T-79.514 Special Course on Cryptology / Privacy Preserving Data Mining

Cryptographic Randomized Response Techniques

Ella Bingham

October 27, 2003

Abstract

In this note, we briefly review the manuscript "Cryptographic Randomized Response Techniques" by Ambainis, Jakobsson and Lipmaa. Cryptographic details are omitted in this review, and instead we focus on motivating the approach taken in the paper and on giving necessary background information for non-cryptographers.

1 Introduction

The paper discusses a setting where privacy is a precondition for the exchange of information. Such settings are encountered in e.g. elections and polls, where the respondent is only willing to reveal her personal information to a system if it can be guaranteed that it is not revealed to anybody. In an election, the votes given by the respondents are typically stored so that no connection can be made between a specific vote and a respondent. Also, the elections are typically organized by authorities whom the respondents trust, and a decision to participate in an election is not affected by concerns of privacy. However, the situation is quite different in a poll. The respondent typically has no specific reason to trust to the organizing party; the practical situation involves interaction directly between the respondent and the interviewer; the number of respondents may be significantly smaller than in an election, making it easier to draw conclusions regarding individual respondents; statistical estimates instead of exact counts are both sufficient and desirable as poll results. For these reasons, designing privacy preserving poll systems is highly motivated, although a large literature exists on election systems.

2 Randomized response techniques

The approach to privacy preservation taken in the paper is that of randomized response techniques (RRT). In the original RRT, the interviewer has no access to the actual opinion or private information given by the respondent. It is assumed that the question is posed as one for which a simple "yes/no" answer suffices, such as "Do you belong to a stigmatizing group A?". The respondent is given e.g. a biased coin and asked to tell the truth if the coin gives heads, and lie otherwise. (The coin is biased in such a way that the probability p_{ct} of telling the truth is larger than 1/2.) The interviewer, of course, cannot see the result of the coin toss, but by knowing p_{ct} he can later estimate the proportion of the population belonging to group A; this is often sufficient in a poll. Let π_A be the true proportion. It is estimated as follows. The a prior probability of answering "yes" is

$$p_{yes} = p_{ct} \cdot \pi_A + (1 - p_{ct})(1 - \pi_A) \qquad (1)$$

and its unbiased estimator is $\widehat{p_{yes}} = L/N$ where L respondents out of a total of N respondents answer "yes". Then the unbiased estimator for π_A is

$$\widehat{\pi_A} = \frac{p_{ct} - 1}{2p_{ct} - 1} + \frac{L}{N} \cdot \frac{1}{2p_{ct} - 1}. \qquad (2)$$

We will say that a respondent is of type $t = 1$ if she belongs to group A, and $t = 0$ otherwise.

Two alternative RRTs are also described in the paper. In the innocuous question method, the respondent is given two questions: the one of interest in the poll, and another completely irrelevant that does not include any privacy concerns. The form of possible answers to both questions must be the same, e.g. "yes/no". The respondent chooses between the two questions by a toss of a biased coin. The second alternative, polychtomous RRT, involves a question with multiple mutually exclusive answers $A_1, \ldots, A_m$, some of which are harmless and some of which the respondent typically wants to keep as a secret. A simple, but not only, solution is to ask the respondent to answer truthfully with a probability p_{ct} and answer A_i with a probability p_i, where all probabilities $p_{ct}, p_1, \ldots, p_m$ are fixed in advance and sum to 1.

There is an inherent problem with the pure RRT. A respondent may be willing to cheat by not lying even if asked to (or similarly, not telling the truth when asked to), or by not answering certain questions. There might be a philosophical reason to this: a human being does not want to belong to the minority. If a respondent learns (based on other people's answers) that she will belong to the minority, she might be unwilling to reveal that, as minorities are sometimes regarded as stigmatizing. In an election, people are often willing to vote the candidate who is leading the polls. Refusing to lie means that the probability of telling the truth is different from p_{ct}, which biases the estimation of π_A. Also, refusing to answer biases the estimate. It would be therefore valuable to have a procedure in which a respondent does not get any private information from the interviewer (such as the percentage of π_A among other respondents, or the estimate of the answer computed by the interviewer for this particular respondent), or a procedure in which a respondent does not get extra benefit from not obeying p_{ct} (such as biasing the estimate of π_A). For these reasons, the authors present cryptographic versions of the RRT. We will discuss them in the following section.

3 Cryptographic RRT

3.1 Privacy

The authors present Cryptographic RRT (CRRT) whose procedure (1) guarantees the privacy of the respondent, and (2) guarantees that the respondent cannot determine the outcome of the protocol before the end (otherwise she could refuse to answer to certain questions and thus bias the poll); so (2) can be interpreted as the privacy of the interviewer. Also, the procedure should be correct, meaning that in the end of the protocol, $\mathcal{I}$ either halts or receives his private output. The paper presents both strongly secure protocols (which are privacy-preserving for both parties and correct) and weakly secure (which are privacy-preserving for the respondent only and correct, but are simpler to understand and construct).

