Enhancing Privacy in Participatory Sensing Applications with Multidimensional Data

Michael M. Groat\textsuperscript{a,}\textsuperscript{*}, Benjamin Edwards\textsuperscript{a}, James Horey\textsuperscript{b}, Wenbo He\textsuperscript{c}, Stephanie Forrest\textsuperscript{a}

\textsuperscript{a}University of New Mexico, Albuquerque, New Mexico 87131, USA
\textsuperscript{b}Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA
\textsuperscript{c}McGill University, Montreal, Quebec H3A 2T5, Canada

Abstract
Participatory sensing applications rely on individuals to share local and personal data with others to produce aggregated models and knowledge. In this setting, privacy is an important consideration, lack of which could discourage widespread adoption of many applications. We present a privacy-preserving participatory sensing scheme for continuous and multivariate categorical data which uses negative surveys. Our algorithms conserving energy by avoiding computationally expensive encryption and key management techniques. An implementation on Android smart phones illustrate how the approach can protect the privacy of a participant’s multidimensional data while allowing useful aggregate information to be collected.

Keywords: multidimensional data; negative surveys; privacy protection; participatory sensing applications

1. Introduction
Participatory sensing applications [1] sense, collect, analyze, and share local information or knowledge collected from a large population of people, enabling a wide range of applications such as urban planning [2], public health [3], and vehicular transportation monitoring [4, 5]. In these applications, the privacy of those carrying sensing devices who are willing to share their information should be respected, especially when information travels across open wireless networks. On the other hand, it is desirable to generate high quality data for policymakers, researchers, and the public. Hence, trade-offs exist between protecting the privacy of the participants’ data and the utility gained from examining this content. This trade-off must also consider the energy efficiency of sensing devices because of their resource-constrained nature.

Existing approaches for protecting 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 information with other participants or trust a central collection server.

Our approach applies negative surveys to multivariate categorical data, where categories might be symbolic values (e.g., hair color, race) or a coarse-graining of numerical data into bins. A set of categories forms a proper partition over each dimension. Individual participants disguise data by reporting for each dimension a category from the set complement of the sensed category. A base station is then able to reconstruct the original distribution of sensed categories from this disguised data [4]. This approach avoids complicated encryption and key management schemes, thus conserving energy on the participants’ devices.

Multidimensional data are well suited for wireless sensor networks (WSNs) and participatory sensing applications, and can include several different environmental values along with time and location data. We seek to preserve the privacy of multidimensional data where all dimensions are sensitive. Particular values from one dimension might reveal information about another through correlation analysis. Non-sensitive dimensions can remain un-perturbed if (1) the data cannot be linked to a particular individual and (2) there are no correlations between sensitive and non-sensitive dimensions.

Using privacy and utility metrics extended from Huang and Du [11], we quantify the trade-offs between the accuracy of the reconstruction and the amount of privacy protected. These privacy and utility metrics, and some terminology, are borrowed from the privacy-preserving data mining (PPDM) field. This is because WSNs are very 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) Since sensors work more with people, their data are naturally sensitive. The terms accuracy, reconstruction error, and utility are used interchangeably, as are the terms disguise, perturb, and negate.

Our threat model assumes the base station as an honest but curious [12, 13] entity. That is, we assume it faithfully follows the network protocols but could mischievously try to collect information to use against the nodes. Additional threats come from eavesdroppers listening to radio communications who try to intercept packets. We assume all devices are equipped with sensors for data capture.

One of the limitations of previous work with negative surveys was the requirement for a large number of participants to reconstruct the data accurately [4, 14]. A slight increase in the number of categories requires a significant increase in the number of participants to maintain a given level of utility. The problem is compounded when data are multi-dimensional. We present a method called dimensional adjustment that controls this error, reducing the number of required participants. It accomplishes this by sacrificing a small amount of privacy to gain a greater amount of utility, typically 2.5 times more.

Negative surveys can be applied to continuous data as well to reconstruct
probability density functions. We compare this technique to random data perturbation (RDP) used in privacy preserving data mining. We show that our method can better reconstruct the original distribution if enough samples are present. In addition, negative surveys can better reconstruct probability density functions that are discontinuous and have derivatives that are discontinuous.

To illustrate our protocols we implemented them on Android smart phones. The University of New Mexico campus was divided into 24 different location categories and samples of the volume level were taken from the phone’s microphone and divided into 3 different levels. Since the campus is surround by streets with heavy traffic, the goal of the experiment is to distinguish the noisier boundary locations from the inner quieter locations on campus.

