solutions: perturbation, k-anonymity, secure multi-party computation, and homomorphic encryption. We review these briefly, focusing on random response techniques, including a specific instance known as "negative surveys", which serve as important background for the new algorithms and results presented in this paper.

In data mining, data values are typically hidden by perturbing individual data or query results [14–16]. To obtain accurate results, these methods typically assume that the distribution of data/noise is known ahead of time. However, as shown by Kargupta et al. [14] and Huang et al. [16], certain types of data perturbation might not preserve privacy well.

The k-anonymization technique [6,7,9,10] makes a data value from a participant indistinguishable from $k-1$ other items. It was originally designed for privacy-preserving data mining, but in participatory sensing applications individual participants can sense and share their own data. Thus, there is limited potential to mix individual participants' data as required for k-anonymity.

Secure multi-party computation (SMC) [25–27] methods specify a joint computation among a set of involved peers. This is problematic in a participatory sensing setting, because of high communication or computation overhead when the participant population is large.

[1] An earlier version of the paper appeared in PERCOM'12. We have expanded the paper to include the following material. (1) Implementation of multidimensional negative surveys (MDNSs) on smart phones. This includes comparing the energy use of the node protocol with and without encryption on a phone, and conducting an experiment to map noise levels on the University of New Mexico campus. (2) A comparison of negative surveys that operate on continuous data to random data perturbation. (3) A section detailing how the original data distribution affects our privacy and utility metrics. (4) A section using the Kronecker technique to confirm the correctness of our metrics in terms of the variance and covariance of MDNS. (5) Increased detail about the dimensional adjustment technique, including an explanation of how errors can be magnified with negative surveys, and a proof that dimensional adjustment always improves utility. Finally, we have expanded the literature review of earlier.

There is a growing body of work developing techniques for aggregating data that have been encrypted using homomorphic functions [18,19], which allows a user to calculate some aggregate values (e.g., summation) using the encrypted values. Encryption is energy intensive, which can limit its applicability to resource-constrained devices, and only some aggregate functions can be computed using this approach. A slightly different approach to privacy-preserving data aggregation is taken in SMART [20], which slices individual data, sending the slices through the aggregation network, and reassembling the data pieces later. However, SMART requires the availability of neighboring peers, which may not be available in our setting.

In participatory sensing applications, data points are often tagged with location information, and a rich set of location-based privacy and anonymity rules has been developed for this situation [28,29]. These schemes, however, typically hide or perturb single-dimensional continuous location information; see, for example, Horey et al. [4]. For multidimensional data, privacy-preservation involves trade-offs among accuracy (or information completeness), computational complexity, and the level of anonymity. For example, Aggarwal et al. [7] shows the curse of high dimensionality for k-anonymization in data mining, even if $k = 2$.

Dwork et al. [30] introduced the term *pan-private* in the context of streaming algorithms which can protect the state of information inside a node. This is useful for protecting against node capture attacks that examine internal data. However, it assumes a secure stream as a precondition of the algorithm. In contrast, the work reported here protects the stream of information in transit. Pan-private algorithms, however, are preferable for complex aggregates such as the t-incidence items, the t-cropped mean, and the fraction of k-heavy hitters [30].

Differential privacy [31] aims to provide the maximal accuracy of responses for users querying a statistical database, while minimizing the ability of these users to identify records in the database. Differential privacy assumes that a trusted server handles and responds to the queries, while our approach does not assume that the server is trustworthy.

2.1. Randomized response techniques and negative surveys

Randomized response techniques (RRTs) disguise data by perturbing a categorical value to another value. For example, in a survey of ethnicity if a participant is Hispanic, the response could be randomly perturbed to new value, such as Asian. A *perturbation matrix*, denoted M, gives the probabilities of perturbing category i to category j. It is an $\alpha \times \alpha$ square matrix, in which each entry $M_{i,j}$ is the probability of responding with category j when category i is detected.

Finding the optimal M that balances privacy and utility has been the subject of earlier research [32,33]. Warner described the RRT for binary data [34]; however, it can be extended to categorical data [35] using the following perturbation matrix, which gives an initial suggestion for M:

$$M = \begin{pmatrix} p & \frac{1-p}{\alpha-1} & \cdots \\ \frac{1-p}{\alpha-1} & p & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}, \tag{1}$$

where p is the probability that a category remains unchanged. Similar schemes such as the uniform perturbation (UP) [35], and the framework for high-accuracy privacy-preserving mining (FRAPP) [33] matrices perform similarly to the Warner scheme [32].

The original data are estimated from the disguised data with the following equation:

$$\widehat{A} = M^{-1}\widehat{Y}, \tag{2}$$

where $\widehat{Y} = (Y_1, \ldots, \underset{\frown}{Y_\alpha})^\tau$ and Y_i is the number of disguised values in the ith category. Since this is an unbiased maximum likelihood estimate, $\widehat{A}$ approaches the original distribution as the population size increases. Eq. (2) is known as the matrix inversion approach. An iterative approach is given by Agrawal et al. [35] but is not developed for multiple dimensions.

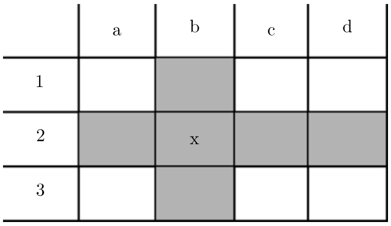
We review a special case of the Warner scheme, called negative surveys [4,36,37], which uses a perturbation matrix containing zeros on the diagonal entries and equal values everywhere else, i.e., $p = 0$ in Eq. (1). We call these matrices *negative survey perturbation matrices (NSPMs)*.

A negative survey applied to WSNs consists of two protocols. The *node protocol* maps sensed data to its negative representation. Each node chooses a category it did not sense with uniform probability and returns that "negative" information to the base station. The *base station protocol* reconstructs the original data at the base station. Instead of Eq. (2), the following simpler equation [36] can be used:

$$A_i = N - (\alpha - 1)Y_i, \tag{3}$$

where A_i is the reconstructed number of values in category i, and Y_i is the reported perturbed number of values in category i, with $1 \leq i \leq \alpha$. N is the total number of sensed values. Eq. (3) has time complexity $O(\alpha)$, compared to $O(\alpha^2)$ for Eq. (2) (ignoring matrix inversion), while still remaining an unbiased maximum likelihood estimate.

Negative surveys [36] are inspired by the human immune system's method of detecting non-self, and are closely related to *negative databases* (NDBs) [38–40]. NDBs are an alternative representation of information that store the set complement



	a	b	c	d
1		▨		
2	▨	x	▨	▨
3		▨		

Fig. 1. Example of positive and negative multidimensional space for two dimensions. A sensor that reads $\langle 2, b\rangle$ from its environment selects among the white cells for a value to report to the base station.

of data instead of the actual data. Both negative surveys and NDBs store or report data in this way, yet NDBs differ because the information is stored in a compressed form that adds additional security. It is provably NP-complete, by a reduction to 3SAT, to try and extract the positive database from the NDB. Negative surveys, unlike NDBs, do not store the entire set of the strings representing the data complement.

Gaussian negative surveys (GNSs) [13] also reduce the number of participants needed for accurate negative survey reconstruction. Xie et al. propose a special perturbation matrix in which each column is represented as a Gaussian distribution with the mean centered over the original category, which is represented as zero. With location data, this perturbs an individual's location a Gaussian random distance away from the original location. This special perturbation matrix eliminates the need for reconstruction at the base station. However, GNSs with location data do not protect privacy as well as traditional negative surveys do. The privacy guarantee of an individual participant depends on the variance of the Gaussian distributions in the perturbation matrix. This variance must be small enough to maintain an acceptable level of utility and number of participants; however, smaller values do not perturb a location a sufficient amount of distance. This may make it easier for an adversary to determine the general location of an individual participant. It is not until the variance is increased to cover more than the entire column of the perturbation matrix that GNSs approach the same privacy guarantee as traditional negative surveys.

Quercia et al. [41] propose a randomized response technique similar to our scheme. Instead of perturbing a location to a different location, each location is perturbed to a yes or no bit with a probability that includes whether a participant is at that location. For each sample, a bit vector that depends on the size of all locations, $O(\mathit{number\ of\ locations})$, is transmitted, while our technique sends a smaller vector, $O(\mathit{log}(\mathit{number\ of\ locations}))$.

MDNSs could use a scheme similar to that of Horey et al. [42] in which locations are nested inside other locations. Each level would represent a dimension. In Horey et al., levels were divided into four categories; however, MDNSs can use an arbitrary and varying number of levels. This can increase the total coverage in the cell phone simulation from a city to a metropolitan area.

3. Protocols

Before describing the multidimensional node and base station protocols, we introduce some notation. The collection of participatory sensing application users is known as the *population*. For the entire population, X, Y, and A are D-dimensional matrices which represent the counts of the categories of the original, disguised, and reconstructed data sets, respectively. For example, if $D = 3$ (i.e., we are sensing three different environmental variables), then $X(a, b, c)$, $Y(a, b, c)$, and $A(a, b, c)$ are counts of the number of times the ath, bth, and cth category appear in a data set. Vectors such as $\vec{x} = \langle a, b, c\rangle$ indicate a specific index into either X, Y, or A. An individual participant senses vector $\vec{x}^+ = \langle x_1^+, x_2^+, \ldots, x_D^+\rangle$ from its environment. Sensed values that are real numbers are quantized into categories, if necessary. Each $x_i^+ \in \vec{x}^+$, where $1 \le i \le D$, expresses that category x_i was sensed in dimension i. x_i is drawn from a set of categories $C_i = 1, 2, \ldots, \alpha_i$ that forms a proper partition over the data in dimension i, and α_i is the total number of categories for dimension i. The "+" in $\vec{x}^+$ denotes the positive or sensed categorical information, as opposed to the negated or perturbed information represented as $\vec{x}^-$. Subscripts in $\vec{x}_i$ denote the dimension (the ith dimension), while superscripts in $\vec{x}^1$ denote an instance of $\vec{x}$.

3.1. Node protocol

There are three phases to the node protocol.

1. *Sensing:* A node senses a multidimensional value $\vec{x}^+$ from its environment and quantizes it into categories if necessary.
2. *Negation:* For each $x_i^+ \in \vec{x}^+$, the node selects uniformly at random a category x_i^- to report to the base station from the set $\{C_i - \{x_i\}\}$, where "-" denotes set difference. Hence, $x_i^- \neq x_i^+$. It does this for each dimension, creating the perturbed vector $\vec{x}^-$. The probability of selecting a category in dimension i is $\frac{1}{\alpha_i - 1}$, where α_i is the number of categories in dimension i. For example, in Fig. 1, a node has sensed $\vec{x}^+ = \langle 2, b\rangle$ from its environment, and must choose among the white cells, for instance $\vec{x}^- = \langle 3, c\rangle$, for a negative value to report back to the base station.
3. *Transmission:* After negation, the node sends $\vec{x}^-$ to the base station either immediately, when queried, or according to another protocol. For this paper, we assume no data aggregation in the network.

