ANONYMOUS AND PRIVATE COMMUNICATION

Draft of April 12, 2015 at 01 : 45

BY

PETER KAIROUZ

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Doctoral Committee:

Professor Pramod Viswanath, Chair
Professor Nikita Borisov
Professor Bruce Hajek
Professor Sewoong Oh
Professor R. Srikant

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The Internet has shaped our daily lives. On one hand, social networks like Facebook and Twitter allow people to share their precious moments and opinions with virtually anyone around the world. On the other hand, services like Google, Netflix, and Amazon allow people to look up information, watch movies, and shop online anytime, anywhere. However, with the ability to surf the web efficiently comes the danger of being monitored. Totalitarian governments worldwide monitor their populations with ease [1]. More mundanely, people often post content online, only to later suffer repercussions due to uncontrolled information spread [2].

There is an increasing tension between the need to share data and the need to preserve the privacy and anonymity of Internet users. The need for privacy appears in two main contexts: the *metadata privacy context*, as in when individuals want to broadcast their opinions anonymously, and the *data privacy context*, as in when individuals want to communicate with a potentially malicious data analyst.

1.1 The Metadata Privacy Context

In a free society, people have the right to consume and distribute information without being monitored or surveilled. It is a basic right to be able to express one's opinion *anonymously* [3]. The demand for anonymity is evident from the explosive growth of anonymous *chatrooms* and *social networks* like Rooms, Secret, Whisper, and Yik Yak [4, 5, 6, 7]. Anonymity is particularly important in nations with authoritarian governments, where the right to free expression and the personal safety of message authors hinge on anonymity. Whether one's fear is of judgment or personal danger, the ability to anonymously, quickly, and efficiently spread content is becoming a priority.

Anonymous communication has been a popular research topic for decades, starting with the famous dining cryptographers' problem. Chaum's seminal work on DC nets spawned a great deal of literature on anonymous message sharing [8, 9, 10]. Our work differs from the existing literature by considering a different adversarial model (i.e., we allow collusion between adversarial nodes), a different class of solutions based on spreading models rather than encoding, and an arbitrary network structure (instead of a fully connected network as in [8]).

In a parallel vein, "anonymous" secret-sharing applications like Secret, Whisper, and Yik Yak hide authorship information from other users. However, authorship information is stored on centralized servers, which may be accessible to governmental agencies, hackers, advertisers, and of course, the company itself [11, 12]. In these applications, users' real names and contact information are hidden by the application itself, either via a pseudonym or no name at all. Secret, Whisper, and Yik Yak share a user's posts with members of her extended online network or geographic neighbors without revealing her identity. Rooms allows users to join and create chat rooms under a pseudonym (as is standard in most chat rooms). Despite the rise in popularity of these services, they do not provide true anonymity. Messages and authors can be linked on the centralized servers that store content for these networks—servers that may be visible to government agencies, hackers, and of course the company itself. Indeed, a recent article by the Guardian newspaper revealed that Whisper retains users' posts indefinitely (including deleted posts), along with timestamps and geographical locations [12]. Furthermore, Whisper employees personally monitor user activity and provide that data to the U.S. Department of Defense and the FBI. Our work differs by being fully distributed and offering *provable* anonymity guarantees.

We wish to distinguish our work from anonymous *point-to-point* communication, like Tor [13], Freenet [14], Free Haven [15], and Tarzan [16]. These services are particularly successful due to their distributed nature and routing algorithms. These systems are point-to-point in the sense that they allow Alice to communicate with Bob without Bob learning that Alice is at the other end. Our problem differs because we want to communicate with *everyone* under the constraints established by an underlying network (in our case, a social network). One could use a point-to-point anonymous tool like Tor to send the message to a public forum. This solution works if the public

forum can reach the whole network. However, most people do not subscribe to truly public forums due to overwhelming spam. Spreading content over a fixed network whenever users *approve* a message significantly decreases the risk of spreading irrelevant content, and is therefore more appealing to users. Thus the network inherently inhibits the spread of messages.

1.1.1 Completed Work

To overcome the shortcomings of centralized systems, we focus on distributed network architectures. Distributed systems lack a central point of failure and are difficult to monitor. Moreover, the incentive structures in such systems are better aligned for protecting privacy than they are for centralized service providers; that is, privacy-minded network participants may help one another achieve privacy, whereas centralized services often have a financial incentive to track user activities (e.g. targeted advertisement).

In Chapter 2, we design novel *spreading mechanisms* that disseminate messages quickly over a distributed network while preserving the anonymity of the original message author against strong adversaries. We study a global adversary which has side information on the underlying connectivity of the distributed network. Moreover, the adversary can track who has received the objectionable message. To counteract such an adversary, the system designer can control the message spreading mechanism. That is, when Alice is ready to forward a message (e.g., after she clicks "like"), the designer can control when and to whom Alice will forward this message. The system designer consequently wishes to design a spreading mechanism that makes it hard to detect the content source, while spreading the message as quickly as possible given network connectivity constraints. We provide information-theoretic guarantees, which ensure that the probability of an adversary identifying the true message author is always minimal. Precisely, we show that a simple spreading mechanism, called *adaptive diffusion*, enables the message author to hide perfectly among all nodes with the message. This work will appear in SIGMETRICS 2015 [17].

1.1.2 Proposed Research

Going next, we plan to design distributed anonymous message spreading mechanisms for more realistic adversarial models. In a more realistic setting, the adversary can be modeled as a colluding set of network nodes (spies) trying to estimate the most likely author. Spy nodes can track when and from whom they receive objectionable messages. Our initial investigations show that adaptive diffusion is capable of hiding the message author perfectly under this more general adversarial model.

Further, we plan to implement our secret sharing algorithms by building an open source, P2P secret-sharing mobile messaging applications that protects the anonymity of authors—much like Secret and Whisper, except distributed, open-source, and offering provable guarantees of anonymity. Users will be able to anonymously compose and disseminate messages over a social graph. The spreading models we have designed will provide strong anonymity guarantees against realistic adversaries.

We are in the process of building WildFire, the first truly anonymous social networking application. To communicate between users, WildFire uses an existing secure communication framework called Tox [18]. Tox is an open-source, P2P anonymous communication tool, much like a privacy-aware Skype; it shares its name with its underlying communication protocol. The Tox codebase (written in C) features the ability to share text messages, files, and video. We predict the latter two will be particularly useful in times of political dissent. Like TCP, the Tox communication protocol supports the lossless transmission of packets over IP, with the additional advantage that all communications are encrypted. We plan to overlay our anonymous spreading algorithms on this foundation.

1.2 The Data Privacy Context

Privacy is a fundamental individual right. Traditionally, individual information access was limited and corresponding privacy violations were essentially local, both temporally and geographically. In the era of big data, massive amounts of data about individuals are collected both voluntarily and involuntarily. With the ready ability to search for information and correlate it across distinct sources, privacy violation takes on an ominous note in this

information age.

Classical approaches to providing privacy guarantees involve anonymizing user information. While this seems to be a reasonable approach to protect the privacy of individuals, it is not infallible to correlation attacks: by correlating the anonymized database with another (perhaps publicly available) deanonymized database, a user's privacy could still be divulged. Early work in 1997 by Sweeney [19] demonstrated such an attack by correlating anonymized health records released by the state of Massachusetts with voter registration records. Similar deanonymization attacks have been routinely conducted in the ensuing years, despite the adoption of more sophisticated anonymization strategies [20]: AOL search logs (reported by NYTimes in 2006), Netflix collaborative filtering contest [21], Kaggle recommender system contest of Flickr data [22], and surname inference from genome datasets [23] are instances that have received widespread attention.

1.2.1 Completed Work

While correlation attacks using currently available databases are already devastating for anonymization techniques, an even larger issue is that anonymization is susceptible to future data releases. A way out of these limitations is to release randomized data.

Local differential privacy has recently surfaced as a strong measure of privacy in contexts where personal information remains private even from data analysts. Working in a setting where both the data providers and data analysts want to maximize the utility of statistical analyses performed on the released data, we study the fundamental trade-off between local differential privacy and utility. This trade-off is formulated as a constrained optimization problem: maximize utility subject to local differential privacy constraints.

In Chapter 3, we identify the *combinatorial structure* of the family of optimal privatization mechanisms for a broad class of information theoretic utility functions such as mutual information and f-divergences. We further prove that for a given utility function and privacy level, solving the privacy-utility maximization problem is equivalent to solving a finite-dimensional linear program, the outcome of which is the optimal privatization mechanism. However, solving this linear program can be computationally expensive since

it has a number of variables that is exponential in the size of the alphabet the data lives in. To account for this, we show that two simple privatization mechanisms are universally optimal in the high and low privacy regimes, and well approximate the intermediate regime. This work was partially presented at NIPS 2014 [24] and is currently under review by the Journal of Machine Learning Research (JMLR) [25].

1.2.2 Proposed Research

Going next, we plan to apply our fundamental results to real world applications. We would like to design optimal *data-dependent* privatization mechanisms for the release of medical and smart meter data. In addition, we plan to build open source applications that allow users (patients and energy consumers) to privately share their personal data with third party data analytic agencies.

CHAPTER 2

ANONYMOUS SOCIAL NETWORKING

2.1 Introduction

Microblogging platforms form a core aspect of the fabric of the present Internet; popular examples include Twitter and Facebook. Users propagate short messages (texts, images, videos) through the platform via local friendship links. The forwarding of messages often occurs through built-in mechanisms that rely on user input, such as clicking "like" or "share" with regards to a particular post. Brevity of message, fluidity of user interface, and trusted party communication combine to make these microblogging platforms a major communication mode of modern times. There has been tremendous recent interest in the privacy implications of these platforms, as evidenced by the explosive growth of *anonymous microblogging* platforms, like Secret [5], Whisper [6] and Yik Yak [7]. These platforms enable users to share messages with their friends, without leaking the identity of the message author. In such applications, it is crucial to keep anonymous the identity of the user who initially posted the message.

Existing anonymous messaging services store both messages and authorship information on centralized servers, which makes them vulnerable to government subpoenas, hacking, or direct company access. A more robust solution would be to store this information in a distributed fashion; each node would know only its own friends, and message authorship information would never be transmitted to any party. Distributed systems are more robust to monitoring due to lack of central points of failure. However, even under distributed architectures, simple anonymous messaging protocols (such as those used by commercial anonymous microblogging apps) are still vulnerable against an adversary with side information, as proved in recent advances in rumor source detection. In this work, we study in depth a basic building

block of the messaging protocol that would underpin truly anonymous social media – *broadcast a single message on a contact network with the goal of obfuscating the source* under strong adversarial conditions. We refer to social graphs among the users of the anonymous social media as contact networks.

A natural strategy to obscure source identification by an adversary would be to spread the message as fast as possible (with reliable connection to infrastructure like the Internet, this could be in principle done nearly instantaneously). If all users receive the message instantaneously, any user is equally likely to have been the source. However, this strategy is not available in many of the key real-life scenarios we are considering. For instance, in social networks, messages are spread based on users approving the message via liking, sharing or retweeting (to enable social filtering and also to avoid spamming) – this scenario naturally has inherent random delays associated with when the user happens to encounter the message and whether or not she decides to "like" the message. Indeed, standard models of rumor spreading in networks explicitly model such random delays via a *diffusion* process: messages are spread independently over different edges with a fixed probability of spreading (discrete time model) or an exponential time to spread (continuous time model).

2.1.1 Related Work

Anonymous communication has been a popular research topic for decades, starting with the famous dining cryptographers' (DC) problem. This work diverges from the vast literature on this topic [8, 9, 26, 10, 27]. We consider statistical spreading models rather than cryptographic encodings, accommodate computationally unbounded adversaries, and arbitrary network structures rather than a fully connected network.

Anonymous point-to-point communication, where a sender communicates with a receiver without the receiver learning the sender's identity, has also been well studied; examples include Tor [13], Freenet [14], Free Haven [15], and Tarzan [16]. Instead, we address the problem of broadcasting a message over an underlying contact network (in our case, a social network).

Within the realm of statistical message spreading models, the problem of detecting the origin of an epidemic or the source of a rumor has been

studied under the *diffusion* model. Recent advances in [28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38] show that it is possible to identify the source within a few hops with high probability. Drawing an analogy to epidemics, we refer to a person who has received the message as 'infected' and the act of passing the message as 'spreading the infection'. Consider an adversary who has access to the underlying *contact network* of friendship links and the snapshot of infected nodes at a certain time. The problem of finding the source of a rumor, first posed in [28], naturally corresponds to graph centrality based inference algorithms: for a continuous time model, [28, 29] used the rumor centrality measure to correctly identify the source after time T (with probability converging to a positive number for large d-regular and random trees and with probability proportional to $1/\sqrt{T}$ for lines). The probability of identifying the source increases even further when multiple infections from the same source are observed [30]. With multiple sources of infections, spectral methods have been proposed for estimating the number of sources and the set of source nodes in [31, 32]. When infected nodes are allowed to recover as in susceptible-infected-recovered (SIR) model, Jordan centrality was proposed in [33, 34] to estimate the source. In [34], it is shown that the Jordan center is still within a bounded hop distance from the true source with high probability, independent of the number of infected nodes. Under natural and diffusion-based message spreading – as seen in almost every content-sharing platform today – an adversary with some side-information can identify the rumor source with high confidence. We overcome this vulnerability by asking the reverse question: can we design messaging protocols that spread fast while protecting the anonymity of the source?

2.1.2 System and Adversarial Models

We focus on anonymous social media built atop an underlying contact network, such as Secret [5] over the network of phone contacts and Facebook friends. In such systems, the designer has some control over the spreading rate, by introducing artificial delays on top of the usual random delays due to users' approval of the messages. We model this physical setup as a discrete-time system, where any individual receiving a message approves it immediately at the next timestep, at which point the protocol determines

how much delay to introduce before sending the message to each of her uninfected neighbors. Given this control, the system designer wishes to design a spreading protocol that makes inference on the source of the message difficult. The assumption that all nodes are willing to approve and pass the message is not new. Such assumptions are common in the analysis of rumor spreading [28, 29, 34], and our deviation from those standard models is that we are operating in discrete time and approvals are immediate.

Following the adversarial model assumed in rumor source detection [28, 29, 34], we assume the adversary knows the whole underlying contact network and, at a certain time, it observes a snapshot of the state of all the nodes, i.e. who has received the message thus far. This adversary is strong in the sense that the whole contact network is revealed as well as the state of everyone in the network, but it is also limited in the sense that the adversary is not aware of when a particular node received the message or from whom. This model captures an adversary that is able to indirectly observe the contents of users' devices without actively compromising the devices; for instance, if the message in question contains the time and location of a protest, then the adversary learns a snapshot of the infection at a given point in time by observing who attends the protest. This adversarial model also captures an adversary that is able to monitor the network state more closely, but only at a high cost. As such, it cannot afford to continuously monitor state. We design a new anonymous messaging protocol, which we call *adaptive diffusion*, that is inherently distributed and provides strong anonymity guarantees under this adversarial model. We discuss other plausible adversarial models in Section 2.5.

2.1.3 Spreading Models

At time $t = 0$, a single user $v^* \in V$ starts to spread a message on a contact network $G = (V, E)$ where users and contacts are represented by nodes and edges, respectively. Upon receiving the message, a node can send the message to any of its neighbors. We assume a discrete-time system and model the delays due to user approval and intermittent network access via a deterministic delay of one time unit. Therefore, a message always propagates with a delay of at least one time unit. Our goal is to introduce appropri-

ate random delays into the system in order to obfuscate the identity of the source v^*. After T timesteps, let $V_T \subseteq V$, G_T, and $N_T \triangleq |V_T|$ denote the set of infected nodes, the subgraph of G containing only V_T, and the number of infected nodes, respectively. At a certain time T, an adversary observes the infected subgraph G_T and produces an estimate $\hat{v}$ of the source v^* of the message (with probability of detection $P_D = \mathbb{P}(\hat{v} = v^*)$). Since the adversary is assumed to not have any prior information on which node is likely to be the source, we use the maximum likelihood estimator

$$\hat{v}_{\mathrm{ML}} = \arg\max_{v \in G_T} \mathbb{P}(G_T|v). \tag{2.1}$$

We wish to achieve the following performance metrics.

(*a*) We say a protocol has an *order-optimal rate of spread* if the expected time for the message to reach n nodes scales linearly compared to the time required by the fastest spreading protocol.

(*b*) We say a protocol achieves a *perfect obfuscation* if the probability of source detection for the maximum likelihood estimator conditioned on n nodes being infected is bounded by

$$\mathbb{P}\big(\, \hat{v}_{\mathrm{ML}} = v^* | N_T = n \,\big) \;\; = \;\; \frac{1}{n} + o\Big(\frac{1}{n}\Big) \,. \tag{2.2}$$

2.1.4 Key Insights

Figure 2.1 (left) illustrates an example of the spread when the message is propagated immediately upon reception. The source is indicated by a solid circle. This scheme spreads the message fast but the source is trivially identified as the center of the infected subgraph if the contact network is an infinite tree. This is true independent of the infection size. Even if we introduce some randomness at each node, the source will still be identified within a few hops. This is due to both the fact that the source is close to some notion of the center of the infected subgraph [28, 34].

Since we do not know a priori when the adversary is going to attack, the main challenge is to ensure that the source is equally likely to be anywhere in the infection at *any given time*. Figure 2.1 (right) illustrates the main idea of our approach: we intentionally break the symmetry around the source.

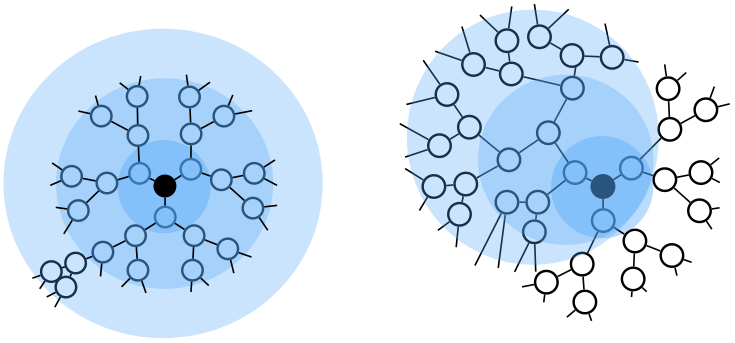


Figure 2.1: Illustration of a spread of infection when spreading immediately (left) and under the adaptive diffusion (right).

This is achieved by combining two insights illustrated in the two warm-up examples in Section 2.2. The first insight is that nodes farther away from the source should spread the infection faster. The second insight is that the spread should be coordinated in order to maintain a symmetric structure centered around a 'virtual source' node. This leads to the source node being anywhere in the infected subgraph with equal probability.

2.1.5 Contributions

We introduce a novel messaging protocol, which we call *adaptive diffusion*, with provable author anonymity guarantees against strong adversaries. Our protocol is inherently distributed and spreads messages fast, i.e., the time it takes adaptive diffusion to reach n users is at most twice the time it takes the fastest spreading scheme which immediately passes the message to all its neighbors.

We further prove that adaptive diffusion provides perfect obfuscation of the source when applied to regular tree contact networks. The source hides perfectly within all infected users, i.e., the likelihood of an infected user being the source of the infection is equal among all infected users. For a more general class of graphs which can be finite, irregular and have cycles, we provide results of numerical experiments on real-world social networks and synthetic networks showing that the protocol hides the source at nearly the best possible level of obfuscation.

2.1.6 Organization

The remainder of this chapter is organized as follows. To warm up, we introduce, in Section 2.2, two messaging protocols customized for lines and trees. Combining the key insights of these two approaches, we introduce, in Section 2.3, a new messaging protocol called *adaptive diffusion* and analyze its performance theoretically and empirically, in Section 2.4. Section 2.5 discusses limitations and future work.

2.2 Warm-up Examples

In this section, we discuss two special contact networks as warm-up examples: a line and a regular tree with degree larger than two. We provide two fully-distributed, customized messaging protocols, one for each case, and show that these protocols spread messages quickly while effectively hiding the source. However, these protocols fail to protect the identity of the source when applied to a broader class of contact networks. In particular, Protocol 1 developed for line contact networks reveals the source with high probability when applied to a regular tree with degree larger than two. Similarly, Protocol 2 developed for tree contact networks reveals the source with high probability when applied to a line. In Section 2.3, we introduce a novel messaging protocol, which we call *adaptive diffusion*, that combines the key ideas behind the two approaches presented in this section.

2.2.1 Spreading on a line

Given the contact network of an infinite line, consider the following deterministic spreading protocol. At time $t = 1$, the source node infects its left and right neighbors. At $t \geq 2$, the leftmost and rightmost infected nodes spread the message to their uninfected neighbors. Thus, the message spreads one hop to the left and one hop to the right of the true source at each timestep. This scheme spreads as fast as possible, infecting $N_T = 2T + 1$ nodes at time T, but the source is trivially identified as the center of the infection.