Roadmap: The remainder of this paper is structured as follows. Section 2 gives background information on negative surveys and randomized response techniques. Our protocols are presented in Section 3, and Section 4 describes the privacy and utility metrics used in the analysis. Section 5 gives the benefits of using negative surveys. Dimensional adjustment is introduced and analyzed in Section 6. Section 7 compares MDNS on continuous data to random data perturbation, and Section 8 gives an implementation of MDNS on Android smart phones. We discuss the implementation and speculate on how to improve its performance and security in Section 9. Section 10 discusses related work, while the final section gives future work and our conclusions.

2. Background

This section reviews background material on randomized response techniques and a specific instance of these, negative surveys in their single dimensional form. Randomized response techniques (RRTs) disguise data by perturbing a categorical value to another value. For example, if race is Hispanic, it could be perturbed to Asian. A perturbation matrix, denoted $M$, gives the probabilities of perturbing category $i$ to category $j$. It is an $\alpha$ by $\alpha$ square matrix where the columns sum to one and $\alpha$ is the number of categories.

Finding the optimal $M$ that balances both privacy and utility has been the subject of earlier research [11, 15]. Warner described the RRT for binary data [16], however, it can be extended to categorical data [17] using the following perturbation matrix, which gives an initial suggestion for $M$:

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

where $p$ is the probability of a category remaining unchanged.

The original data can then be estimated from the disguised data with the following equation:

$$\hat{A} = M^{-1}\hat{Y},$$
where \( \hat{Y} = (Y_1, ..., Y_\alpha)^T \) and \( Y_i \) is the number of disguised values in the \( i^{th} \) category. Since this is an unbiased maximum likelihood estimate, \( \hat{A} \) approaches the original distribution as the population grows. Equation (2) is known as the matrix inversion approach. An iterative approach is given by Agrawal et al. [17] but is not developed for multiple dimensions.

We review a special case of the Warner scheme called negative surveys [4, 18, 19] which use a specialized perturbation matrix containing zeros on the diagonal entries and equal values everywhere else, with the columns summing to one, i.e., \( p = 0 \) in Equation (1). We call these matrices negative survey perturbation matrices (NSPMs).

A negative survey applied to WSNs consists of two protocols. The first, or node protocol, maps the sensed data into its negative representation. To do this, each node chooses a category it did not sense with uniform probability and returns that negative information to the base station. The second, or base station protocol, reconstructs the original data at the base station. Instead of Equation (2), the following simpler equation [18] can be used:

\[
\forall i \mid A_i = N - (\alpha - 1) \cdot Y_i,
\]

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. Equation (3) has time complexity \( O(\alpha) \), compared to \( O(\alpha^2) \) for Equation (2) (ignoring matrix inversion), while still remaining an unbiased maximum likelihood estimate.

3. Protocols

Before we describe the multidimensional node and base station protocols, we will 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 \( D=3 \) then \( X(a, b, c) \), \( Y(a, b, c) \), and \( A(a, b, c) \) are counts of all the values that occur in the \( a^{th} \), \( b^{th} \), and \( c^{th} \) category in the first, second, and third dimensions. Vectors such as \( \vec{x}=<a, b, c> \) indicate a specific index in either \( X \), \( Y \), or \( A \). An individual participant senses vector \( \vec{x}^+ = <x_1^+, x_2^+, \ldots, x_D^+> \) from its environment. Sensed real values are quantized into categories, if necessary. Each \( x_i^+ \in \vec{x}^+ \) where \( 1 \leq i \leq D \), reflects 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 form a proper partition over the data in dimension \( i \), and \( \alpha_i \) 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 \( i^{th} \) dimension), while superscripts in \( \vec{x}^i \) 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 perturbed category in dimension $i$ is $\frac{1}{\alpha_i - 1}$, where $\alpha_i$ is the number of categories in dimension $i$. For example in Figure 1, a node has sensed $\vec{x}^+ = <2, b>$ from its environment, and must choose among the white cells, for instance $\vec{x}^- = <3, c>$, 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 chapter, 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 slight increase in resource cost to compute and transmit the perturbed value. Hence, the node protocol saves resources compared to encryption methods which also spend energy on key distribution and management [4].

3.2. Base Station Protocol

The base station protocol 1) collects the reported data, $Y$, and 2) estimates the original distributions of sensed values, $A$, with a reconstruction algorithm. Since this protocol is relatively trivial, we will focus on the reconstruction algorithm for the rest of the section. We introduce a natural multidimensional extension to the single dimensional equation, and later present a time optimization.

