A Bayesian approach for on-line Sum/Count/Max/Min Auditing on boolean data

Bice Cavallo* and Gerardo Canfora**

* Department of Constructions and Mathematical Methods in Architecture
University of Naples Federico II, Italy

** Department of Engineering
University of Sannio, Benevento, Italy

Abstract. We consider the problem of auditing databases that support statistical sum/count/max/min queries to protect the privacy of sensitive information. We study the case in which the domain of the sensitive information is the boolean set. Principles and techniques developed for the privacy of statistical databases in the case of continuous attributes do not always apply here. We provide a probabilistic framework for the on-line auditing and we show that sum/count/min/max queries can be audited by means of a Bayesian network.

1 Introduction

A number of disclosure control methods to protect a Statistical Database (SDB) have been proposed in the literature (see [1] for a survey). We focus on auditing [7], [8], [10], [11], [12], [15], and particularly on the on-line auditing over boolean data. On-line auditing entails that queries are answered one by one in sequence and an auditor has to determine whether the SDB is compromised by answering a new query.

Reference [7] considers the on-line sum, max, and mixed sum/max auditing problems. Both the on-line sum and the on-line max problems have efficient auditing algorithms. However, the mixed sum/max problem is NP-hard. References [4], [5], [6] and [10] deal with on-line max/min auditing.

Most of the work in this area assumes that the confidential data are real-valued and unbounded (see [12]). In certain important applications, however, data may have discrete values, or have maximum or minimum values, that are fixed a priori and frequently attainable. In these cases, traditional methods for maintaining privacy are inadequate.

As an example, let us consider a dataset with n records, and assume that there is a sensitive field X. For each $i \in \{1, \dots n\}$, the value x_i must not be disclosed. The system

$$\begin{cases} x_1 + x_2 + x_4 = 1, \\ x_2 + x_3 = 1, \\ x_1 + x_3 = 1, \end{cases}$$

is secure if the variables are real, but it is not secure if they are boolean, because in this case the values of all variables are determined. In reference [11], the authors study the sum auditing problem over boolean attributes and propose an algorithm that approximates the auditing problem.

Following the approach introduced in [4], [2], [5] and [6], in reference [3] we provide a probabilistic analysis for variables in a sum query; we use a Bayesian Network (BN) for dealing with sum auditing on boolean data.

The original contribution of this paper is threefold:

1. to provide a formal basis for probabilistic sum/count/max/min auditing on a boolean domain;
2. to show that sum/count/min/max queries can be audited by means of a BN;
3. to optimize the BN proposed in [3]. In a first time, we reduce the CPT size of the BN encoding a query of size l, from $O(2^l)$ to $O(l^3)$, by means of a parent divorcing or a temporal transformation, then we furtherly reduce the CPT size at run-time, given the answer to the current query.

The paper is organized as follows: Section 2 introduces the notation and definitions used in the paper; Section 3 provides a formal basis for probabilistic auditing on a boolean domain; Section 4 proposes a BN for the on-line sum/count/max/min auditing; finally, Section 5 provides concluding remarks and directions for future work.

2 Notation and preliminaries

Let T be a dataset with n records, X the sensitive field and $D = \{0, 1\}$ the domain of X.

Moreover, let us assume that:

- a sum query of size equal to l is represented by the set $Q = \{x_{i_1}, \ldots, x_{i_l}\}$. For instance, $Q = \{x_2, x_3, x_5\}$ encodes $x_2 + x_3 + x_5$, and $l = |Q| = 3$;
- s is the answer to a sum query Q, that is $\sum_{x_i \in Q} x_i = s$;
- the sensitive data x_i are n independent variables;
- each x_i has the same probability distribution, that is $P(x_i = 1) = p$ and $P(x_i = 0) = 1 - p$, for each $i \in \{1, \ldots, n\}$, with $p \in [0, 1]$.

In the on-line auditing, given a sequence of queries $\{Q_1, Q_2, ..., Q_{t-1}\}$, the corresponding answers $\{s_1, s_2, ..., s_{t-1}\}$ provided to an user, and the current query Q_t, the auditor has to decide if to deny Q_t, or provide the answer s_t; no value of x_i has to be disclosed.

