Privacy-preserving Analysis of Correlated Data

Yizhen Wang YIW248@ENG.UCSD.EDU
UC San Diego, 9500 Gilman Drive, La Jolla CA 92093

Shuang Song SHS037@ENG.UCSD.EDU
UC San Diego, 9500 Gilman Drive, La Jolla CA 92093

Kamalika Chaudhuri KAMALIKA@ENG.UCSD.EDU
UC San Diego, 9500 Gilman Drive, La Jolla CA 92093

Abstract

Many modern machine learning applications involve sensitive correlated data, such private information on users connected together in a social network, and measurements of physical activity of a single user across time. However, the current standard of privacy in machine learning, differential privacy, cannot adequately address privacy issues in this kind of data.

This work looks at a recent generalization of differential privacy, called Pufferfish, that can be used to address privacy in correlated data. The main challenge in applying Pufferfish to correlated data problems is the lack of suitable mechanisms. In this paper, we provide a general mechanism, called the Wasserstein Mechanism, which applies to any Pufferfish framework. Since the Wasserstein Mechanism may be computationally inefficient, we provide an additional mechanism, called Markov Quilt Mechanism, that applies to some practical cases such as physical activity measurements across time, and is computationally efficient.

1. Introduction

Machine-learning is increasingly done on personal data, and consequently, it is extremely important to design methods that can learn while still preserving privacy of individuals in the sensitive datasets. For the past several years, the de facto standard of privacy in machine learning has been differential privacy (Dwork et al., 2006), and there is a growing body of work on differentially private learning algorithms (Chaudhuri et al., 2011; Jain et al., 2012; Kifer et al., 2012; Chaudhuri et al., 2012; Hardt and Roth, 2013; Wang et al., 2015; Duchi et al., 2013; Bassily et al., 2014; Imtiaz and Sarwate, 2015). The typical setting in these works is that each (independently drawn) training example corresponds to a single individual's private value, and the goal is to learn a supervised or unsupervised model while adding enough noise to hide the evidence of the participation of a single individual in the dataset.

Many machine learning applications however increasingly involve a very different setting – correlated data – privacy issues in which have been largely ignored by this literature. An example is data on the flu status of people connected together in a social network; if a fairly large subset of these people are children attending the same school, then their flu status are highly correlated. Another example is data from measurements of physical activity of a single subject across time; if these measurements are made at small intervals, then they are highly correlated as human activities change slowly over time. Differential privacy is usually insufficient to address privacy challenges in such data. For example, in the flu status case, differential privacy with parameter $\epsilon = 1$ would advocate adding noise with standard deviation $\sqrt{2}$ to the number of people who have flu. While this hides the evidence of the flu status of a single independent individual, it will not hide the evidence of flu of a child in the school if the school size is big enough. Similarly, in the physical activity example, a modified form of differential privacy, called entry-privacy, will advocate adding noise with standard deviation $2\sqrt{2}$ to the histogram of activities; again, this will not hide evidence of the fact that the individual was sitting at a specific time as the same activity is likely to continue for several measurements. Thus to deal with this kind of correlated data, we need a different notion of privacy.

A generalized version of differential privacy called Pufferfish that captures some of these issues has been recently proposed by (Kifer and Machanavajjhala, 2014). The main idea is to specify what kind of privacy protection is required through three components. These are $\mathcal{S}$, a set of secrets that represents what may need to be hidden, $\mathcal{Q}$, a set of se-

cret pairs that represents pairs of secrets that the adversary should not be able to distinguish between and finally Θ, a class of distributions that can plausibly model correlation in the data. For example, in the flu status case, secrets would be the flu statuses of each person, and the secret pair set would be the set of all pairs of the form (Alice has flu, Alice does not have flu). The distribution class Θ could be the class of all distributions that induce a certain amount of correlation between the flu statuses of people in the connected components. In the physical activity example, the secrets will be the activity at each time point t, secret pairs will be pairs of the form (Activity a at time t, Activity b at time t), for all activity pairs a and b and all t. Assuming that activities transition in a Markovian fashion, Θ will be the set of all Markov Chains over activities. (Kifer and Machanavajjhala, 2014) has shown that Pufferfish is a generalization of differential privacy, and differential privacy essentially is equivalent to Pufferfish when we want to keep each individual's private value secret and when Θ consists of all product distributions over the individuals' values. To address privacy issues in correlated data, we adopt Pufferfish as our privacy definition.