Single dimensional negative surveys reconstruct their data with Equation (3) [4, 19]. A natural extension to $D$ dimensions is given as:

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

(4)
where \( \Gamma(\vec{x}, k) \) is given as:

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

(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 a NSPM. As an example, Equation (4) with \( D=3 \) is given as:

\[
\forall a, b, c \mid A(a, b, c) = \sum_{\vec{x}} Y(\vec{x}) - (\alpha_1 - 1) \sum_{\vec{x} \text{ s.t.} \atop x_1 = a} Y(\vec{x}) - (\alpha_2 - 1) \sum_{\vec{x} \text{ s.t.} \atop x_2 = b} Y(\vec{x})
\]

\[
- (\alpha_3 - 1) \sum_{\vec{x} \text{ s.t.} \atop y_1 = c} Y(\vec{x}) + (\alpha_1 - 1)(\alpha_2 - 1) \sum_{\vec{x} \text{ s.t.} \atop x_1 = a, x_2 = b} Y(\vec{x}) + (\alpha_1 - 1)(\alpha_3 - 1) \sum_{\vec{x} \text{ s.t.} \atop x_1 = a, x_3 = c} Y(\vec{x})
\]

\[
+ (\alpha_2 - 1)(\alpha_3 - 1) \sum_{\vec{x} \text{ s.t.} \atop x_2 = b, x_3 = c} Y(\vec{x}).
\]

(6)

The time complexity of Equation (4) is given as:

\[
O \left( \prod_{i=1}^{D} \alpha_i \cdot \sum_{i=1}^{D} \binom{D}{i} (D - i)\alpha_{\text{max}} \right).
\]

(7)

where \( \alpha_{\text{max}} \) is the maximum number of categories among the dimensions. There are \( \sum_{i=1}^{D} \binom{D}{i} \) total \( Y \) terms that require \( (D - i) \) calculations. \( \alpha_{\text{max}} \) 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.

### 3.2.1. Optimization

We now present a more efficient alternative given in Algorithm 1 to the previous reconstruction process, Equation (4). It computes the same result, \( A \), however, it uses a memoization technique to improve running time and can be used with any perturbation matrix, not just a NSPM. The inputs to the algorithm 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 that dimension; \( \tau \) is a function similar to transpose that takes a row, column, hyper-row, or hyper-column, and transforms it into
Algorithm 1 Reconstruction Optimization for D Dimensions

1: function reconstruct_matrix(Y, D, F, M)
2:     R = Y
3:     for δ ∈ [1 : D] do
4:         update_dim(R, D, [], δ, F, M)
5:     end for
6:     return R
7: end function

9: function update_dim(R, D, index, δ, F, M)
10:     if length(index) = D then
11:         R(index) ← M_{δ}^{-1} * R(index)^{T}
12:     else if len(index) + 1 = δ then
13:         new_index ← index.append([:])
14:         update_dim(R, D, new_index, δ, F, M)
15:     else
16:         for i ∈ [1 : F(length(index) + 1)] do
17:             new_index ← index.append([i])
18:             update_dim(R, D, new_index, δ, F, M)
19:         end for
20:     end if
21: end function

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:

\[ O \left( \sum_{i=1}^{D} \left[ \prod_{j=1, j \neq i}^{D} \alpha_{i}^{2} \alpha_{j} \right] \right) = O \left( \sum_{i=1}^{D} \alpha_{i} \prod_{i=1}^{D} \alpha_{i} \right), \]  \hspace{1cm} (8)

ignoring the cost of matrix inversion for each \( M_{\delta} \). 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. This can have implications in privacy-preserving data mining where there is minimal research on reconstructing perturbed multidimensional categorical data. If only NSPMs are used for each dimensional perturbation matrix, the cost of Algorithm 1 reduces to:

\[ O \left( D \cdot \prod_{i=1}^{D} \alpha_{i} \right), \]  \hspace{1cm} (9)

because Line 11 in Algorithm 1 is replaced with the simpler Equation (3). The complexity denoted by Equation (9) is clearly an improvement over the complexity denoted by Equation (8). This is because parts of Equation (4) are cached in matrix R each time a vector in R is updated (Line 11).
4. Privacy and Utility Metrics

The multidimensional privacy and utility metrics were extended from the single dimensional metrics given by Huang and Du [11]. Like their single dimensional counterpart, the multidimensional formulations apply to any perturbation matrix, not necessarily a NSPM. The privacy metric ranges from [0,1], while the utility metric ranges from [0, +\infty]. For both metrics a lower value is more desirable.