We consider the following definition of probabilistic compromise:

Definition 1. *[6] A privacy breach occurs if and only if a sensitive data is disclosed with probability greater or equal to a given tolerance probability tol. If a sensitive data is disclosed with tol = 1, then the SDB is fully compromised.*

2.1 Bayesian networks

A BN is a probabilistic graphical model that represents a set of variables and their probabilistic dependencies [14]. A BN, also called a belief net, is a directed acyclic graph (DAG), which consists of nodes to represent variables and arcs to represent dependencies between variables. Arcs, or links, also represent causal influences among the variables. The strength of an influence between variables is represented by the conditional probabilities which are summarized in a conditional probability table (CPT). If there is an arc from node A to another node B, A is called a parent of B, and B is a child of A. The set of parent nodes of a node X is denoted $parents(X)$. The size of the CPT of a node X depends on the number s of its states, the number r of $parents(X)$, and the number s_j of parent states, in the following way:

$$size(CPT) = s \cdot \prod_{j=1}^{r} s_j. \tag{1}$$

For every possible combination of parent states, there is an entry listed in the CPT. Thus, for a large number of parents the CPT will expand drastically. If the node X has no parents, its local probability distribution is said to be *unconditional*, otherwise it is *conditional*. If the value of a node is observed, then the node is said to be an *evidence* node.

In order to add prior knowledge on a BN, we can add likelihood; adding likelihood is what we do when the user learns something about the state of the BN, which can be entered into a node. The simplest form is the evidence, that is, the probability that a state is 1 while the probability of each other state is 0. In general, likelihood has value in $[0, 1]$ and represents the probability of a state. Obviously, the sum of all probabilities is necessarily 1.

3 A probabilistic approach for the on-line auditing

By assumptions in Section 2, a sum query Q of length l on boolean data is described by a binomial distribution with parameters l and p, that is $Q \sim B(l, p)$. Thus, Proposition 1 and Corollary 1 hold true:

Proposition 1. *[3] Let Q be a sum query of length l. Then, for $k \in \{0, \ldots, l\}$:*

$$P(\sum_{x_i \in Q} x_i = k) = \binom{l}{k} \cdot p^k \cdot (1-p)^{l-k}. \tag{2}$$

Corollary 1. *The mean value and the variance of $\sum_{x_i \in Q} x_i$ are:*

$$\mu[\sum_{x_i \in Q} x_i] = lp, \qquad \sigma[\sum_{x_i \in Q} x_i] = lp(1-p).$$

Example 1. Let us assume $Q_1 = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ and $p = \frac{1}{2}$. Then, the probabilities $P(\sum_{i=1}^{7} x_i = k)$, for $k \in \{0, \ldots, 7\}$, are provided in Table 1. The mean value and the variance of $\sum_{i=1}^{7} x_i$ are:

$$\mu[\sum_{i=1}^{7} x_i] = \frac{7}{2} \tag{3}$$

$$\sigma[\sum_{i=1}^{7} x_i] = \frac{7}{4}. \tag{4}$$

k	$P(\sum_{i=1}^{7} x_i = k)$
0	0.0078125
1	0.0546875
2	0.1640625
3	0.2734375
4	0.2734375
5	0.1640625
6	0.0546875
7	0.0078125

Table 1. $P(\sum_{i=1}^{7} x_i = k)$ with $p = \frac{1}{2}$.

In order to deal with sum auditing, we have to check whenever a privacy breach occurs after the answer s to a sum query Q; for each $x_i \in Q$, Proposition 2 allows us to compute $P(x_i | \sum_{x_i \in Q} x_i)$.

Proposition 2. *[3] Let Q be a sum query of length equal to l. For each $x_i \in Q$, the following posterior probability holds true:*

$$P(x_i = 1 | \sum_{x_i \in Q} x_i = s) = \frac{s}{l}. \tag{5}$$

Example 2. If $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 = 3$ then $P(x_i = 1 | \sum_{i=1}^{7} x_i = 3) = \frac{3}{7} = 0.4286$.

Remark 1. Let Q be a sum query of length equal to l, $m = min\{x_i\}_{x_i \in Q}$ and $M = max\{x_i\}_{x_i \in Q}$. Since the domain of the sensitive field is $D = \{0, 1\}$, the following considerations are straightforward:

1. $\sum_{x_i \in Q} x_i = s$ if and only if there are s values equal to 1, thus a sum query is equivalent to a count-query;
2. if $\sum_{x_i \in Q} x_i = 0$ (resp. $\sum_{x_i \in Q} x_i = l$) then the auditor has to deny the answer because each $x_i = 0$ (resp. each $x_i = 1$);

3. $(m = 0 \Leftrightarrow \sum_{x_i \in Q} x_i < l)$ and $(m = 1 \Leftrightarrow \sum_{x_i \in Q} x_i = l)$, thus, for each $x_i \in Q$, $P(x_i = 1|m = 1) = 1$;
4. $(M = 0 \Leftrightarrow \sum_{x_i \in Q} x_i = 0)$ and $(M = 1 \Leftrightarrow \sum_{x_i \in Q} x_i > 0)$, thus, for each $x_i \in Q$, $P(x_i = 0|M = 0) = 1$.