Adding a little bit of randomness can significantly decrease the probability of detection. Consider a discrete time *random diffusion* model with a

parameter $p \in (0,1)$ where at each time t, an infected node infects its uninfected neighbor with probability p. Using the analysis from [28] where the continuous time version of this protocol was studied, we can show that this protocol spreads fast, infecting $\mathbb{E}[N_T] = 2pT + 1$ nodes on average at time T. Further, the probability of source detection $P_D = \mathbb{P}(\hat{v}_{\mathrm{ML}} = v^*)$ for the maximum likelihood estimator scales as $1/\sqrt{p(1-p)T}$. With $p = 1/2$ for example, this gives a simple messaging protocol with a probability of source detection vanishing at a rate of $1/\sqrt{T}$.

In what follows, we show that with an appropriate choice of *time-dependent* randomness, we can achieve almost perfect source obfuscation without sacrificing the spreading rate. The key insight is to add randomness such that all the infected nodes are (almost) equally likely to have been the origin of the infection (see Figure 2.2 and Equation (2.6)). This can be achieved by adaptively choosing the spreading rate such that *the farther away the infection is from the source the more likely it is to spread*. We now apply this insight to design precisely how fast the spread should be for each infected node at any time step. A node v is designed to infect a neighbor at time $t \in \{1, 2, \ldots\}$ with probability

$$p_{v,t} \triangleq \frac{\delta_H(v, v^*) + 1}{t + 1}, \tag{2.3}$$

where $\delta_H(v, v^*)$ is the hop distance between an infected node v at the boundary of infection and the source v^*. The details of this spreading model are summarized in Protocol 1.

Protocol 1 Spreading on a line

Require: contact network $G = (V, E)$, source v^*, time T
Ensure: infected subgraph G_T

$G_0 \leftarrow \{v^*\}$
$\delta_H(v^* - 1, v^*) \leftarrow 1$ and $\delta_H(v^* + 1, v^*) \leftarrow 1$
$t \leftarrow 1$
for $t \leq T$ **do**
    $v \leftarrow$ rightmost node in G_t
    draw a random variable $X \sim U(0, 1)$
    if $X \leq (\delta_H(v, v^*) + 1)/(t + 1)$ **then**
        $G_t \leftarrow G_{t-1} \cup \{v + 1\}$
        $\delta_H(v + 1, v^*) \leftarrow \delta_H(v, v^*) + 1$
    $v \leftarrow$ leftmost node in G_T
    draw a random variable $Y \sim U(0, 1)$
    if $Y \leq (\delta_H(v, v^*) + 1)/(t + 1)$ **then**
        $G_t \leftarrow G_{t-1} \cup \{v - 1\}$
        $\delta_H(v - 1, v^*) \leftarrow \delta_H(v, v^*) + 1$
    $t \leftarrow t + 1$

The next proposition shows that this protocol achieves two of the main goals of an anonymous messaging protocol: order-optimal spreading rate and close to perfect obfuscation.

Proposition 2.2.1 *Suppose that the underlying contact network G is an infinite line, and one node v^* in G starts to spread a message according to Protocol 1 at time $t = 0$. At a certain time $T \geq 0$ an adversary estimates the location of the source v^* using the maximum likelihood estimator $\hat{v}_{\mathrm{ML}}$ defined in Equation* (2.1)*. The following properties hold for Protocol 1:*

(*a*) *the expected number of infected nodes at time T is $\mathbb{E}[N_T] = T + 1$;*

(*b*) *the probability of source detection at time T is upper bounded by*

$$\mathbb{P}(\hat{v}_{\mathrm{ML}} = v^*) \;\leq\; \frac{2T + 1}{(T + 1)^2}\ ;\ \textit{and} \tag{2.4}$$

(*c*) *the expected hop-distance between the true source v^* and its estimate $\hat{v}_{\mathrm{ML}}$ is lower bounded by*

$$\mathbb{E}[\delta_H(v^*, \hat{v}_{\mathrm{ML}})] \;\geq\; \frac{T^3}{9(T + 1)^2}\ . \tag{2.5}$$

The proof of the above proposition can be found in [17]. Compared to

the (fastest-spreading) deterministic spreading model with a spreading rate of $N_T = 2T + 1$, Protocol 1 is slower by a factor of 2. This type of constant-factor loss in the spreading rate is inevitable: the only way to deviate from the deterministic spreading model is to introduce appropriate delays. The probability of detection is $2/\mathbb{E}[N_T] + o(1/\mathbb{E}[N_T])$, which is almost perfect obfuscation up to a factor of 2. Further, the expected distance of the true source from the ML source estimate scales linearly with the size of the infection $\mathbb{E}[N_T]$, which is the best separation one can hope to achieve.

To illustrate the power of Protocol 1, we consider a fixed T and a finite ring graph of size larger than $2T + 1$, and compare the protocol to a simple random diffusion. If the source v^* is chosen uniformly at random on the ring and its message is spread according to Protocol 1, then the probability of the source being detected given a set of infected nodes V_T is

$$\mathbb{P}\big(v^* = k \,\big|\, V_T\big) \;=\; \frac{1}{|V_T|} + O\Big(\frac{1}{|V_T|^2}\Big)\,, \tag{2.6}$$

for all $k \in V_T$ and $|V_T| \leq 2T + 1$. This follows from the exact computation of the posterior distribution which is omitted for brevity. For an example with $|V_T| = 101$, Figure 2.2 illustrates how Protocol 1 flattens the posterior distribution compared to the random diffusion model. When messages are sent according to the random diffusion model, the source can only hide in the central part, which has width $O(\sqrt{T})$, leading to a probability of source detection on the order of $1/\sqrt{T}$ [28].

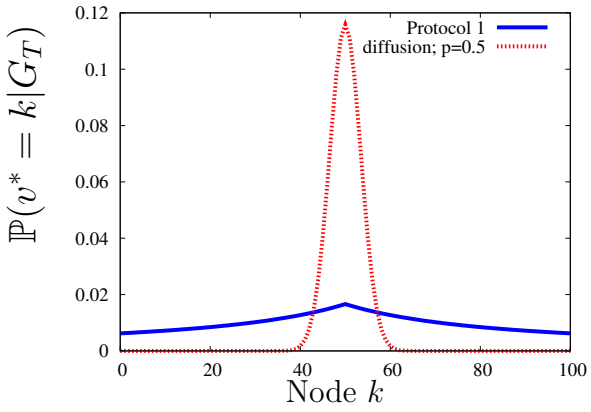


Figure 2.2: Protocol 1 has a close to uniform posterior distribution $\mathbb{P}(v^* = k \,|\, V_T = \{1, \ldots, 101\})$.

On an infinite line, Protocol 1 provides maximum protection, since the

probability of detection scales as $1/\mathbb{E}[N_T]$ for any T. When Protocol 1 is applied to regular trees with degree larger than two, the infected subgraph contains exponentially many paths starting at v^* of length close to T. In such cases, the Jordan center (i.e., the node with the smallest maximum distance to every other node in the graph) matches the source with positive probability, as shown in Figure 2.3 for different d-regular trees.

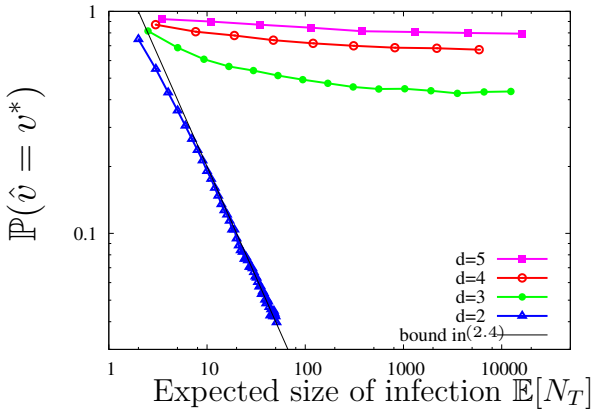


Figure 2.3: Detection probability versus the average size of infection on regular trees using Jordan center estimator. On a line ($d = 2$) the Jordan center is equal to the ML estimate.

2.2.2 Spreading on a regular tree

Consider the case when the underlying contact network is an infinite d-regular tree with d larger than two. Analogous to the line network, the fastest spreading protocol infects all the uninfected neighbors at each timestep. This spreads fast, infecting $N_T = 1 + d((d-1)^T - 1)/(d-2)$ nodes at time T, but the source is trivially identified as the center of the infected subtree. In this case, the infected subtree is a balanced regular tree where all leaves are at equal depth from the source.

Now consider a random diffusion model. At each timestep, each uninfected neighbor of an infected node is independently infected with probability p. In this case, $\mathbb{E}[N_T] = 1 + pd((d-1)^T - 1)/(d-2)$, and it was shown in [28] that the probability of correct detection for the maximum likelihood estimator of the rumor source is $\mathbb{P}(\hat{v}_{\mathrm{ML}} = v^*) \geq C_d$ for some positive constant C_d that only depends on the degree d. Hence, the source is only hidden in a constant

number of nodes close to the center, even when the total number of infected nodes is arbitrarily large.

We now present a protocol that spreads the message fast ($N_T = O((d-1)^{T/2})$) and hides the source within a constant fraction of the infected nodes ($\mathbb{P}(\hat{v}_{\mathrm{ML}} = v^*) = O(1/N_T)$). This protocol keeps the infected subtree balanced: at any time t, all the leaves of the infected subtree are at the same hop distance from its center. Further, as we will see next, the leaves of the infected subtree are equally likely to have been the source. Figure 2.4 illustrates how this protocol spreads a message on a regular tree of degree 3. At $t = 1$, node 0 (the message author) infects one of its neighbors (node 1 in this example) uniformly at random. Node 1 will be referred to as the virtual source at $t = 1$. The virtual source at time t is the center of the infected subtree at time $2t$. At $t = 2$, node 1 infects all its uninfected neighbors, making the infected subgraph G_2 a balanced tree with node 1 at the center. Among the uninfected neighbors of node 1, one node is chosen to be the new virtual source (node 2 in the example). The message then spreads to the uninfected neighbors of node 2 at time $t = 3$, and then to their neighbors at time $t = 4$ making G_4 a balanced tree with node 2 at the center. Notice that any given time t, all leaves are equally likely to have been the source. This follows form the symmetric structure of G_t.

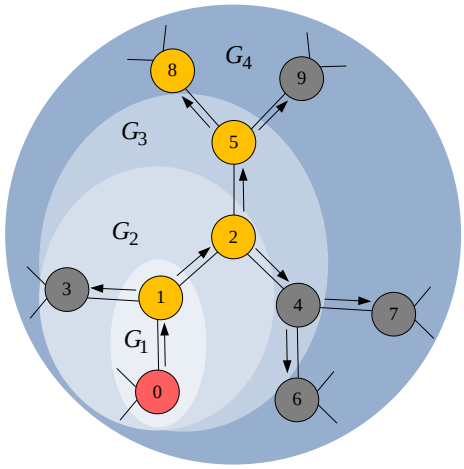


Figure 2.4: Spreading on a tree. The red node is the message source. Yellow nodes denote nodes that have been, are, or will be the center of the infected subtree.

The distributed implementation of this spreading algorithm is given in Protocol 2.

Protocol 2 Spreading on a tree

Require: contact network $G = (V, E)$, source v^*, time T
Ensure: infected subgraph G_T

$G_0 \leftarrow \{v^*\}$
$s_{1,v^*} \leftarrow 0$ and $s_{2,v^*} \leftarrow 0$
v^* selects one of its neighbors u at random
$G_1 \leftarrow G_0 \cup \{u\}$
$s_{1,u} \leftarrow 1$ and $s_{2,u} \leftarrow 1$
$t \leftarrow 2$
for $t \leq T$ **do**
    for all $v \in G_{t-1}$ with $s_{2,v} > 0$ **do**
        if $s_{1,v} = 1$ **then**
            v selects one of its uninfected neighbors u at random
            $G_t \leftarrow G_{t-1} \cup \{u\}$
            $s_{1,u} \leftarrow 1$ and $s_{2,u} \leftarrow s_{2,v} + 1$
            $s_{1,v} \leftarrow 0$
        else
            for all uninfected neighboring nodes w of v **do**
                $G_t \leftarrow G_{t-1} \cup \{w\}$
                $s_{1,w} \leftarrow 0$ and $s_{2,w} \leftarrow s_{2,v} - 1$
                $s_{2,v} \leftarrow 0$
    $t \leftarrow t + 1$

Protocol 2 ensures that the source can hide among the leaf nodes of the infected subtree, i.e. all leaves are equally likely to have been the source. Since a significant fraction of the infected nodes are at the leaf, this protocol achieves an almost *perfect obfuscation*.

Proposition 2.2.2 *Suppose that the underlying contact network G is an infinite d-regular tree with $d > 2$, and one node v^* in G starts to spread a message according to Protocol 2 at time $t = 0$. At a certain time $T \geq 1$ an adversary estimates the location of the source v^* using the maximum likelihood estimator $\hat{v}_{\mathrm{ML}}$. Then the following properties hold for Protocol 2:*
(a) *the number of infected nodes at time $T \geq 1$ is at least*

$$N_T \;\geq\; \frac{(d-1)^{(T+1)/2}}{d-2}\,; \tag{2.7}$$

(b) *the probability of source detection for the maximum likelihood estimator at time T is*

$$\mathbb{P}\big(\hat{v}_{\mathrm{ML}} = v^*\big) \;=\; \frac{d-1}{2+(d-2)N_T}\;;\;\textit{and} \tag{2.8}$$

(c) *the expected hop-distance between the true source v^* and its estimate $\hat{v}$ is lower bounded by*

$$\mathbb{E}[\delta_H(v^*, \hat{v}_{\mathrm{ML}})] \;\geq\; \frac{T}{2}\,. \tag{2.9}$$

The proof of the above proposition can be found in [17]. Equation (2.7) shows that the spreading rate of Protocol 2 is $O((d-1)^{T/2})$, which is slower than the deterministic spreading model that infects $O((d-1)^T)$ nodes at time T. This is inevitable, as we explained in relation to Proposition 2.2.1.

Although this protocol spreads fast and provides an almost perfect obfuscation on a tree with degree larger than two, it fails when the contact network is a line. There are only two leaves in a line, so at any given time T, the source can be detected with probability $1/2$, independent of the size of the infected subgraph. Another drawback of this approach, is that even in the long run, not every node receives the message. For instance, the neighbors of the source node that are not chosen in the first step are never infected. In the following section, we address these issues and propose a new messaging protocol that combines the key ideas of both spreading models presented in this section.

2.3 Adaptive Diffusion

Section 2.2 showed that by changing the infection rate and direction based on state variables, the source can hide from the adversary. In particular, the messaging protocols presented in Sections 2.2.1 and 2.2.2 provide provable anonymity guarantees for line graphs and d-regular trees with $d > 2$, respectively. However, Protocol 1 fails to protect the source on graphs with larger degree. Similarly, Protocol 2 fails to protect the source on a line and does not pass the message to some of the nodes. To overcome these challenges, we use ideas from Protocol 1 (nodes farther away from the source spread message

faster) and from Protocol 2 (keep the infected subgraph balanced and keep the source closer to the leaves), to design a protocol that achieves perfect obfuscation and spreads fast on all regular trees (including lines). We call this protocol *adaptive diffusion* to emphasize the fact that unlike diffusion, the protocol adapts the infection rate and direction as a function of time.

We step through the intuition of the adaptive diffusion spreading model with an example, partially illustrated in Figure 2.5. Suppose that the underlying contact network is an infinite d-regular tree. As illustrated in Figure 2.5, we ensure that the infected subgraph G_t at any even timestep $t \in \{2, 4, \ldots\}$ is a balanced tree of depth $t/2$, i.e. the hop distance from any leaf to the root (or the center of the graph) is $t/2$. We call the root node of G_t the "virtual source" at time t, and denote it by v_t. We use $v_0 = v^*$ to denote the true source. To keep the regular structure at even timesteps, we use the odd timesteps to transition from one regular subtree G_t to another one G_{t+2} with depth incremented by one.

Figure 2.5 illustrates two sample evolutions of infection, as per adaptive diffusion. The source $v^* = 0$ starts the infection at $t = 0$. At time $t = 1$, node 0 infects node 1 and passes the virtual source token to it, i.e. $v_2 = 1$ (we only define virtual sources for even timesteps). At time $t = 1$, node 1 infects its uninfected neighbors, nodes 2 and 3. Notice that it requires two timesteps to infect nodes $\{1, 2, 3\}$ in order to spread infection to G_2, which is a balanced tree of depth $v_2 = 1$ rooted at node 1. At time $t = 3$, the adaptive diffusion protocol has two choices, either to pass the virtual source token to

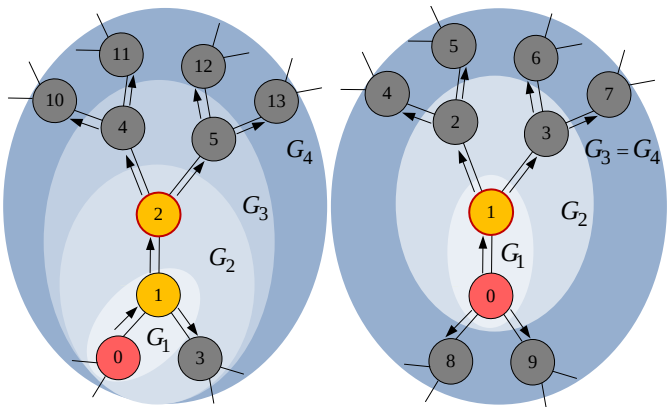


Figure 2.5: Adaptive diffusion over regular trees. Yellow nodes indicate the set of virtual sources (past and present), and for $T = 4$, the virtual source node is outlined in red.

one of node 1's neighbors that is not a previous virtual source, for example node 2 (Figure 2.5 left), or to keep the virtual source at node 1 (Figure 2.5 right). In the former case, it again takes two timesteps to spread infection to G_4, which is a balanced tree of depth 2 rooted at node $v_4 = 2$. In the latter case, only one timestep is required but we add one time delay to be consistent with the previous case. Hence, $G_3 = G_4$ which is a balanced tree of depth 2 rooted at node $v_4 = 1$. Such a random process can be defined as a time-inhomogenous (time-dependent) Markov chain over the state defined by the location of the current virtual source $\{v_t\}_{t\in\{0,2,4,\ldots\}}$.

By the symmetry of the underlying contact network (which we assume is an infinite d-regular tree) and the fact that the next virtual source is chosen uniformly at random among the neighbors of the current virtual source, it is sufficient to consider a Markov chain over the hop distance between the true source v^* and v_t, the virtual source at time t. Therefore, we design a Markov chain over the state

$$h_t \;\; = \;\; \delta_H(v^*, v_t) \; ,$$

for even t. Figure 2.5 shows an example with $(h_2, h_4) = (1, 2)$ on the left and $(h_2, h_4) = (1, 1)$ on the right.

At every even timestep, the protocol randomly determines whether to keep the virtual source token ($h_{t+2} = h_t$) or to pass it ($h_{t+2} = h_t + 1$). Using ideas from Section 2.2.1, we will construct an time-inhomogeneous Markov chain over $\{h_t\}_{t\in\{2,4,6,\ldots\}}$ by choosing appropriate transition probabilities as a function of time t and current state h_t. For an even t, we denote this probability by

$$\alpha_d(t, h) \;\; \triangleq \;\; \mathbb{P}\big(\, h_{t+2} = h_t | h_t = h \, \big) \; , \tag{2.10}$$

where the subscript d denotes the degree of the underlying contact network. For the running example, at $t = 2$, the virtual source remains at the current node (right) with probability $\alpha_3(2, 1)$, or passes the virtual source to a neighbor with probability $1 - \alpha_3(2, 1)$ (left). The parameters $\alpha_d(t, h)$ fully describe the transition probability of the Markov chain defined over $h_t \in \{1, 2, \ldots, t/2\}$. Let $p^{(t)} = [p_h^{(t)}]_{h\in\{1,\ldots,t/2\}}$ denote the distribution of the state of the Markov chain at time t, i.e. $p_h^{(t)} = \mathbb{P}(h_t = h)$. The state tran-

sition can be represented as the following $((t/2)+1) \times (t/2)$ dimensional column stochastic matrices:

$$p^{(t+2)} = \begin{bmatrix} \alpha_d(t,1) & & & & \\ 1-\alpha_d(t,1) & \alpha_d(t,2) & & & \\ & 1-\alpha_d(t,2) & \ddots & & \\ & & \ddots & \alpha_d(t,t/2) & \\ & & & 1-\alpha_d(t,t/2) & \end{bmatrix} p^{(t)} \ . \tag{2.11}$$

We treat h_t as strictly positive, because at time $t = 0$, when $h_0 = 0$, the virtual source is always passed. Thus, $h_t \geq 1$ afterwards. We design the the parameters $\alpha_d(t,h)$ to achieve perfect hiding. Precisely, at all even t, we desire $p^{(t)}$ to be

$$p^{(t)} = \frac{d-2}{(d-1)^{t/2}-1} \begin{bmatrix} 1 \\ (d-1) \\ \vdots \\ (d-1)^{t/2-1} \end{bmatrix} \in \mathbb{R}^{t/2} \ , \tag{2.12}$$

for $d > 2$ and for $d = 2$, $p^{(t)} = (2/t)\mathbf{1}_{t/2}$ where $\mathbf{1}_{t/2}$ is all ones vector in $\mathbb{R}^{t/2}$. There are $d(d-1)^{h-1}$ nodes at distance h from the virtual source, and by symmetry all of them are equally likely to have been the source:

$$\begin{aligned} \mathbb{P}(G_T | v^*, \delta_H(v^*, v_t) = h) &= \frac{1}{d(d-1)^{h-1}} p_h^{(t)} \\ &= \frac{d-2}{d((d-1)^{t/2}-1)} \ , \end{aligned}$$

for $d > 2$, which is independent of h. Hence, all the infected nodes (except for the virtual source) are equally likely to have been the source of the origin. This statement is made precise in Eq. (2.15).