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 [11] theorize the that MAP estimate is the “best that adversaries can achieve when their estimation is consistent,” and gives an upper bound on an adversary’s achievement. We extend their single dimensional metric to multiple dimensions as follows:

\[ Privacy = \sum_{\forall \chi \in \chi_\alpha} P(Y|\widehat{X}_\chi) \cdot P(\widehat{X}_\chi), \] (10)

where

\[ \widehat{X}_\chi = \arg \max_{\forall \chi \in \chi_\alpha} P(\chi|Y). \] (11)

Equation (11) calculates for Equation (10) the optimal maximum a posteriori estimate for a given index of \( Y \). This is the index that has 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, privacy is generalized to \( k \)-indistinguishability. An item is \( k \) indistinguishable if it cannot be distinguished better than guessing from \( k \) other items. A participant’s reported data in a single-dimensional negative survey with \( \alpha \) categories will have \( k \)-indistinguishability value of \( \alpha - 1 \). An individual’s data in a multidimensional negative survey with categories \( \alpha_1, \alpha_2, ..., \alpha_D \) will have a \( k \)-indistinguishability value of \( (\alpha_1 - 1) \cdot (\alpha_2 - 1) \cdot ... \cdot (\alpha_D - 1) \). This is different from \( k \)-anonymity in WSNs [6, 9, 10, 20, 21] 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 [11]. Since \( A \) is a 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 a particular
amount. Utility improves the closer $A$ gets to $X$. Hence, the mean square error (MSE), given as follows, is used to quantify utility.

\[ \text{MSE} = E[(A - X)^2] \]  

(12)

Huang and Du [11] actualized this equation by replacing $X$ with the mean of $A$ to get the variance of $A$. They then apply the variance operation to both sides of Equation (2) and formulate a theorem to compute the MSE. This theorem is extended to multiple dimensions with the following equation:

\[
\text{Utility} = \frac{1}{\alpha_1 \cdots \alpha_D} \sum_{\vec{x}} \text{MSE}(X = \vec{x}) \\
= \frac{1}{\alpha_1 \cdots \alpha_D} \sum_{\vec{x}} E[(P(A = \vec{x}) - P(X = \vec{x}))^2] \\
= \frac{1}{\alpha_1 \cdots \alpha_D} \sum_{\vec{x}} \left( \sum_{\vec{x}} \left[ \mu(\vec{x}, \vec{x})^2 \cdot \text{var}(\vec{x}) \right] \\
+ \sum_{\vec{x}^k, \vec{x}^l \text{ s.t. } \vec{x}^k \neq \vec{x}^l, \forall \gamma} \left[ 2 \cdot \mu(\vec{x}, \vec{x}^k) \cdot \mu(\vec{x}, \vec{x}^l) \cdot \text{cov}(\vec{x}^k, \vec{x}^l) \right] \right),
\]

(13)

where

\[
\mu(\vec{x}^m, \vec{x}^n) = \prod_{d=1}^{D} M^{-1}_d(x^m_d, x^n_d),
\]

(14)

denotes the product of an element from each dimensional perturbation inverse matrix where the row and column correspond to the categories in the $d^{th}$ dimension of $\vec{x}^m$ and $\vec{x}^n$ respectively. \text{var} and \text{cov} are given as:

\[
\text{var}(\vec{x}) = \frac{1}{N} \cdot P(Y = \vec{x}) \cdot (1 - P(Y = \vec{x}))
\]

\[
\text{cov}(\vec{x}, \vec{x}^l) = -\frac{1}{N} \cdot P(Y = \vec{x}) \cdot P(Y = \vec{x}^l),
\]

(15)

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

\[
\text{var}_{MDSN}(\vec{x}) = \left( \left[ \prod_{i=1}^{D} \alpha_i \right] - 1 \right) \frac{1}{N - 1} \cdot P(Y = \vec{x}) \cdot (1 - P(Y = \vec{x}))
\]

\[
\text{cov}_{MDSN}(\vec{x}, \vec{x}^l) = -\left( \left[ \prod_{i=1}^{D} \alpha_i \right] - 1 \right) \frac{1}{N - 1} \cdot P(Y = \vec{x}) \cdot P(Y = \vec{x}^l),
\]

(16)
4.3. Trade-offs Between Privacy and Utility

The underlying distribution, $X$, affects the utility and privacy. The following normalized version of Shannon’s entropy will help illustrate the effects:

$$S = \frac{-\sum \bar{x} P(X = \bar{x}) \log_2 P(X = \bar{x})}{\log_2 \prod_{i=1}^{D} \alpha_i},$$

(17)

where $S$ ranges from $[0, 1]$. For example, a spiked distribution (all elements in one category) has the lowest normalized entropy, $S = 0$, and provides the worse privacy, but the highest utility. A uniform distribution, which has the highest normalized entropy, $S = 1$ provides the worse utility, but the best privacy. All other distributions fall in between. 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 on utility, 0.786 to 0.802, which correspond to 1.6% of the privacy metric. Since this effect on utility is so small, it was incorrectly assumed in Groat et al. [22] that utility is independent of the underlying distribution. However, it is safe to make this assumption because the number of categories and the number participants predominately decide the metric’s value. This allows WSN designers to determine the utility of negative surveys a priori.