Since the number of bits that is required to transmit either the positive or negative data is identical, there is only a small increase in resource cost to compute and transmit the perturbed value, due to selecting a random category to report. Hence, the node protocol saves resources compared to encryption methods that spend energy encrypting data as well as on key distribution and management [4].

3.2. Base station protocol

The base station protocol first collects the reported data, Y, and then estimates the original distributions of sensed values, A, with a reconstruction algorithm. Since this protocol is straightforward, we focus on the reconstruction algorithm in the following. First, we introduce a natural multidimensional extension to the single-dimensional equation, then present a time optimization, and finally give an algorithmic simplification.

The reconstruction algorithm is not trivial. A naive approach would reconstruct each dimension independently. However, this would reveal no information about the joint distribution between dimensions. For example, in Fig. 1, the number of reported 2s and bs would be known individually, but the number of reports of $\langle 2, b\rangle$ would be unknown.

3.2.1. Natural extension

Single-dimensional negative surveys reconstruct their data with Eq. (3) [4,37]. A natural extension to D dimensions is given as

$$\forall\vec{x} \mid A(\vec{x}) = N + \sum_{k=1}^{D}(-1)^k \cdot \Gamma(\vec{x}, k), \tag{4}$$

where $\Gamma(\vec{x}, k)$ is given as

$$\Gamma(\vec{x}, k) = \sum_{\substack{d\in \\ B(\{1,\ldots,D\},k)}} \left(\left[\prod_{j\in d}(\alpha_j - 1)\right] \cdot \sum_{\substack{\vec{y}\ s.t. \\ y_i \in \vec{x}, \\ \forall i\in d}} Y(\vec{y}) \right), \tag{5}$$

and $B(\{1, \ldots, D\}, k)$ is all the k-length possible combinations of members of $\{1, \ldots, D\}$. For example, $B(\{1, 2, 3\}, 2)$ is $\{\{1, 2\},\{1, 3\},\{2, 3\}\}$. $Y(\vec{y})$ is the count of the reported disguised sensed values that have categories specified by d from $\vec{x}$. Each dimension must use an NSPM. As an example, Eq. (4) with $D = 3$ is given as

$$\begin{aligned}\forall a, b, c \mid A(a, b, c) = {} & \sum_{\vec{x}} Y(\vec{x}) - (\alpha_1 - 1) \sum_{\substack{\vec{x}\ s.t. \\ x_1=a}} Y(\vec{x}) - (\alpha_2 - 1) \sum_{\substack{\vec{x}\ s.t. \\ x_2=b}} Y(\vec{x}) \\ & - (\alpha_3 - 1) \sum_{\substack{\vec{x}\ s.t. \\ y_x=c}} Y(\vec{x}) + (\alpha_1 - 1)(\alpha_2 - 1) \sum_{\substack{\vec{x}\ s.t \\ x_1=a, \\ x_2=b}} Y(\vec{x}) + (\alpha_1 - 1)(\alpha_3 - 1) \sum_{\substack{\vec{x}\ s.t., \\ x_1=a, \\ x_3=c}} Y(\vec{x}) \\ & + (\alpha_2 - 1)(\alpha_3 - 1) \sum_{\substack{\vec{x}\ s.t. \\ x_2=b, \\ x_3=c}} Y(\vec{x}) - (\alpha_1 - 1)(\alpha_2 - 1)(\alpha_3 - 1) \sum_{\substack{\vec{x}\ s.t. \\ x_1=a, \\ x_2=b, \\ x_3=c}} Y(\vec{x}).\end{aligned} \tag{6}$$

The time complexity of Eq. (4) is given as

$$O\left(\left[\prod_{i=1}^{D}\alpha_i\right] \cdot \left[\sum_{i=1}^{D}\binom{D}{i}(D - i)\alpha_{max}\right]\right), \tag{7}$$

where α_{max} is the maximum number of categories among the dimensions. There are in total $\sum_{i=1}^{D}\binom{D}{i}$ Y terms that require $(D - i)$ calculations. α_{max} guarantees that enough calculations are accounted for. Since $\binom{D}{0} + \binom{D}{1} + \cdots + \binom{D}{D} = 2^D$, this equation is exponential with respect to the number of dimensions.

3.2.2. Time optimization

We can optimize the reconstruction algorithm shown in Eq. (4) with a more efficient algorithm, presented in Algorithm 1. This algorithm uses memoization to improve the running time and can generalize to any perturbation matrix, not just an NSPM. The inputs to Algorithm 1 are D, the number of dimensions; Y, the D-dimensional matrix of reported disguised values; $F = [\alpha_1, \ldots, \alpha_D]$, a list of the number of categories for each dimension; and $M = [M_1, \ldots, M_D]$, the perturbation matrices for each dimension. The symbol ":" denotes a slice operator, an operation on a matrix designating every element in the dimension which it appears; τ is a function similar to transpose that takes a row, column, hyper-row, or hyper-column,

Algorithm 1 Reconstruction Optimization for D Dimensions

```
1:  function RECONSTRUCT_MATRIX($Y, D, F, M$)
2:      $R = Y$
3:      for $\delta \in [1 : D]$ do
4:          update_dim($R, D, [], \delta, F, M$)
5:      end for
6:      return $R$
7:  end function
8:
9:  function UPDATE_DIM($R, D, index, \delta, F, M$)
10:     if $length(index) = D$ then
11:         $R(index) \leftarrow M_\delta^{-1} * R(index)^\tau$
12:     else if $len(index) + 1 = \delta$ then
13:         $new\_index \leftarrow index.append([:])$
14:         update_dim($R, D, new\_index, \delta, F, M$)
15:     else
16:         for $i \in [1 : F(length(index) + 1)]$ do
17:             $new\_index \leftarrow index.append([i])$
18:             update_dim($R, D, new\_index, \delta, F, M$)
19:         end for
20:     end if
21: end function
```

and transforms it into a vector appropriate for matrix multiplication. *index* is constructed to be a vector of length D, with a single instance of ":". When used as an index into R, it returns a vector.

The time complexity of Algorithm 1 is

$$O\left(\sum_{i=1}^{D}\left[\prod_{j=1, j\neq i}^{D} \alpha_i^2 \alpha_j\right]\right) = O\left(\sum_{i=1}^{D} \alpha_i^2 \cdot \prod_{i=1}^{D} \alpha_i\right), \tag{8}$$

ignoring the cost of matrix inversion for each M_δ. Intuitively, this complexity is based on a matrix multiplication with every possible vector in Y. Each update of R from Line 11 in Algorithm 1 stores information back in R for other overlapping vectors to use, thus reducing the total amount of computation. If the NSPM is used for each dimensional perturbation matrix, the cost of Algorithm 1 reduces to

$$O\left(D \cdot \prod_{i=1}^{D} \alpha_i\right), \tag{9}$$

because Line 11 in Algorithm 1 is replaced with the simpler Eq. (3). Eq. (9) is clearly an improvement over Eq. (7) and arises because parts of Eq. (4) are cached in matrix R each time a vector in R is updated (Line 11).

3.2.3. Algorithmic simplification

The *Kronecker technique* converts a multidimensional negative survey to a single dimension. Although the complexity cost is greater than that of Eq. (9), this technique simplifies the software engineering of MDNSs and allows the use of single-dimensional metrics that obtain the same values as their multidimensional counterparts.

The Kronecker technique uses a perturbation matrix, M, that is the *Kronecker product* [43] of the individual perturbation matrices for each dimension, given as

$$M' = (((M_1 \otimes M_2) \otimes M_3) \cdots \otimes M_D), \tag{10}$$

where $\otimes$ is the Kronecker product operator. The Kronecker product of two matrices is the tensor product with respect to a standard choice of basis. Y is transformed into a new vector Y', an $n \times 1$ vector, where n is the product of the number of categories in each dimension. For example, if Y has three dimensions with four, three, and two categories each, Y' is given as

$$Y' = \begin{bmatrix} Y(1,1,1) \\ Y(1,1,2) \\ Y(1,2,1) \\ Y(1,2,2) \\ Y(1,3,1) \\ \vdots \\ Y(4,3,2) \end{bmatrix}. \tag{11}$$

To obtain the estimated distribution, A, Y is multiplied with $\left(M'\right)^{-1}$ according to Eq. (2). A is then transformed in an inverse manner to how Y' is obtained. Care must be taken for the order of transformation to correspond with the correct order in which the Kronecker products were applied. This technique, however, is not optimal, because the time complexity is given as

$$O\left(\left[\prod_{i=1}^{D}\alpha_i\right]^2\right), \tag{12}$$

ignoring matrix inversion, which is minimal because of the mixed-product property [43].

4. Privacy and utility metrics

Using privacy and utility metrics extended from Huang and Du [32], we quantify the trade-offs between the accuracy of the reconstruction and the level of privacy. Like their single-dimensional counterpart, the multidimensional formulations apply to any perturbation matrix, not necessarily an NSPM. The privacy metric ranges from 0 to 1, while the utility metric ranges from 0 to infinity. For both metrics lower values are desirable.

These metrics and related terminology are borrowed from the privacy-preserving data mining (PPDM) field. WSNs are similar to PPDM for the following reasons. (1) Sensor nodes contain many streams of data and have multiple environmental sensors, making them well suited for multiple dimensions. (2) Participatory sensing applications that work with people are naturally sensitive. We use the terms accuracy, reconstruction error, and utility interchangeably, and similarly the terms disguise, perturb, and negate are used interchangeably.

4.1. Privacy metric

The privacy metric measures the probability of guessing the original data from the disguised values, and is based on the *maximum a posteriori* (MAP) estimate. Huang and Du [32] theorize that the MAP estimate is the "best that adversaries can achieve when their estimation is consistent", and it gives an upper bound on an adversary's threat. We extend their single-dimensional metric to multiple dimensions as follows:

$$Privacy = \sum_{\substack{\Upsilon \in Y(\vec{x}) \\ \forall \vec{x}}} P(\Upsilon | \widehat{X_\chi}) \cdot P(\widehat{X_\chi}), \tag{13}$$

where

$$\widehat{X_\chi} = \arg \max_{\substack{\chi \in X(\vec{x}) \\ \forall \vec{x}}} P(\chi | Y). \tag{14}$$

Eq. (14) calculates for Eq. (13) the optimal MAP estimate for a given index of Y. We report privacy using the index that gives the maximum probability in $P(X|Y)$ (the maximum index in each column of $P(X|Y)$).