A basic requirement is that at the end of the protocol, the participants will have no information about the private inputs and outputs of their partners, except for what can be deduced from their own private inputs and outputs. In particular, the interviewer $\mathcal{I}$ will not learn the type $t_{\mathcal{R}}$ of the respondent $\mathcal{R}$, and similarly $\mathcal{R}$ will not learn the output $r_{\mathcal{R}}$ computed by $\mathcal{I}$.

As seen above, we denote by $r_{\mathcal{R}} \in \{0, 1\}$ the output that $\mathcal{I}$ computes for $\mathcal{R}$. It is similar to the "yes" answer in the basic RRT discussed in the beginning of Section 2 of this survey: the interviewer estimates π_A using Formula (2) where L is now the number of $r_{\mathcal{R}} = 1$ values in the population.

The authors also discuss related cryptographic work (biased coin flipping, binary symmetric channels, oblibious transfer, oblivious function evaluation, private information retrieval, data randomization etc.) which we omit in this survey and in the presentation.

3.2 Background for non-cryptographers

As the audience of the seminar includes a few non-cryptographers, some necessary background for the cryptographic methods presented in the paper is given here. (Taken from Goldwasser and Bellare: Lecture Notes on Cryptography, link available from the course web page)

A *group* is a set G together with some operation $*$ which obeys

1. If a, $b \in G$ then $a * b \in G$

2. $(a * b) * c = a * (b * c)$

3. There is an identity element I such that $I * a = a * I = a \, \forall a \in G$

4. Every $a \in G$ has an inverse a^{-1} such that $a * a^{-1} = a^{-1} * a = I$

For example, for integers under addition, $I = 0$ and $a^{-1} = -a$. For real numbers under multiplication, $I = 1$ and $a^{-1} = 1/a$.

A subset $S \subseteq G$ is called a *subgroup* if it is a group in its own right, under the same operation $*$.

We will use $\mathbb{Z}_p$ which is the set of integers modulo an integer p: $\mathbb{Z}_p = \{0, \ldots, p-1\}$. In other words, if we divide any integer by p then the remainder is in $\mathbb{Z}_p$.

Let G be a group and $g \in G$. Let $\langle g \rangle = \{g^i : i \geq 0\}$ be the set of the powers of g. We say that g is a *generator* of G if $\langle g \rangle = G$.

For example, consider $G = \{1, 2, 4, 5, 7, 8\} \subset \mathbb{Z}_9$. 2 is a generator of G:
$\langle 2 \rangle = \{2^0, 2^1, 2^2, 2^3, 2^4, 2^5, \ldots\} = \{1, 2, 4, 8, 7, 5\}$

If g is a generator, then for any $y \in G$ there is a unique $i \in \{0, \ldots, m-1\}$ (where m is the number of elements in G) such that $g^i = y$. This i equals $\log_g(y)$ and takes exponential time to find.

3.3 Protocol 1

Protocol 1 contains a variant of the Naor-Pinkas 1-out-of-n oblivious transfer and Pedersen's commitment scheme as subroutines. The protocol is roughly as follows. First some basic properties:

- p and q are primes such that q divides $p-1$. The public key consists of g and h that are two generators of G that is a unique subgroup of $\mathbb{Z}_p$, of size q.
- The encryption is based on the fact that even if g and h are known, their mutual logarithms $\log_g h$ and $\log_h g$ are hard to compute in $\mathbb{Z}_p$. Thus $g^\mu h^v$ is hard to invert (here μ is the message, and v is picked at random from $\mathbb{Z}_q$).
- $n \in \mathbb{N}$ is the size of an imaginary database that is constructed in the protocol, and $\ell \in \mathbb{N}$ such that $p_{ct} = \ell/n > 1/2$.

Then the precomputation step:

- The respondent $\mathcal{R}$ prepares n random bits $\mu_i \in \{0, 1\}$ for $i = 1, \ldots, n$, such that $\sum_i \mu_i = \ell$ if her type is $t = 1$ and $\sum_i \mu_i = n - \ell$ if $t = 0$. (Thus $p_{ct} = \ell/n$ is the probability that a randomly picked bit equals her type). Additionally, she sets $\mu_{n+1} \leftarrow t - 1$.
- The interviewer $\mathcal{I}$ chooses $\sigma \in \{1, \ldots, n\}$

And the interactive step:

- $\mathcal{I}$ picks a and b at random from $\mathbb{Z}_q$ and sends g^a, g^b and $g^{ab-\sigma+1}$ to $\mathcal{R}$.
- $\mathcal{R}$ repeats the following for all $i \in \{1, \ldots, n\}$: Pick r_i and s_i at random from $\mathbb{Z}_q$. (This r_i is not related to $r_\mathcal{R}$, $\mathcal{I}$'s private output.) Compute $w_i \leftarrow g^{r_i}(g^a)^{s_i} = g^{r_i + as_i}$ and $v_i \leftarrow (g^b)^{r_i}(g^{ab-\sigma+1}g^{i-1})^{s_i} = g^{(r_i+as_i)b+(i-\sigma)s_i}$, and use v_i as a key to encrypt the answer μ_i to y_i using $y_i \leftarrow g^{\mu_i}h^{v_i}$. Send w_i and y_i to $\mathcal{I}$.
- $\mathcal{I}$ computes w_σ^b (note that when $i = \sigma$ above, then the key v_i is w_i^b) and thus gets the key v_σ to decrypt y_σ by first computing $g^{\mu_\sigma} \leftarrow y_\sigma / h^{w_\sigma^b}$ and then computing μ_σ from that. (This is the only answer that $\mathcal{I}$ can decrypt. With probability p_{ct}, this is 1, and he will conclude $r_\mathcal{R} = 1$; with probability $1 - p_{ct}$, this is 0 and $r_\mathcal{R} = 0$.)
- $\mathcal{R}$ must now prove that she created a correct database $\{\mu_1, \ldots, \mu_{n+1}\}$. This can be done very efficiently by using noninteractive zero-knowledge arguments (details are seen in the paper).
- $\mathcal{I}$ verifies the arguments, and halts if the verification fails.

3.4 Protocol 2

Protocol 2 is based on an idea that if $\mu, \nu \in \{0, 1, \ldots, n-1\}$ and $i \in \{0, 1, \ldots, d-1\}$ and $\ell \in \mathbb{N}$, then at least one of the integers $\mu + \eta + i\ell \mod n$ must be in the interval $[0, \ell - 1]$ and at least one of them must be in $[\ell, n-1]$.

The basic properties to start with are quite similar to Protocol 1, except that $p_{ct} = \ell/n$ might need to have a very specific value, and $d = \lceil 1/(1 - p_{ct}) \rceil$.

The precomputation step is

- $\mathcal{R}$ chooses a random $\mu \in \{0, 1, \ldots, n-1\}$.
- $\mathcal{I}$ chooses random $\nu \in \{0, 1, \ldots, n-1\}$ and $\sigma \in \{0, 1, \ldots, d-1\}$.

The interactive step is

- $\mathcal{R}$ commits to t and μ and sends the commitments to $\mathcal{I}$.

- $\mathcal{I}$ chooses a random ρ and commits to σ by setting $y \leftarrow C_K(\sigma; \rho)$. He sends nu and y to $\mathcal{R}$, together with a zero-knowledge argument that y is a commitment of some $i \in \{0, 1, \dots, d-1\}$.
- $\mathcal{R}$ verifies the argument. She computes for all $i \in \{0, 1, \dots, d-1\}$ a value μ'_i such that $\mu'_i = t$ if and only if $(\mu + \nu + i\ell \mod n) < \ell$. She signs y and sends her signature together with all μ'_i and a zero-knowledge argument
- $\mathcal{I}$ sets $r_{\mathcal{R}} \leftarrow \mu'_\sigma$, accompanied with $\mathcal{R}$'s signature on the commitment, so that both $\mathcal{R}$ and third parties can verify it.

3.5 Quantum cryptographic RRT

The authors also present a quantum cryptographic RRT protocol which allows using p_{ct} that is not a rational number, and which provides a relaxed form of information-theoretic security for both parties. Lower bounds are given that restrict the benefit that a cheater can obtain: even if $\mathcal{R}$ is dishonest, her vote only counts as $\leq \sqrt{2}$ votes; and if $\mathcal{I}$ gets to know $\mathcal{R}$'s private input with some probability, he is also caught cheating with another probability. The authors claim that the protocol can implemented using contemporary technology and no non-existing quantum technology is needed.

In the end of the paper, the authors present protocols for other RRT's and discuss extensions of their approaches.

4 Conclusion

The paper describes cryptographic protocols for secure polling. For a noncryptographic reader, the paper is quite difficult to comprehend, as many details, definitions and explanations are omitted. The situation is analogous to cryptographers reading data mining papers.

From a data mining point of view, the protocols do not do any harm as long as the percentages of π_A in the population can still be computed, as seems to be the case in this approach.

However, in a typical data mining situation one is not interested in the mere π_A but other behaviour of people belonging to group A. For example, if a person belongs to group A, then which products she buys in the supermarket.

Also, the procedures described in the paper might be slow if there are e.g. 10^6 respondents — a figure not uncommon in data mining applications.