Proposition 3. *Let $Q \subseteq \{x_1, \dots x_n\}$, with $|Q| = l$ and $M = max\{x_i\}_{x_i \in Q}$. Then, for each $x_i \in Q$, the following equalities hold true:*

$$P(M = 1) = 1 - (1 - p)^l, \quad P(x_i = 1|M = 1) = \frac{p}{1 - (1 - p)^l}.$$

Proof. By item 4. of Remark 1 and Proposition 1, we have:

$$P(M = 1) = P(\sum_{x_i \in Q} x_i > 0) = 1 - \binom{l}{0} p^0 (1 - p)^l = 1 - (1 - p)^l.$$

Finally, by applying Bayes' Theorem, for each $x_i \in Q$, we have:

$$P(x_i = 1|M = 1) = \frac{P(M = 1|x_i = 1)P(x_i = 1)}{P(M = 1)} = \frac{p}{1 - (1 - p)^l}.$$

A result analogous to Proposition 3 can be proved for $m = min\{x_i\}_{x_i \in Q} = 0$.

Let Q_1 and Q_2 be disjoint sum queries, then Proposition 4 holds true.

Proposition 4. *Let Q_1 and Q_2 be disjoint sum queries of length l_1 and l_2 respectively. Then, for $k \in \{0, \dots, l_1 + l_2\}$:*

$$P(\sum_{x_i \in Q_1 \cup Q_2} x_i = k) = \binom{l_1 + l_2}{k} \cdot p^k \cdot (1 - p)^{l_1 + l_2 - k}. \tag{6}$$

$$P(x_i = 1|\sum_{x_i \in Q_1} x_i = s_1, \sum_{x_i \in Q_2} x_i = s_2) = \begin{cases} \frac{s_1}{l_1} & if\ x_i \in Q_1 \\ \frac{s_2}{l_2} & if\ x_i \in Q_2. \end{cases} \tag{7}$$

Proof. Let $X \sim B(m, p)$ and $Y \sim B(n, p)$ be independent binomial variables with the same probability p. By $X + Y \sim B(m + n, p)$ and Proposition 1, equation (6) is achieved. Equation (7) follows by Proposition 2.

Let Q_1 and Q_2 be sum queries such that $Q_2 \subseteq Q_1$, then Proposition 5 and Proposition 6 hold true.

Proposition 5. *Let Q_1 and Q_2 be sum queries of length l_1 and l_2 respectively, such that $Q_2 \subseteq Q_1$ and $\sum_{x_i \in Q_2} x_i = s_2$. Then, for each $k \in \{s_2, \dots, l_1 - l_2 + s_2\}$:*

$$P(\sum_{x_i \in Q_1} x_i = k|\sum_{x_i \in Q_2} x_i = s_2) = \binom{l_1 - l_2}{k - s_2} \cdot p^{k - s_2} \cdot (1 - p)^{l_1 - l_2 - (k - s_2)}. \tag{8}$$

Moreover, let us assume $\sum_{x_i \in Q_1} x_i = s_1$*, then:*

$$P(x_i = 1 | \sum_{x_i \in Q_1} x_i = s_1, \sum_{x_i \in Q_2} x_i = s_2) = \begin{cases} \frac{s_2}{l_2} & if\ x_i \in Q_2 \\ \frac{s_1 - s_2}{l_1 - l_2} & if\ x_i \in Q_1 \setminus Q_2. \end{cases} \quad (9)$$

Proof. By $P(\sum_{x_i \in Q_1} x_i = k | \sum_{x_i \in Q_2} x_i = s_2) = P(\sum_{x_i \in Q_1 \setminus Q_2} x_i = k - s_2)$, Proposition 1 and Proposition 2.

Proposition 6. *Let* Q_1 *and* Q_2 *be sum queries of length* l_1 *and* l_2 *respectively, such that* $Q_2 \subseteq Q_1$ *and* $\sum_{x_i \in Q_1} x_i = s_1$*. Then:*

$$P(\sum_{x_i \in Q_2} x_i = k | \sum_{x_i \in Q_1} x_i = s_1) = \frac{\binom{l_1 - l_2}{s_1 - k} \cdot \binom{l_2}{k}}{\binom{l_1}{s_1}},$$

for each integer k *such that* $max\{s_1 - (l_1 - l_2), 0\} \leq k \leq min\{s_1, l_2\}$*.*

Proof. The proof is omitted for lack of space, it is given in an extended working version of this paper available from the authors.

Let Q_1 and Q_2 be sum queries such that $Q_1 \cap Q_2 \neq \emptyset$, then Proposition 7 generalizes equation (9).