If an adversary has no prior knowledge of the underlying distribution, we propose that privacy generalizes to k-indistinguishability. We define an item to be k-indistinguishable if it cannot be identified with higher probability than guessing from $k-1$ other items. A participant's reported data in a single-dimensional negative survey with α categories has a k-indistinguishability value of $\alpha-1$. An individual's data in a multidimensional negative survey with categories $\alpha_1, \alpha_2, \cdots, \alpha_D$ will have a k-indistinguishability value of $(\alpha_1 - 1) \cdot (\alpha_2 - 1) \cdot \cdots \cdot (\alpha_D - 1)$. This is different from k-anonymity in WSNs [6,9,10,44,45], which preserves location information and measures the ability of an adversary to distinguish a participant from a set of $k-1$ nearby participants.

4.2. Utility metric

Utility, also known as accuracy or reconstruction error, measures the difference between the original, X, and reconstructed, A, data distributions. We use the following reasoning from Huang and Du [32]. Since A is an unbiased maximum likelihood estimate of X, the mean of the estimate A is identical to the original distribution X. Yet, each specific estimate A deviates from X by some amount. The closer A is to X, the higher A's utility. Hence, the mean square error (MSE), given as follows, is used to quantify utility:

$$MSE = E[(A - X)^2]. \tag{15}$$

Huang and Du [32] actualized this equation by replacing X with the mean of A to estimate A's variance. Using Eq. (2), they equate the variance of A and $M^{-1}Y$, and state a theorem to compute the MSE. We extend this theorem to multiple dimensions with the following equation:

$$\begin{aligned} Utility &= \frac{1}{\alpha_1 \cdot \cdots \cdot \alpha_D} \sum_{\vec{x}^i} MSE(X = \vec{x}^i) \\ &= \frac{1}{\alpha_1 \cdot \cdots \cdot \alpha_D} \sum_{\vec{x}^i} E[(P(A = \vec{x}^i) - P(X = \vec{x}^i))^2] \\ &= \frac{1}{\alpha_1 \cdot \cdots \cdot \alpha_D} \sum_{\vec{x}^i} \left(\sum_{\vec{x}^j} \left[\mu(\vec{x}^i, \vec{x}^j)^2 \cdot var(\vec{x}^j) \right] + \sum_{\substack{\vec{x}^k, \vec{x}^l \text{ s.t.} \\ \vec{x}^k_\gamma \neq \vec{x}^l_\gamma, \forall \gamma}} \left[2 \cdot \mu(\vec{x}^i, \vec{x}^k) \cdot \mu(\vec{x}^i, \vec{x}^l) \cdot cov(\vec{x}^k, \vec{x}^l) \right] \right), \end{aligned} \tag{16}$$

where

$$\mu(\vec{x}^m, \vec{x}^n) = \prod_{d=1}^{D} M_d^{-1}(\vec{x}^m_d, \vec{x}^n_d) \tag{17}$$

denotes the product of the elements from the inverse of the perturbation matrix for each dimension where the row and column correspond to the categories in the dth dimension of $\vec{x}^m$ and $\vec{x}^n$ respectively. var and cov are given as

$$\begin{aligned} var(\vec{x}^i) &= \frac{1}{N} \cdot P(Y = \vec{x}^i) \cdot (1 - P(Y = \vec{x}^i)) \\ cov(\vec{x}^i, \vec{x}^j) &= -\frac{1}{N} \cdot P(Y = \vec{x}^i) \cdot P(Y = \vec{x}^j), \end{aligned} \tag{18}$$

which are very similar to the actual variance and covariance of an MDNS, which are given as

$$\begin{aligned} var_{MDSN}(\vec{x}^i) &= \frac{\left(\left[\prod_{i=1}^{D} \alpha_i \right] - 1 \right)^2}{N-1} \cdot P(Y = \vec{x}^i) \cdot (1 - P(Y = \vec{x}^i)) \\ cov_{MDSN}(\vec{x}^i, \vec{x}^j) &= -\frac{\left(\left[\prod_{i=1}^{D} \alpha_i \right] - 1 \right)^2}{N-1} \cdot P(Y = \vec{x}^i) \cdot P(Y = \vec{x}^j). \end{aligned} \tag{19}$$

4.3. Experimental study of trade-offs between privacy and utility

The underlying distribution, X, affects utility and privacy. We use the following normalized version of Shannon's entropy to illustrate the effects:

$$S = \frac{-\sum_{\vec{x}} P(X = \vec{x}) \log P(X = \vec{x})}{\log \left(\prod_{i=1}^{D} \alpha_i \right)}, \tag{20}$$

where S is in [0, 1]. For example, a spiked distribution (all elements in one category) has the lowest normalized entropy, $S = 0$, and provides the worst privacy, but the highest utility. A uniform distribution, which has the highest normalized entropy, $S = 1$, provides the worst utility, but the best privacy. All other distributions fall between these two extremes. However, the underlying distribution affects privacy significantly more than utility. For example, the spiked and uniform distributions span 87.4% of the entire privacy metric. These two distributions span a significantly smaller range of utility, 0.786 to 0.802, which corresponds to 1.6% of the privacy metric. Since this effect on utility is so small, Groat et al. [46] interpreted utility to be independent of the underlying distribution. This is a reasonable simplification because the number of categories and the number of participants have such a dominant effect on the metric's value. This allows WSN designers to determine the utility of negative surveys without any knowledge of the distribution of data they are trying to collect.

5. Dimensional adjustment improves efficiency

In this section, we introduce a technique called *dimensional adjustment (DA)*, which reduces the number of participants required to maintain reasonable utility level. It accomplishes this by increasing the number of dimensions, yet it maintains the overall number of categories. This addresses the magnification of errors that previous work with single-dimensional data [4,13] observed. A significant increase in the number of samples or participants is needed to maintain a reasonable reconstruction accuracy as the number of categories increases. This limitation is compounded with multiple dimensions and has limited negative surveys to applications with a small number of categories. In this section, we propose dimensional adjustment to address this challenge, discuss the privacy and utility trade-offs, present a novel explanation for the magnification of errors, and illustrate that it will always improve utility.

Table 1
Two negative surveys of 10,000 total categories and 1000,000 participants. The second uses dimensional adjustment.

	One dimension of 10,000 categories	Six dimensions of 5 × 5 × 5 × 5 × 4 × 4 categories
Utility	0.00100	0.00014
Privacy	0.01457	0.01960

5.1. How dimensional adjustment operates

DA increases utility by accepting a slight decrease in privacy for a given number of participants. It accomplishes this by distributing the same overall number of categories over an increasing number of dimensions. For example, if an original one-dimensional negative survey contains 64 categories, it can be remapped to 2 dimensions of 8 categories each; 2 dimensions of 4 and 16 categories; or any number of dimensions where the product of the number of categories in each dimension equals 64. Remapping dimensions is easy to implement, and is similar to base conversion with variable bases.

Splitting data into multiple dimensions with a smaller number of categories for each dimension improves the reconstruction accuracy (utility). Having fewer dimensions with a larger number of categories makes the utility worse (higher utility value). Intuitively, accuracy is related to Fig. 1 and the ratio of the white squares (negative information) to the total number of squares. As the number of dimensions grows, and the number of distinct categories remains the same, this ratio decreases, reducing the possible number of cells for perturbed data, which increases the accuracy of reconstruction. The next section discusses these trade-offs.

5.2. Trade-off analysis

Using dimensional adjustment to transform a low-dimensional survey into a high-dimensional survey does not occur without trade-offs. For example, a one-dimensional negative survey with 64 categories provides the best privacy but the worst utility. However, having the same data transformed to having six dimensions with two categories each provides the worst privacy but the best utility. The relationship between privacy and utility is usually nonlinear, providing an opportunity to sacrifice a small amount of one for a significant gain in the other. For example, in Table 1 with 1000,000 samples and 10,000 categories, we see that privacy degrades 34% while utility improves 86%.

Using Table 1 and modeling equations for privacy and utility, we further illustrate these trade-offs. Without loss of generality, the normal distribution is used as the original distribution, X. We can use a simple linear model to estimate the utility of a single-dimensional negative survey given the number of participants N and the number of categories α:

$$Utility_{model} = (\alpha - 2)/N. \tag{21}$$

Eq. (21) has an R^2 value of 0.9999. Similarly, using the values from Table 1, we can model privacy given the number of categories in a single-dimensional survey.

$$Privacy_{model} = \frac{2.5}{(\log_2(\alpha))^2 + 1.5}, \tag{22}$$

which has an R^2 value of 0.976.

Using the two models in Eqs. (21) and (22), we can estimate the population size, or number of categories needed to achieve the same metrics that the multidimensional negative survey achieves in Table 1. Eq. (21) indicates that a single-dimensional survey of 10,000 categories and 71,414,286 participants is equivalent to an MDNS using DA of 10,000 categories with 1000,000 participants. The equation also indicates that a single-dimensional negative survey of 10,000 categories and 1000,000 participants is equivalent to an MDNS using DA of 142 categories and 1000,000 participants. Dimensional adjustment can reconstruct more categories with fewer data than the equivalent one-dimensional survey. Similarly, Eq. (22) indicates that to achieve better privacy than the multidimensional survey with a single-dimensional survey we would need at least 2397 categories.

Using this privacy and utility modeling, we can further illustrate the trade-offs in DA. Privacy degrades when the number of categories is reduced because there is less anonymity among fewer categories. Utility, however, increases when the number of categories decreases. We see above a degradation of privacy from 10,000 to 2397 categories, but an improvement of utility from 10,000 to 142 categories. This is a degradation of 70.0% for privacy, but an improvement of 98.58% for utility. These percentages will not linearly correlate with the privacy and utility metrics, but they do show how the privacy–utility trade-off is favorable for dimensional adjustment.

5.3. Magnification of errors

Horey et al. [4] use extensive simulation to suggest there is an almost linear increase in the number of participants required to maintain a given utility as the number of categories increases when utility is measured with the relative root

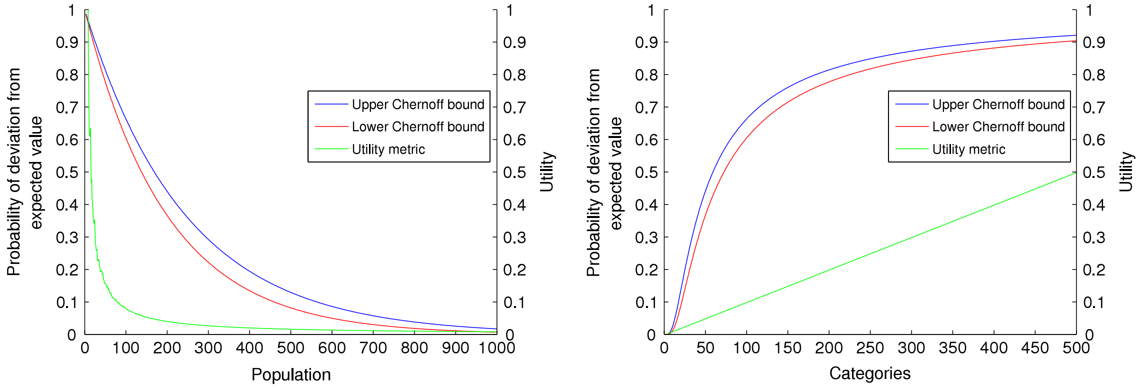


Fig. 2. Chernoff upper and lower bounds showing the probability that the bins in the Y data set deviated one standard deviation from their expected value. Categories are fixed (left) at 10 and population is fixed (right) at 1000. The utility metric is included (green), yet it does not model the variations in the disguised data, Y, as well. Note that lower utility values are more desirable.

mean square error (RMSE). When utility is measured using the more common MSE, as illustrated in Fig. 2, the relation appears linear. However, these metrics are misleading, because they do not model how the disguised distribution, Y, deviates from its expected value. Xie et al. [13] suggest that this magnification of errors arises from the discrete nature of integer counts. Our reconstruction algorithm assumes that the expected disguised counts are real. In the worst case where all rounding discrepancies are the same direction, this bounds the error for any category by $\pm O(\alpha^2)$ and the overall utility by $O(\alpha^3)$. These two results suggest that utility will increase linearly in practice, and at worst cubicly.