Together with the desired probability distribution in Equation (2.12), this gives a recursion over t and h for computing the appropriate $\alpha_d(t,h)$'s. After some algebra and an initial state $p^{(2)} = 1$, we get that the following choice ensures the desired Equation (2.12):

$$\alpha_d(t,h) = \begin{cases} \frac{(d-1)^{t/2-h+1}-1}{(d-1)^{t/2+1}-1} & \text{if } d > 2 \\ \frac{t-2h+2}{t+2} & \text{if } d = 2 \end{cases} \tag{2.13}$$

With this choice of parameters, we show that adaptive diffusion spreads fast, infecting $N_t = O((d-1)^{t/2})$ nodes at time t and each of the nodes except for the virtual source is equally likely to have been the source.

Theorem 2.3.1 *Suppose the contact network is a d-regular tree with $d \geq 2$, and one node v^* in G starts to spread a message according to Protocol 3 at time $t = 0$. At a certain time $T \geq 0$ an adversary estimates the location of the source v^* using the maximum likelihood estimator $\hat{v}_{\mathrm{ML}}$. The following properties hold for Protocol 3:*

(a) *the number of infected nodes at time T is*

$$N_T \geq \begin{cases} \frac{2(d-1)^{(T+1)/2}-d}{(d-2)} + 1 & \text{if } d > 2 \\ T+1 & \text{if } d = 2 \end{cases} \tag{2.14}$$

(b) *the probability of source detection for the maximum likelihood estimator at time T is*

$$\mathbb{P}\left(\hat{v}_{\mathrm{ML}} = v^*\right) \leq \begin{cases} \frac{d-2}{2(d-1)^{(T+1)/2}-d} & \text{if } d > 2 \\ (1/T) & \text{if } d = 2 \end{cases} \tag{2.15}$$

(c) *the expected hop-distance between the true source v^* and its estimate $\hat{v}_{\mathrm{ML}}$ under maximum likelihood estimation is lower bounded by*

$$\mathbb{E}[d(\hat{v}_{\mathrm{ML}}, v^*)] \geq \frac{d-1}{d}\frac{T}{2}. \tag{2.16}$$

Protocol 3 describes the details of the implementation of adaptive diffusion. The first three steps are always the same. At time $t = 1$, the rumor source v^* selects, uniformly at random, one of its neighbors to be the virtual source v_2 and passes the message to it. Next at $t = 2$, the new virtual source v_2 infects all its uninfected neighbors forming G_2 (see Figure 2.5). Then node v_2 chooses to either keep the virtual source token with probability $\alpha_d(2, 1)$ or to pass it along.

If v_2 chooses to remain the virtual source i.e., $v_4 = v_2$, it passes 'infection messages' to all the leaf nodes in the infected subtree, telling each leaf to infect all its uninfected neighbors. Since the virtual source is not connected to the leaf nodes in the infected subtree, these infection messages get relayed

Protocol 3 Adaptive Diffusion

Require: contact network $G = (V, E)$, source v^*, time T, degree d
Ensure: set of infected nodes V_T

$V_T \leftarrow \{v^*\}$, $h \leftarrow 0$, $v_0 \leftarrow v^*$
v^* selects one of its neighbors u at random
$V_T \leftarrow V_T \cup \{u\}$, $h \leftarrow 1$, $v_1 \leftarrow u$
let $N(u)$ represent u's neighbors
$V_T \leftarrow V_T \cup N(u) \setminus \{v^*\}$, $v_2 \leftarrow v_1$
$t \leftarrow 3$
for $t \leq T$ **do**
    v_{t-1} selects a random variable $X \sim U(0, 1)$
    if $X \leq \alpha_d(t-1, h)$ **then**
        for all $v \in N(v_{t-1})$ **do**
            Infection Message(G,v_{t-1},v,G_T)
    else
        v_{t-1} randomly selects $u \in N(v_{t-1}) \setminus \{v_{t-2}\}$
        $h \leftarrow h + 1$
        $v_t \leftarrow u$
        for all $v \in N(v_t) \setminus \{v_{t-1}\}$ **do**
            Infection Message(G,v_t,v,V_T)
            if $t + 1 > T$ **then**
                break
            Infection Message(G,v_t,v,V_T)
    $t \leftarrow t + 2$
procedure Infection Message(G,u,v,V_T)
    if $v \in V_T$ **then**
        for all $w \in N(v) \setminus \{u\}$ **do**
            Infection Message(G,v,w,G_T)
    else
        $V_T \leftarrow V_T \cup \{v\}$

by the interior nodes of the subtree. This leads to N_t messages getting passed in total (we assume this happens instantaneously). These messages cause the rumor to spread symmetrically in all directions at $t = 3$. At $t = 4$, no more spreading occurs.

If v_2 does *not* choose to remain the virtual source, it passes the virtual source token to a randomly chosen neighbor v_4, excluding the previous virtual source (in this example, v_0). Thus, if the virtual source moves, it moves away from the true source by one hop. Once v_4 receives the virtual source token, it sends out infection messages. However, these messages do not get passed back in the direction of the previous virtual source. This causes the infection to spread asymmetrically over only one subtree of the infected graph (G_3 in left panel of Figure 2.5). In the subsequent timestep ($t = 4$), the virtual source remains fixed and passes the same infection messages again. After this second round of asymmetric spreading, the infected graph is once again symmetric about the virtual source v_4 (G_4 in left panel of Figure 2.5).

2.4 General Contact Networks

We study adaptive diffusion on general networks, and empirically show that adaptive diffusion hides the identity of the source when the underlying graph is cyclic, irregular, and finite.

2.4.1 Irregular tree networks

We first consider tree networks with potentially different degrees at the vertices. Although the degrees are irregular, we still apply the adaptive diffusion with $\alpha_{d_0}(t, h)$'s chosen for a specific d_0 that might be mismatched with the graph due to degree irregularities. There are a few challenges in this degree-mismatched adaptive diffusion. First, finding the maximum likelihood estimate of the source is not immediate, due to degree irregularities. Second, it is not a priori clear which choice of d_0 is good. We first show an efficient message passing algorithm for computing the maximum likelihood source estimate. Using this estimate, we show, via simulations, that adaptive diffusion successfully hides the source and detection probability is not too sensitive to the choice of d_0.

Efficient ML estimation. To keep the discussion simple, we assume that T is even. The same approach can be naturally extended to odd T. Since the spreading pattern in adaptive diffusion is entirely deterministic given the sequence of virtual sources at each time step, computing the likelihood $\mathbb{P}(G_T|v^* = v)$ is equivalent to computing the probability of the virtual source moving from v to v_T over T time steps. On trees, there is only one path from v to v_T and since we do not allow the virtual source to "backtrack", we only need to compute the probability of every virtual source sequence $(v_0, v_2, \ldots, v_T)$ that meets the constraint $v_0 = v$. Due to the Markov property exhibited by adaptive diffusion, we have $\mathbb{P}(G_T|\{(v_t, h_t)\}_{t\in\{2,4,\ldots,T\}}) = \prod_{\substack{t<T-1\\ t \text{ even}}} \mathbb{P}(v_{t+2}|v_t, h_t)$, where $h_t = \delta_H(v_0, v_t)$. For t even, $\mathbb{P}(v_{t+2}|v_t, h_t) = \alpha_d(t, h_t)$ if $v_t = v_{t+2}$ and $\frac{1-\alpha_d(t,h_t)}{\deg(v_t)-1}$ otherwise. Here $\deg(v_t)$ denotes the degree of node v_t in G. Given a virtual source trajectory $\mathcal{P} = (v_0, v_2, \ldots, v_T)$, let $\mathcal{J}_\mathcal{P} = (j_1, \ldots, j_{\delta_H(v_0,v_T)})$ denote the timesteps at which a new virtual source is introduced, with $1 \le j_i \le T$. It always holds that $j_1 = 2$ because after $t = 0$, the true source chooses a new virtual source and $v_2 \neq v_0$. If the virtual source at $t = 2$ were to keep the token exactly once after receiving it (it flips a biased coin at the end of $t = 2$), then $j_2 = 6$, and so forth. To find the likelihood of a node being the true source, we sum over *all* such trajectories

$$\mathbb{P}(G_T|v_0) = \sum_{\mathcal{J}_\mathcal{P}:\mathcal{P}\in\mathcal{S}(v_0,v_T,T)} \underbrace{\frac{1}{\deg(v_0)} \prod_{k=1}^{\delta_H(v_0,v_T)-1} \frac{1}{\deg(v_{j_k})-1}}_{A_{v_0}} \times \underbrace{\prod_{\substack{t<T\\ t \text{ even}}} \left(\mathbb{1}_{\{t+2\notin\mathcal{J}_\mathcal{P}\}}\alpha_d(t, h_t) + \mathbb{1}_{\{t+2\in\mathcal{J}_\mathcal{P}\}}(1 - \alpha_d(t, h_t))\right)}_{B_{v_0}}, \tag{2.17}$$

where $\mathbb{1}$ is the indicator function and

$$\mathcal{S}(v_0, v_T, T) = \{\mathcal{P} : \mathcal{P} = (v_0, v_2, \ldots, v_T) \text{ is a valid trajectory}\}.$$

Intuitively, part A_{v_0} of the above expression is the probability of choosing the set of virtual sources specified by $\mathcal{P}$, and part B_{v_0} is the probability of keeping or passing the virtual source token at the specified timesteps. Equation (2.17) holds for both regular and irregular trees. Since the path

between two nodes in a tree is unique and part A_{v_0} multiplies the degree of each node once in that path, A_{v_0} is identical for all trajectories $\mathcal{P}$. Pulling A_{v_0} out of the summation, we wish to compute the summation over all valid paths $\mathcal{P}$ of part B_{v_0} (for ease of exposition, we will use B_{v_0} to refer to this whole summation). Although there are combinatorially many valid paths, we can simplify the formula in Equation (2.17) for the particular choice of $\alpha_d(t,h)$'s defined in (2.13).

Proposition 2.4.1 *Suppose that the underlying contact network $\tilde{G}$ is an infinite tree with degree of each node larger than one. One node $\tilde{v}^*$ in $\tilde{G}$ starts to spread a message at time $t = 0$ according to Protocol 3 with the choice of $d = d_0$. At a certain even time $T \geq 0$, the maximum likelihood estimate of $\tilde{v}^*$ given a snapshot of the infected subtree $\tilde{G}_T$ is*

$$\arg \max_{v \in \tilde{G}_T \setminus \tilde{v}_T} \quad \frac{d_0}{\deg(v)} \left\{ \prod_{v' \in p(\tilde{v}_T, v) \setminus \{\tilde{v}_T, v\}} \frac{d_0 - 1}{\deg(v') - 1} \right\}, \tag{2.18}$$

where $\tilde{v}_T$ is the (Jordan) center of the infected subtree $\tilde{G}_T$, $p(\tilde{v}_T, v)$ is the unique path from $\tilde{v}_T$ to v, and $\deg(v')$ is the degree of node v'.

Consider the following observation in Figure 2.6, which was spread using the adaptive diffusion (Protocol 3) with a choice of $d_0 = 2$. Then, the Equation (2.18) can be computed easily for each node, which gives $[1/2, 1, 0, 1, 2/3, 1/2, 1/2, 1/4]$ for node $[1, 2, 3, 4, 5, 6, 7, 8]$. Hence nodes 2 and 4 are most likely. Intuitively nodes whose path to the center have small degrees are more likely. However, if we repeat this estimation assuming $d_0 = 4$, then Equation (2.18) gives $[3, 2, 0, 2, 4/3, 3, 3, 3/2]$. In this case, nodes 1, 6, and 7 are most likely. Intuitively, when d_0 is large, the adaptive diffusion tends to put the source closer to the leaf, and hence the leaf nodes are more likely to have been the source.

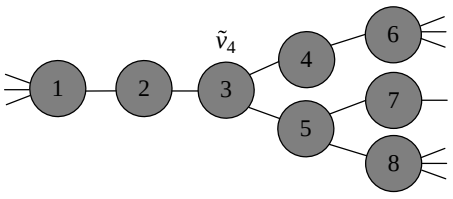


Figure 2.6: Irregular tree $\tilde{G}_4$ with virtual source $\tilde{v}_4$.

Proof 1 *We first make two observations: (a) Over regular trees,* $\mathbb{P}(G_T|u) = \mathbb{P}(G_T|w)$ *for any* $u \neq w \in G_T$*, even if they are different distances from the virtual souce. (b) Part* B_{v_0} *is identical for regular and irregular graphs, as long as the distance from the candidate source node to* v_T *is the same in both and the same* $\alpha_{d_0}(t,h)$*'s are used. That is, let* $\tilde{G}_T$ *denote an infected subtree over an* irregular *tree network, with virtual source* $\tilde{v}_T$*, and* G_T *will denote a* regular *infected subtree with virtual source* v_T*. For candidate sources* $\tilde{v}_0 \in \tilde{G}_T$ *and* $v_0 \in G_T$*, if* $\delta_H(\tilde{v}_T, \tilde{v}_0) = \delta_H(v_T, v_0) = h$*, then* $B_{v_0} = B_{\tilde{v}_0}$*. So to find the likelihood of* $\tilde{v}_0 \in \tilde{G}_T$*, we can solve for* $B_{\tilde{v}_0}$ *using the likelihood of* $v_0 \in G_T$*, and compute* $A_{\tilde{v}_0}$ *using the degree information of every node in the infected, irregular subgraph.*

We now solve for $B_{\tilde{v}_0}$*. Note that over regular graphs,* $A_v = 1/(d_0\,(d_0 - 1)^{\delta_H(v,v_T)-1})$*, where* d_0 *is the degree of the regular graph. If* G *is a regular tree, Equation* (2.17) *still applies. A crucial fact is that the* $\alpha_{d_0}(t,h)$*'s have been designed such that the likelihood are equal for all node in the regular tree. Hence,*

$$\mathbb{P}(G_T|v_0) = \underbrace{\frac{1}{d_0(d_0-1)^{\delta_H(v_0,v_T)-1}}}_{A_{v_0}} \times B_{v_0}\ , \tag{2.19}$$

is a constant that does not depend on v_0*. This gives* $B_{v_0} \propto (d_0-1)^{\delta_H(v_T,v_0)}$*. From observation (b), we have that* $B_{\tilde{v}_0} = B_{v_0}$*. Putting these together, we get that for a* $\tilde{v}_0 \in \tilde{G}_T \setminus \{\tilde{v}_T\}$*,*

$$\begin{aligned}\mathbb{P}(\tilde{G}_T|\tilde{v}_0) &= A_{\tilde{v}_0}\, B_{\tilde{v}_0} \\ &\propto \frac{(d_0-1)^{\delta_H(\tilde{v}_T,\tilde{v}_0)}}{\deg(\tilde{v}_0)\prod_{\tilde{v}'\in p(\tilde{v}_T,\tilde{v}_0)\setminus\{\tilde{v}_0,\tilde{v}_T\}}(\deg(\tilde{v}')-1)}\end{aligned}$$

After some scaling, and using that $|p(\tilde{v}_T,\tilde{v}_0)| = \delta_H(\tilde{v}_T,\tilde{v}_0)+1$*, this gives the formula in Equation* (2.18)*.*

Implementation and numerical simulations. We provide an efficient message passing algorithm for computing the ML estimate in Equation (2.18), which is naturally distributed. We then use this estimator to simulate message spreading for random irregular trees and show that the obfuscation is not too sensitive to the choice of d_0.

$A_{\tilde{v}_0}$ can be computed efficiently for irregular graphs with a simple message-passing algorithm. In this algorithm, each node $\tilde{v}$ multiplies its degree infor-

Algorithm 4 ML estimator of (2.18)

Input: infected network $\tilde{G}_T = (\tilde{V}_T, \tilde{E}_T)$, virtual source $\tilde{v}_T$, time T, the spreading model parameter d_0

Output: $\operatorname{argmax}_{\tilde{v}\in\tilde{V}_T} \mathbb{P}(\tilde{G}_T|\tilde{v}^* = \tilde{v})$

$P_{\tilde{v}} \triangleq \mathbb{P}(\tilde{G}_T|\tilde{v}^* = \tilde{v})$.
$P_{\tilde{v}_T} \leftarrow 0$
$A_{\tilde{v}} \leftarrow 1$ for $\tilde{v} \in \tilde{V}_T \setminus \{\tilde{v}_T\}$
$A_{\tilde{v}_T} \leftarrow 0$
$A \leftarrow$ Degree Message(G_T, $\tilde{v}_T$, $\tilde{v}_T$, A)
$\mathbb{P}(G_T|v_{leaf}) \leftarrow \frac{1}{d_0(d_0-1)^{T/2-1}} \prod_{\substack{t<T \\ t \text{ even}}} (1 - \alpha_{d_0}(t, \frac{t}{2}))\}$
for all $\tilde{v} \in \tilde{V}_T \setminus \{\tilde{v}_T\}$ **do**
    $h \leftarrow \delta_H(\tilde{v}, \tilde{v}_T)$
    $B_{\tilde{v}} \leftarrow \mathbb{P}(G_T|v_{leaf}) \cdot d_0 \cdot (d_0 - 1)^{h-1}$
    $P_{\tilde{v}} \leftarrow A_{\tilde{v}} \cdot B_{\tilde{v}}$
return $\operatorname{argmax}_{\tilde{v}\in\tilde{V}_T} P_{\tilde{v}}$
procedure DEGREE MESSAGE($\tilde{G}_T$, $\tilde{u}$, $\tilde{v}$, A)
    for all $\tilde{w} \in N(\tilde{v}) \setminus \{\tilde{u}\}$ **do**
        if $\tilde{v} = \tilde{u}$ **then**
            $A_{\tilde{w}} \leftarrow A_{\tilde{v}}/deg(\tilde{w})$
            Degree Message($\tilde{G}_T$, $\tilde{v}$, $\tilde{w}$, A)
        else
            if $\tilde{v}$ is not a leaf **then**
                $A_{\tilde{w}} \leftarrow A_{\tilde{v}} \cdot deg(\tilde{v})/(deg(\tilde{w}) \cdot (deg(\tilde{v}) - 1))$
                Degree Message($\tilde{G}_T$, $\tilde{v}$, $\tilde{w}$, A)
    return A

mation by a cumulative likelihood that gets passed from the virtual source to the leaves. Thus if there are $\tilde{N}_T$ infected nodes in $\tilde{G}_T$, then $A_{\tilde{v}_0}$ for every $\tilde{v}_0 \in \tilde{G}_T$ can be computed by passing $O(\tilde{N}_T)$ messages. This message-passing is outlined in procedure 'Degree Message' of Algorithm 4, which we use to find A_5 in our example. In this algorithm, the virtual source $\tilde{v}_T = 3$ starts by setting $A_2 = \frac{1}{2}$, $A_4 = \frac{1}{2}$, and $A_5 = \frac{1}{3}$. Our example stops here, but to compute other other values of $A_{\tilde{w}}$, the message passing continues. Each of the nodes $\tilde{v} \in N(3)$ in turn sets $A_{\tilde{w}}$ for *their* children $\tilde{w} \in N(\tilde{v})$; this is done by dividing $A_{\tilde{v}}$ by $deg(\tilde{w})$ and replacing the factor of $\frac{1}{deg(\tilde{v})}$ in $A_{\tilde{v}}$ with $\frac{1}{deg(\tilde{v}-1)}$. For example, node 5 would set $A_7 = \frac{A_5}{2} \cdot \frac{3}{2}$. This step is applied recursively until reaching the leaves.