Proposition 7. *Let* Q_1 *and* Q_2 *be sum queries of length* l_1 *and* l_2 *respectively, such that* $Q_1 \cap Q_2 \neq \emptyset$*, and* $\mu_{s_1 s_2}$ *equal to* $\mu[\sum_{x_i \in Q_1 \cap Q_2} x_i]$ *given* $\sum_{x_i \in Q_1} x_i = s_1$ *and* $\sum_{x_i \in Q_2} x_i = s_2$*. Then:*

$$P(x_i = 1 | \sum_{x_i \in Q_1} x_i = s_1, \sum_{x_i \in Q_2} x_i = s_2) = \begin{cases} \frac{s_1 - \mu_{s_1 s_2}}{|Q_1 \setminus Q_2|} & if\ x_i \in Q_1 \setminus Q_2; \\ \frac{\mu_{s_1 s_2}}{|Q_1 \cap Q_2|} & if\ x_i \in Q_1 \cap Q_2; \\ \frac{s_2 - \mu_{s_1 s_2}}{|Q_2 \setminus Q_1|} & if\ x_i \in Q_2 \setminus Q_1. \end{cases} \quad (10)$$

Proof. The proof is omitted for lack of space, it is given in an extended working version of this paper available from the authors.

4 A Bayesian network for the on-line auditing

In reference [3], we propose a BN for dealing with sum auditing. We build the BN for the on-line sum auditing at run-time, that is, we update the BN after each user query and decide whether or not to answer the query. Each sum query $Q = \{x_{i_1}, \ldots, x_{i_l}\}$, with answer s, is implemented by means of a family, that is a child node with l parents:

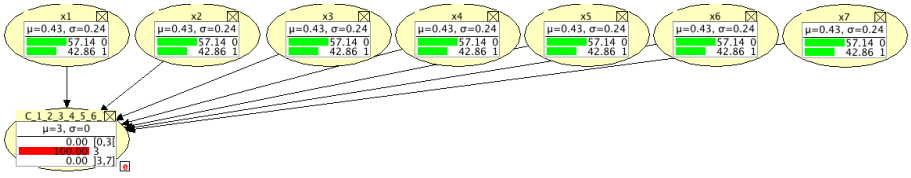

Fig. 1. $P(x_i = 1 | \sum_{x_i \in Q} x_i = 3) = 0.4286$.

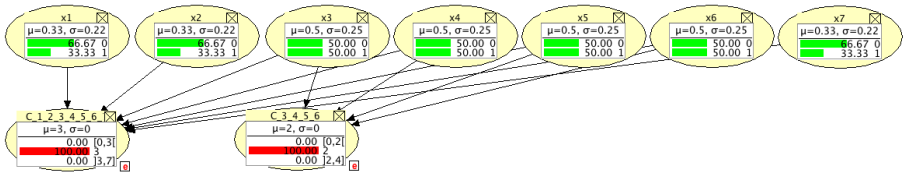

Fig. 2. $P(x_i|, \sum_{i=1}^{7} x_i = 3,\ \sum_{i=3}^{6} x_i = 2)$

- the l parents encode the sensitive variables, thus each parent has two states, that are 0 and 1;
- the child node encodes the sum $x_{i_1} + \ldots + x_{i_l}$, thus, this node has three states: $[0, s[,\ s,\ ]s, l]$. Inserting evidence on the second state, we compute $P(x_i | \sum_{x_i \in Q} x_i = s)$.

Example 3. Let us consider Example 2. The sum query is represented by BN in Figure 1. If the user submits a second query $\sum_{i=3}^{6} x_i$, with answer equal to 2, then the BN is updated as in Figure 2. Thus, if the tolerance value *tol* is chosen greater than 0.6666 then, by Definition 1, the privacy is not breached.

In this section, we propose an optimization of the BN proposed in [3]; the optimized BN is able to audit count, max and min queries, on the boolean domain, in addition to sum queries, and to compute the probabilities provided, in a formal way, in Section 3.

4.1 Complexity analysis and Bayesian Network transformations

In this section, we perform an analysis about the CPT size of the model proposed in [3] and provide a more efficient solution.

Let B be a family, that is n independent causes that contribute independently to a common effect, and d the number of the states of each node. Then, by equation (1), the total CPT size of the family is $d^{n+1} + nd$, that is $O(d^{n+1})$, and, for a large number of parents, the CPT will expand drastically.
By exploiting causal independence among several random variables, B can be decomposed in such way that its CPT size decreases. Two well known transformations are: parent divorcing [13] and temporal transformation [9]. Parent divorcing constructs a binary tree in which each node encodes a binary operator. Temporal transformation constructs a linear decomposition tree in which