We introduce an alternative analysis using Chernoff bounds which more precisely explains deviation in the expected utility. These bounds express the probability that the counts in Y differ from their expectation by a specified quantity; we use one standard deviation in the following analysis for clarity. Because reconstruction of A depends only on Y, these bounds can be directly applied to utility. This analysis is more rigorous than the simulations provided in [4], and more realistic and general than the worst case provided in [13].

Counts in Y that are closer to their expected values give a better reconstructed distribution, A, that is closer to the original data set, X. Since negative surveys are similar to the balls and bins problem [47], its notation (balls are sensed values and bins are categories) will be used for the rest of this section. In a negative survey with an original distribution, X, the balls in category X_i must be distributed among the other $\alpha - 1$ bins. This is a series of Bernoulli trials, which can be represented as a binomial distribution. Chernoff bounds approximate a generalization of the binomial distribution and are good at representing their tails far from the mean. The expected number of balls in the disguised bins, Y, is calculated by first taking the inverse function of Eq. (3) as follows:

$$E[Y_i] = \frac{N - A_i}{\alpha - 1}. \tag{23}$$

Since this is a maximum likelihood estimate, A_i can be replaced with X_i.

The Chernoff upper and lower bounds, which determine the probability that a bin will be filled with δ more or fewer balls than the expected value, are represented as follows:

$$\begin{aligned} P[X_i > E[Y] + \delta] &= \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\frac{N-X_i}{\alpha-1}} \\ P[X_i < E[Y] - \delta] &= \left(\frac{e^{\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\frac{N-X_i}{\alpha-1}}. \end{aligned} \tag{24}$$

Without loss of generality (and for graphing), δ is fixed at one normalized standard deviation of the binomial distribution, given as

$$\delta = \sqrt{\left(\frac{1}{\alpha}\right) \cdot \left(1 - \frac{1}{\alpha}\right)}. \tag{25}$$

Fig. 2 gives the results of Chernoff bounds when the number of categories is fixed at 10 and the population varies, and when the population is fixed at 1000 and the number of categories varies. These two figures use a single-dimensional negative survey, but a multidimensional negative survey would behave similarly where αin Eq. (24) is replaced with $(\alpha_1 - 1) \cdot (\alpha_2 - 1) \cdot \cdots \cdot (\alpha_D - 1)$.

In Fig. 2 (left), when the population increases, the probability of a bin in Y being filled with more balls than a standard deviation from its expected value decreases. If the y-axis is scaled logarithmically, the Chernoff bounds form a straight line, suggesting that an increase in participants exponentially (−0.70 in Fig. 2 left) decreases the deviation of Y from its expected value. Fig. 2 (right) shows that, when the number of categories increases, the probability of a bin deviating from its expected value grows almost logarithmically. To maintain constant probability of this deviation, the population needs to increase as follows:

$$N_{new} = \alpha_{increased} \cdot \log(N), \tag{26}$$

where $\alpha_{increased}$ is the increase in number of categories, and N is the original population. This means that the needed number of participants increases linearly with the number of categories. Any base can be used for the log. However, lower values give lower Chernoff bound probabilities. Although the probability of deviation from the expected value grows almost logarithmically with an increase in number of categories, the initial increase (from 3 to 150 categories) is significant. This significance could explain the magnification of errors associated with negative surveys.

5.4. Dimensional adjustment always improves utility

In this section we argue that DA always improves utility. Assuming that no negative survey has fewer than three categories,[2] the limit of the Chernoff upper bound as α goes to three from the right is given below:

$$\lim_{x\to 3^+} \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\frac{N-X_i}{\alpha-1}} = \left(e^{\delta}(\delta+1)^{(-\delta-1)}\right)^{\frac{N-X_i}{2}}. \tag{27}$$

The derivative of the Chernoff upper bound is given as

$$\frac{\partial}{\partial\alpha}\left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\frac{N-X_i}{\alpha-1}} = -\frac{\log(e^{\delta}(\delta+1)^{-\delta-1})(N-X_i)(e^{d}(\delta+1)^{-\delta-1})^{\frac{N-X_i}{\alpha-1}}}{(\alpha-1)^2}, \tag{28}$$

which is always positive when α, N, δ, and X_i are greater than zero, and $\delta < 1$ and $X_i < N$. The lower bound behaves similarly. Eq. (28) means that, as α increases, the Chernoff bound is monotonically increasing. Hence, any α lower than another will always have a lower probability. A lower probability means that the bins are closer to their expected value. Since dimensional adjustment always reduces the overall number of categories, it will always tighten the distribution of values for each bin in Y, i.e., the bins will be closer to their expected amount. This improves the reconstructed distribution (A values closer to X), which improves utility.

Additionally, as the population goes to infinity, the limit of the Chernoff bound approaches 0, as given below:

$$\lim_{N\to\infty} \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\frac{N-X_i}{\alpha-1}} = 0. \tag{29}$$

This illustrates how reconstruction is a maximum likelihood estimate. More participants tightens the bins in Y around the expected values.

6. Simulations and implementations

In this section, we present two MDNS simulations in Matlab, and describe an implementation on Android smart phones. The simulation and implementation illustrate the effectiveness of the previous protocols, how they can be adapted to real-world scenarios, and trade-offs between energy and privacy.

6.1. Cell phone radiation threat detection simulation

Participatory sensing could potentially help detect and locate radiation threats in a city, such as the detonation of a dirty bomb, loss of radioactive material, or spread of radiation from a nuclear reactor accident. In this scenario, we assume that cell phones are equipped with radiation monitors and GPS devices. Locations are quantized into different quadrants, each with a unique label. We also assume that individuals care about the privacy of their locations. We show that with reasonable parameter assumptions (number of locations, number of discernible radiation levels, and participants), multidimensional negative surveys can maintain location confidentiality and identify locations containing radiation threats, if they exist.

Cell phones are ideal for radiation detection, and the United States Department of Homeland Security has considered their use [48]. If radiation sensors were installed at fixed locations, they might be tampered with or adversaries could avoid them, which is more difficult with cell phones because they are owned by many independent individuals. As an incentive to promote participation, aggregate information could be disseminated freely to participants. Since readings from an individual cell phone might not be as accurate as the combined readings from a larger population, access to aggregate information would be advantageous. And, for an event such as the Fukushima Daiichi nuclear accident, participants might prefer to send the unperturbed data and receive more accurate readings. Either way, in such a situation, immediate feedback would be beneficial, especially to determine if radiation has spread further than publicly acknowledged.

[2] A survey with two categories is simply the bitwise inverse of the data and provides no privacy, while a survey with one category is not very interesting.

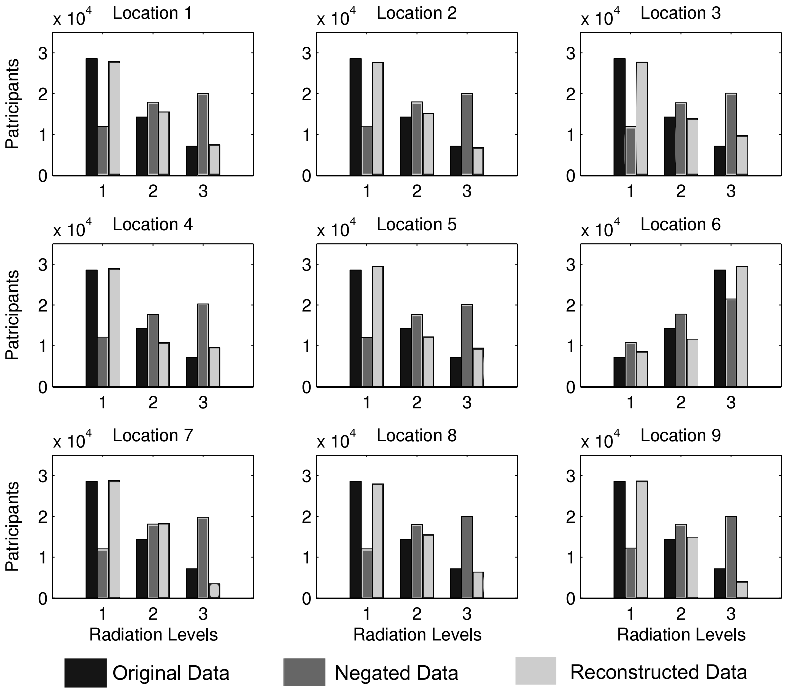


Fig. 3. Histograms for a multidimensional negative survey of nine locations and three radiation levels. The *y*-axis measures the number of participants per level of radiation (the *x*-axis). Location 6 is suspicious since its radiation levels form a threat distribution. The other locations have non-threat distributions.

6.1.1. Simulation setup

Before we explain the simulation setup, we give a small example of a geographic area divided into a 3 × 3 grid, shown in Fig. 3. The total population of cell phones (participants) is 450,000, and it is equally divided among the nine locations. In the actual simulation, we do not assume a uniform population distribution, and instead follow a more realistic model given by Bertaud et al. [49]. We simulate three radiation levels: low, medium, and high. Depending on the level of radiation, each location's distribution of reported levels will be shifted lower or higher. For example, in Fig. 3, location 6 contains a *threat distribution*, illustrated by the black histogram. This distribution, exponentially shifted towards the higher range, contains 28,571 participants in the high radiation level, half that number or 14,286 participants in the medium radiation level, and 7143 in the low level. Benign locations, characterized by the *non-threat distributions*, are shown in black at the other locations. These distributions are distributed in the reverse order, 28,571 participants in the low radiation level, etc. We observe in Fig. 3 that the reconstructed distribution shown in light gray resembles the original distribution enough that important decisions could be made such as where to send response teams.

Negative surveys occasionally generate negative estimates for some categories. This statistical artifact arises when the expected contribution for a particular category exceeds the actual reported total for that category. As the number of samples increases, the number of negative estimates decreases, along with the reconstruction error. If negative values appear in the reconstructed distribution, these can be mapped to zero. If the total number of participants must be consistent, then the negative amount can be deducted in equal amounts from the rest of the categories. For example, if we had a negative survey of ten categories and category one was reconstructed to negative nine, this value could be changed to zero and one value removed from each category between two and ten.