We now solve for $B_{\tilde{v}_0}$. Note that over regular graphs, $A_{v_0} = 1/d_0 \cdot (d_0 - 1)^{1-\delta_H(v_0,v_T)}$, where d_0 is the degree of the regular graph. If $v_{\text{leaf}} \in G_T$ is a leaf node and G is a regular tree, we get

$$\mathbb{P}(G_T|v_{\text{leaf}}) = \underbrace{\frac{1}{d_0(d_0-1)^{T/2-1}}}_{A_{v_{\text{leaf}}}} \underbrace{\prod_{\substack{t<T \\ t \text{ even}}} (1 - \alpha_{d_0}(t, \frac{t}{2}))}_{B_{v_{\text{leaf}}}} \tag{2.20}$$

This is because the virtual source must have moved $T/2$ times for the true source to be a leaf. If candidate node $\tilde{v}_0$ is $h < T/2$ hops from $\tilde{v}_T$, then to solve for $B_{\tilde{v}_0}$ in (2.17), we make use of observations (1) and (2). Specifically, from observation (1), we have that for node v_0 with $\delta_H(v_0, v_T) = h < T/2$ over a *regular* tree,

$$\begin{aligned} \mathbb{P}(G_T|v_0) &= \mathbb{P}(G_T|v_{\text{leaf}}) \\ &= \frac{1}{d_0 \cdot (d_0-1)^{h-1}} B_{v_0}. \end{aligned} \tag{2.21}$$

From observation (2), we have that $B_{\tilde{v}_0} = B_{v_0}$. Now that we know B_{v_0}, we can multiply it by $A_{\tilde{v}_0}$ to obtain $\mathbb{P}(\tilde{G}_T|\tilde{v}_0)$. So to solve for B_5 in our example, we compute $\mathbb{P}(G_T|v_{leaf})$ for a 3-regular graph at time $T = 4$. This gives $\mathbb{P}(G_4|v_{leaf}) = A_{v_{leaf}} \cdot B_{v_{leaf}} = \frac{1}{6} \cdot (1 - \alpha_3(2,1)) = \frac{1}{9}$. Thus $B_5 = \mathbb{P}(G_4|v_{leaf}) \cdot d_0 \cdot (d_0-1)^{h-1} = \mathbb{P}(G_4|v_{leaf}) \cdot 3 \cdot (2)^0 = \frac{1}{3}$. This gives $\mathbb{P}(\tilde{G}_4|5) = A_5 \cdot B_5 = \frac{1}{9}$. The same can be done for other nodes in the graph to find the maximum likelihood source estimate.

We tested adaptive diffusion over random trees – each node's degree was i.i.d., drawn from a fixed distribution. Figure 2.7 illustrates the results of our simulations for three degree distributions, averaged over 10,000 trials. This plot shows that the probability of detection decays nearly at a rate of $1/\mathbb{E}[N_T]$ implying nearly perfect anonymity. Moreover, the message spreads exponentially quickly over the trees.

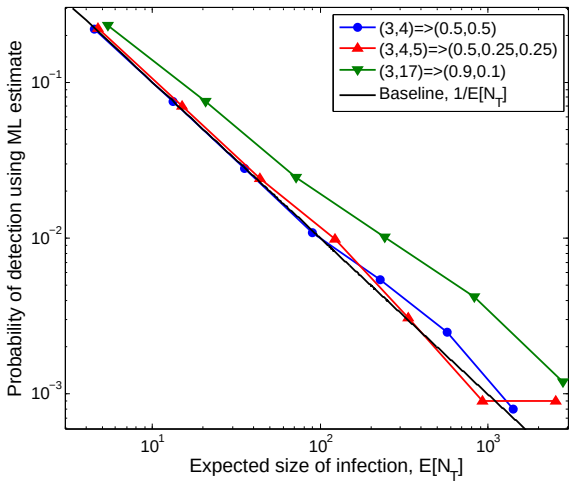


Figure 2.7: The probability of detection by the maximum likelihood estimator decays approximately as $O(1/\mathbb{E}[N_T])$ when adaptive diffusion is run over irregular trees.

2.4.2 Real World Networks

To understand how the adaptive diffusion algorithm fares in realistic scenarios which involve cycles, have irregular degrees, and is finite, we ran the adaptive diffusion algorithm over an underlying connectivity network of 10,000 Facebook users in New Orleans circa 2009, as described by the Facebook WOSN dataset [39]. We eliminated all nodes with fewer than three friends (this approach is taken by Secret so users cannot guess which of their friends originated the message), which left us with a network of 9,502 users. Over this underlying network, we selected a node uniformly at random as the rumor source, and spread the message using adaptive diffusion setting with $d_0 = \infty$, which means that the virtual source is always passed to a new node. This choice is to make the ML source estimation faster, and other choices of d_0 could outperform this naive choice. To preserve the symmetry of our

constructed trees as much as possible, we constrained each infected node to infect a maximum of three other nodes in each time step. We also give the adversary access to the undirected infection *subtree* (which explicitly identifies all pairs of nodes such that one node spread the infection to the other), which is overlaid on the underlying contact network which is not necessarily a tree. We demonstrate in simulation (Figure 2.8) that even with this strong side information, the adversary—which has access to the infected subgraph—can only identify the true message source with a small probability.

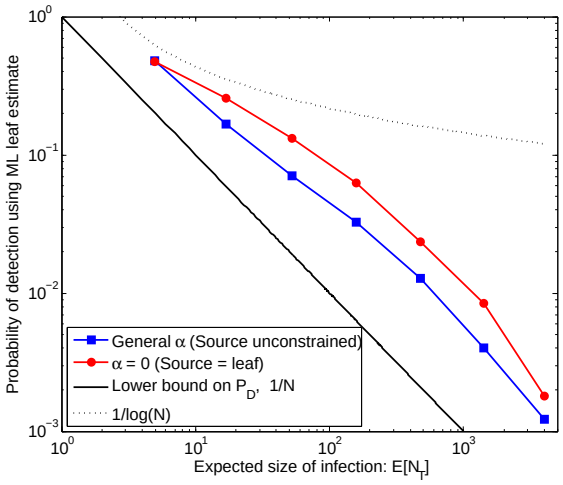


Figure 2.8: Near-ML probability of detection for the Facebook graph with adaptive diffusion.

Using the naive method of enumerating every possible message trajectory, it is computationally expensive to find the exact ML source estimate since there are 2^T possible trajectories, depending on whether the virtual source stayed or moved at each time step. We note that if the true source is one of the leaves, we can closely approximate the ML estimate *among all leaf nodes*, using the same procedure as described in 2.4.1, with one small modification: in graphs with cycles, the term $(deg(v_{j_k}) - 1)$ from equation 2.17 should be substituted with $(deg_u(v_{j_k}) - 1)$, where $deg_u(v_{j_k})$ denotes the number of uninfected neighbors of v_{j_k} at time j_k. Loops in the graph cause this value to be time-varying, and also dependent on the location of v_0, the candidate true source. We did not approximate the ML estimate for non-leaves because the simplifications used in Section 2.4.1 to compute the likelihood no longer hold, leading to an exponential increase in the problem dimension. This approach is only an approximation of the ML estimate because the virtual source could

move in a loop over the social graph (i.e., the same node could be the virtual source more than once, in nonadjacent time steps).

On average, adaptive diffusion reached 96 percent of the network within 10 time steps.

We also computed the average distance of the true source from the estimated source *over the infected subtree* (Figure 2.9). We see that as time progresses, so does the hop distance of the estimated source from the true source. In social networks, nearly everyone is within a very small number of hops (say, 6 hops [40]) from everyone else, so this computation is not as informative in this setting.

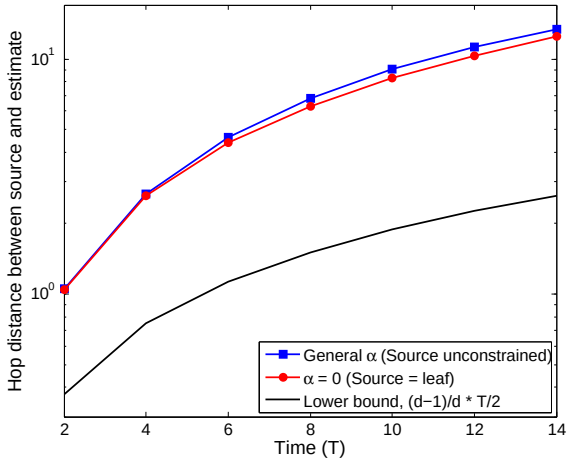


Figure 2.9: Hop distance between true source and estimated source over infection subtree for adaptive diffusion over the Facebook graph.

2.5 Discussion

Besides the adversarial model studied in this chapter, anonymous messaging applications face challenges under alternative adversarial models that can occur in practice. For instance, (a) an adversary that has corrupted a subset of network nodes through malware, bribery, or Sybil node creation can spy on the metadata on those compromised nodes on the timing of receiving a message and who the sender is (b) an adversary might create malware which prevents some nodes to follow the messaging protocol, or (c) an adversarial network provider can monitor all network activity and analyze this activity retroactively.

All these adversarial attacks increase the chance of the source being identified, which is a challenging problem for designing anonymous protocols. To a large extent, de-anonymization is an arms race in which there is always side information for an adversary to exploit. The point is to make that exploitation as expensive and difficult as possible, thereby preventing it from scaling. Within this arms race, anonymous spreading protocols ensure that adversaries cannot use message propagation patterns as a weapon.

CHAPTER 3

LOCAL DIFFERENTIAL PRIVACY

3.1 Introduction

In statistical analyses involving data from individuals, there is an increasing tension between the need to share the data and the need to protect sensitive information about the individuals. For example, users of social networking sites are increasingly cautious about their privacy, but still find it inevitable to agree to share their personal information in order to benefit from customized services such as recommendations and personalized search [41, 42]. There is a certain utility in sharing data for both data providers and data analysts, but at the same time, individuals want *plausible deniability* when it comes to sensitive information.

For such applications, there is a natural core optimization problem to be solved. Assuming both the data providers and analysts want to maximize the utility of the released data, how can they do so while preserving the privacy of participating individuals? The formulation and study of a framework addressing this fundamental tradeoff is the focus of this chapter.

3.1.1 Local differential privacy

The need for data privacy appears in two different contexts: the *local privacy* context, as in when individuals disclose their personal information (e.g., voluntarily on social network sites), and the *global privacy* context, as in when institutions release databases of information of several people or answer queries on such databases (e.g., US Government releases census data, companies like Netflix release proprietary data for others to test state of the art data analytics). In both contexts, privacy is achieved by *randomizing* the data before releasing it. We study the setting of local privacy, in which data

providers do not trust the data collector (analyst). Local privacy dates back to [43], who proposed the *randomized response* method to provide plausible deniability for individuals responding to sensitive surveys.

A natural notion of privacy protection is making inference of information beyond what is released hard. *Differential privacy* has been proposed in the global privacy context to formally capture this notion of privacy [44, 45, 46]. In a nutshell, differential privacy ensures that an adversary should not be able to reliably infer whether or not a particular individual is participating in the database query, even with unbounded computational power and access to every entry in the database except for that particular individual's data. Recently, [47] extended the notion of differential privacy to the local privacy context. Formally, consider a setting where there are n data providers each owning a data X_i defined on an input alphabet $\mathcal{X}$. The X_i's are independently sampled from some distribution P_ν parameterized by ν. A statistical privatization mechanism Q is a conditional distribution that maps $X_i \in \mathcal{X}$ stochastically to $Y_i \in \mathcal{Y}$, where $\mathcal{Y}$ is an output alphabet possibly larger than $\mathcal{X}$. The Y_i's are referred to as the privatized (sanitized) views of X_i's. In a non-interactive setting where all X_i's are independently sampled from the same distribution, the same privatization mechanism Q is used by all individuals. This setting is shown in Figure 3.1 for a special case with $n = 2$. For some non-negative ε, we follow the definition of [47] and say that a mechanism Q is *ε-locally differentially private* if

$$\sup_{S\subset\mathcal{Y},x,x'\in\mathcal{X}} \frac{Q(S|x)}{Q(S|x')} \;\leq\; e^{\varepsilon}\,, \tag{3.1}$$

where $Q(S|x) = \mathbb{P}(Y_i \in S|X_i = x)$ represents the privatization mechanism. This ensures that for small values of ε, given a privatized data Y_i, it is (almost) equally likely to have come from any data, i.e. x or x'. A small value of ε means that we require a high level of privacy and a large value corresponds to a low level of privacy. At one extreme, for $\varepsilon = 0$, the privatized output must be independent of the private data, and on the other extreme, for $\varepsilon = \infty$, the privatized output can be made equal to the private data.

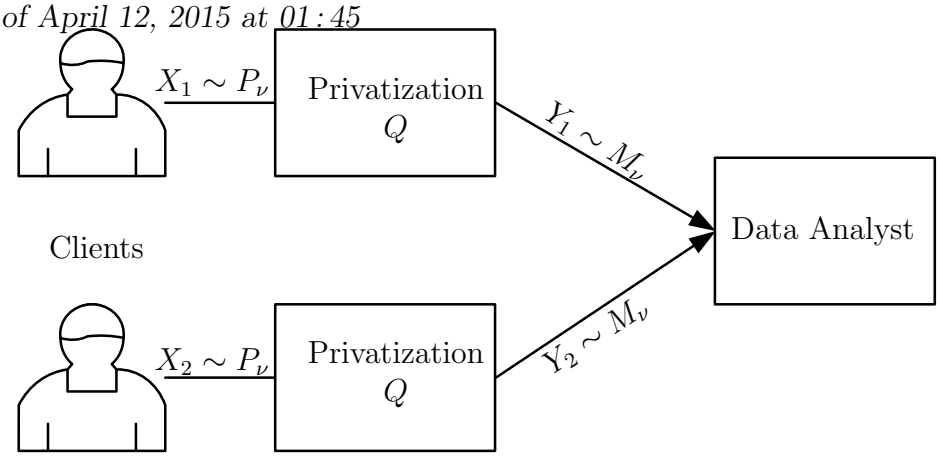


Figure 3.1: Client server model

3.1.2 Information theoretic utilities for statistical analysis

In analyses of statistical databases, the analyst is interested in the *statistics* of the data as opposed to individual records. Naturally, the utility should also be measured in terms of the distribution rather than sample quantities. Concretely, consider a client-server setting, where each client with data X_i sends a privatized version of the data Y_i, via a non-interactive ε-locally differentially private privatization mechanism Q. Assume all the clients use the same privatization mechanism denoted by Q, and each client's data is an i.i.d. sample from a distribution P_ν for some parameter ν. Given the privatized views $\{Y_i\}_{i=1}^n$, the data analyst wants to make inferences based on the induced marginal distribution

$$M_\nu(S) \equiv \sum_{x \in \mathcal{X}} Q(S|x) P_\nu(x) \,, \tag{3.2}$$

for $S \subseteq \mathcal{Y}$. We consider a broad class of convex utility functions, and identify the class of optimal mechanisms, which we call *staircase mechanisms*, in Section 3.2. We apply this framework to two specific applications: (a) hypothesis testing where the utility is measured in Kullback-Leibler divergence (Section 3.3) and (b) information preservation where the utility is measured in mutual information (Section 3.4).

In the binary hypothesis testing setting, $\nu \in \{0, 1\}$; therefore, X can be generated by one of two possible distributions P_0 and P_1. The power to discriminate data generated from P_0 to data generated from P_1 depends on the 'distance' between the marginals M_0 and M_1. To measure the ability of such statistical discrimination, our choice of utility of a particular priva-

tization mechanism Q is an information theoretic quantity called Csiszár's f-divergence defined as

$$D_f(M_0||M_1) = \sum_{x\in\mathcal{X}} f\Big(\frac{M_0(x)}{M_1(x)}\Big) M_1(x) \,, \tag{3.3}$$

for some convex function f such that $f(1) = 0$. The Kullback-Leibler (KL) divergence $D_{\mathrm{kl}}(M_0||M_1)$ is a special case with $f(x) = x\log x$, and so is the total variation $\|M_0 - M_1\|_{\mathrm{TV}}$ with $f(x) = (1/2)|x-1|$. Such f-divergences capture the quality of statistical inference, such as minimax rates of statistical estimation or error exponents in hypothesis testing [48]. As a motivating example, suppose a data analyst wants to test whether the data is generated from P_0 or P_1 based on privatized views $Y_1, \ldots, Y_n$. According to Chernoff-Stein's lemma, for a bounded type I error probability, the best type II error probability scales as $e^{-n\, D_{\mathrm{kl}}(M_0||M_1)}$. Naturally, we are interested in finding a privatization mechanism Q that minimizes the probability of error by solving the following constraint maximization problem

$$\begin{array}{ll} \underset{Q}{\text{maximize}} & D_{\mathrm{kl}}(M_0||M_1) \\ \text{subject to} & Q \in \mathcal{D}_\varepsilon \end{array} \,, \tag{3.4}$$

where $\mathcal{D}_\varepsilon$ is the set of all ε-locally differentially private mechanisms satisfying (3.1).

In the information preservation setting, X is generated from an underlying distribution P. We are interested in quantifying how much information can be preserved when releasing a private view of the data. In other words, the data provider would like to release an ε-locally differentially private view Y of X that preserves the amount of information in X as much as possible. The utility in this case is measured by the mutual information between X and Y

$$I(X;Y) = \sum_{\mathcal{X}}\sum_{\mathcal{Y}} P(x)\, Q(y|x) \log\left(\frac{Q(y|x)}{\sum_{l\in\mathcal{X}} P(l)\, Q(y|l)}\right). \tag{3.5}$$

Mutual information, as the name suggests, measures the mutual dependence between two random variables. It has been used as a criterion for feature selection and for determining the similarity between two different clusterings

of a dataset, in addition to many other applications in signal processing and machine learning. To characterize the fundamental tradeoff between privacy and information preservation, we solve the following constrained maximization problem

$$\begin{array}{ll} \underset{Q}{\text{maximize}} & I(X;Y) \\ \text{subject to} & Q \in \mathcal{D}_\varepsilon \end{array}, \tag{3.6}$$

where $\mathcal{D}_\varepsilon$ is the set of all ε-locally differentially private mechanisms satisfying (3.1).

Motivated by such applications in statistical analysis, our goal is to provide a general framework for finding optimal privatization mechanisms that maximize information theoretic utilities under local differential privacy. We demonstrate the power of our techniques in a very general setting that includes both hypothesis testing and information preservation.

3.1.3 Our contributions

We study the fundamental tradeoff between local differential privacy and a rich class of convex utility functions. This class of utilities includes several information theoretic quantities such as mutual information and f-divergences. The privacy-utility tradeoff is posed as a constrained maximization problem: maximize utility subject to local differential privacy constraints. This maximization problem is (a) nonlinear: the utility functions we consider are convex in Q; (b) non-standard: we are maximizing instead of minimizing a convex function; and (c) infinite dimensional: the space of all differentially private mechanisms is uncountable. We show, in Theorem 3.2.2, that for all utility functions considered and any privacy level ε, a *finite* family of *extremal* mechanisms (a subset of the corner points of the space of privatization mechanisms), which we call *staircase* mechanisms, contains the optimal privatization mechanism. We further prove, in Theorem 3.2.4, that solving the original problem is equivalent to solving a linear program, the outcome of which is the optimal staircase mechanism. However, solving this linear program can be computationally expensive since it has $2^{|\mathcal{X}|}$ variables. To account for this, we show that two simple staircase mechanisms (the binary and randomized response mechanisms) are optimal in the high and low privacy regimes, respectively, and well approximate the intermediate regime.

This contributes an important progress in the differential privacy area, where the privatization mechanisms have been few and almost no exact optimality results are known. As an application, we show that the effective sample size reduces from n to $\varepsilon^2 n$ under local differential privacy in the context of hypothesis testing.

We also study the fundamental tradeoff between utility and approximate differential privacy, a generalized notion of privacy that was first introduced in [49]. The techniques we develop for differential privacy do not generalize to approximate differential privacy. To account for this, we use the operational interpretation of approximate differential privacy (developed in [50]) to prove that a simple mechanism maximizes utility for all levels of privacy when the data is binary.

3.1.4 Related work

Our work is closely related to the recent work of [47] where an upper bound on $D_{\mathrm{kl}}(M_0||M_1)$ was derived under the same local differential privacy setting. Precisely, Duchi et. al. proved that the KL-divergence maximization problem in (3.4) is at most $4(e^{\varepsilon}-1)^2\|P_1-P_2\|_{TV}^2$. This bound was further used to provide a minimax bound on statistical estimation using information theoretic converse techniques such as Fano's and Le Cam's inequalities. Such tradeoffs also provide tools for comparing various notions of privacy [51].

In a similar spirit, we are also interested in maximizing information theoretic quantities of the marginals under local differential privacy. We generalize the results of [47], and provide stronger results in the sense that we (a) consider a broader class of information theoretic utilities; (b) provide explicit constructions of the optimal mechanisms; and (c) recover the existing result of [47, Theorem 1] (with a stronger condition on ε).

Our work provides a formal connection to information-theoretical notion of privacy, where privacy loss is defined as information leakage. Information leakage has been widely studied as a practical notion of privacy [52, 53]. Such a connection to differential privacy has been studied only indirectly through comparisons to how much distortion is incurred under the two notions of privacy [54]. Given a privatization mechanism, mutual information privacy is measured by the mutual information between the data and the released

output, i.e. $I(X;Y)$. We show that under ε-locally differentially, mutual information is bounded by $I(X;Y) = 0.5\varepsilon^2 \max_{A\subseteq\mathcal{X}} P(A)P(A^c) + O(\varepsilon^3)$. Moreover, we provide an explicit privatization mechanism that achieves this bound.