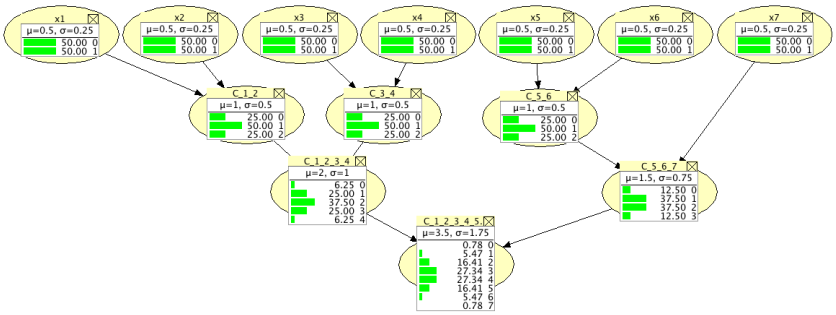

Fig. 3. Parent divorcing for $\sum_{i=1}^{7} x_i$. Before evidence on sum node ($p = \frac{1}{2}$).

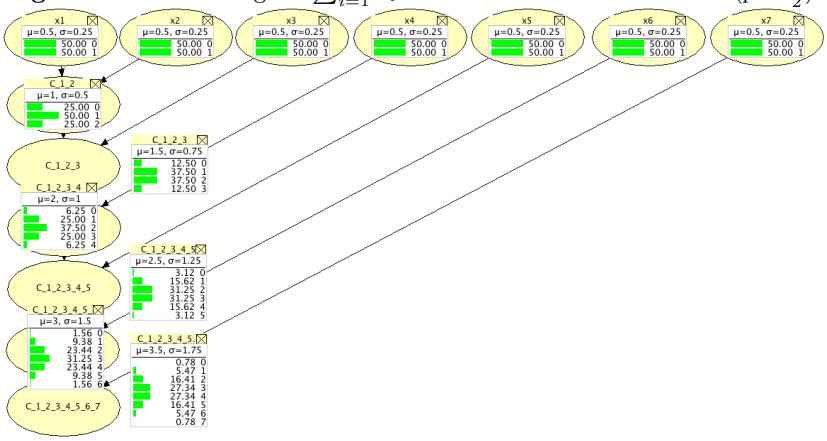

Fig. 4. Temporal transformation for $\sum_{i=1}^{7} x_i$. Before evidence on sum node ($p = \frac{1}{2}$).

each node encodes a binary operator.
By applying these transformations, the child node is transformed in $n-1$ nodes and, each of these nodes has d states and 2 parents with d states; in this way the total CPT size decreases from $O(d^{n+1})$ to $O(nd^3)$, thus, these transformations reduce the complexity from exponential to linear in the family size.

Since x_i are independent random variables and sum is an associative operation, the model proposed in [3] can be optimized by means of such kinds of transformations. In particular, the CPT size of a family encoding a single sum query in [3] is $3 \cdot 2^l + 2l$, that is $O(3 \cdot 2^l)$ where l is the length of the query. We stress that the experimentation, carried on in [3] for determining a link between the tolerance value and the probability to deny, was performed on a boolean dataset with 300 records, and, for each tolerance value, we generated in a random way 150 different queries of length less than 6; but, for bigger values, the memory requirement increases drastically.

Example 4. Parent divorcing and temporal transformation for BN in Figure 1 are shown in Figure 3 and Figure 4.

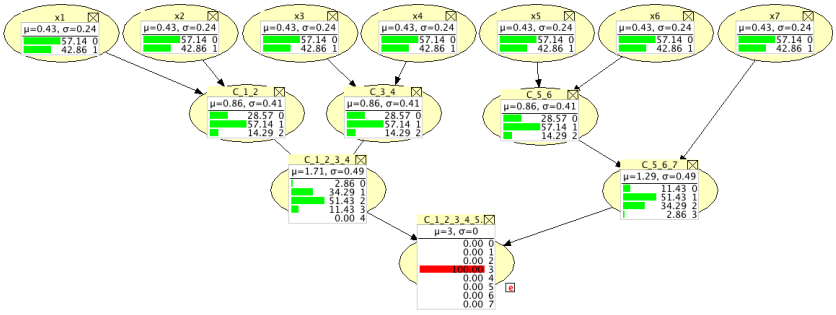

Fig. 5. Parent divorcing for $\sum_{i=1}^{7} x_i$. After evidence on sum node, $P(x_i = 1 | \sum_{i=1}^{7} x_i = 3) = \frac{3}{7} = 0.4286$.