San Francisco, which has roughly 46.7 square land miles, is our example city. We chose the number of distinct locations to be 48, which works well with DA due to its high number of composites. A hexagonal grid was used in which each location covers roughly one square land mile. This size would allow a response team with more powerful equipment (such as helicopters equipped with radiation detectors) to pinpoint the exact location of a threat.

San Francisco has a population of about 815,000. We vary the number of participants from 100,000 to 500,000 in increments of 100,000. The spatial distribution of participants follows a standard urban model taken from Bertaud et al. [49], where the population is concentrated at the central business district, and is gradually reduced further from this center.

Radiation levels were divided into three categories. While experiments show that more categories would increase the granularity of the data, they would not improve accuracy. We found that fewer radiation levels gives more accurate results. However, there is a limit. If there were only two radiation levels, privacy would be lost, and adversaries could determine a user's location, if a threat existed.

Each participant's cell phone, when queried, samples the environment for the radiation level and notes its location. It then perturbs this information according to Section 3.1 and sends the perturbed values to the base station. After the base

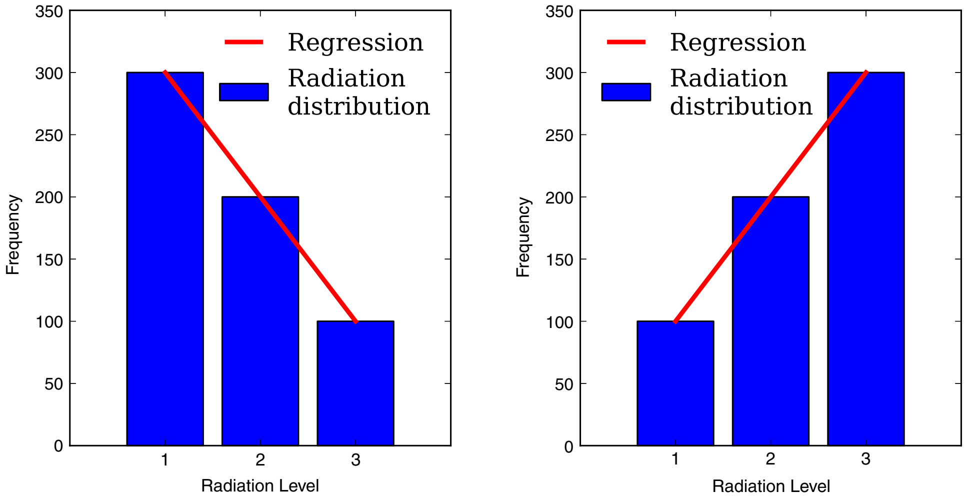


Fig. 4. Example of how linear regression on a histogram can be used to determine threat distributions. A negative slope (left) indicates a non-threat environment with predominately low radiation levels. In contrast, a positive slope (right) indicates a threat environment where there are many reports of high radiation levels.

station collects the perturbed data (one sample from each cell phone), it reconstructs the original distribution according to the protocols from Section 3.2.

The base station determines if a threat exists by computing the linear regression at each location from its reconstructed histogram of radiation levels, assuming that the histogram bins are one unit apart. Fig. 4 illustrates this technique. Ideally, a location reporting elevated radiation levels will have a positive slope from the linear regression, and a location with a non-threat distribution will have a negative slope. The slope thresholds that distinguish threats from non-threats were defined to minimize the overall number of errors. These thresholds could be adjusted to favor one error type over another. For example, one strategy might send response teams to investigate false positives, rather than allowing a false negative to slip through. We chose the thresholds *a posteriori*, but in a real deployment these values would be chosen *a priori*, with additional domain knowledge.

We ran the simulation 1000 times for each increment of participants, assigning the threat distribution to a random location in 500 of the runs. In the other 500 runs we assigned a non-threat distribution to all locations.

6.1.2. Results and analysis

Table 2 summarizes the results, showing the number of false positives and false negatives. Accuracy in the third column is the percentage of true positives that correctly determined the threat location. The average privacy and utility metrics are also shown. Since we are calculating an unbiased maximum likelihood estimate, more participants reduces the number of errors and increases the reconstruction accuracy.

Because accuracy was low for a single dimension, shown in the first four rows of Table 2, we used DA from Section 5. Table 2 shows results with a single location dimension of 48 categories factored into two dimensions of 6 and 8 categories; three dimensions of 4, 4, and 3 categories; and four dimensions of 2, 2, 4, and 3 categories. With four dimensions, we obtained 100% accuracy with 200,000 or more participants.

Radiation monitoring with cell phones would be practical for an example city such as San Francisco, with 46 square land miles and a population of around 900,000. Using the reconstruction Algorithm 1, MDNSs with DA can accurately determine whether a radiation threat exists, and if so, at which location.

6.2. Reconstructing continuous values

In addition to categorical data such as locations and radiation levels, MDNSs can be applied to continuous data such as temperature or humidity. In this subsection, we reconstruct the probability density functions of different underlying distributions and compare the parameters of these distributions to the original parameters.

6.2.1. Simulation setup

Any fixed point number can be represented as a collection of categories by labeling each digit's position (1, 10, 100, ...) with a value ranging from zero to nine. Thus, a fixed-point number with n digits can be treated as an n-dimensional negative survey, with each dimension having ten potential categories; it is then straightforward to apply the protocols presented previously, as illustrated in Algorithm 2.

Table 2
Cell phone radiation detection simulation. Each test was run 1000 times. 500 runs contained a radiation threat in a random location and for the other 500 runs no locations contained a threat.

Number of samples	Number of false negatives	Number of false positives	Accuracy of true positives % (Ratio)	Average privacy	Average utility
One locational dimension with 48 categories					
100,000	246	246	5.5 (14/254)	0.0282	4.54E−04
200,000	244	245	7.8 (20/256)	0.0252	2.27E−04
300,000	244	244	18.0 (26/256)	0.0241	1.51E−04
400,000	241	242	18.9 (49/259)	0.0234	1.13E−04
500,000	238	239	19.9 (52/262)	0.0230	9.08E−05
Two locational dimensions with 8 categories					
100,000	246	248	11.8 (30/254)	0.0350	1.90E−04
200,000	250	251	21.2 (53/250)	0.0319	9.48E−05
300,000	222	222	31.3 (87/278)	0.0307	6.32E−05
400,000	199	199	39.2 (119/301)	0.0300	4.74E−05
500,000	194	194	44.4 (136/306)	0.0296	3.79E−05
Three locational dimensions with 4 × 4 × 3 categories					
100,000	203	205	56.6 (168/297)	0.0586	3.09E−05
200,000	139	139	81.7 (295/361)	0.0554	1.54E−05
300,000	87	87	92.5 (382/413)	0.0542	1.03E−05
400,000	58	58	94.3 (417/442)	0.0535	7.71E−06
500,000	37	37	98.3 (455/463)	0.0531	6.17E−06
Four locational dimensions with 2 × 2 × 4 × 3 categories					
100,000	17	17	99.4 (480/483)	0.1444	4.41E−06
200,000	0	0	100 (500/500)	0.1411	2.20E−06
300,000	0	0	100 (500/500)	0.1398	1.47E−06
400,000	0	0	100 (500/500)	0.1392	1.10E−06
500,000	0	0	100 (500/500)	0.1387	8.81E−07

Algorithm 2 MDNSs on Continuous Data

```
    % Leading zeros can be applied to sensed value $SV$ if needed.
    % || denotes concatenation.
    % “-" denotes set difference.
    % $m_1$ may be chosen from a limited set.
for each numeral $n_i$ in sensed value $SV$, where $SV = n_1||n_2||\dots||n_x$ do
    $m_i \leftarrow$ Select from $\{0, 1, \dots, 9\} - \{n_i\}$ with uniform probability.
end for
return $m_1||m_2||\dots||m_x$
```

We generated values from two probability distributions, rounding each value so that the contained only two and three significant digits. Without loss of generality, we used the following normal and exponential distributions:

$$\mathcal{N}(\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{\sigma}}, \qquad \mathcal{E}(\mu) = \frac{1}{\mu} e^{-\frac{x}{\mu}}. \tag{30}$$

Tests used $\mathcal{N}(500, 100)$ and $\mathcal{E}(100)$, and we truncated the tails of the distribution at 0 and 1000. To perturb a sensed value, a random digit not equal to the actual digit is reported for each position. We varied the number of samples from the two distributions from 1000 to 9 billion in exponential increments.

The base station reconstructs the frequency of each number of the D most significant digits (the probability density function) of the underlying data according to the protocols in Section 3. The parameters from the reconstructed data are determined using a maximum likelihood estimate and then are compared to the original parameters used to construct the data.

6.2.2. Results and analysis

We calculated the difference between the estimated and original parameter and divided by the original parameter. Each data point is an average of 20 runs. Fig. 5 show the results. It suggests that a theoretical maximum accuracy depends on the parameter type, the distribution, and the number of dimensions. All parameters are within 5% of the original parameter values after 200,000 sensed values.

6.2.3. Comparison of MDNSs on continuous data to random data perturbation

Random data perturbation (RDP) algorithms proposed by Agrawal, et al. [50] and later Zhang et al. [51], perturb continuous data by adding noise drawn from a known distribution. The perturbed data are then reconstructed to an approximation of

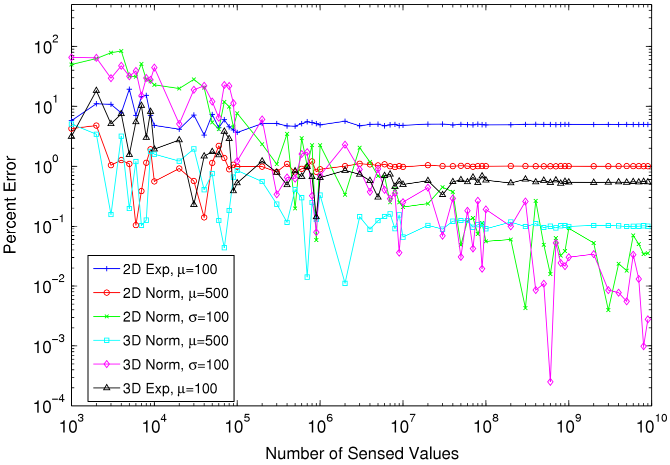


Fig. 5. Parameter reconstruction error from the continuous negative survey simulation measured as a percentage difference from the original parameter.

the original data using an iterative algorithm based on Bayes' theorem. RDP adds a randomized value, r_i, drawn from a known distribution over a finite range, to a datum, x_i. Reconstruction uses an expectation maximization (EM) algorithm [52] that provably converges to the maximum likelihood estimate of the original distribution [53]. RDP techniques assume that the data and the noise are drawn from a continuous domain.