The main challenge in directly applying Pufferfish to correlated data is the lack of suitable mechanisms. While mechanisms are known for specific Pufferfish frameworks (Kifer and Machanavajjhala, 2014; He et al., 2014), there is no general mechanism, and moreover, there is no mechanism in existing work that applies to correlated data. Our first contribution in this work is to provide a generic mechanism that applies to any Pufferfish framework and provides privacy. Our mechanism, called the Wasserstein Mechanism, is an analogue of the popular global sensitivity mechanism for differential privacy, and exploits a novel connection between privacy and a certain form of the Wasserstein distance between measures.

The Wasserstein Mechanism, due to its extreme generality, is computationally inefficient, and a natural question to ask is whether we can come up with a more computationally efficient mechanism, at least for some cases of practical interest. To address this issue, we turn our attention to the case when correlation between the variables can be described by a Bayesian network, and the goal is to hide the private value of each variable. We provide a second mechanism, called Markov Quilt Mechanism, that can exploit properties of the Bayesian network to reduce the computational complexity. As a case study, we apply our mechanism to a Markov Chain, which models the physical activity measurement problem, and we show that this mechanism runs in $\mathcal{O}(n^3)$ time in the size of the chain n.

2. The Privacy Model

2.1. Differential Privacy

We begin with defining differential privacy, which was introduced in (Dwork et al., 2006), and has now become a gold standard for privacy in machine learning.

Definition 2.1. *A (randomized) privacy mechanism $M(\cdot)$ is said to be ϵ-differentially private if for any set of outcomes $S \subseteq Range(M)$, and any datasets D and D' that differ in the private value of a single individual, we have:*

$$\Pr(M(D) \in S) \leq e^{\epsilon} \Pr(M(D') \in S). \tag{1}$$

The parameter ϵ is called the privacy budget, and is usually set to be a small constant. Differential privacy ensures that the participation of a single individual in the dataset does not alter the probability of any outcome by much. This, in turn, means that if we have an adversary who has prior information on the data and an individual Alice, then participation of Alice in the data will not change the extra information he gains about Alice from observing the output of a differentially-private algorithm on the data (Kifer and Machanavajjhala, 2011; Dwork et al., 2006).

Finally, a variant of differential privacy, called *entry-privacy*, enforces (1) for datasets D and D' that differ in a single *entry* or *feature* of an individual. This means that a change in the value of a fixed feature of any individual Alice in the dataset (for example, Alice's disease status) does not change the probability of any outcome by more than a factor of e^{ϵ} when the values of all other features for Alice remain the same.

Recent work has developed a number of generic mechanisms to enforce differential privacy which apply under different conditions; see (Sarwate and Chaudhuri, 2013; Dwork and Roth, 2013) for surveys. The most popular mechanism is the Global Sensitivity Mechanism.

Definition 2.2. *(Global Sensitivity Mechanism) The global sensitivity of a function F is defined as:*

$$GS(F) = \max_{(D,D')} |F(D) - F(D')|,$$

where D, D' are two data sets that differ by a single individual's private value. Given a dataset D, a function F, and a privacy budget ϵ, the global sensitivity mechanism outputs:

$$M(D) = F(D) + Z,$$

where $Z \sim Lap(GS(F)/\epsilon)$.

Here Lap(α) is a Laplace random variable with mean 0 and scale parameter α. (Dwork et al., 2006) shows that this mechanism is ϵ-differentially private.

2.2. Differential Privacy for Correlated Data

It was shown by (Kifer and Machanavajjhala, 2011) that differential privacy will erase the evidence of a single individual's private value only when individuals in the data are independent. There is thus a potential for privacy leaks when individuals' private values are correlated.

Let us consider two concrete examples when this happens. First, suppose we have a dataset $X = \{X_1, \ldots, X_n\}$ where X_i is an indicator variable for whether person i has flu. Our goal is to release (an approximation to) the number of people in the dataset who have flu, while an adversary would like to determine if Alice in the dataset has flu or not.

As changing one person's flu status changes the number of patients by 1, the Global Sensitivity Mechanism (Dwork et al., 2006) will add noise with standard deviation $\approx \frac{1}{\epsilon}$ to the true output value. If the flu status of the people are independent, then this measure will be sufficient to preserve privacy. However now suppose that a certain number of people in the dataset work together, and Alice is in the connected component. Moreover, flu is highly contagious, and passes on between interacting people with probability 0.5. In this case, the noise added due to differential privacy is not sufficient to hide the evidence of Alice's disease status, and we need more noise.