4.4. Summary of Metrics

These metrics, and the variance and covariance, were verified using the Kronecker technique, a technique that converts a multidimensional negative survey to a single dimension. The perturbation matrix, $M$, of the Kronecker technique is the Kronecker product [23] of the individual perturbation matrices for each dimension, given as:

$$M_{Kron} = (((M_1 \otimes M_2) \otimes M_3) \ldots \otimes M_D),$$

(18)

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 marshalled into 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 4, 3, and 2 categories each, it is given as:

$$Y_{MARSHALLED} = \begin{bmatrix} Y(1,1,1) \\ Y(1,1,2) \\ Y(1,2,1) \\ \vdots \\ Y(4,3,2) \end{bmatrix}. \quad (19)$$

To obtain the estimated distribution, $A$, $Y$ is multiplied with $M_{Kron}^{-1}$ according to Equation (2). $A$ is then de-marshalled. Care must be taken to the order
of marshalling to correspond with the correct order that the Kronecker products were applied. This technique, however, is not optimal because the time complexity is given as:

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

(20)

ignoring matrix inversion which is minimal because of the mixed-product property [23]. Although the complexity cost is greater than Equation (9), the Kronecker technique simplifies the software engineering of MDNSs for simulations and allows the use of single dimensional metrics to verify their multidimensional counterparts. The variance and covariance were also verified with the Kronecker technique.

5. Benefits of a Negative Survey Perturbation Matrix

NSPMs are well suited for participatory sensing applications and WSNs, therefore we identify several reasons to base our scheme on them:

1. **Algorithms are simplified at the nodes and base station:** MDNSs that use NSPMs are appropriate for resource-constrained devices because perturbation is simplified at the sensors. $M$ does not need to be stored or used in the perturbation process. Negative values are simply chosen with equal probabilities, which can be beneficial if there are a large number of categories. Additionally, with Algorithm 1, reconstruction at the base station is simplified.

2. **All samples are guaranteed to be perturbed:** If a perturbation matrix has non-zero values on the diagonal, there is a small chance of a record maintaining all of its original values. This could be viewed as a privacy breach even if it only occurs in 1 record out of a million [24]. With NSPMs this is eliminated.

3. **Utility can be assumed to be independent of the prior distribution:** Sensors often do not know the distributions of the environment they will be deployed in. However, in order to find the best perturbation matrix this distribution needs to be known a priori. For example, Huang and Du [11] use generic algorithms to evolve an optimal perturbation matrix. These genetic algorithms used privacy and utility metrics as their fitness functions. Since their privacy metric is based on the underlying original distribution, $X$, the only way in their scheme to evolve the best perturbation matrix is to know $X$. Since the underlying distributions are typically not known in participatory sensing applications, NSPMs are a good initial choice because utility can be known a priori.

4. **NSPMs allow Dimensional Adjustment** NSPMs allow for dimensional adjustment (explained in the next section) which is not possible with other perturbation matrices.
While some may argue that NPSMs are not the most optimal perturbation matrix [11], they are appropriate for protecting privacy in resource constrained devices because the algorithms are efficient, all samples are guaranteed to be perturbed, utility can be known \textit{a priori}, they allow for dimensional adjustment.

6. Dimensional Adjustment

In this section we introduce a technique called \textit{dimensional adjustment (DA)} that reduces the number of participants necessary to maintain a reasonable amount of utility. Previous work with single dimensional data [4, 14] observed that an inherently large number of samples or participants are required to maintain a reasonable reconstruction accuracy when the number of categories increases. This is only compounded with multiple dimensions and has limited negative surveys to applications of a small number of categories. In this section, we discuss the magnification of errors, propose dimensional adjustment to address this challenge, discuss the privacy and utility trade-offs, and illustrate that it will always improve utility.

6.1. Magnification of Errors

Horey et al. [4] suggests the relationship between an increase in categories and the number of participants needed to maintain the same utility is almost linear. Because utility was measured with the relative RMSE, this give the appearance of an almost linear relationship. When measured with the MSE as illustrated in Figure 3 the relation is indeed linear. However, these metrics are misleading because they do not model how the disguised distribution, \( Y \), deviates from its expected value, nor do they give upper and lower bounds on this deviation. Xie et al. [14] suggests that the magnification of errors is due to counting with integers and give the cause as the gap between the floor and ceiling of the value \( \frac{X_i}{\alpha - 1} \). This gap introduces errors in the reconstruction process which is increased with the total number of categories. They give an upper and lower bound of the error with a specific category to be \( \pm (\alpha - 1)^2 \). However, this is also related to using the RMSE to measure utility.