Previous research [4] compared the original negative survey method to a modified categorical RDP technique. However, the accuracy of the comparison was compromised by modifying the original RDP technique for categorical data. Here, we redo the comparison using MDNSs on continuous data to the original RDP technique.

We implemented the original RDP technique proposed by Agrawal and Srikant [50] using the same data granularity as the MDNSs. We used a triangle, plateau, and step distribution as the original data, hoping to capture the ability of RDP and MDNSs on discontinuities. A moving average of size 7 was used on the reconstructed distributions from the negative surveys to help smooth the data. However, with enough participants this is not necessary.

Both MDNS and RDP have strengths and weaknesses. MDNSs require a large number of samples to reconstruct the probability density function (PDF) accurately. For example in Fig. 6 with 100,000,000 samples, the root mean square error (RMSE) of RDP for the triangle distribution was 419,732.8, while with MDNS it was 227,941.2, a 45.7% improvement. For the plateau distribution with the same number of participants, the RMSE with RDP was 63,521.3, while with MDNS it was 24,153.6, a 62.0% improvement. However, with 100,000 samples, MDNSs could not reconstruct the PDFs enough to distinguish between the two different distributions. However, MDNSs handle discontinuous probability density functions (PDFs) and PDFs whose derivatives are discontinuous better than RDP, as illustrated in Fig. 7. In addition, RDP's stopping criterion is problematic [50], and it takes a considerably longer time to run than MDNSs.

6.3. Implementation on physical devices

We implemented the MDNS node protocol as an Android smart phone application and performed two experiments. For both experiments, location information was obtained through GPS satellites only. Although this consumes more energy than other methods such as multilateration of radio signals between cellular towers, we required precise location information because experiments were performed in a small area on the University of New Mexico (UNM) campus. GPS information was represented as categories of longitude and latitude. Volume levels from the phone's microphone were divided into three categories for the third dimension. The phones sampled the volume level and its location, perturbed this information, and sent it back to the base station, which was implemented in Java on a Computer Science Department server.

We performed two experiments. The first was designed to test the energy cost of communicating data with and without encryption. We used the Secure Socket Layer (SSL) because it is popular and easily implemented on the smart phones. Messages were sent using SSL until the battery was drained to half its level. We then recharged the phone and the experiment was repeated without SSL. Without affecting the results, the phone stayed at the same location. To maintain an equal test bed, all other non-essential applications were regularly closed every 15 min and experiments were performed in the middle of the night to prevent unwanted calls or text messages.

We were able to send 8612 (16.22% more) messages without encryption than the 7401 messages with SLL. It took 159.2 min to send these message with SSL, while it took 143.6 min with no encryption. This means that approximately 4.836 more milliseconds was spent per message encrypting. While this is not a formal detailed verification of energy use, it does illustrate how encryption uses extra energy.

The second experiment used MDNSs to identify the noisiest locations on the University of New Mexico campus. Latitude and longitude were divided into four and six categories respectively for a total of 24 different locations. Because of the limited

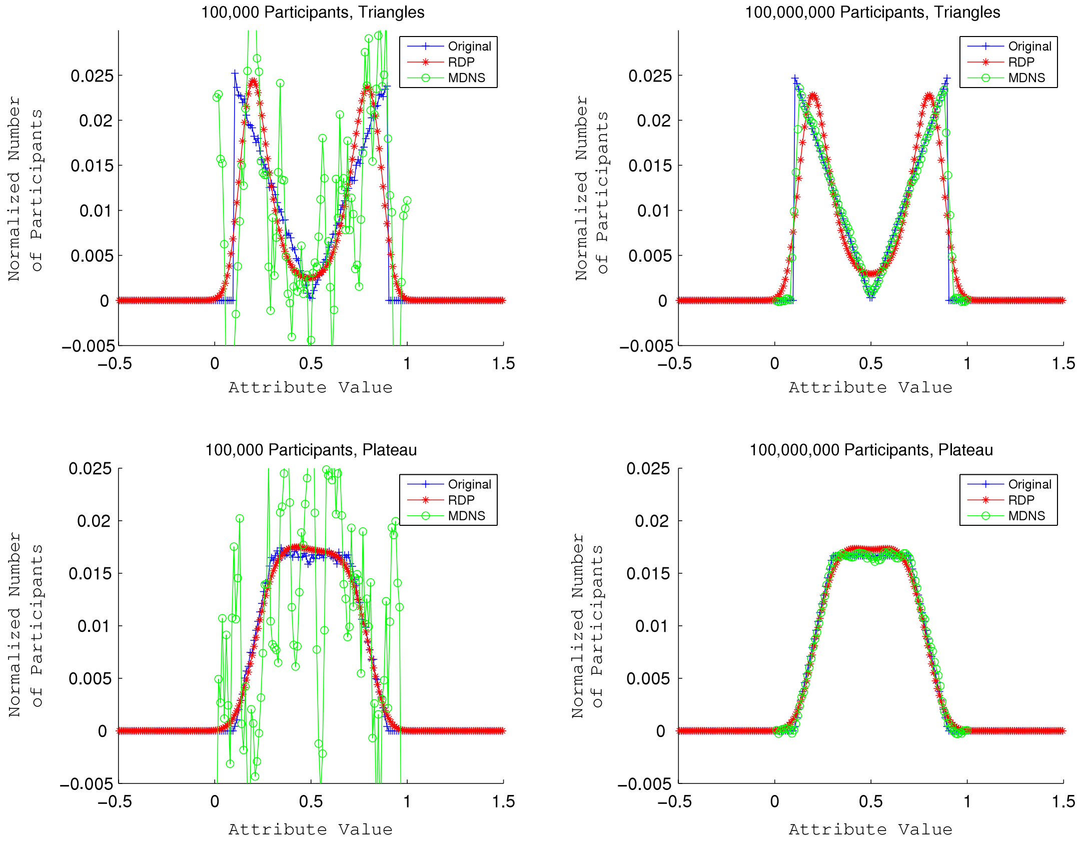


Fig. 6. Comparison of RDP to MDNS distribution reconstruction. Note that MDNSs have difficulties reconstructing the distribution with an insufficient number of participants (left top and bottom). Root mean square error of is 45.7% (right top), and 62.0% (right bottom) better with MDNS than RDP.

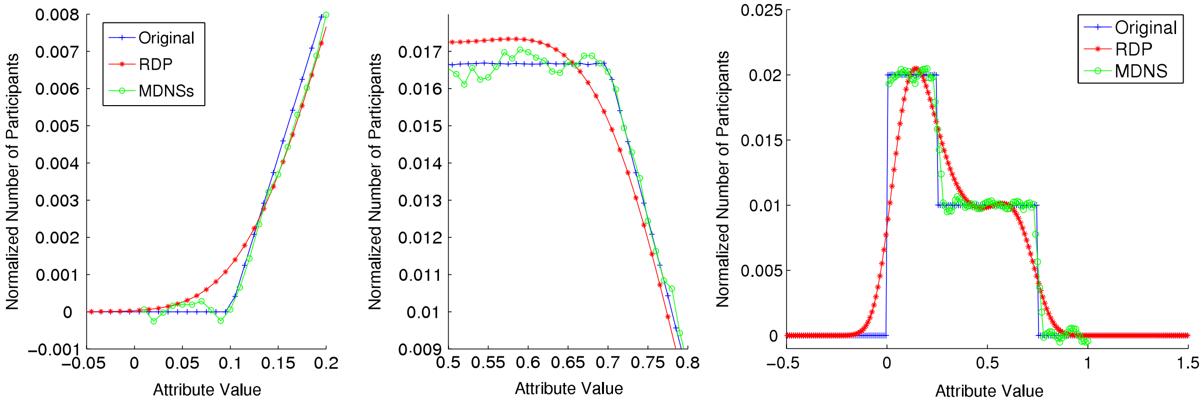


Fig. 7. Comparison of MDNSs and RDP. MDNSs have an easier time with a discontinuous PDF, and PDFs whose derivatives are discontinuous.

number of samples, longitude was dimensionally adjusted to two dimensions of two and three categories. Approximately an equal amount of time was spent in each location. Sound was sampled from the phone's microphone and quantized into three different volume levels. Levels were calibrated to three different categories that correspond to a room with normal conversations, outside in a quiet environment, and outside next to rush-hour traffic. Since the campus is surrounded on four sides by busy streets, we expected the highest noise from the surrounding 18 locations.

The results of the experiment are shown in Fig. 8. 7433 samples were collected from three smart phones on campus from 4 pm to 6 pm. The phones collected data continuously while the owners walked a predetermined path on the campus interior. We are able to separate the noisier boundary locations from the quieter locations inside the campus. Noisier locations are shown in the figure as brighter red, while quieter locations are shown as brighter green. Black represents the mean. Color in the figure was determined from the slope of a linear regression with the reconstructed histogram of volume

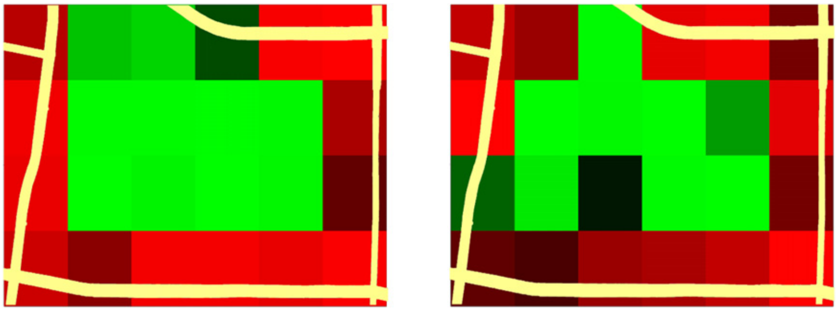


Fig. 8. Results of the implementation of MDNSs on smart phones. Left, the original distribution of the environment, X. Right, the reconstruction distribution, A. Bright red are noisier locations while bright green are quieter locations. Black represents the median noisy location. Yellow denotes the main roads. One can determine from the reconstructed distribution that the noisier locations correspond to those that contain the roads. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

levels at each location, assuming that the histogram bins are one unit apart; see Fig. 4. Ideally, positive slopes indicate noisy environments while negative slopes indicate quieter environments. The slopes were used as inputs in Fig. 8.

7. Discussion

The previous sections present algorithms, evaluation metrics, simulations, and a prototype implementation for enhancing the privacy of participatory sensing applications. Our approach is notable because it is computation and energy efficient and does not rely on encryption or a trusted base station. In this section, we summarize the tunable trade-offs among granularity (precision of data), accuracy (utility), and privacy, which were illustrated by the simulations and Android implementation. We then discuss the strengths and limitations of our current work, indicating areas for future extensions.

7.1. Trade-offs

The balance between privacy, data granularity, and reconstruction accuracy can be adjusted to meet the needs of particular applications as follows.