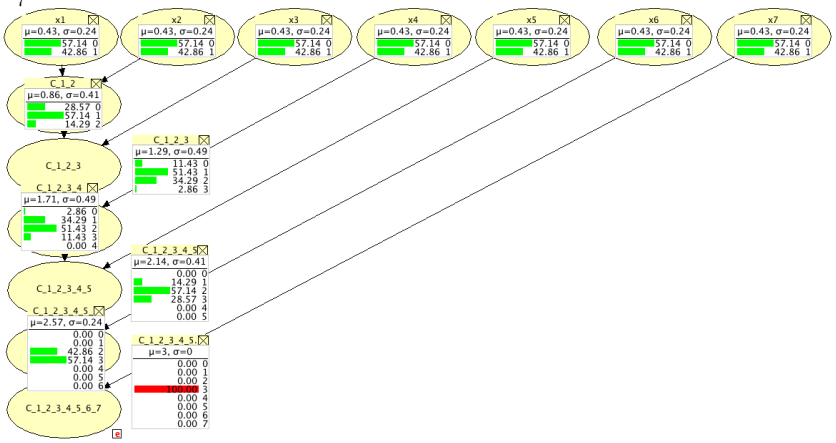

Fig. 6. Temporal transformation for $\sum_{i=1}^{7} x_i$. After evidence on sum node, $P(x_i = 1 | \sum_{i=1}^{7} x_i = 3) = \frac{3}{7} = 0.4286$.

In both parent divorcing and temporal transformation, for each node encoding a sum query, before inserting evidence, we find again binomial distribution in equation (2), and mean value μ and variance σ as in Corollary 1 (e.g., for $p = \frac{1}{2}$, the node encoding $\sum_{i=1}^{7} x_i$ has probability distribution as in Table 1, and $\mu[\sum_{i=1}^{7} x_i]$ and $\sigma[\sum_{i=1}^{7} x_i]$ as in equations (3) and (4), respectively).

By means of these transformations, CPT size of a BN for a sum query decreases from $O(2^l)$ to $O(l^3)$. By inserting evidence on the node encoding the sum query $\sum_{x_i \in Q} x_i$, we find again probabilities in Proposition 2.

Example 5. Let us consider Example 4. Then, by inserting evidence on the node encoding $\sum_{i=1}^{7} x_i$, for each x_i, we obtain the same probabilities computed in Example 2 and showed in Figure 1 (see Figure 5 and Figure 6).

By Proposition 6, the BNs, obtained applying a parent divorcing or a temporal transformation, may be furtherly optimized by unifying the states with

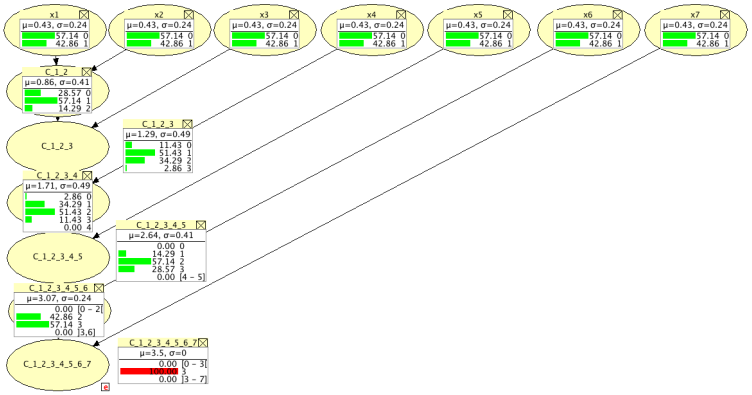

Fig. 7. Optimized temporal transformation for $\sum_{i=1}^{7} x_i$.

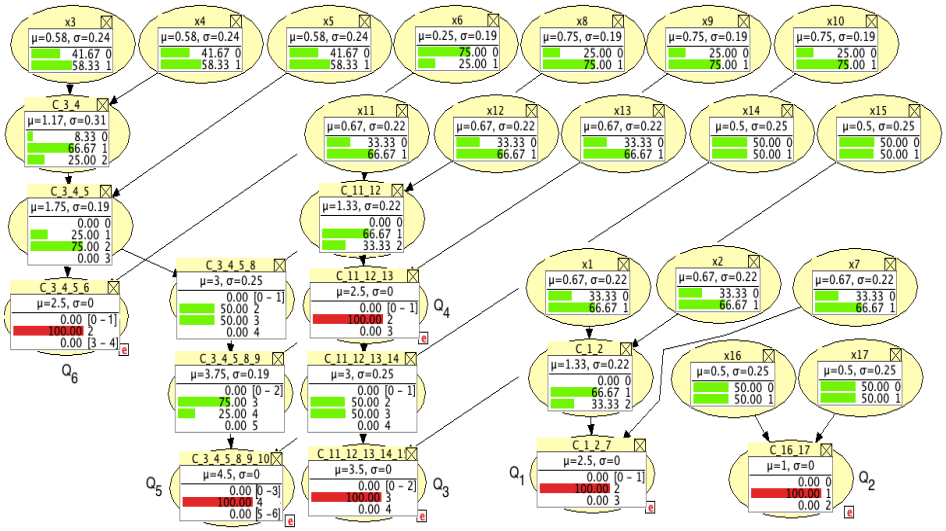

Fig. 8. Sequence of sum/count queries by means of optimized temporal transformations.

probability equal to 0. For instance, the BN in Figure 6 is transformed in the BN in Figure 7.