We introduce an alternative explanation using Chernoff bounds which can better model the deviation of \( Y \) from its expected value. Closer values of \( Y \) to its expected value gives a reconstructed distributed, \( A \), closer to the original distribution, \( X \). Since negative surveys are similar to the balls and bins problem, 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 Equation (3) as follows:

\[
E[Y_i] = \frac{N - A_i}{\alpha - 1}.
\]
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 $\delta$ more or less balls than the expected value, are represented in as follows:

$$
P[X_i > E[Y] + \delta] = \left( \frac{e^\delta}{(1 + \delta)^{(1+\delta)}} \right)^{\frac{N - X_i}{\alpha - 1}}
$$

$$
P[X_i < E[Y] - \delta] = \left( \frac{e^\delta}{(1 - \delta)^{(1-\delta)}} \right)^{\frac{N - X_i}{\alpha - 1}},
$$

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

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

Figure 3 gives the results of Chernoff bounds where categories are fixed at 10 and the population varies, and where the population is fixed at 1,000 and categories vary. These two figures use a single dimensional negative survey, but a multidimensional negative survey would behave similarly where $\alpha$ in Equation (22) is replaced with all possible categories a sensed value can be perturbed to, for example, $(\alpha_1 - 1) \cdot (\alpha_2 - 1) \cdot \ldots \cdot (\alpha_D - 1)$.

In Figure 3 (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 (albeit a small exponent) decreases the deviation of $Y$ from its expected value. Figure 3 (right) also shows when the number of categories increases, the probability grows almost logarithmically. To maintain a constant probability, the population needs...
to increase as follows:

\[ N = \alpha_{\text{increased}} \cdot \log(N), \quad (24) \]

which increases the needed number of participants linearly with the number of categories. Although the probability of deviation from the expected value grows almost logarithmically, an initial increase (up to 150) in categories significantly increases the deviation of \( Y \) from its expected value, which is compounded with multiple dimensions. This significant initial increase could explain the magnification of errors associated with negative surveys.

6.2. Dimensional Adjustment Protocols

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 reconstruction accuracy (utility). Having less dimensions with a larger number of categories makes utility worse (higher utility value). Intuitively, accuracy is related to Figure 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.

6.3. Trade-off Analysis

Trade-offs exist between a high number of dimensions with a low number of categories, versus a low number of dimensions with a high number of categories. A one-dimensional negative survey with 64 categories provides the best privacy but the worst utility, compared to 6 dimensions with 2 categories each, which 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 1,000,000 samples and 10,000 categories, we see 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, \( X \). The multidimensional negative survey that uses 6 dimensions and 1,000,000 participants is comparable to a single dimensional negative survey that uses 71,414,286 participants. This is calculated by setting the following utility modeling equation for a single dimension:

\[ Utility_{\text{model}} = (\alpha - 2)/N, \quad (25) \]
Table 1: Two negative surveys of 10,000 total categories and 1,000,000 participants. The second uses dimensional adjustment.

<table>
<thead>
<tr>
<th></th>
<th>1 dimension of</th>
<th>6 dimensions of</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10,000 categories</td>
<td>5x5x5x5x4x4 categories</td>
</tr>
<tr>
<td>utility</td>
<td>0.00100</td>
<td>0.00014</td>
</tr>
<tr>
<td>privacy</td>
<td>0.01457</td>
<td>0.01960</td>
</tr>
</tbody>
</table>

to 1.40E-04 (from Table 1), α to 10,000, and solving for N. The same multidimensional negative survey is also equivalent to a single dimension using 142 categories. This is calculated by setting Equation (25) to 1.40E-04, N to 1,000,000, and solving for α. Equation (25) has an $R^2$ value of 0.9999.

We perform a similar analysis for the privacy equivalence with Table 1. Modeling equations for the privacy metric are dependent on the underlying distribution, and therefore cannot be determined until the underlying distribution is known. We use the following privacy modeling equation assuming the normal distribution:

$$P_{rivacy\_model} = \frac{2.5}{(\log_2(\alpha))^2 + 1.5}, \quad (26)$$

which has an $R^2$ value of 0.976. The multidimensional negative survey in Table 1 is equivalent in privacy to using a single-dimensional negative survey of 2,397 categories, yet previously it had the same utility as 142 categories.

Using this privacy and utility modeling, we can further illustrate the trade-offs in DA. Privacy degrades when the number of categories are reduced because there is less anonymity among fewer categories. Utility, however, increases when categories decrease. We see above a degradation of privacy from 10,000 to 2,397 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.

### 6.4. Proof of Utility Increase in Dimensional Adjustment

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

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

The derivative of the Chernoff upper bound is given as:

$$\frac{\partial}{\partial \alpha} \left( \frac{e^\delta}{(1 + \delta)^{1+\delta}} \right)^{\frac{N - X_i}{\alpha - 1}} = \frac{\log(e^{\delta(\delta + 1)^{(\delta-1)}}(N - X_i)(e^{\delta(\delta + 1)^{(\delta-1)}} - 1))}{(\alpha - 1)^{2}} \quad (28)$$
which is always positive when $\alpha$, $N$, $\delta$, $X_i$ are greater than zero and $\delta < 1$, and $X_i < N$. The lower bound is not given but behaves similar. Equation (28) means that as $\alpha$ increases, the Chernoff bound is monotonically increasing. Hence, any $\alpha$ lower than another will always have a lower probability. A lower probability means that the bins that 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.