- Data granularity: Collecting information with finer granularity generally enhances the quality of the data, provided the number of participants scales accordingly. Given a constant number of participants, however, there is a trade-off between data granularity and reconstruction error. In some settings is may be preferable to report data with less precision (e.g., fewer locations, each covering a larger area) in exchange for higher accuracy of the reconstructed distributions.
- Privacy: Privacy increases with granularity. As the number of categories increases, it becomes more difficult to guess the category that any individual has sensed [54].
- Accuracy: With a fixed level of granularity, reconstruction error can be reduced by adding more participants.

7.2. Strengths

NSPMs enhance privacy in resource-constrained devices. The algorithms are efficient, all samples are guaranteed to be perturbed, utility metrics can be well-approximated regardless of the data distribution, they can be tuned using dimensional adjustment to improve performance when the number of participants is limited, and they are the optimal Warner scheme when the data distribution is not known *a priori*.

1. *Efficiency:* Because the algorithms at the nodes and the base station are so simple, the method is energy and computation efficient. The time complexity of the node protocol is $O(1)$ for each sensed value. M does not need to be stored or used in the perturbation process at the nodes. The base station's time complexity is the product of the number of categories for each dimension.
2. *All samples are guaranteed to be perturbed:* Our algorithms use a perturbation matrix with zeros on all diagonal entries. If the matrix has non-zero values on the diagonal, a record could in principle be reported with all of its original values. In some cases this would be viewed as a privacy breach even if it only occurred in one record out of a million [15].
3. *Utility is nearly independent of prior data distribution:*The distribution of values in the environment is often not known before sensors are deployed. In Section 4.3, we show that with an NSPM the effect of the underlying original distribution, X, on utility is small enough to safely assume it is independent. Other methods, however, require prior knowledge of the data distribution for computing an optimal perturbation matrix. For example, Hung and Du [32] argue that NPSMs (or any Warner scheme) are not the most optimal perturbation matrices for maximizing both privacy and utility. They use genetic algorithms to evolve an optimal perturbation matrix, taking privacy and utility metrics as components of the fitness function. Since their privacy metric assumes an underlying original distribution, X, the only way their scheme can evolve the best perturbation matrix is by knowing X.

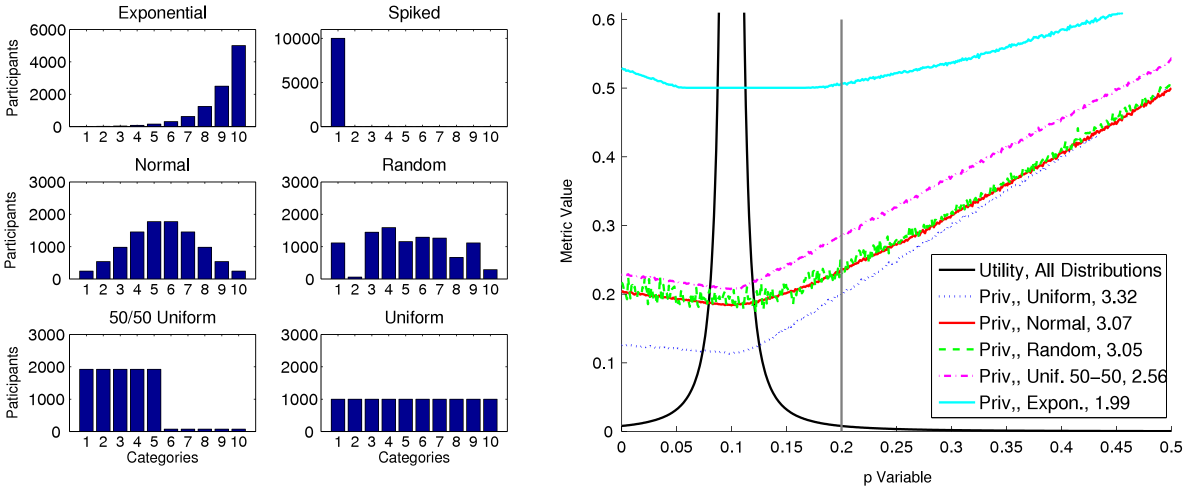


Fig. 9. (Left) Six representative distributions with 10 categories and 10,000 participants used to compute the data in the left panel. (Right) Privacy and utility values using different values of p in Eq. (1). Each curve represents the privacy value of a different underlying distribution, listed in the legend with its Shannon entropy. Utility is nearly the same for all six different distributions and is plotted once. We exclude the spiked distribution because it has a privacy value of 1 for all p.

4. *Dimensional adjustment improves the performance of NSPMs when the number of participants is limited:* Other perturbation matrices do not have straightforward implementations of this idea.
5. *NSPMs are the optimal Warner scheme when the prior data distribution is not known:* Fig. 9 illustrates NSPM's optimality when using 10 categories. The left panel of Fig. 9 shows several data distributions (uniform, normal, random, uniform 50/50, exponential, and spiked) with various Shannon entropies. The right panel plots privacy and utility metrics (y-axis) against different values of p (x-axis) in Eq. (1). The spiked distribution is omitted from the right panel as its privacy is 1 for all values of p. Because utility is largely independent of the underlying distribution, the values appear as a single curve for all distributions. The underlying distribution does, however, affect privacy. The uniform distribution provides the best privacy (lower values are better) as it has the highest entropy. The figure also illustrates the trade-offs between privacy and utility as p varies. As p approaches 0.1, utility increases asymptotically for all distributions. This occurs at 0.1 because if they are ten categories $p = 0.1$ implies that random values are being reported independent of what is sensed. While values close to 0.1 also provide excellent privacy, because of the asymptotic increase in utility (higher is worse), they are not viable parameter settings. Moving away from 0.1 improves utility symmetrically, but privacy does not degrade symmetrically. Because of this, $p = 0$ provides the same utility as $p = 0.2$ with better privacy. Generally, lower p values will provide a better utility/privacy trade-off; however, this breaks down when the underlying distribution does not have sufficient entropy, as is demonstrated by the exponential distribution.

In the cell phone implementation, because the data are perturbed, it is almost impossible for the collection server to determine a participant's true location. (We note that a cell phone tower could potentially reveal the node's location, but the base station cannot determine individual locations from its own information.) Most, if not all, encryption methods must eventually trust the final recipient of the data. In contrast, our method does not require such trust because the data are never "decrypted". Our method also does not incur the extra computational and energy costs associated with encryption/decryption algorithms and the additional communication overhead required to transmit encrypted data. Finally, it does not require a key distribution and management system.

7.3. Limitations and caveats

In the following, we discuss how some limitations of the method and the experiments, and how they might be addressed.

A sensor node might be captured by an adversary and report faulty data, either reporting the original sensed value or biasing the reported value in other ways. Similarly, if human inputs were solicited directly (e.g., please send us the name of one candidate you did not vote for), their (negative) answers might be biased. This issue can be addressed if we know how the negative answers are distributed, for example by adjusting the perturbation matrices used in reconstruction, M_δ, for each dimension δ, with the correct probabilities. A more extreme approach would authenticate the packets to ensure that they originate from a secure sensor.

If a sensor is stationary, moving slowly, or following a regular pattern, an adversary might be able to infer its location through long-term monitoring of the transmitted (negative) values. This would be especially important if an ID were transmitted with the data. This threat can be mitigated if participants respond to a base station query only if their location has changed since the last query.

In the cell phone simulation each node reports directly to the base station. Routing in traditional wireless sensor networks is often organized as a tree with the base station at the root. Data aggregation in the network can improve efficiency but is challenging if privacy is important. This is an area of active investigation; see, for example, the work of Castelluccia et al. [47] and Groat et al.'s [55] KIPDA algorithm.

For MDNSs that operate on continuous data with a small number of participants, repeated samples from the participants can be accumulated at the base station over a long period of time. For many participatory sensing applications, we would expect a long period of sensing. Furthermore, DA can improve accuracy of continuous MDNSs data by using a lower base or radix for the samples. Also, if the first dimension (the most significant digit) contains a smaller range of categories, e.g. 0–5, this dimension can use fewer categories.

We have examined only one NSPM, but there are many other possible matrices that contain zeros on the diagonal, although in some cases matrix inversion would be impossible. Future work could also examine the limits of dimensional adjustment on real-world data sets with large numbers of categories. Finally, our methods are currently restricted to reconstructing the distribution of original data. Other aggregate reconstructions could be investigated in the future.

8. Conclusion

Information such as physical locations, driving speeds, or medical information, can have devastating effects if intercepted by adversarial parties. This information can be collected through multidimensional negative surveys of participants' devices which perturb data for participatory sensing applications, providing high levels of privacy. The privacy problem addressed here is challenging, because (1) users may not trust the information collection server, and (2) embedded or sensor devices may have limited resources. Thus, we do not rely on standard encryption schemes or key distribution and management, because resources are limited on embedded or sensor devices. An advantage of our work is that privacy and accuracy can be managed by simply tuning parameters of the protocols. And, if the base station receives sufficient information, an aggregate distribution in multiple dimensions can be reconstructed efficiently.

Our results have implications for privacy-preserving data mining where there is minimal research on reconstructing perturbed multidimensional categorical data. The dimensional adjustment technique described in Section 5 reduces the reconstruction error when there is a small number of participants. Two simulations and an implementation on smart phones illustrated the feasibility of the protocols.

In each of our application scenarios, economies of scale were leveraged to achieve accurate reconstruction while protecting privacy and placing very low communication and computation overheads on the sensor nodes. These scenarios are relevant to public health safety, where, for example, they could be extended to collect body temperatures and other data for detecting disease outbreaks such as influenza. Similarly, patterns of financial transactions could be tracked, while protecting the content of the transactions themselves. Sensors on vehicles could monitor air pollution, while protecting the location and velocity information of the participants.

Acknowledgments

The authors thank Esponda, Lane, Rammohan, and Williams for their suggestions, insights, and ideas. MG acknowledges support from Motorola, Inc., Eli Lilly and Company, and NSF grant HRD-0622930; WH acknowledges support from DOE NNSA grant DE-FG52-06NA27494; SF acknowledges the partial support of NSF (grants CCF-0621900, CCR-0331580, SHF-0905236), AFOSR MURI grant FA9550-07-1-0532, and DARPA grant P-1070-113237.