While there is a vast literature on differential privacy, exact optimality results are only known for a few cases. The typical recipe is to propose a differentially private mechanism inspired by the work of [44, 45, 55] and [56], and then establish its near-optimality by comparing the achievable utility to a converse, for example in principal component analysis [57, 58, 59, 60], linear queries [61, 62], logistic regression [63] and histogram release [64]. In this work, we take a different route and solve the utility maximization problem *exactly*.

Optimal differentially private mechanisms are known only in a few cases. [65] showed that the geometric noise adding mechanism is optimal (under a Bayesian setting) for monotone utility functions under count queries (sensitivity one). This was generalized by Geng et. al. (for a worst-case input setting) who proposed a family of mechanisms and proved its optimality for monotone utility functions under queries with arbitrary sensitivity [66, 67, 68]. The family of optimal mechanisms was called *staircase mechanisms* because for any y and any neighboring x and x', the ratio of $Q(y|x)$ to $Q(y|x')$ takes one of three possible values e^{ε}, $e^{-\varepsilon}$, or 1. Since the optimal mechanisms we develop also have an identical property, we retain the same nomenclature.

3.1.5 Organization

The remainder of this chapter is organized as follows. In Section 3.2, we introduce the family of staircase mechanisms, prove its optimality for a broad class of convex utility functions, and study its combinatorial structure. In Section 3.3, we study the problem of private hypothesis testing and prove that two staircase mechanisms, the binary and randomized response mechanisms, are optimal for KL-divergence in the high and low privacy regimes, respectively, and (nearly) optimal the intermediate regime. We show, in Section 3.4, similar results for mutual information. In Section 3.5, we study approximate local differential privacy, a more general notion of local pri-

vacy. Finally, we conclude this chapter with a few interesting and nontrivial extensions in Section 3.6.

3.2 Main Results

In this section, we provide a formal definition for staircase mechanisms and show that they are the optimal solutions to optimization problems of the form (3.8). Using the structure of staircase mechanisms, we propose a combinatorial representation of staircase mechanisms. This allows us to reduce the infinite dimensional nonlinear program of (3.8) to a linear program with $2^{|\mathcal{X}|}$ variables. Potentially, for any instance of the problem, one can solve this linear program to obtain the optimal privatization mechanism, albeit with significant computational challenges since the number of variables scales exponentially in the alphabet size. To address this issue, we prove, in Sections 3.3 and 3.4, that two simple staircase mechanisms, which we call the binary mechanism and the randomized response mechanism, are optimal in the high and low privacy regimes, respectively, and well approximate the intermediate regime.

3.2.1 Optimality of staircase mechanisms

For an input alphabet $\mathcal{X}$ with $|\mathcal{X}| = k$, we represent the set of ε-locally differentially private mechanisms that lead to output alphabets $\mathcal{Y}$ with $|\mathcal{Y}| = \ell$ by

$$\mathcal{D}_{\varepsilon,\ell} = \mathcal{Q}_{k\times\ell} \cap \left\{ Q \ : \ \forall\, x, x' \in \mathcal{X}, S \subseteq \mathcal{Y}, \ \Big| \ \ln \frac{Q\left(S|x\right)}{Q\left(S|x'\right)} \ \Big| \leq \varepsilon \right\},$$

where $\mathcal{Q}_{k\times\ell}$ denotes the set of all $k \times \ell$ dimensional conditional distributions. The set of all ε-locally differentially private mechanisms is given by

$$\mathcal{D}_{\varepsilon} = \cup_{\ell\in\mathbb{N}} \mathcal{D}_{\varepsilon,\ell}. \tag{3.7}$$

The set of all conditional distributions acting on $\mathcal{X}$ is given by $\mathcal{Q} = \cup_{\ell\in\mathbb{N}} \mathcal{Q}_{k,\ell}$.

We consider two types of utility functions, one for the hypothesis testing setup and another for the mutual information setup. In the hypothesis testing

setup, the utility is a function of the privatization mechanism and two priors defined on the input alphabet. Namely, $U(P_0, P_1, Q) : \mathbb{S}^k \times \mathbb{S}^k \times \mathcal{Q} \to \mathbb{R}_+$, where P_0 and P_1 are positive priors defined on $\mathcal{X}$ and $\mathbb{S}^k$ is the $(k-1)$-dimensional probability simplex. P_ν is said to be positive if $P_\nu(x) > 0$ for all $x \in \mathcal{X}$. In the information preservation setup, the utility is a function of the privatization mechanism and a prior defined on the input alphabet. Namely, $U(P, Q) : \mathbb{S}^k \times \mathcal{Q} \to \mathbb{R}_+$, where P is a positive prior defined on $\mathcal{X}$. For notational convenience, we will use $U(Q)$ to refer to both $U(P, Q)$ and $U(P_0, P_1, Q)$.

Definition 3.2.1 (Sublinear Functions) *A function $\mu(z) : \mathbb{R}^k \to \mathbb{R}$ is said to be sublinear if the following two conditions are met*

1. *$\mu(\gamma z) = \gamma \mu(z)$ for all $\gamma \in \mathbb{R}_+$.*
2. *$\mu(z_1 + z_2) \leq \mu(z_1) + \mu(z_2)$ for all $z_1, z_2 \in \mathbb{R}$.*

Let Q_y be the column of Q corresponding to $Q(y|\cdot)$ and μ be any sublinear function. We are interested in utilities that can be decomposed as a summation of sublinear functions. We study the *fundamental tradeoff between privacy and utility* by solving the following constrained maximization problem

$$\begin{aligned} \underset{Q}{\text{maximize}} \quad & U(Q) = \sum_{y \in \mathcal{Y}} \mu(Q_y) \\ \text{subject to} \quad & Q \in \mathcal{D}_\varepsilon \end{aligned} \quad . \tag{3.8}$$

This includes maximization over information theoretic quantities of interest in statistical estimation and hypothesis testing such as mutual information, total variation, KL-divergence, and χ^2-divergence [48]. Since sub-linearity implies convexity in this case, this is in general a complicated nonlinear program: we are maximizing (instead of minimizing) a convex function in Q; further, the dimension of Q might be unbounded: the optimal privatization mechanism Q^* might produce an infinite output alphabet $\mathcal{Y}$. The following theorem proves that one never needs an output alphabet larger than the input alphabet in order to achieve the maximum utility, and provides a combinatorial representation of the optimal solution.

Theorem 3.2.2 *For any sublinear function μ and any $\varepsilon \geq 0$, there exists an optimal mechanism Q^* maximizing the utility in* (3.8) *over all ε-locally differentially private mechanisms, such that*

(a) *the output alphabet size is at most the input alphabet size, i.e.* $|\mathcal{Y}| \leq |\mathcal{X}|$*; and*

(b) *for all* $y \in \mathcal{Y}$*, and* $x, x' \in \mathcal{X}$

$$\left| \ln \frac{Q^*(y|x)}{Q^*(y|x')} \right| \in \{0, \varepsilon\} \; . \tag{3.9}$$

The first claim of bounded alphabet size is more generally true for any general utility $U(Q)$ that is convex in Q (not necessarily decomposing into a sum of sublinear functions as in (3.8)). The second claim establishes that there is an optimal mechanism with an extremal structure; the absolute value of the log-likelihood ratios can only take one of the two extremal values: zero or e^ε (see Figure 3.2 for example). We refer to such a mechanism as a staircase mechanism, and define the *family of staircase mechanisms* formally as

$$\mathcal{S}_\varepsilon \; \equiv \; \{Q \,|\, \text{satisfying (3.9)}\} \; .$$

For all choices of $U(Q) = \sum_{\mathcal{Y}} \mu(Q_y)$ and any $\varepsilon \geq 0$, Theorem 3.2.2 implies that the family of staircase mechanisms includes the optimal solutions to maximization problems of the form (3.8). Notice that staircase mechanisms are ε-locally differentially private, since any Q satisfying (3.9) implies that $Q(y|x)/Q(y|x') \leq e^\varepsilon$.

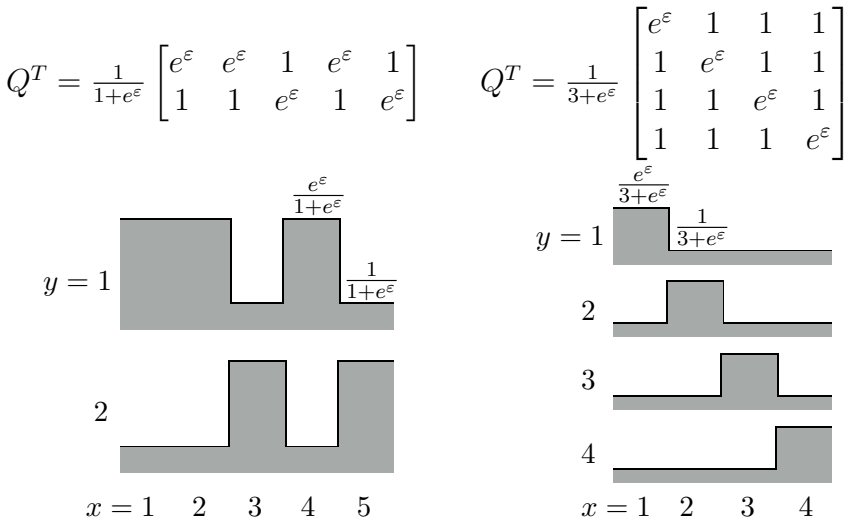


Figure 3.2: Examples of staircase mechanisms: the binary (left) and the randomized response (right) mechanisms.

For global differential privacy, we can generalize the definition of staircase mechanisms to hold for all neighboring database queries x, x' (or equivalently within some sensitivity), and recover all known existing optimal mechanisms. Precisely, the geometric mechanism shown to be optimal in [65], and the mechanisms shown to be optimal in [66, 67] (also called staircase mechanisms) are special cases of the staircase mechanisms defined above. We believe that the characterization of these extremal mechanisms and the analysis techniques developed in this chapter can be of independent interest to researchers interested in optimal mechanisms for global privacy and more general utilities.

3.2.2 Combinatorial representation of staircase mechanisms

Now that we know staircase mechanisms are optimal, we can try to combinatorially search for the best staircase mechanism for an instance of the function μ and a fixed ε. To this end, we give a simple representation of all staircase mechanisms, exploiting the fact that they are scaled copies of a finite number of patterns.

Let $Q \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{Y}|}$ be a staircase mechanism, and $k = |\mathcal{X}|$ denote the input alphabet size. Then, from the definition of staircase mechanisms, $Q(y|x)/Q(y|x') \in \{e^{-\varepsilon}, 1, e^{\varepsilon}\}$ and each column $Q(y|\cdot)$ must be proportional to one of the canonical staircase patterns we define next.

Definition 3.2.3 (Staircase Pattern Matrix) *Let b_j be the k-dimensional binary vector corresponding to the binary representation of j for $j \leq 2^k - 1$. A matrix $S^{(k)} \in \{1, e^{\varepsilon}\}^{k\times 2^k}$ is called a staircase pattern matrix if the j-th column of $S^{(k)}$ is $S^{(k)}_j = (e^{\varepsilon} - 1)\, b_{j-1} + \mathbb{1}$, for $j \in \{1, \ldots, 2^k\}$. Each column of $S^{(k)}$ is a staircase pattern.*

When $k = 3$, there are $2^k = 8$ staircase patterns and the staircase pattern matrix is given by

$$S^{(3)} = \begin{bmatrix} 1 & 1 & 1 & 1 & e^{\varepsilon} & e^{\varepsilon} & e^{\varepsilon} & e^{\varepsilon} \\ 1 & 1 & e^{\varepsilon} & e^{\varepsilon} & 1 & 1 & e^{\varepsilon} & e^{\varepsilon} \\ 1 & e^{\varepsilon} & 1 & e^{\varepsilon} & 1 & e^{\varepsilon} & 1 & e^{\varepsilon} \end{bmatrix}.$$

For all values of k, there are exactly 2^k such patterns, and any column $Q(y|\cdot)$

of Q, a staircase mechanism, is a scaled version of one of the columns of $S^{(k)}$. Using this pattern matrix, we will show that we can represent (an equivalence class of) any staircase mechanism Q as

$$Q \;=\; S^{(k)}\Theta\,, \tag{3.10}$$

where $\Theta = \mathrm{diag}(\theta)$ is a $2^k \times 2^k$ diagonal matrix and θ is a 2^k-dimensional vector representing the scaling of the columns of $S^{(k)}$. We can now formulate the problem of maximizing the utility as a linear program and prove their equivalence.

Theorem 3.2.4 *For any sublinear function μ and any $\varepsilon \geq 0$, the nonlinear program of* (3.8) *and the following linear program have the same optimal value*

$$\begin{aligned} \underset{\theta\in\mathbb{R}^{2^k}}{\textit{maximize}} \quad & \sum_{j=1}^{2^k} \mu(S_j^{(k)})\theta_j = \mu^T\theta \\ \textit{subject to} \quad & S^{(k)}\theta \;=\; \mathbb{1} \\ & \theta \geq 0\,, \end{aligned} \tag{3.11}$$

and the optimal solutions are related by (3.10)*.*

The infinite dimensional nonlinear program of (3.8) is now reduced to a finite dimensional linear program. The constraints in (3.11) ensure that we get a valid probability matrix $Q = S^{(k)}\Theta$ with rows that sum to one. One could potentially solve this LP with 2^k variables but its computational complexity scales exponentially in the alphabet size $k = |\mathcal{X}|$. For practical values of k this might not always be possible. However, in the following sections, we prove that in the high privacy regime ($\varepsilon \leq \varepsilon^*$ for some positive ε^*), there is a single optimal mechanism, which we call the *binary mechanism*, which dominates over all other mechanisms in a very strong sense for all utility functions of practical interest.

In order to understand the above theorem, observe that both the objective function and differential privacy constraints are invariant under *permutation* (or relabelling) of the columns of a privatization mechanism Q. Similarly, both the objective function and differential privacy constraints are invariant under *merging/splitting* of outputs with the same pattern. To be specific,

consider a privatization mechanism Q and suppose there exist two outputs y and y' that have the same pattern, i.e. $Q(y|\cdot) = C\,Q(y'|\cdot)$ for some positive constant C. Then, we can consider a new mechanism Q' by merging the two columns corresponding to y and y'. Let y'' denote this new output. It follows that Q' satisfies the differential privacy constraints and the resulting utility is also preserved. Precisely, using the fact that $Q(y|\cdot) = C\,Q(y'|\cdot)$, it follows that

$$\mu(Q_y) + \mu(Q_{y'}) \;\; = \;\; \mu((1+C)Q_y) \;\; = \;\; \mu(Q'_{y''}) \;,$$

by the homogeneity of μ. We can naturally define equivalence classes for staircase mechanisms that are equivalent up to a permutation of columns and merging/splitting of columns with the same pattern:

$$[Q] = \{Q' \in \mathcal{S}_\varepsilon \,|\, \exists \text{ a sequence of permutations and merge/split of columns from } Q' \text{ to } Q\} \;.$$

To represent an equivalence class, we use a mechanism in $[Q]$ that is ordered and merged to match the patterns of the pattern matrix $S^{(k)}$. For any staircase mechanism Q, there exists a possibly different staircase mechanism $Q' \in [Q]$ such that $Q' = S^{(k)}\Theta$ for some diagonal matrix Θ with nonnegative entries. Therefore, to solve optimization problems of the form (3.8), we can restrict our attention to such representatives of equivalent classes. Further, for privatization mechanisms of the form $Q = S^{(k)}\Theta$, the objective function takes the form $\sum_j \mu(S^{(k)}_j)\theta_j$, a simple linear function of Θ.

3.3 Hypothesis Testing

In this section, we study the fundamental tradeoff between local privacy and hypothesis testing. In this setting, there are n individuals each with data X_i from a distribution P_ν for a fixed $\nu \in \{0, 1\}$. Let Q be a non-interactive privatization mechanism guaranteeing ε-local differential privacy. The output of the privatization mechanism Y_i is distributed according to the induced marginal M_ν defined in (3.2). With a slight abuse of notation, we will use M_ν and P_ν to represent both probability distributions and probability mass

functions. The power to discriminate data from P_0 to the data from P_1 depends on the 'distance' between the marginals M_0 and M_1. To measure the ability of such statistical discrimination, our choice of utility of a privatization mechanism Q is an information theoretic quantity called Csiszár's f-divergence defined as

$$D_f(M_0||M_1) = \sum_{\mathcal{Y}} M_1(y) f\Big(\frac{M_0(y)}{M_1(y)}\Big) = U(P_0, P_1, Q) = U(Q) \ , \quad (3.12)$$

for some convex function f such that $f(1) = 0$. The Kullback-Leibler (KL) divergence $D_{\mathrm{kl}}(M_0||M_1)$ is a special case of f-divergence with $f(x) = x \log x$, and total variation $\|M_0 - M_1\|_{\mathrm{TV}}$ is a special case with $f(x) = (1/2)|x-1|$. Note that the f-divergence is not a distance since it might not be symmetric or satisfy triangular inequality. We are interested in characterizing the optimal solution to

$$\underset{Q \in \mathcal{D}_\varepsilon}{\text{maximize}} \qquad D_f(M_0||M_1) \ , \quad (3.13)$$

where $\mathcal{D}_\varepsilon$ is the set of all ε-locally differentially private mechanisms defined in (3.7).

A motivating example for this choice of utility is the Neyman-Pearson hypothesis testing framework [69]. Given the privatized views $\{Y_i\}_{i=1}^n$, the data analyst wants to test whether they are generated from M_0 or M_1. Let the null hypothesis be $H_0 : Y_i$'s are generated from M_0, and the alternative hypothesis $H_1 : Y_i$'s are generated from M_1. For a choice of rejection region $R \subseteq \mathcal{Y}^n$, the probability of false alarm (type I error) is $\alpha = M_0^n(R)$ and the probability of miss detection (type II error) is $\beta = M_1^n(\mathcal{Y}^n \setminus R)$. Let $\beta^\delta = \min_{R \subseteq \mathcal{Y}^n, \alpha < \alpha^*} \beta$ denote the minimum type II error achievable while keeping type I error rate at most α^*. According to Chernoff-Stein lemma [69], we know that

$$\lim_{n \to \infty} \frac{1}{n} \log \beta^{\alpha^*} = -D_{\mathrm{kl}}(M_0||M_1) \ .$$

Suppose the analyst knows P_0, P_1, and Q. Then in order to achieve optimal asymptotic error rate, one would want to maximize the KL divergence of the induced marginals, over all ε-locally differentially private mechanisms Q. The results we present in this section (Theorems 3.3.1 and 3.3.4 to be

precise) provide an explicit construction of optimal mechanisms in high and low privacy regimes. Using those optimality results, we prove a fundamental limit on the achievable error rates under differential privacy. Precisely, with data collected from an ε-locally differentially privatization mechanism, one cannot achieve an asymptotic type II error smaller than

$$\begin{aligned}\lim_{n\to\infty}\frac{1}{n}\log\beta^{\alpha^*} &\geq -\frac{(1+\delta)(e^\varepsilon-1)^2}{(e^\varepsilon+1)}\|P_0-P_1\|_{\mathrm{TV}}^2\\ &\geq -\frac{(1+\delta)(e^\varepsilon-1)^2}{2(e^\varepsilon+1)}D_{\mathrm{kl}}(P_0||P_1)\ ,\end{aligned}$$

whenever $\varepsilon\leq\varepsilon^*$, where ε^* is dictated by Theorem 3.3.1 and $\delta>0$ is some positive constant. In the equation above, the second inequality follows from Pinsker's inequality. Since $(e^\varepsilon-1)^2=O(\varepsilon^2)$ for small ε, the effective sample size is now reduced from n to $\varepsilon^2 n$. This is the price of privacy. In the low privacy regime where $\varepsilon\geq\varepsilon^*$, for ε^* dictated by Theorem 3.3.4, one cannot achieve an asymptotic type II error smaller than

$$\lim_{n\to\infty}\frac{1}{n}\log\beta^{\alpha^*}\geq -D_{\mathrm{kl}}(P_0||P_1)+(1-\delta)G(P_0,P_1)e^{-\varepsilon}\ .$$

3.3.1 Optimal staircase mechanisms

From the definition of $D_f(M_0||M_1)$, we have that

$$D_f(M_0||M_1)=\sum_{\mathcal{Y}}(P_1^TQ_y)f(P_0^TQ_y/P_1^TQ_y)=\sum_{\mathcal{Y}}\mu\left(Q_y\right),$$

where $P_\nu^TQ_y=\sum_{\mathcal{X}}P_\nu\left(x\right)Q\left(y|x\right)$ and $\mu\left(Q_y\right)=(P_1^TQ_y)f(P_0^TQ_y/P_1^TQ_y)$. For any $\gamma>0$,

$$\begin{aligned}\mu\left(\gamma Q_y\right) &= \left(P_1^T(\gamma Q_y)\right)f\left(P_0^T\left(\gamma Q_y\right)/P_1^T\left(\gamma Q_y\right)\right)\\ &= \gamma\left(P_1^TQ_y\right)f\left(P_0^TQ_y/P_1^TQ_y\right)\\ &= \gamma\mu\left(Q_y\right).\end{aligned}$$

Moreover, since the function $\phi(z,t)=tf\left(\frac{z}{t}\right)$ is convex in (z,t) for $0\leq z,t\leq 1$, then μ is convex in Q_y. Convexity and homogeniety together imply sublinearlity. Therefore, Theorems 3.2.2 and 3.2.4 apply to $D_f(M_0||M_1)$ and we have that staircases are optimal.