As a second example, consider a time-series $X = \{X_1, X_2, \ldots\}$ where X_t denotes a discrete physical activity (e.g, running, sleeping, sitting, etc) of a person at time t. Our goal is to release (an approximate) histogram of activities, while the adversary would like to determine what the person was doing at a specific time t.

Looking at each X_t as an entry, then we can use entry-differential privacy, which would propose adding noise with standard deviation $\approx 1/\epsilon$ to each bin of the histogram. However, time-series data is highly correlated across consecutive time periods. Thus, this amount of noise is not sufficient to hide the evidence of their physical activity status, and we need to add more noise.

2.3. The Pufferfish Privacy Framework

To account for correlated data, (Kifer and Machanavajjhala, 2014) introduce a novel generalization of the differential privacy framework called Pufferfish.

A Pufferfish privacy framework has three parameters – a set $\mathcal{S}$ of secrets, a set $\mathcal{Q} \subseteq \mathcal{S} \times \mathcal{S}$ of secret pairs, and a class of data distributions Θ.

$\mathcal{S}$ consists of possible facts about the data that we wish to hide, and secrets could refer to a single individual's private data or part thereof. $\mathcal{Q}$ is the set of secret pairs that the privacy algorithm wishes to be indistinguishable. Finally, Θ represents a set of distributions that plausibly generate the data, and the set Θ thus controls the amount of allowable correlation in the data. Each $\theta \in \Theta$ represents a belief an adversary may hold about the data, and our goal is to ensure indistinguishability in the face of these beliefs.

How do we model the two examples we described above in Pufferfish? In the flu example, the set of secrets $\mathcal{S}$ is everyone's disease status. A plausible set of secret pairs $\mathcal{Q}$ is $\{(s_{i,0}, s_{i,1}) : i = 1, \ldots, n\}$ where $s_{i,j}$ means person i has flu status j, $j \in \{0, 1\}$. If only some people in the data are privacy-sensitive, $\mathcal{Q}$ can consist of pairs $(s_{i,0}, s_{i,1})$ where i ranges over all the privacy-sensitive people. Θ is a set of network models that model spread of the disease in question. For example, Θ could be all models where connections represent social interaction, and flu spreads across connections with probability between $[0, 0.5]$. Observe that this example could also be modeled using *group differential privacy* – where we enforce (1) for datasets D and D' which differ in the disease status of an entire group of connected individuals; however, Pufferfish allows for a more nuanced model that accounts for the diffusion rate as well as the connected individuals.

In the physical activity example, the set of secrets $\mathcal{S}$ is the activity status at all times t. $\mathcal{Q}$ is the set of all pairs $(s_{t,a}, s_{t,b})$ where a, b lie in the set of activities, and where $s_{t,a}$ means that activity a happened at time t. Finally, Θ would be a set of time series models, such as Markov or semi-Markov models that capture how people switch between activities. Observe that another plausible way to model this example is via pan-privacy (Dwork et al., 2010a), which models a streaming setting where a new individual arrives at each time period, and the goal is continually release a private statistic with time. However, this setting does not capture this example as pan-privacy still (implicitly) assumes that the arriving individuals are independent of individuals existing in the data.

(Kifer and Machanavajjhala, 2014) shows that in general, we cannot expect to have privacy against all possible distributions and still retain utility. As a result it is essential to select Θ wisely; if Θ is too restrictive, then we may not have privacy against legitimate prior knowledge on the part of the adversary, and if Θ is too large, then the resulting privacy mechanisms will have little utility.

Definition 2.3. *A privacy mechanism M is said to be ϵ-Pufferfish private with Pufferfish parameters $(\mathcal{S}, \mathcal{Q}, \Theta)$ if for datasets $X \sim \theta$ where $\theta \in \Theta$ and for all secret pairs $(s_i, s_j) \in \mathcal{Q}$, and for all $w \in Range(M)$, we have:*

$$e^{-\epsilon} \leq \frac{P(M(X) = w | s_i, \theta)}{P(M(X) = w | s_j, \theta)} \leq e^{\epsilon} \tag{2}$$

when s_i and s_j are such that $P(s_i|\theta) \neq 0$, $P(s_j|\theta) \neq 0$.

Observe that unlike (1), the probability in (2) is with re-

spect to the randomness in the mechanism *and* $X \sim \theta$; to emphasize this, we use the notation X instead of D in the definition; in general, we use D to denote a specific dataset. An alternative interpretation of the definition is for $X \sim \theta$, $\theta \in \Theta$, for all $(s_i, s_j) \in \mathcal{Q}$, and for all $w \in \text{Range}(M)$, we have:

$$e^{-\epsilon} \leq \frac{P(s_i|M(X) = w, \theta)}{P(s_j|M(X) = w, \theta)} \Big/ \frac{P(s_i|\theta)}{P(s_j|\theta)} \leq e^{\epsilon}. \quad (3)$$

In other words, knowledge of the output $M(X)$ does not affect the ratio of likelihood of s_i and s_j, compared to the initial belief.