7. Comparison of MDNSs to Random Data Perturbation

MDNSs can operate on continuous data as well. This is accomplished by treating every digit in a datum as a dimension of 10 categories, 0-9. Each digit is then perturbed to something other than itself. For example, 536 could be perturbed to 74, but not 339. If a range of values consists of more digits than desired dimensions, the first $D$ most significant digits can be used, and the rest can be ignored, however, granularity will decrease.

Random Data Perturbation (RDP) algorithms proposed by Agrawal, et al. [25] and later Zhang et al. [26] perturb continuous data by adding noise drawn from a known distribution. The perturbed data is then reconstructed to an approximation of 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 [27] that provably converges to the maximum likelihood estimate of the original distribution [28]. RDP techniques assume that the data and the noise are drawn from a continuous domain. However, with negative surveys, data is quantized into discrete categories. Previous research compared the original negative survey method to a modified categorical RDP technique [4]. It is possible the accuracy of comparison was lost in the modification of RDP. In this section we compare negative surveys on continuous data to the original RDP technique.

Both MDNS and RDP have their strengths and weaknesses. MDNSs require a large number of samples to reconstruction the probability density function (PDF) accurately\(^1\). For example in Figure 4 with 100,000,000 samples, the root mean square error (RMSE) of RDP for the triangles 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. Although, MDNSs handle discontinuous probability density functions (PDFs) and PDFs whose derivatives

\(^1\)A moving average (size 7) was used on the negative surveys to help reconstruct and smooth the data better. With enough participants this is not necessary.
Figure 3: Comparison of RDP to MDNS distribution reconstruction. Note, MDNSs have difficulties reconstructing the distribution with an insufficient number of participants (left top and bottom). Root mean square error is 45.7% (right top), and 62.0% (right bottom) better with MDNS than RDP.

Figure 4: Comparison of MDNSs to RDP. MDNSs have an easier time with a discontinuous PDF, and PDF’s whose derivatives are discontinuous.

are discontinuous better than RDP as illustrate in Figure 5. In addition, RDP’s stopping criteria is problematic [25], and takes a considerable more amount of time to run than MDNSs.

8. Implementations on Physical Devices

We implemented the MDNS node protocol as an Android smartphone application and performed two experiments. For both experiments, location information was obtained through GPS satellites only. While this consumes more
energy, we required a finer granularity of location information because experiments were performed on the University of New Mexico's (UNM) campus. GPS information was divided into two dimensions, longitude and latitude. Volume levels from the phone’s microphone were used as another dimension and was divided into three different categories. The phones sample the microphone and its location, perturb this information, and send it back to the base station, which was implemented in Java on a CS department server.

The first experiment was designed to test the energy use of sending messages with and without encryption. We decided to use the Secure Socket Layer for the encryption technique because it is widely used and easily implemented in Java on the smart phones. Messages were first sent with SSL until the battery was drained of half its charge. This was repeated without SSL. Without affecting the results, the phone stayed at the same location. To maintain an equal test bed, all other applications were regularly killed with Advanced Task Killer from Rechill and experiments were performed in the middle of the night to prevent unwanted calls or text messages.

We were able to send 8612, or 16.22% more messages sent without encryption than the 7401 messages with encryption. It took 159.2 minutes to send these message with SSL, while it took 143.6 minutes with no encryption. This means that approximately 4.836 more milliseconds was spent per message for the extra security. This experiment was repeated with similar results, i.e. the number of messages sent differed by at most 4 messages. [ToDo: This is with the SSL option only. Need to run with the non SSL option.]

We designed the second experiment to determine the noisiest locations on the University of New Mexico’s campus. Latitude and longitude were divided into 4 and 6 categories respectively for a total of 24 different locations. An almost uniform amount of time was spent in each location. Sound was sampled from the phone’s microphone and quantized into three different volume levels. Since the volume reported by the operating system does not correspond directly to decibels, it was calibrated manually to 3 different categories that correspond to a room with normal conversations, outside in a quit environment, and outside next to rush hour traffic. Since the campus is surrounded by busy streets, and we performed the experiment during rush hour, we expected the highest noise from the surrounding 18 locations.