References

[1] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, M.B. Srivastava, Participatory sensing, in: World Sensor Web Workshop, in: ACM Sensys, Boulder, CO, 2006.
[2] A.T. Campbell, S.B. Eisenman, N.D. Lane, E. Miluzzo, R.A. Peterson, People-centric urban sensing, in: Proc. 2nd Annu. Int. Workshop on Wireless Internet, WICON '06, Boston, MA, 2006, p. 18.
[3] J. Corburn, Confronting the challenges in reconnecting urban planning and public health, American Journal of Public Health 94 (4) (2004) 541–549.
[4] J. Horey, M.M. Groat, S. Forrest, F. Esponda, Anonymous data collection in sensor networks, in: 4th Annu. Int. Conf. Mobile and Ubiquitous Systems: Computing, Networking and Services, Philadelphia, PA, 2007, pp. 1–8.
[5] C. Sharp, S. Schaffert, A. Woo, N. Sastry, C. Karlof, S. Sastry, D. Culler, Design and implementation of a sensor network system for vehicle tracking and autonomous interception, in: Proc. 2nd European Workshop Wireless Sensor Networks, Istanbul, Turkey, 2005, pp. 93–107.
[6] A. Meyerson, R. Williams, On the complexity of optimal K-anonymity, in: Proc. 23rd ACM SIGMOD-SIGACT-SIGART Symp. Principles of Database Systems, Paris, France, 2004, pp. 223–228.
[7] C.C. Aggarwal, On *k*-Anonymity and the curse of dimensionality, in: Proc. 31st Int. Conf. Very Large Data Bases, Trondheim, Norway, 2005, pp. 901–909.
[8] F. Furfaro, G.M. Mazzeo, D. Saccà, A probabilistic framework for building privacy-preserving synopses of multi-dimensional data, in: Proc. 20th Int. Conf. Scientific and Statistical Database Management, Hong Kong, China, 2008, pp. 114–130.
[9] L. Sweeney, Achieving *k*-anonymity privacy protection using generalization and suppression, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10 (5) (2002) 571–588.
[10] P. Samarati, L. Sweeney, Protecting privacy when disclosing information: *k*-anonymity and its enforcement through generalization and suppression, in: Proceedings of the IEEE Symposium on Research in Security and Privacy, 1998.
[11] O. Goldreich, Basic Applications, in: Foundations of Cryptography, vol. 2, Cambridge University Press, New York, NY, 2004.
[12] V. Bozovic, D. Socek, R. Steinwandt, V.I. Villanyi, Multi-authority attribute based encryption with honest-but-curious central authority, in: IACR Eprint Archive, 2009, http://eprint.iacr.org/2009/083.

[13] H. Xie, L. Kulik, E. Tanin, Privacy-aware collection of aggregate spatial data, Data & Knowledge Engineering 70 (6) (2011) 576–595.

[14] H. Kargupta, S. Datta, Q. Wang, K. Sivakumar, On the privacy preserving properties of random data perturbation techniques, in: Proc. 3rd IEEE Int. Conf. Data Mining, Melbourne, FL, 2003, pp. 99–106. http://dx.doi.org/10.1109/ICDM.2003.1250908.

[15] A. Evfimievski, R. Srikant, R. Agrawal, J. Gehrke, Privacy preserving mining of association rules (invited journal version), Journal of Information Systems 29 (4) (2004) 343–364.

[16] Z. Huang, W. Du, B. Chen, Deriving private information from randomized data, in: Proc. 2005 ACM SIGMOD Int. Conf. Management of Data, Baltimore, MD, 2005, pp. 37–48.

[17] B. Pinkas, Cryptographic techniques for privacy-preserving data mining, SIGKDD Explorations Newsletter 4 (2) (2002) 12–19.

[18] J. Girao, D. Westhoff, M. Schneider, CDA: Concealed data aggregation for reverse multicast traffic in wireless sensor networks, in: Proc. 40th IEEE Int. Conf. Communications, Seoul, Korea, 2005.

[19] C. Castelluccia, E. Mykletum, G. Tsudik, Efficient aggregation of encrypted data in wireless sensor networks, in: 2nd Annu. Int. Conf. Mobile and Ubiquitous Systems: Networking and Services, San Diego, CA, 2005, pp. 109–117.

[20] W. He, X. Liu, H. Nguyen, K. Nahrstedt, T. Abdelzaher, Pda: Privacy-preserving data aggregation in wireless sensor networks, in: 26th Annual IEEE Conference on Computer Communications, INFOCOM 2007, Anchorage, Alaska, 2007, pp. 2045–2053.

[21] R.K. Ganti, N. Pham, Y.-E. Tsai, T.F. Abdelzaher, Poolview: Stream privacy for grassroots participatory sensing, in: Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, SenSys '08, ACM, New York, NY, USA, 2008, pp. 281–294.

[22] T. Feng, C. Wang, W. Zhang, L. Ruan, Confidentiality protection schemes for data aggregation in sensor networks, IEEE INFOCOM (2008).

[23] L. Lu, J. Han, L. Hu, Y. Liu, L.M. Ni, Dynamic key-updating: Privacy-preserving authentication for RFID systems, in: Proc. 5th Annu. IEEE Int. Conf. Pervasive Computing and Communications, White Plains, NY, 2007, pp. 13–22.

[24] N. Subramanian, K. Yang, W. Zhang, D. Qiao, ElliPS: A privacy preserving scheme for sensor data storage and query, in: Proc. 28th IEEE Int. Conf. Computer Communication, Rio de Janeiro, Brazil, 2009, pp. 936–944.

[25] R. Cramer, I. Damgård, S. Dziembowski, On the complexity of verifiable secret sharing and multiparty computation, in: Proc. 32nd Annu. ACM Symp. Theory of Computing, OR, Portland, 2000, pp. 325–334.

[26] J. Halpern, V. Teague, Rational secret sharing and multiparty computation: Extended abstract, in: Proc. 36th Annu. ACM Symp. Theory of Computing, Chicago, IL, 2004, pp. 623–632.

[27] Q. Huang, H.J. Wang, N. Borisov, Privacy-preserving friends troubleshooting network, in: Proc. 12th Annu. Symp. Network and Distributed Systems Security, San Diego, CA, 2005, pp. 245–257.

[28] A. Pingley, W. Yu, N. Zhang, X. Fu, W. Zhao, Cap: A context-aware privacy protection system for location-based services., in: The 29th Int'l Conference on Distributed Computing Systems, ICDCS 2009, IEEE Computer Society, 2009, pp. 49–57.

[29] J. Meyerowitz, R. Roy Choudhury, Hiding stars with fireworks: Location privacy through camouflage, in: Proceedings of the 15th Annual International Conference on Mobile Computing and Networking, MobiCom '09, ACM, New York, NY, USA, 2009, pp. 345–356. http://doi.acm.org/10.1145/1614320.1614358.

[30] C. Dwork, M. Naor, T. Pitassi, G. Rothblum, S. Yekhanin, Pan-private streaming algorithms, in: Proc. 1st Symp. Innovations in Computer Science, Beijing, China, 2010, pp. 66–80.

[31] C. Dwork, Differential privacy, in: Proc. 33rd Int. Colloq. Automata, Languages and Programming, Part II, Venice, Italy, 2006, pp. 1–12.

[32] Z. Huang, W. Du, OptRR: Optimizing randomized response schemes for privacy-preserving data mining, in: Proc. IEEE 24th Int. Conf. Data Engineering, Cancun, Mexico, 2008, pp. 705–714.

[33] S. Agrawal, J.R. Haritsa, A framework for high-accuracy privacy-preserving mining, in: Proc. 21st Int. Conf. Data Engineering, Tokyo, Japan, 2005, pp. 193–204.

[34] S.L. Warner, Randomized response: A survey technique for eliminating evasive answer bias, Journal of the American Statistical Association 60 (309) (1965) 63–69.

[35] R. Agrawal, R. Srikant, D. Thomas, Privacy preserving olap, in: Proc. 2005 ACM SIGMOD Int. Conf. Management of Data, Baltimore, MD, 2005, pp. 251–262.

[36] F. Esponda, Negative surveys, ArXiv Mathematics e-Prints arXiv:math/0608176.

[37] F. Esponda, V.M. Guerrero, Surveys with negative questions for sensitive items, Statistics & Probability Letters 79 (15) (2009) 2456–2461.

[38] F. Esponda, E.S. Ackley, P. Helman, H. Jia, S. Forrest, Protecting data privacy through hard-to-reverse negative databases, International Journal of Information Security 6 (6) (2007) 403–415.

[39] F. Esponda, S. Forrest, P. Helman, Enhancing privacy through negative representations of data, Tech. rep., University of New Mexico, 2004.

[40] F. Esponda, Everything that is not important: Negative databases, IEEE Computational Intelligence Magazine.

[41] D. Quercia, I. Leontiadis, L. McNamara, C. Mascolo, J. Crowcroft, Spotme if you can: Randomized responses for location obfuscation on mobile phones, in: ICDCS, 2011, pp. 363–372.

[42] J. Horey, S. Forrest, M. Groat, Reconstructing spatial distributions from anonymized locations, in: ICDE Workshop on Secure Data Management on Smartphones and Mobiles, Washington D.C., 2012.

[43] R.A. Horn, C.R. Johnson, Topics in Matrix Analysis, Cambridge University Press, Cambridge, UK, 1991, Ch. 4: Matrix equations and the Kronecker product, pp. 239–297.

[44] M. Gruteser, G. Schelle, A. Jain, R. Han, D. Grunwald, Privacy-aware location sensor networks, in: HOTOS'03: Proceedings of the 9th conference on Hot Topics in Operating Systems, USENIX Association, Berkeley, CA, USA, 2003, 28–28.

[45] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, A. Zhu, Approximation Algorithms for K-Anonymity, Journal of Privacy Technology.

[46] M.M. Groat, B. Edwards, J. Horey, W. He, S. Forrest, Enhancing privacy in participatory sensing applications with multidimensional data, in: Proceedings of the Tenth Annual IEEE International Conference on Pervasive Computing and Communications, Lugano, Switzerland, 2012, pp. 144–152.

[47] C. Castelluccia, C. Soriente, ABBA: A balls and bins approach to secure aggregation in WSNs, in: 6th Int. Symp. Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, Berlin, Germany, 2008, pp. 185–191.

[48] FOXNEWS.com, Homeland security looking into cell phones as anti-terror device, 2007, http://www.foxnews.com/story/0,2933,270033,00.html.

[49] A. Bertaud, S. Malpezzi, The spatial distribution of population in 48 world cities: Implications for economies in transition, unpublished manuscript (2003).

[50] R. Agrawal, R. Srikant, Privacy-preserving data mining, in: Proc. 2000 ACM SIGMOD Int. Conf Management of Data, Dallas, TX, 2000, pp. 439–450.

[51] S. Zhang, J. Ford, F. Makedon, Deriving private information from randomly perturbed ratings, in: SDM, 2006.

[52] A.P. Dempster, N.M. Laird, D.B. Rubin, Compressed sensing, Journal of the Royal Statistical Society 39 (1) (1977) 1–38.

[53] D. Agrawal, C.C. Aggarwal, On the design and quantification of privacy preserving data mining algorithms, in: Proceedings of the 20th symposium on principles of database systems, ACM, New York, NY, USA, 2001, pp. 247–255.

[54] F. Esponda, Negative representations of information, Ph.D. thesis, University of New Mexico (2005).

[55] M.M. Groat, W. He, S. Forrest, KIPDA: *k*-Indistinguishable privacy-preserving data aggregation in wireless sensor networks, in: Proc. 30th IEEE Int. Conf. Computer Communications, Shanghai, China, 2011, pp. 2024–2032.