4.2 A Bayesian Network for on-line sum/count auditing

Like in reference [3], we build the BN for the on-line sum auditing at run-time. Moreover, by Remark 1, we are able to deal with sum/count auditing; thus, given a sequence of sum/count queries $\{Q_1, Q_2, ..., Q_{t-1}\}$, the corresponding answers $\{s_1, s_2, ..., s_{t-1}\}$ provided to an user, and the current query Q_t, the auditor has to decide if to deny Q_t, or provide the answer s_t; no value of x_i has to be disclosed.

By Section 4.1, each query can be encoded by an optimized parent divorcing or temporal transformation; in this section we consider optimized temporal transformations.

Example 6. Let us consider the tolerance value $tol = 0.8$ and the following sequence of sum/count queries:

$$Q_1 = \{x_1, x_2, x_7\},\ Q_2 = \{x_{16}, x_{17}\},\ Q_3 = \{x_{11}, x_{12}, x_{13}, x_{14}, x_{15}\},$$

$$Q_4 = \{x_{11}, x_{12}, x_{13}\},\ Q_5 = \{x_3, x_4, x_5, x_8, x_9, x_{10}\},\ Q_6 = \{x_3, x_4, x_5, x_6\}.$$

The BN encoding this sequence is shown in Figure 8. As $Q_1 \cap Q_2 = \emptyset$, in according to Proposition 4, we have:

$$P(x_i = 1 | s_1 = 2,\ s_2 = 1) = \begin{cases} \frac{s_1}{l_1} = \frac{2}{3} & if\ x_i \in Q_1 \\ \\ \frac{s_2}{l_2} = \frac{1}{2} & if\ x_i \in Q_2. \end{cases}$$

Moreover, as $Q_4 \subset Q_3$, in according to equation (9) in Proposition 5, we have:

$$P(x_i = 1 | s_3 = 3, s_4 = 2) = \begin{cases} \frac{s_4}{l_4} = \frac{2}{3} & if\ x_i \in Q_4 \\ \\ \frac{s_3 - s_4}{l_3 - l_4} = \frac{3-2}{5-3} = \frac{1}{2} & if\ x_i \in Q_3 \setminus Q_4. \end{cases}$$

Finally, as $Q_5 \cap Q_6 \neq \emptyset$, in according to Proposition 7, we have:

$$P(x_i = 1 | s_5 = 4, s_6 = 2) = \begin{cases} \frac{s_5 - \mu_{s_5 s_6}}{|Q_5 \setminus Q_6|} = \frac{4-1.75}{3} = 0.75 & if\ x_i \in Q_5 \setminus Q_6; \\ \\ \frac{\mu_{s_5 s_6}}{|Q_5 \cap Q_6|} = \frac{1.75}{3} = 0.58\overline{3} & if\ x_i \in Q_5 \cap Q_6; \\ \\ \frac{s_6 - \mu_{s_5 s_6}}{|Q_6 \setminus Q_5|} = \frac{2-1.75}{1} = 0.25 & if\ x_i \in Q_6 \setminus Q_5. \end{cases}$$

Since each sensitive value is disclosed with probability less than tol, the privacy is not breached.
We stress that the CPT size of the BN in Figure 8 is 302(*8 bytes). The same queries, represented by means of the BN proposed in [3], require a CPT size equal to 1206(*8 bytes).

4.3 A Bayesian Network for on-line sum/count/max/min auditing

By Remark 1 and Proposition 3, the BN used for on-line sum/count auditing can be used for max and min queries in addition to sum and count queries; of course, in order to preserve the privacy, the min value (resp. max) has to be 0 (resp. 1).

Example 7. Let us consider $Q = \{x_{11}, x_{12}, x_{13}, x_{14}, x_{15}\}$, with prior probabilities $P(x_i = 1) = \frac{1}{2}$. If the user asks the max value $M = max\{x_i\}_{x_i \in Q}$, and $M = 1$, then, in according to Proposition 3, with $p = \frac{1}{2}$, we have:

$$P(x_i = 1 | M = 1) = \frac{16}{31} = 0.5161 \quad \forall x_i \in Q.$$

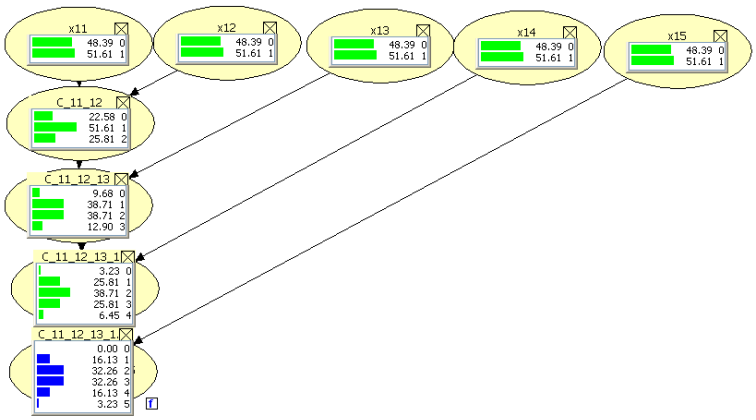

Fig. 9. Max query.

For computing these probabilities, it is enough to add likelihood on the node encoding $x_{11}+x_{12}+x_{13}+x_{14}+x_{15}$, that is $P(x_{11}+x_{12}+x_{13}+x_{14}+x_{15}=0)=0$ (see Figure 9).

5 Conclusions and future work

We propose a Bayesian approach for dealing with on-line sum/count/max/min auditing; we study the case in which the domain of the sensitive information is the boolean set. In particular, we:

- provide a formal basis for probabilistic sum/count/max/min auditing on a boolean domain;
- show that sum/count/min/max queries can be audited by means of a BN;
- optimize the BN proposed in [3]. In a first time, we reduce the CPT size of the BN encoding a query of size l, from $O(2^l)$ to $O(l^3)$, by means of a parent divorcing or a temporal transformation, then we furtherly reduce the CPT size at run-time, given the answer to the current query.

Our future work will be directed to:

- evaluate total clique size of the junction tree in addition to CPT size of the BN for the on-line sum/count/max/min auditing;
- model dependent sensitive variables with different probability distributions;
- provide a formal basis for probabilistic sum/count/max/min auditing on a discrete domain, in addition to the boolean one, and audit, on this domain, sum/count/min/max queries by means of a BN.

References

1. Adam, N.R., Worthmann, J.C.: Security-control methods for statistical databases: a comparative study. ACM Computing Surveys (CSUR) Volume 21(4) (1989)

2. Canfora, G., Cavallo, B.: A bayesian approach for on-line max auditing. In: Proocedings of The Third International Conference on Availability, Reliability and Security (ARES). pp. 1020–1027. IEEE Computer Society Press (2008)
3. Canfora, G., Cavallo, B.: A probabilistic approach for on-line sum-auditing. In: Proocedings of 2010 International Conference on Availability, Reliability and Security. pp. 303–308. IEEE Computer Society Press (2010)
4. Canfora, G., Cavallo, B.: A bayesian approach for on-line max and min auditing. In: Proocedings of International workshop on Privacy and Anonymity in Information Society (PAIS). pp. 12–20. ACM DL (2008)
5. Canfora, G., Cavallo, B.: Reasoning under uncertainty in on-line auditing. In: Privacy in Statistical Databases, Lecture Notes in Computer Science. vol. 5262, pp. 257–269. Springer-Verlag Berlin Heidelberg (2008)
6. Canfora, G., Cavallo, B.: A bayesian model for disclosure control in statistical databases. Data & Knowledge Engineering 68(11), 1187–1205 (2009)
7. Chin, F.Y.: Security problems on inference control for sum, max, and min queries. Journal of the ACM 33(3), 451–464 (1986)
8. Chin, F.Y., Ozsoyoglu, G.: Auditing and inference control in statistical databases. IEEE Transaction on Software Engineering SE-8(6), 574–582 (1982)
9. Heckerman, D.: Causal independence for knowledge acquisition and inference. In Proceedings of Ninth Conference on Uncertainty in Artificial Intelligence pp. 122–127 (1993)
10. Kenthapadi, K., Mishra, N., Nissim, K.: Simulatable auditing. In PODS pp. 118–127 (2005)
11. Kleinberg, J., Papadimitriou, C., Raghavan, P.: Auditing boolean attributes. Journal of Computer and System Sciences 66(1), 244–253 (2003)
12. Malvestuto, F.M., Mezzini, M., Moscarini, M.: Auditing sum-queries to make a statistical database secure. ACM Transactions on Information and System Security (TISSEC) 9(1), 31–60 (2006)
13. Olesen, K.G., Kjaerulff, U., Jensen, F., Jensen, F.V., Falck, B., Andreassen, S., Andersen, S.K.: A munin network for the median nerve - a case study in loops. Applied Artificial Intelligence 3(2-3), 385–403 (1989)
14. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann, San Francisco, CA, USA (1998)
15. Reiss, S.P.: Security in databases: A combinatorial study. Journal of the ACM 26(1), 45–57 (1979)