For a given P_0 and P_1, the *binary mechanism* is defined as a staircase mechanism with only two outputs $y \in \{0, 1\}$ satisfying (see Figure 3.2)

$$Q(0|x) = \begin{cases} \frac{e^\varepsilon}{1+e^\varepsilon} & \text{if } P_0(x) \geq P_1(x) \,, \\ \frac{1}{1+e^\varepsilon} & \text{if } P_0(x) < P_1(x) \,. \end{cases} \tag{3.14}$$

$$Q(1|x) = \begin{cases} \frac{e^\varepsilon}{1+e^\varepsilon} & \text{if } P_0(x) < P_1(x) \,, \\ \frac{1}{1+e^\varepsilon} & \text{if } P_0(x) \geq P_1(x) \,. \end{cases} \tag{3.15}$$

Although this mechanism is extremely simple, perhaps surprisingly, we will establish that this is the optimal mechanism when high level of privacy is required. Intuitively, the output is very noisy in the high privacy regime, and we are better off sending just one bit of information that tells you whether your data is more likely to have come from P_0 or P_1.

Theorem 3.3.1 *For any pair of distributions P_0 and P_1, there exists a positive ε^* that depends on P_0 and P_1 such that for any f-divergences and any positive $\varepsilon \leq \varepsilon^*$, the binary mechanism maximizes the f-divergence between the induced marginals over all ε-locally differentially private mechanisms.*

This implies that in the high privacy regime, which is a typical setting studied in much of differential privacy literature, the binary mechanism is a universally optimal solution for all f-divergences in (3.13). In particular this threshold ε^* is *universal*, in that it does not depend on the particular choice of which f-divergence we are maximizing. This is established by proving a very strong statistical dominance using Blackwell's celebrated result on comparisons of statistical experiments [70]. In a nutshell, we prove that any ε-differentially private mechanism for sufficiently small ε, and can be simulated from the output of the binary mechanism. Hence, the binary mechanism dominates over all other mechanisms and at the same time achieves the maximum divergence. A similar idea has been used previously in [50] to exactly characterize how much privacy degrades under composition.

The optimality of binary mechanisms is not just for high privacy regimes. The next theorem shows that it is *the* optimal solution of (3.13) for all ε, when the objective function is the total variation $D_f(M_0||M_1) = \|M_0 - M_1\|_{\text{TV}}$.

Theorem 3.3.2 *For any pair of distributions P_0 and P_1, and any $\varepsilon \geq 0$, the binary mechanism maximizes total variation of the induced marginals M_0 and M_1 among all ε-locally differentially private mechanisms.*

When maximizing the KL divergence between the induced marginals, we show that the binary mechanism still achieves good performance for $\varepsilon \leq C$ where C now does not depend on P_0 and P_1. For a special case of KL divergence, let OPT denote the maximum value of (3.13) and BIN denote the KL divergence when the binary mechanism is used. The next theorem shows that

$$\text{BIN} \;\geq\; \frac{1}{2(e^{\varepsilon}+1)^2}\text{OPT} \;.$$

Theorem 3.3.3 *For any ε and for any pair of distributions P_0 and P_1, the binary mechanism is an $1/(2(e^{\varepsilon}+1)^2)$ approximation of the maximum KL divergence of the induced marginals M_0 and M_1 among all ε-locally differentially private mechanisms.*

Note that $2(e^{\varepsilon}+1)^2 \leq 32$ for $\varepsilon \leq 1$, and for any $\varepsilon \leq 1$ which is the typical regime of interest in differential privacy, we can always use the simple binary mechanism and the resulting divergence is at most a constant factor away from the optimal.

The *randomized response mechanism* is defined as a staircase mechanism with the same set of outputs as the input, $\mathcal{Y} = \mathcal{X}$, satisfying (see Figure 3.2)

$$Q(y|x) \;=\; \begin{cases} \frac{e^{\varepsilon}}{|\mathcal{X}|-1+e^{\varepsilon}} & \text{if } y = x \;, \\ \frac{1}{|\mathcal{X}|-1+e^{\varepsilon}} & \text{if } y \neq x \;. \end{cases} \tag{3.16}$$

It is a randomization over the same alphabet, and we are more likely to give an honest response. We view it as a multiple choice generalization of the randomized response method proposed by [43], assuming equal level of sensitivity for all choices. We establish that this is the optimal mechanism when low level of privacy is required. Intuitively, the noise is small in the low privacy regime, and we want to send as much information about our current data as allowed, but no more. For a special case of maximizing KL divergence, we show that the *randomized response mechanism* is the optimal solution of (3.13) in the low privacy regime ($\varepsilon \geq \varepsilon^*$).

Theorem 3.3.4 *There exists a positive ε^* that depends on P_0 and P_1 such that for any P_0 and P_1, and all $\varepsilon \geq \varepsilon^*$, the randomized response mechanism maximizes the KL divergence between the induced marginals over all ε-locally differentially private mechanisms.*

3.3.2 Numerical Experiments

A typical approach for achieving ε-local differential privacy is to add geometric noise with appropriately chosen variance. For an input with alphabet size $|\mathcal{X}| = k$, this amounts to relabelling the input as integers $\{1, \ldots, k\}$ and adding geometric noise, i.e., $Q(y|x) = ((1-\varepsilon^{1/(k-1)})/(1+\varepsilon^{1/(k-1)}))\varepsilon^{|y-x|/(k-1)}$. The output is then truncated at 1 and k to preserve the support.

For 100 instances of randomly chosen P_0 and P_1 over input alphabet of size $|\mathcal{X}| = 6$, we compare the average performance of the binary, randomized response, and the geometric mechanisms to the optimal staircase mechanism. The optimal staircase mechanism is computed by solving the linear program in Equation (3.11) for each fixed pair (P_0, P_1) and ε. We plot (in Figure 3.3, left) the average performance measured by the normalized divergence $D_{\mathrm{kl}}(M_0||M_1)/D_{\mathrm{kl}}(P_0||P_1)$ for all 4 mechanisms. The average is taken over the 100 instances of P_0 and P_1. In the low privacy (large ε) regime, the randomized response achieves optimal performance as predicted, which converges to one. In the high privacy regime (small ε), the binary mechanism achieves optimal performance as predicted. In all regimes, both mechanisms significantly improve over the geometric mechanism.

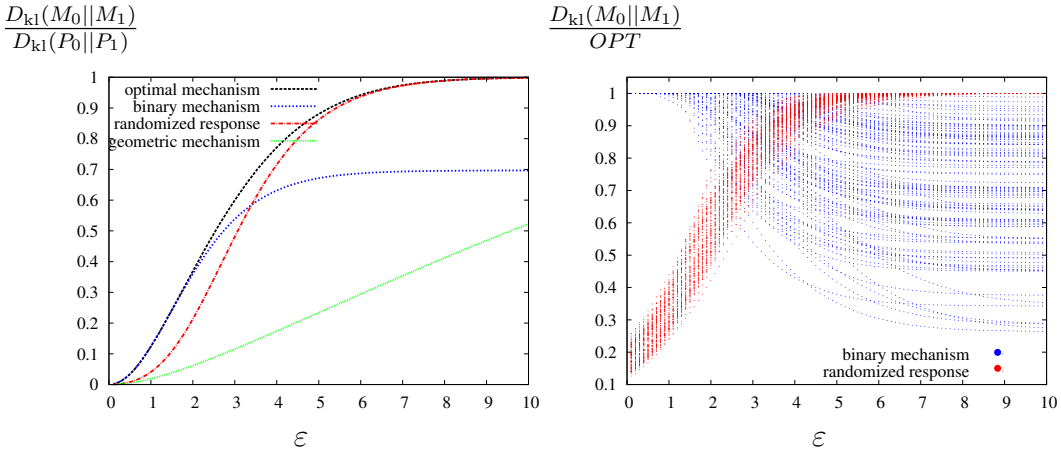


Figure 3.3: The binary and randomized response mechanisms are optimal in the high-privacy (small ε) and low-privacy (large ε) regimes, respectively, and improve over the geometric mechanism significantly (left). When the regimes are mismatched, $D_{\mathrm{kl}}(M_0||M_1)$ under these mechanisms can be as bad as 10% of the optimal one (right).

To illustrate how much worse the binary and the randomized response mechanisms can be (relative to the optimal extremal mechanism), we plot

(in Figure 3.3, right) the divergence under each mechanism normalized by the divergence under the optimal mechanism. This is done for all 100 instances of P_0 and P_1. In all instances, the binary mechanism is optimal for small ε and the randomized response mechanism is optimal for large ε. However, $D_{\mathrm{kl}}(M_0||M_1)$ under the randomized response mechanism can be as bad as 10% of the optimal one (for small ε). Similarly, $D_{\mathrm{kl}}(M_0||M_1))$ under the binary mechanism can be as bad as 25% of the optimal one (for large ε). To overcome this issue, we propose the following simple strategy: use the better among these two mechanisms. The performance of this strategy is illustrated in Figure 3.4. For various input alphabet size $|\mathcal{X}| \in \{3, 4, 5, 6\}$, we plot the performance of this mixed strategy for each value of ε and each of the 100 randomly generated instances of P_0 and P_1. This mixed strategy achieves 60% of the optimal divergence for all instances. Further, it is not sensitive to the size of the alphabet k. This strategy provides a good mechanism that can be readily used in practice for any value of ε.

3.3.3 Lower bounds

In this section, we provide converse results on the fundamental limit of differentially private mechanisms; these results follow from our main theorems and are of independent interest in other applications where lower bounds in statistical analysis are studied [71, 61, 72, 73]. For example, a bound similar to the one we present next was used to provide converse results on the sample complexity for statistical estimation with differentially private data in [47].

Corollary 3.3.5 *For any $\varepsilon \geq 0$, let Q be any conditional distribution that guarantees ε-local differential privacy. Then, for any pair of distributions P_0 and P_1 and any positive $\delta > 0$, there exists a positive ε^* that depends on P_0 and P_1 and δ such that for any $\varepsilon \leq \varepsilon^*$ the induced marginals M_0 and M_1 satisfy the bound*

$$D_{\mathrm{kl}}\big(M_0||M_1\big) + D_{\mathrm{kl}}\big(M_1||M_0\big) \;\leq\; \frac{2(1+\delta)(e^{\varepsilon}-1)^2}{(e^{\varepsilon}+1)} \left\|P_0 - P_1\right\|_{\mathrm{TV}}^2 .$$

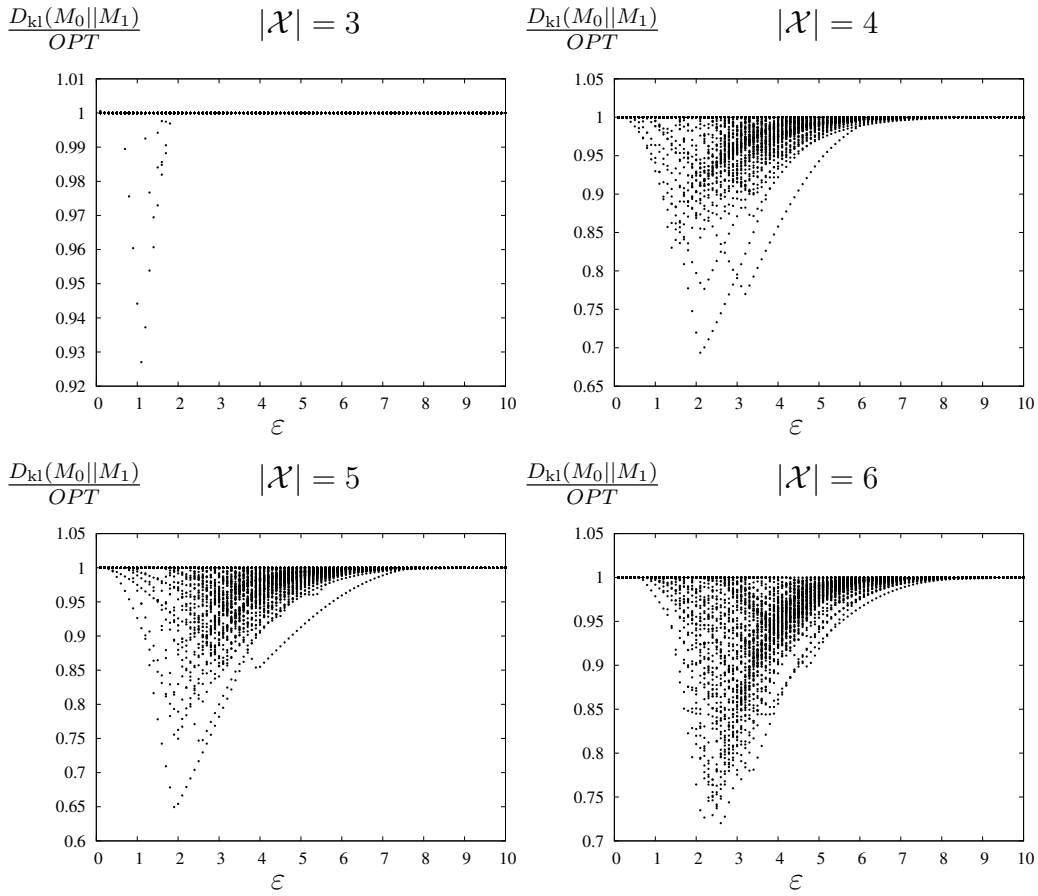


Figure 3.4: For varying input alphabet size $|\mathcal{X}| \in \{3, 4, 5, 6\}$, at least 60% of the optimal divergence can be achieved by taking the better one between the binary and the randomized response mechanisms.

This follows from Theorem 3.3.1 and observing that the binary mechanism achieves

$$\begin{aligned} & D_{\mathrm{kl}}\big(M_0||M_1\big) \\ &= \frac{(e^\varepsilon - 1)P_0(T) + 1}{e^\varepsilon + 1} \log\Big(\frac{1 + (e^\varepsilon - 1)P_0(T)}{1 + (e^\varepsilon - 1)P_1(T)}\Big) \\ &\qquad + \frac{(e^\varepsilon - 1)P_0(T^c) + 1}{e^\varepsilon + 1} \log\Big(\frac{1 + (e^\varepsilon - 1)P_0(T^c)}{1 + (e^\varepsilon - 1)P_1(T^c)}\Big) \\ &= \frac{(e^\varepsilon - 1)^2}{e^\varepsilon + 1}(P_0(T) - P_1(T)) + O(\varepsilon^3) \\ &= \frac{(e^\varepsilon - 1)^2}{e^\varepsilon + 1} \big\|P_0 - P_1\big\|_{\mathrm{TV}}^2 + O(\varepsilon^3)\,, \end{aligned} \tag{3.17}$$

where $T \subseteq \mathcal{X}$ is the set of x such that $P_0(x) \geq P_1(x)$. Compared to [47, Theorem 1], we recover their bound of $4(e^\varepsilon - 1)^2 \|P_0 - P_1\|_{\mathrm{TV}}^2$ with a smaller constant. We want to note that Duchi et al.'s bound holds for all values of ε and uses a different technique of bounding the KL divergence directly, however no achievable mechanism has been provided. We instead provide an explicit mechanism, that is optimal in high privacy regime.

Similarly, in the low privacy regime, we can show the following converse result.

Corollary 3.3.6 *For any $\varepsilon \geq 0$, let Q be any conditional distribution that guarantees ε-local differential privacy. Then, for any pair of distributions P_0 and P_1 and any positive $\delta > 0$, there exists a positive ε^* that depends on P_0 and P_1 and δ such that for any $\varepsilon \geq \varepsilon^*$ the induced marginals M_0 and M_1 satisfy the bound*

$$D_{\mathrm{kl}}\big(M_0||M_1\big) + D_{\mathrm{kl}}\big(M_1||M_0\big) \;\leq\; D_{\mathrm{kl}}(P_0||P_1) - (1-\delta)G(P_0,P_1)e^{-\varepsilon} \,,$$

where $G(P_0, P_1) = \sum_{\mathcal{X}} (1 - P_0(x)) \log(P_1(x)/P_0(x))$.

This follows directly from Theorem 3.3.4 and observing that the randomized response mechanism achieves

$$D_{\mathrm{kl}}(M_0||M_1) = D_{\mathrm{kl}}(P_0||P_1) - G(P_0,P_1)e^{-\varepsilon} + O(e^{-2\varepsilon}) \,. \tag{3.18}$$

Similarly, for total variation, we can get the following converse result. This follows from Theorem 3.3.2 and explicitly computing the total variation achieved by the binary mechanism.

Corollary 3.3.7 *For any $\varepsilon \geq 0$, let Q be any conditional distribution that guarantees ε-local differential privacy. Then, for any pair of distributions P_0 and P_1, the induced marginals M_0 and M_1 satisfy the bound $\big\|M_0 - M_1\big\|_{\mathrm{TV}} \leq ((e^\varepsilon - 1)/(e^\varepsilon + 1)) \big\|P_0 - P_1\big\|_{\mathrm{TV}}$, and equality is achieved by the binary mechanism.*

Figure 3.5 illustrates the gap between the divergence achieved by the geometric mechanism described in the previous section and the optimal mechanisms (the binary mechanism for the high privacy regime and the randomized response mechanism for the low privacy regime). For each instance of the 100 randomly generated P_0 and P_1 over input of size $k = 6$, we plot the

resulting divergence $D_{\mathrm{kl}}(M_0||M_1)$ as a function of $\|P_0 - P_1\|_{\mathrm{TV}}$ for $\varepsilon = 0.1$, and as a function of $D_{\mathrm{kl}}(P_0||P_1)$ for $\varepsilon = 10$. The binary and the randomized response mechanisms exhibit the scaling predicted by Equation (3.17) and (3.18), respectively.

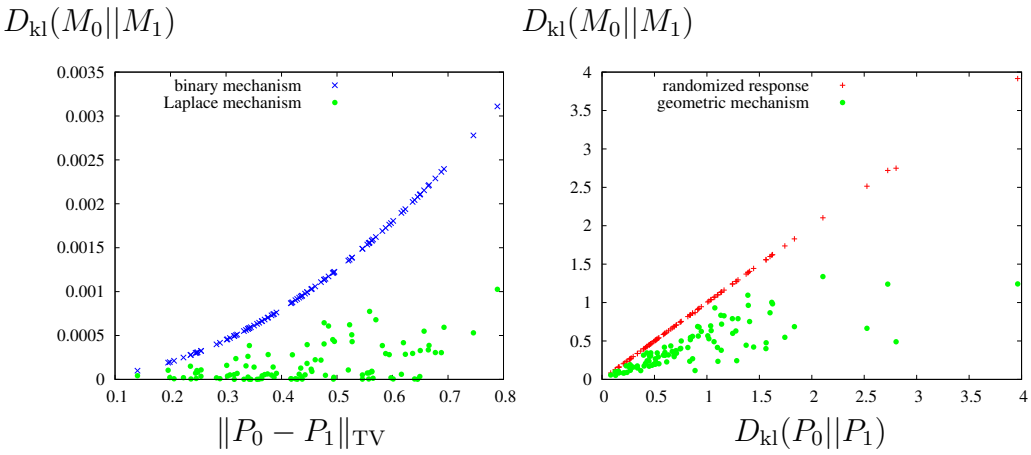


Figure 3.5: For small $\varepsilon = 0.1$ (left) the binary mechanism achieves the optimal KL divergence, which scales as Equation (3.17). For large $\varepsilon = 10$ (right) the randomized response achieves the optimal KL divergence, which scales as Equation (3.18). Both mechanisms improve significantly over the geometric mechanism.

3.4 Information Preservation

In this section, we study the fundamental tradeoff between local privacy and mutual information. Consider a random variable X distributed according to P. The information content in X is captured by entropy

$$H(X) = -\sum_{x} P(x) \log P(x).$$

We are interested in releasing a differentially private version of X represented by Y. The random variable Y should preserve the information content of X as much as possible while meeting the local differential privacy constraints. Similar to the hypothesis testing setting, we will show that a variant of the binary mechanism is optimal in the high privacy regime and the randomized response mechanism is optimal in the low privacy regime.