The results of the experiment are shown in Figure 6. 7,433 samples were collected from 3 smart phones. We are able to determine the noisier boundary locations from the quieter locations inside the campus. Noisier locations are brighter red, while quieter locations are brighter green. Black represents the mean. Color in the figure was determined by the slope of a linear regression with the reconstructed histogram of volume levels at each location, assuming histogram bins are one unit apart. Ideally a positive slop will indicate noisy environments. The campus was walked around 4 in the afternoon to sample the rush hour traffic. The two bright green squares along central ave could be due that this location was traversed during the later end of rush hour, or that cars were stopped at red lights.

We discovered from the experiment that duplicate packets need to be pre-
vented. This could be solved with a unique user id and packet id numbers. This gives the base station the option of authenticating participants’ information, although this would vary from the insecure but energy cheap techniques such as simply reporting a unique id, to the more secure and energy intensive techniques such as signing the information with a asymmetric encryption. Data integrity is left for future work.

If cell phones were equipped with radiation monitoring sensors, they could monitor a city for possible radiation threats such as unexploded dirty bombs, escaped radiation from a nuclear reactor accident, or lost or stolen medical waste, while not revealing individuals’ locations. 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. However, 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.
9. Discussion

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. Most, if not all, encryption methods must eventually trust the final recipient of the data. In contrast, our method does not require such trust. Further, it does not incur the extra computational cost of encryption and the additional communication overhead to transmit encrypted data, nor the extra cost of key distribution and management.

Sometimes smart phones or nodes will be captured or masquerade as legitimate nodes, continually sending responses when queried. These nodes might try to corrupt the aggregate information by not following or altering the node protocol by reporting the original sensed value or favoring some categories over others. This can be addressed by adjusting in the reconstruction process the perturbation matrices, $M_δ$, for each dimension $δ$, with the correct probabilities that categories were favored.

Some participants’ locations might either be constant (if they are not moving around) or follow regular patterns. If a single cell phone were to report its negative location regularly, an adversary might be able to infer its positive location through long term monitoring of the transmitted values. This is especially important if an ID is transmitted with the data. This threat can be addressed if participants are asked to respond to a base station query only if their location has changed since the last query, or to limit the amount of information sent to the base station.

10. Related Work

Privacy-preserving algorithms have been developed for data mining [24, 29, 30], data aggregation [4, 31, 32], and other applications [33, 34]. There are four main classes of solutions: perturbation, $k$-anonymity, secure multi-party computation (SMC), and homomorphic encryption. The first class hides data values by perturbing individual data or query results [24, 29, 30]. These methods usually assume that the distribution of data/noise is known to obtain accurate results. However, as shown by Kargupta et al. [29] and Huang et al. [30], certain types of data perturbation might not preserve privacy well. The second class, $k$-anonymization [6, 7, 9], makes a data value or participant indistinguishable from $k-1$ other items. It was originally designed for privacy-preserving data mining, but in participatory sensing applications individual participants sense and share their own data. Hence, there is limited potential to mix individual participants’ data with others’ data. The third class, SMC techniques [35–37], rely on a joint computation among a set of involved peers. This is problematic in participatory sensing applications which may incur a high communication cost.

\footnote{While the cell phone tower could reveal the node’s location, the base station cannot determine the location from its own information.}
or computation overhead when the participant population is large. The fourth class aggregates data based on homomorphic encryption [31, 32], which allows a user to perform data aggregation on individual data without knowing the data. However, in order to interpret the final aggregation result, a server needs to know which users reported data, which is not always desirable.

Dwork et al. [38] 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 node capture attacks that examine internal data. However, it assumes a secure stream as a precondition of the algorithm, while the work reported in our paper protects the stream of information in transit. Pan-private algorithms, however, work better for complex aggregates such as the t-incidence items, the t-cropped mean, and the fraction of k-heavy hitters [38].

Differential privacy [39] 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 negative surveys, on the other hand, do not assume that the server is trustworthy.

Gaussian negative surveys (GNSs) [14] also reduce the number of participants needed for accurate negative survey reconstruction. Xie et al. propose a special perturbation matrix where 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 negative surveys. 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. [40] 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 depended on the size of all locations, $O(number\ of\ locations)$, is transmitted, while our technique sends a smaller vector, $O(log(number\ of\ locations))$.

11. Future Work and Conclusion

Future work can examine non NSPMs that level the disguised data, with the constraint of zeros on the diagonals. This is not possible for all underlying distributions, because matrix inversion is impossible in some cases. Future work can also examine the limits of dimensional adjustment on real-world data.
sets with large numbers of categories. Although histograms are useful, we are interested in reconstructing other aggregates.

Information such as physical locations, driving speeds, or medical information, can have devastating effects if intercepted by adversarial parties. Multidimensional negative surveys perturb data for participatory sensing applications, providing high levels of privacy. The privacy-preservation 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. Our method scales well because the communication and computation overhead is low for the sensor nodes, especially when compared to expensive encryption schemes.

Acknowledgment

The authors thank Rammohan, Williams, and Esponda 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