Let Q be a non-interactive privatization mechanism guaranteeing ε-local differential privacy. The output of the privatization mechanism Y is distributed according to the induced marginal M given by

$$M(S) \;=\; \sum_{x\in\mathcal{X}} Q(S|x)P(x)\,,$$

for $S \subseteq \mathcal{Y}$. With a slight abuse of notation, we will use M and P to represent both probability distributions and probability mass functions. The information content in Y about X is captured by the well celebrated information theoretic quantity called mutual information. The mutual information between X and Y is given by

$$I\left(X;Y\right) = \sum_{\mathcal{X}}\sum_{\mathcal{Y}} P\left(x\right) Q\left(y|x\right) \log\left(\frac{Q\left(y|x\right)}{\sum_{l\in\mathcal{X}} P\left(l\right) Q\left(y|l\right)}\right) = U\left(P,Q\right) = U\left(Q\right). \tag{3.19}$$

Notice that $I\left(X;Y\right) \leq H\left(X\right)$ and $I\left(X;Y\right)$ is convex in Q [69]. To preserve the information context in X, we wish to choose a privatization mechanism Q such that the mutual information between X and Y is maximized subject to differential privacy constraints. In other words, we are interested in characterizing the optimal solution to

$$\begin{array}{ll} \underset{Q}{\text{maximize}} & I\left(X;Y\right) \\ \text{subject to} & Q \in \mathcal{D}_{\varepsilon} \end{array}, \tag{3.20}$$

where $\mathcal{D}_{\varepsilon}$ is the set of all ε-locally differentially private mechanisms defined in (3.7). The above mutual information maximization problem can be thought of as a conditional entropy minimization problem since $I\left(X;Y\right) = H\left(X\right) - H\left(X|Y\right)$.

3.4.1 Optimal staircase mechanisms

From the definition of $I\left(X;Y\right)$, we have that

$$I\left(X;Y\right) = \sum_{\mathcal{Y}}\sum_{\mathcal{X}} P\left(x\right) Q\left(y|x\right) \log\left(\frac{Q\left(y|x\right)}{P^T Q_y}\right) = \sum_{\mathcal{Y}} \mu\left(Q_y\right),$$

where $P^T Q_y = \sum_{\mathcal{X}} P(x) Q(y|x)$ and

$$\mu(Q_y) = \sum_{\mathcal{X}} P(x) Q(y|x) \log\left(Q(y|x) / P^T Q_y\right).$$

Notice that $\mu(\gamma Q_y) = \gamma \mu(Q_y)$, and by the log-sum inequality, μ is convex. Convexity and homogeneity together imply sublinearity. Therefore, Theorems 3.2.2 and 3.2.4 apply to $I(X;Y)$ and we have that staircase mechanisms are optimal.

For a given P, the *binary mechanism for mutual information* is defined as a staircase mechanism with only two outputs $y \in \{0,1\}$ (see Figure 3.2). Let $T \subseteq \mathcal{X}$ be the set that partitions $\mathcal{X}$ into two partitions, T and T^c, such that $|P(T) - P(T^c)|$ is minimized. Precisely,

$$T \in \arg\min_{A \subseteq \mathcal{X}} \left| P(A) - \frac{1}{2} \right|. \tag{3.21}$$

Observe that there are always multiple choices for T. Indeed, for any minimizing set T, T^c is also a minimizing set since $|P(T) - 1/2| = |P(T^c) - 1/2|$. When there is only one such pair, the binary mechanism is uniquely defined as

$$Q(0|x) = \begin{cases} \frac{e^\varepsilon}{1+e^\varepsilon} & \text{if } x \in T, \\ \frac{1}{1+e^\varepsilon} & \text{if } x \notin T. \end{cases} \quad Q(1|x) = \begin{cases} \frac{e^\varepsilon}{1+e^\varepsilon} & \text{if } x \notin T, \\ \frac{1}{1+e^\varepsilon} & \text{if } x \in T. \end{cases} \tag{3.22}$$

When there are multiple pairs, any pair (T, T^c) can be chosen to define the binary mechanism. All resulting binary mechanisms are equivalent from a utility maximization perspective.

In what follows, we will establish that this simple mechanism is the optimal mechanism in the high privacy regime. Intuitively, in the high privacy regime, we cannot release more than one bit of information, and hence, the input alphabet is reduced to a binary output alphabet. In this case we have to maximize the information contained in the released bit by maximizing its entropy: $T \in \arg\max_{A \subseteq \mathcal{X}} \big(- P(A) \log P(A) - P(A^c) \log P(A^c) \big) = \arg\max_{A \subseteq \mathcal{X}} |P(A) - 1/2|$.

Theorem 3.4.1 *For any distribution P, there exists a positive ε^* that depends on P such that for any positive $\varepsilon \leq \varepsilon^*$, the binary mechanism maximizes the mutual information between the input and the output of a privatization mechanism over all ε-locally differentially private mechanisms.*

This implies that in the high privacy regime, the binary mechanism is the optimal solution for (3.20).

Next, we show that the binary mechanism achieves near-optimal performance for all $(\mathcal{X}, P)$ and $\varepsilon \leq 1$ even when $\varepsilon^* < 1$. Let OPT denote the maximum value of (3.20) and BIN denote the mutual information achieved by the binary mechanism given in (3.22). The next theorem shows that

$$\text{BIN} \;\geq\; \frac{1}{1+e^{\varepsilon}}\text{OPT}\,.$$

Theorem 3.4.2 *For any $\varepsilon \leq 1$ and any distribution P, the binary mechanism is an $(1+e^{\varepsilon})$-approximation of the maximum mutual information between the input and the output of a privatization mechanism among all ε-locally differentially private mechanisms.*

Note that $1 + e^{\varepsilon} \leq 4$ for $\varepsilon \leq 1$ which is a commonly studied regime in differential privacy applications. Therefore, we can always use the simple binary mechanism and the resulting mutual information is at most a constant factor away from the optimal.

In the low privacy regime ($\varepsilon \geq \varepsilon^*$), the *randomized response mechanism* defined in(3.16) is optimal.

Theorem 3.4.3 *There exists a positive ε^* that depends on P such that for any distribution P and all $\varepsilon \geq \varepsilon^*$, the randomized response mechanism maximizes the mutual information between the input and the output of as privatization mechanism over all ε-locally differentially private mechanisms.*

3.4.2 Numerical Experiments

For 100 instances of randomly chosen P defined over input alphabet of size $|\mathcal{X}| = 6$, we compare the average performance of the binary, randomized response, and the geometric mechanisms to the optimal mechanism. We plot (in Figure 3.6, left) the average performance measured by the normalized mutual information $I(X;Y)/H(X)$ for all 4 mechanisms. The average is taken over the 100 instances of P. In the low privacy (large ε) regime, the randomized response achieves optimal performance as predicted, which converges to one. In the high privacy regime (small ε), the binary mechanism achieves optimal performance as predicted. In all regimes, both mechanisms

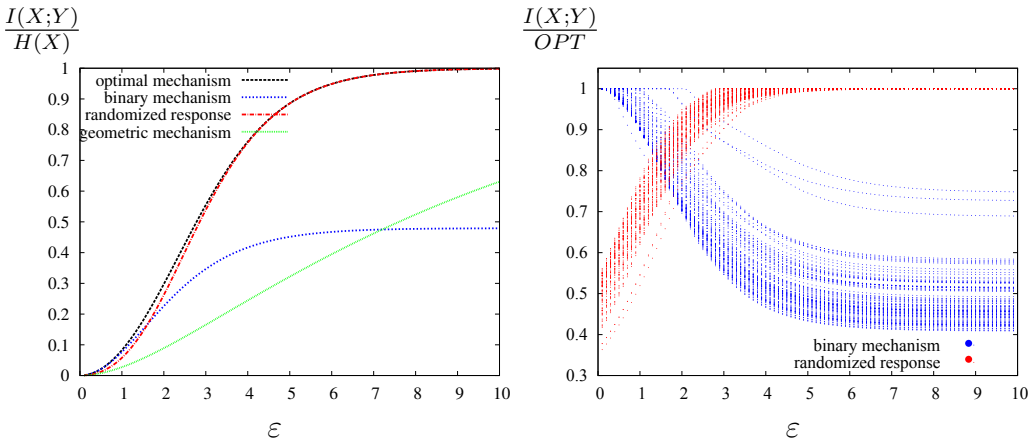


Figure 3.6: The binary and randomized response mechanisms are optimal in the high-privacy (small ε) and low-privacy (large ε) regimes, respectively, and improve over the geometric mechanism significantly (left). When the regimes are mismatched, $I(X;Y)$ under these mechanisms can each be as bad as 35% of the optimal one (right).

significantly improve over the geometric mechanism. To illustrate how much worse the binary and randomized response mechanisms can be (relative to the optimal staircase mechanism), we plot (in Figure 3.6, right) the mutual information under each mechanism normalized by the mutual information under the optimal staircase mechanism. This is done for all 100 instances of P. In all instances, the binary mechanism is optimal for small ε and the randomized response mechanism is optimal for large ε. However, $I(X;Y)$ under the randomized response mechanism can be as bad as 35% of the optimal one (for small ε). Similarly, $I(X;Y)$ under the binary mechanism can be as bad as 40% of the optimal one (for large ε).

For $|\mathcal{X}| \in \{3,4,5,6\}$, we plot (in Figure 3.7) the performance of better between the binary and randomized response mechanisms normalized by the optimal mechanism for all 100 randomly generated instances of P. This mixed strategy achieves at least 75% of the optimal mutual infirmation for all instances of P. Moreover, it is not sensitive to the size of the alphabet $|\mathcal{X}|$.

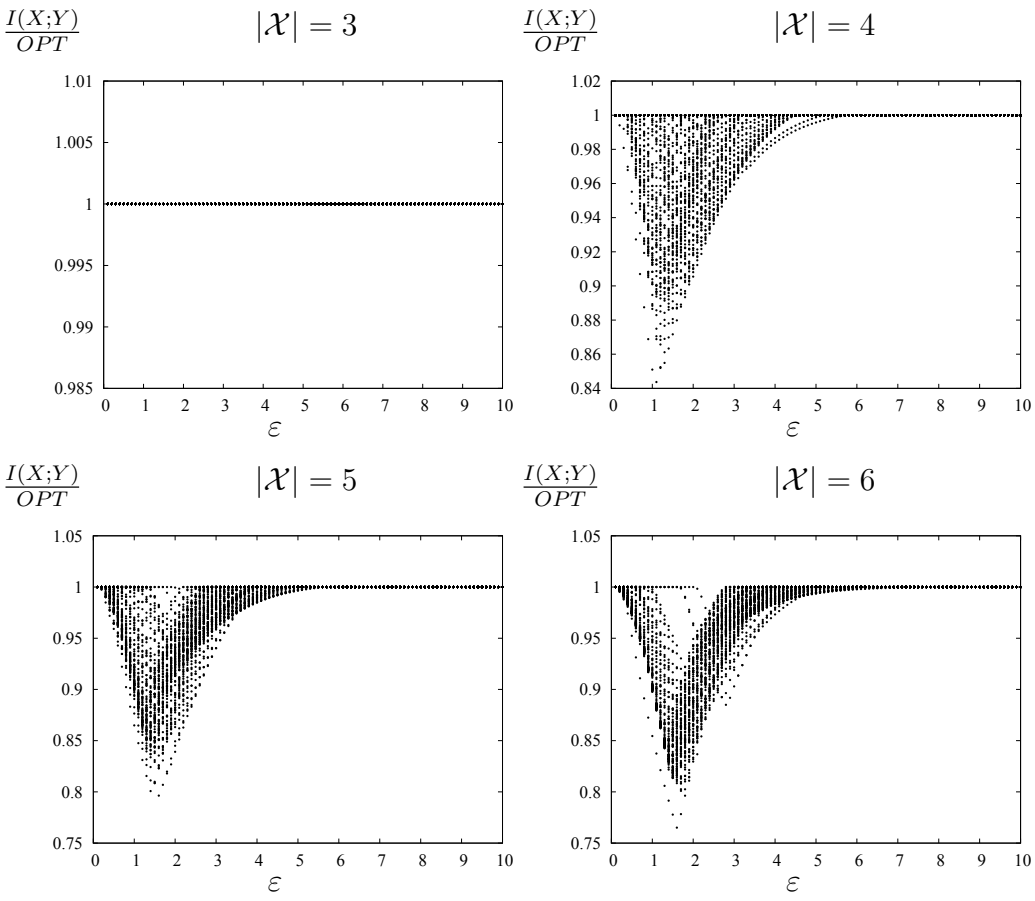


Figure 3.7: For varying input alphabet size $|\mathcal{X}| \in \{3, 4, 5, 6\}$, at least 75% of the maximum mutual information can be achieved by taking the better one between the binary and the randomized response mechanisms.

3.4.3 Lower bounds

In this section, we provide converse results on the fundamental limit of locally differentially private mechanisms when utility is measured via mutual information.

Corollary 3.4.4 *For any $\varepsilon \geq 0$, let Q be any conditional distribution that guarantees ε-local differential privacy. Then, for any distribution P and any positive $\delta > 0$, there exists a positive ε^* that depends on P and δ such that for any $\varepsilon \leq \varepsilon^*$ the following bound holds*

$$I(X;Y) \leq (1+\delta)\frac{1}{2}P(T)P(T^c)\varepsilon^2,$$

where T is defined in (3.21).

This follows from Theorem 3.4.1 (optimality of the binary mechanism) and observing that the binary mechanism achieves

$$\begin{aligned} I(X;Y) &= \frac{1}{e^\varepsilon+1}\left\{P(T)e^\varepsilon \log\frac{e^\varepsilon}{P(T^c)+e^\varepsilon P(T)} + P(T^c)\log\frac{1}{P(T^c)+e^\varepsilon P(T)}\right\} \\ &\quad + \frac{1}{e^\varepsilon+1}\left\{P(T^c)e^\varepsilon \log\frac{e^\varepsilon}{P(T)+e^\varepsilon P(T^c)} + P(T)\log\frac{1}{P(T)+e^\varepsilon P(T^c)}\right\} \\ &= \frac{1}{2}P(T)P(T^c)\varepsilon^2 + O\left(\varepsilon^3\right). \end{aligned} \tag{3.23}$$

Similarly, in the low privacy regime, we can show the following converse result.

Corollary 3.4.5 *For any $\varepsilon \geq 0$, let Q be any conditional distribution that guarantees ε-local differential privacy. Then, for any distributions P and any positive $\delta > 0$, there exists a positive ε^* that depends on P and δ such that for any $\varepsilon \geq \varepsilon^*$ the following bound holds*

$$I(X;Y) \leq H(X) - (1-\delta)(k-1)\varepsilon e^{-\varepsilon}.$$

This follows directly from Theorem 3.4.3 (optimality of the randomized response mechanism) and observing that the randomized response mechanism achieves

$$I(X;Y) = H(X) - (k-1)\varepsilon e^{-\varepsilon} + O(e^{-2\varepsilon}). \tag{3.24}$$

Figure 3.8 illustrates the gap between the mutual information achieved by the geometric mechanism and the optimal mechanisms (the binary mechanism for the high privacy regime and the randomized response mechanism for the low privacy regime). For each instance of the 100 randomly generated P over input of size $k = 6$, we plot the resulting mutual information $I(X;Y)$ as a function of $P(T)\,P(T^c)$ for $\varepsilon = 0.1$, and as a function of $H(X)$ for $\varepsilon = 10$. The binary and the randomized response mechanisms exhibit the scaling predicted by Equations (3.23) and (3.24), respectively.

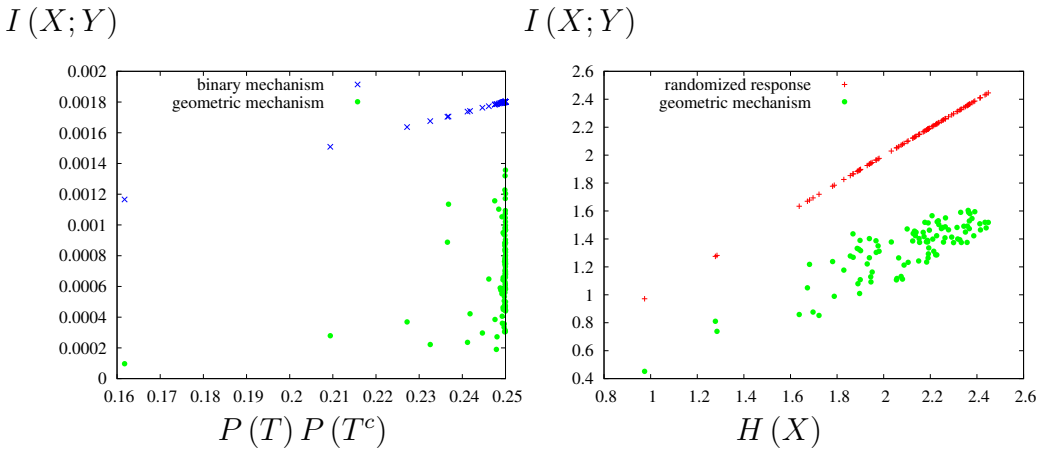


Figure 3.8: For small $\varepsilon = 0.1$ (left) the binary mechanism achieves the optimal mutual information, which scales as Equation (3.23). For large $\varepsilon = 10$ (right) the randomized response mechanism achieves the optimal mutual information, which scales as Equation (3.24). Both mechanisms improve significantly over the geometric mechanism.

3.5 Generalizations to approximate differential privacy

In this section, we generalize the results of the previous sections in the following ways.

1. We consider the class of utility functions that obey the data processing inequality. Consider the composition of two privatization mechanisms $QW = Q \circ W$ where the output of the first mechanism Q is applied to another mechanism W. We say that a utility function $U(\cdot)$ obeys the data processing inequality if the following inequality holds for all

Q and W

$$U(QW) \;\leq\; U(Q)\,.$$

The following proposition, proved in [25], shows that the class of utilities obeying the data processing inequality includes all the utility functions we studied in Section 3.2.

Proposition 3.5.1 *Any utility function that can be written in the form of* $U(Q) = \sum_{\mathcal{Y}} \mu(Q_y)$*, where* μ *is any sublinear function, obeys the data processing inequality.*

2. We consider (ε, δ)-differential privacy which generalizes the notion of ε-differential privacy. (ε, δ)-differential privacy is commonly referred to as approximate differential privacy and it was first introduced in [49]. For the release of a random variable $X \in \mathcal{X}$, we say that a mechanism Q is (ε, δ)-locally differentially private if

$$Q(S|x) \leq e^{\varepsilon} Q(S|x') + \delta, \tag{3.25}$$

for all $S \subseteq \mathcal{Y}$ and all $x, x' \in \mathcal{X}$. Note that ε-local differential privacy is a special case of (ε, δ)-local differential privacy where $\delta = 0$.

3. We prove that the *quaternary mechanism*, defined in Equation (3.26), is optimal for any ε and any δ. This is different from the treatment conducted in the previous sections where we proved the optimality of the binary (randomized response) mechanism for sufficiently small (large) ε and $\delta = 0$.

The treatment in this section, even though more general than the one in previous sections in the ways described above, holds only for binary alphabets (i.e., $|\mathcal{X}| = 2$). Finding optimal privatization mechanisms under (ε, δ)-local differential privacy for larger input alphabets (i.e., $|\mathcal{X}| > 2$) is an interesting open question. Unlike ε-local differential privacy, the privacy constraints under (ε, δ)-local differential privacy no longer decompose into separate constraints on each output y. This makes it difficult to generalize the techniques developed in previous sections of this chapter. However, for the special case of binary input alphabets, we can prove the optimality of one mechanism

for all values of (ε, δ) and all utility functions that obey the data processing inequality.

For a binary random variable $X \in \mathcal{X} = \{0, 1\}$, the *quaternary mechanism* maps X to a quaternary random variable $Y \in \mathcal{Y} = \{0, 1, 2, 3\}$ and is defined as

$$Q_{\mathrm{QT}}(0|x) = \begin{cases} \delta & \text{if } x = 0 \,, \\ 0 & \text{if } x = 1 \,. \end{cases} \quad Q_{\mathrm{QT}}(1|x) = \begin{cases} 0 & \text{if } x = 0 \,, \\ \delta & \text{if } x = 1 \,. \end{cases} \tag{3.26}$$

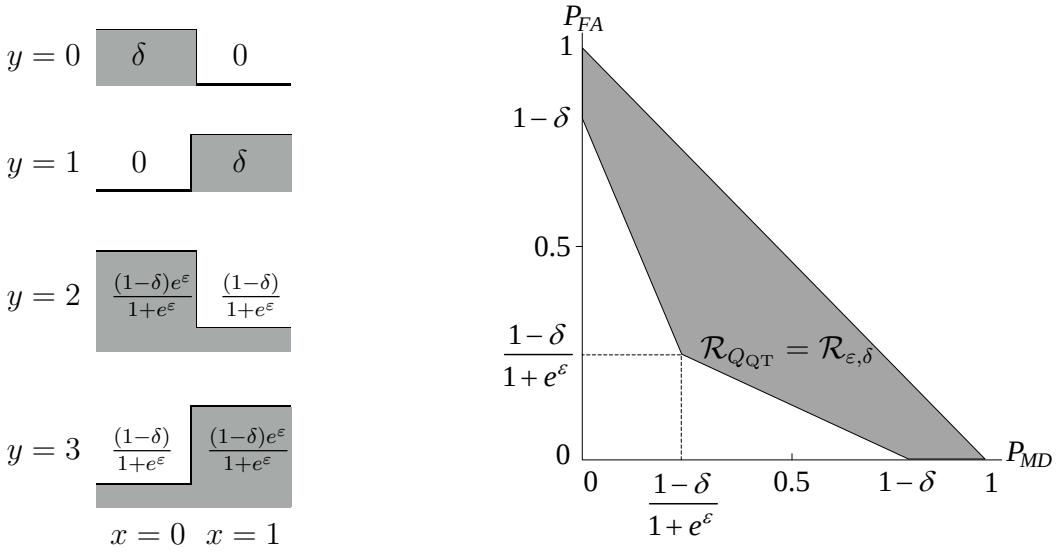
$$Q_{\mathrm{QT}}(2|x) = \begin{cases} (1-\delta)\frac{1}{1+e^{\varepsilon}} & \text{if } x = 0 \,, \\ (1-\delta)\frac{e^{\varepsilon}}{1+e^{\varepsilon}} & \text{if } x = 1 \,. \end{cases} \quad Q_{\mathrm{QT}}(3|x) = \begin{cases} (1-\delta)\frac{e^{\varepsilon}}{1+e^{\varepsilon}} & \text{if } x = 0 \,, \\ (1-\delta)\frac{1}{1+e^{\varepsilon}} & \text{if } x = 1 \,. \end{cases}$$

In other words, the quaternary mechanism passes X unchanged with probability δ and applies the binary mechanism (defined in previous sections) with probability $1 - \delta$. The main result of this section can be stated formally as follows.

Theorem 3.5.1 *If $|\mathcal{X}| = 2$, then for any ε, any δ, and any $U(Q)$ that obeys the data processing inequality, the quaternary mechanism maximizes $U(Q)$ subject to $Q \in \mathcal{D}_{(\varepsilon,\delta)}$, the set of all (ε, δ)-locally differentially private mechanism.*

The proof of Theorem 3.5.1 depends on an *operational definition* of differential privacy which we describe next. Consider a privatization mechanism Q that maps $X \in \{0, 1\}$ stochastically to $Y \in \mathcal{Y}$. Given Y, construct a binary hypothesis test on whether $X = 0$ or $X = 1$. Any binary hypothesis test is completely described by a, possibly randomized, decision rule $\hat{X} : Y \to \{0, 1\}$. The two types of error associated with $\hat{X}$ are *false alarm:* $\hat{X} = 1$ when $X = 0$, and *miss detection:* $\hat{X} = 0$ when $X = 1$. The probability of false alarm is given by $P_{\mathrm{FA}} = \mathbb{P}(\hat{X} = 1|X = 0)$ while the probability of miss detection is given by $P_{\mathrm{MD}} = \mathbb{P}(\hat{X} = 0|X = 1)$. For a fixed Q, the convex hull of all pairs $(P_{\mathrm{MD}}, P_{\mathrm{FA}})$ for all decision rules $\hat{X}$ defines a two-dimensional *error region* where P_{MD} is plotted against P_{FA}. For example, the quaternary mechanism given in Figure 3.9a has an error region $\mathcal{R}_{Q_{\mathrm{QT}}}$ shown in Figure 3.9b.

It turns out that (ε, δ)-local differential privacy imposes the following conditions on the error region of all (ε, δ)-locally differentially private mecha-



(a) Privatization mechanism (b) Error region

Figure 3.9: The quaternary mechanism

nisms

$$P_{\mathrm{FA}} + e^{\varepsilon} P_{\mathrm{MD}} \;\geq\; 1-\delta\,, \quad \text{and} \quad e^{\varepsilon} P_{\mathrm{FA}} + P_{\mathrm{MD}} \;\geq\; 1-\delta\,,$$

for any decision rule $\hat{X}$. These two conditions define an error region $\mathcal{R}_{\varepsilon,\delta}$ shown in Figure 3.9b. Interestingly, the next theorem shows that the converse result is also true.

Theorem 3.5.2 *A mechanism Q is (ε, δ)-locally differentially private if and only if $\mathcal{R}_Q \subseteq \mathcal{R}_{\varepsilon,\delta}$.*

The proof of the above theorem can be found in [50]. Notice that it is no coincidence that $\mathcal{R}_{Q_{\mathrm{QT}}} = \mathcal{R}_{\varepsilon,\delta}$. This property will be essential to proving the optimality of the quaternary mechanism.

Theorem 3.5.2 allows us to benefit from the data processing inequality (DPI) and its converse, which follows from a celebrated result by [70]. These inequalities, while simple by themselves, lead to surprisingly strong technical results. Indeed, there is a long line of such a tradition in the information theory literature (see Chapter 17 of [69]). Consider two privatization mechanisms, $Q^{(1)}$ and $Q^{(2)}$. Let Y and Z denote the output of the mechanisms $Q^{(1)}$ and $Q^{(2)}$, respectively. We say that $Q^{(1)}$ dominates $Q^{(2)}$ if there exists a coupling of Y and Z such that X–Y–Z forms a Markov chain. In other

words, we say $Q^{(1)}$ dominates $Q^{(2)}$ if there exists a stochastic mapping Q such that $Q^{(2)} = Q^{(1)} \circ Q$.

Theorem 3.5.3 *A mechanism $Q^{(1)}$ dominates a mechanism $Q^{(2)}$ if and only if $\mathcal{R}_{Q^{(2)}} \subseteq \mathcal{R}_{Q^{(1)}}$.*

The proof of the above theorem can be found in [70]. Observe that by Theorems 3.5.3 and 3.5.2, and the fact that $\mathcal{R}_{Q_{\mathrm{QT}}} = \mathcal{R}_{\varepsilon,\delta}$, the quaternary mechanism dominates any other differentially private mechanism. In other words, for any differentially private mechanism Q, there exists a stochastic mapping W such that $Q = W \circ Q_{\mathrm{QT}}$. Therefore, for any (ε, δ) and any utility function $U(.)$ obeying the data processing inequality, we have that $U(Q) \leq U(Q_{\mathrm{QT}})$. This finishes the proof of Theorem 3.5.1.

3.6 Discussion

In this chapter, we have considered a broad class of convex utility functions and assumed a setting where individuals cannot collaborate (communicate with each other) before releasing their data. It turns out that the techniques developed in this work can be generalized to find optimal privatization mechanisms in a setting where different individuals can collaborate interactively and each individual can be an analyst [74].

Binary hypothesis testing and information preservation are two canonical problems with a wide range of applications. However, there are a number of non-trivial and interesting extensions to our work.

Correlation among data. In some scenarios the X_i's could be correlated (e.g., when different individuals observe different functions of the same random variable). In this case, the data analyst is interested in inferring whether the data was generated from P_0^n or P_1^n, where P_ν^n is one of two possible joint priors on $X_1, ..., X_n$. This is a challenging problem because knowing X_i reveals information about X_j, $j \neq i$. Therefore, the utility maximization problems for different individuals are coupled in this setting.

Robust and m-ary hypothesis testing. In some cases the data analyst need not have access to P_0 and P_1, but rather two classes of prior distribution P_{θ_0} and P_{θ_1} for $\theta_0 \in \Lambda_0$ and $\theta_1 \in \Lambda_1$. Such problems are studied under the

rubric of universal hypothesis testing and robust hypothesis testing. One possible direction is to select the privatization mechanism that maximizes the worst case utility: $Q^* = \arg\max_{Q\in\mathcal{D}_\varepsilon} \min_{\theta_0\in\Lambda_0,\theta_1\in\Lambda_1} D_f(M_{\theta_0}||M_{\theta_1})$, where M_{θ_ν} is the induced marginal under P_{θ_ν}.

The more general problem of private m-ary hypothesis testing is also an interesting but challenging one. In this setting, the X_i's can follow one of m distributions P_0, P_1, ..., P_{m-1}. Consequently, the Y_i's can follow one of m distributions M_0, M_1, ..., M_{m-1}. The utility can be defined as the average f-divergence between any two distributions: $1/(m(m-1))\sum_{i\neq j} D_f(M_i||M_j)$, or the worst case one: $\min_{i\neq j} D_f(M_i||M_j)$.

Non-exchangeable utility functions. The utility studied in this chapter was measured by functions that are exchangeable, i.e. the utility did not depend on the naming (labelling) of the private and privatized data (X and Y). This made sense for statistical learning applications that depend on information theoretic quantities such as f-divergences and mutual information. However, in some other applications, the utility might be defined over $\mathcal{X}\cup\mathcal{Y}$ in a metric space, where there exists a natural measure of distance (or distortion) between the data points. In this case, we can formulate the problem as a distortion minimization one

$$\text{minimize}_{Q\in\mathcal{D}_\varepsilon} \quad \sum_{x,y} d(x,y)P(x)Q(y|x) \ ,$$

where $d(x,y)$ is some distortion metric. [54] studied this problem, and showed that the mechanism $Q(y|x) \propto e^{\varepsilon(1-d(x,y))}/(k-1+e^\varepsilon)$ achieves near optimal performance when ε is large enough, which is the low privacy regime. Notice that when Hamming distance is used, $d(x,y) = \mathbb{I}(x\neq y)$, this recovers the randomized response mechanism exactly. This provides a starting point for generalizing the search for optimal mechanisms under non-exchangeable utility functions.

REFERENCES

[1] C. Leberknight, M. Chiang, H. Poor, and F. Wong, "A taxonomy of internet censorship and anti-censorship," 2012.

[2] P. Villareal, "Executive loses job after chick-fil-a rant video goes viral," Arizona Daily Star, August 2012.

[3] "The constitutional right to anonymity: Free speech, disclosure and the devil," *The Yale Law Journal*, vol. 70, no. 7, 1961. [Online]. Available: http://www.jstor.org/stable/794351

[4] "Rooms," https://www.rooms.me.

[5] "Secret," https://www.secret.ly.

[6] "Whisper," http://whisper.sh.

[7] "Yik yak," http://www.yikyakapp.com/.

[8] D. Chaum, "The dining cryptographers problem: Unconditional sender and recipient untraceability," *Journal of cryptology*, vol. 1, no. 1, 1988.

[9] H. Corrigan-Gibbs and B. Ford, "Dissent: accountable anonymous group messaging," in *Proc. CCS*. ACM, 2010.

[10] P. Golle and A. Juels, "Dining cryptographers revisited," in *Advances in Cryptology-Eurocrypt 2004*. Springer, 2004.

[11] S. Helmers, "A brief history of anon. penet. fi–the legendary anonymous remailer," *Computer-Mediated Comunication Magazine*, vol. 4.

[12] P. Lewis and D. Rushe, "Revealed: how Whisper app tracks anonymous users," *The Gaurdian*, oct 2014.

[13] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," DTIC Document, Tech. Rep., 2004.

[14] I. Clarke, O. Sandberg, B. Wiley, and T. Hong, "Freenet: A distributed anonymous information storage and retrieval system," in *Designing Privacy Enhancing Technologies*. Springer, 2001.

[15] R. Dingledine, M. Freedman, and D. Molnar, "The free haven project: Distributed anonymous storage service," in *Designing Privacy Enhancing Technologies*. Springer, 2001.

[16] M. Freedman and R. Morris, "Tarzan: A peer-to-peer anonymizing network layer," in *Proc. CCS*. ACM, 2002.

[17] G. Fanti, P. Kairouz, S. Oh, and P. Viswanath, "Spy vs. spy: Rumor source obfuscation," *SIGMETRICS Perform. Eval. Rev.*, 2015, [Accepted].

[18] "Tox," www.tox.im.

[19] L. Sweeney, "Guaranteeing anonymity when sharing medical data, the datafly system." in *Proceedings of the AMIA Annual Fall Symposium*. American Medical Informatics Association, 1997, p. 51.

[20] L. Sweeney, "Achieving k-anonymity privacy protection using generalization and suppression," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 571–588, 2002.

[21] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *Security and Privacy, 2008. SP 2008. IEEE Symposium on*. IEEE, 2008, pp. 111–125.

[22] A. Narayanan, E. Shi, and B. I. Rubinstein, "Link prediction by de-anonymization: How we won the kaggle social network challenge," in *Neural Networks (IJCNN), The 2011 International Joint Conference on*. IEEE, 2011, pp. 1825–1834.

[23] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich, "Identifying personal genomes by surname inference," *Science*, vol. 339, no. 6117, pp. 321–324, 2013.

[24] P. Kairouz, S. Oh, and P. Viswanath, "Extremal mechanisms for local differential privacy," in *Advances in Neural Information Processing Systems*, 2014, pp. 2879–2887.

[25] P. Kairouz, S. Oh, and P. Viswanath, "Extremal mechanisms for local differential privacy," *arXiv preprint arXiv:1407.1338*, 2014.

[26] S. Goel, M. Robson, M. Polte, and E. Sirer, "Herbivore: A scalable and efficient protocol for anonymous communication," Cornell University, Tech. Rep., 2003.

[27] L. von Ahn, A. Bortz, and N. Hopper, "K-anonymous message transmission," in *Proc. CCS*. ACM, 2003.

[28] D. Shah and T. Zaman, "Rumors in a network: Who's the culprit?" *Information Theory, IEEE Transactions on*, vol. 57, no. 8, pp. 5163–5181, Aug 2011.

[29] D. Shah and T. Zaman, "Finding rumor sources on random graphs," *arXiv preprint arXiv:1110.6230*, 2011.

[30] Z. Wang, W. Dong, W. Zhang, and C. W. Tan, "Rumor source detection with multiple observations: Fundamental limits and algorithms," 2014.

[31] B. A. Prakash, J. Vreeken, and C. Faloutsos, "Spotting culprits in epidemics: How many and which ones?" in *ICDM*, vol. 12, 2012, pp. 11–20.

[32] V. Fioriti and M. Chinnici, "Predicting the sources of an outbreak with a spectral technique," *arXiv preprint arXiv:1211.2333*, 2012.

[33] W. Luo, W. Tay, and M. Leng, "How to identify an infection source with limited observations," 2013.

[34] K. Zhu and L. Ying, "A robust information source estimator with sparse observations," *arXiv preprint arXiv:1309.4846*, 2013.

[35] C. Milling, C. Caramanis, S. Mannor, and S. Shakkottai, "Network forensics: Random infection vs spreading epidemic," in *Proceedings of SIGMETRICS*. New York, NY, USA: ACM, 2012. [Online]. Available: http://doi.acm.org/10.1145/2254756.2254784 pp. 223–234.

[36] C. Milling, C. Caramanis, S. Mannor, and S. Shakkottai, "Detecting epidemics using highly noisy data." in *MobiHoc*, 2013, pp. 177–186.

[37] E. A. Meirom, C. Milling, C. Caramanis, S. Mannor, A. Orda, and S. Shakkottai, "Localized epidemic detection in networks with overwhelming noise." 2014.

[38] C. Milling, C. Caramanis, S. Mannor, and S. Shakkottai, "On identifying the causative network of an epidemic." in *Allerton Conference*, 2012, pp. 909–914.

[39] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, "On the evolution of user interaction in facebook," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Social Networks (WOSN'09)*, August 2009.

[40] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow, "The anatomy of the facebook social graph," *arXiv preprint arXiv:1111.4503*, 2011.

[41] A. Acquisti, "Privacy in electronic commerce and the economics of immediate gratification," in *Proceedings of the 5th ACM conference on Electronic commerce*. ACM, 2004, pp. 21–29.

[42] A. Acquisti and J. Grossklags, "What can behavioral economics teach us about privacy," *Digital Privacy*, p. 329, 2007.

[43] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, 1965.

[44] C. Dwork, "Differential privacy," in *Automata, languages and programming*. Springer, 2006, pp. 1–12.

[45] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography*. Springer, 2006, pp. 265–284.

[46] C. Dwork and J. Lei, "Differential privacy and robust statistics," in *Proceedings of the 41st annual ACM symposium on Theory of computing*. ACM, 2009, pp. 371–380.

[47] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *Foundations of Computer Science, 2013 IEEE 54th Annual Symposium on*. IEEE, 2013, pp. 429–438.

[48] A. B. Tsybakov and V. Zaiats, *Introduction to nonparametric estimation*. Springer, 2009, vol. 11.

[49] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in *Advances in Cryptology-EUROCRYPT 2006*. Springer, 2006, pp. 486–503.

[50] S. Oh and P. Viswanath, "The composition theorem for differential privacy," *arXiv preprint arXiv:1311.0776*, 2013.

[51] R. F. Barber and J. C. Duchi, "Privacy and statistical risk: Formalisms and minimax bounds," *arXiv preprint arXiv:1412.4451*, 2014.

[52] K. Chatzikokolakis, T. Chothia, and A. Guha, "Statistical measurement of information leakage," in *Tools and Algorithms for the Construction and Analysis of Systems*. Springer, 2010, pp. 390–404.

[53] L. Sankar, S. R. Rajagopalan, and H. V. Poor, "Utility-privacy tradeoffs in databases: An information-theoretic approach," *Information Forensics and Security, IEEE Transactions on*, vol. 8, no. 6, pp. 838–852, 2013.

[54] W. Wang, L. Ying, and J. Zhang, "On the relation between identifiability, differential privacy and mutual-information privacy," *arXiv preprint arXiv:1402.3757*, 2014.

[55] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *Foundations of Computer Science, 2007. 48th Annual IEEE Symposium on.* IEEE, 2007, pp. 94–103.

[56] M. Hardt and G. N. Rothblum, "A multiplicative weights mechanism for privacy-preserving data analysis," in *Foundations of Computer Science, 2010 51st Annual IEEE Symposium on.* IEEE, 2010, pp. 61–70.

[57] K. Chaudhuri, A. D. Sarwate, and K. Sinha, "Near-optimal differentially private principal components," in *NIPS*, 2012, pp. 998–1006.

[58] J. Blocki, A. Blum, A. Datta, and O. Sheffet, "The johnson-lindenstrauss transform itself preserves differential privacy," in *Foundations of Computer Science, 2012 IEEE 53rd Annual Symposium on.* IEEE, 2012, pp. 410–419.

[59] M. Hardt and A. Roth, "Beating randomized response on incoherent matrices," in *Proceedings of the 44th symposium on Theory of Computing.* ACM, 2012, pp. 1255–1268.

[60] M. Kapralov and K. Talwar, "On differentially private low rank approximation," in *SODA*, vol. 5. SIAM, 2013, p. 1.

[61] M. Hardt and K. Talwar, "On the geometry of differential privacy," in *Proceedings of the 42nd ACM symposium on Theory of computing.* ACM, 2010, pp. 705–714.

[62] M. Hardt, K. Ligett, and F. McSherry, "A simple and practical algorithm for differentially private data release," in *NIPS*, 2012, pp. 2348–2356.

[63] K. Chaudhuri and C. Monteleoni, "Privacy-preserving logistic regression," in *NIPS*, vol. 8, 2008, pp. 289–296.

[64] J. Lei, "Differentially private m-estimators," in *NIPS*, 2011, pp. 361–369.

[65] A. Ghosh, T. Roughgarden, and M. Sundararajan, "Universally utility-maximizing privacy mechanisms," *SIAM Journal on Computing*, vol. 41, no. 6, pp. 1673–1693, 2012.

[66] Q. Geng and P. Viswanath, "The optimal mechanism in differential privacy," *arXiv preprint arXiv:1212.1186*, 2012.

[67] Q. Geng and P. Viswanath, "The optimal mechanism in (ϵ,δ)-differential privacy," *arXiv preprint arXiv:1305.1330*, 2013.

[68] Q. Geng and P. Viswanath, "The optimal mechanism in differential privacy: Multidimensional setting," *arXiv preprint arXiv:1312.0655*, 2013.

[69] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.

[70] D. Blackwell, "Equivalent comparisons of experiments," *The Annals of Mathematical Statistics*, vol. 24, no. 2, pp. 265–272, 1953.

[71] A. Beimel, K. Nissim, and E. Omri, "Distributed private data analysis: Simultaneously solving how and what," in *Advances in Cryptology–CRYPTO 2008*. Springer, 2008, pp. 451–468.

[72] K. Chaudhuri and D. Hsu, "Convergence rates for differentially private statistical estimation," *arXiv preprint arXiv:1206.6395*, 2012.

[73] A. De, "Lower bounds in differential privacy," in *Theory of Cryptography*. Springer, 2012, pp. 321–338.

[74] P. Kairouz, S. Oh, and P. Viswanath, "Optimality of non-interactive randomized response," *arXiv preprint arXiv:1407.1546*, 2014.