(Kifer and Machanavajjhala, 2014) shows that Differential Privacy is a special case of Pufferfish, where $\mathcal{S}$ is the set of all facts of the form *Person i has value x* for $i \in \{1, \ldots, n\}$ and x in a domain $\mathcal{X}$, $\mathcal{Q} = \mathcal{S} \times \mathcal{S}$, and Θ is the set of all distributions where each individuals private value is distributed independently.

3. A General Pufferfish Mechanism

Recent work has developed a large number of general privacy mechanisms (Dwork et al., 2006; McSherry and Talwar, 2007; Chaudhuri et al., 2011; Nissim et al., 2007; Dwork and Lei, 2009; Chaudhuri et al., 2014) that guarantee differential privacy under different conditions. However, while a number of Pufferfish mechanisms for specific frameworks are known (Kifer and Machanavajjhala, 2014; He et al., 2014), there is no generic mechanism that applies to *any* Pufferfish framework. Our first contribution is to provide such a mechanism, analogous to the popular Global Sensitivity Mechanism for differential privacy, that guarantees privacy in any Pufferfish framework.

Specifically, given data represented by random variables X (that may be scalar-valued or vector-valued), a Pufferfish privacy framework $(\mathcal{S}, \mathcal{Q}, \Theta)$, and a function F that maps X into a number, our goal is to design a generic mechanism M that satisfies ϵ-Pufferfish privacy in the *given* framework and approximates $F(X)$.

3.1. The Mechanism

Our proposed mechanism is inspired by the global sensitivity mechanism in differential privacy; recall that this mechanism adds noise to the function F proportional to the sensitivity, which is the worst case distance between $F(D)$ and $F(D')$ where D and D' are two datasets that differ in the value of a single individual. In Pufferfish, the quantities analogous to D and D' are the measures $p(F(X)|s_i, \theta)$ and $p(F(X)|s_j, \theta)$ for a secret pair (s_i, s_j), and therefore, the added noise should be proportional to some distance between these two measures. It turns out that the relevant distance is a form of the Wasserstein distance.

Definition 3.1 (p-Wasserstein Distance)**.** *Let $(\mathcal{X}, d)$ be a Radon space, and μ, ν be two probability measures on $\mathcal{X}$ with finite p-th moment. The p-th Wasserstein distance between μ and ν is defined as:*

$$W_p(\mu, \nu) = \left(\inf_{\gamma \in \Gamma(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{X}} d(x,y)^p \, \mathrm{d}\gamma(x,y) \right)^{1/p}$$
$$= \left(\inf_{\gamma \in \Gamma(\mu,\nu)} \mathbb{E}[d(X,Y)^p] \right)^{1/p}, \quad (4)$$

where $\Gamma(\mu, \nu)$ is the set of all couplings γ over μ and ν.

Intuitively, each $\gamma \in \Gamma(\mu, \nu)$ can be regarded as a way to *shift* probability mass between μ and ν; the cost of a shift γ is $(\mathbb{E}_{(X,Y)\sim\gamma}[d(X,Y)^p])^{1/p}$, and the cost of the min-cost shift is the Wasserstein distance. In particular, we are interested in the ∞-Wasserstein distance $W_\infty(\mu, \nu) = \inf_{\gamma\in\Gamma(\mu,\nu)} \max_{(x,y)\in A} d(x, y)$, where $A = \{(x, y)|\gamma(x, y) \neq 0\}$. We specifically consider the case when $d(x, y) = |x - y|$, and thus $W_\infty(\mu, \nu) = \inf_{\gamma\in\Gamma(\mu,\nu)} \max_{(x,y)\in A} |x - y|$.‘

The Wasserstein Mechanism. Given a Pufferfish framework $(\mathcal{S}, \mathcal{Q}, \Theta)$ and a function F, the Wasserstein Mechanism is as follows.

1. **Inputs:** Pufferfish framework $(\mathcal{S}, \mathcal{Q}, \Theta)$, privacy parameter ϵ, function F, dataset D.
2. For any $(s_i, s_j) \in \mathcal{Q}$ and any $\theta \in \Theta$ such that $\Pr(s_i|\theta) \neq 0$ and $\Pr(s_j|\theta) \neq 0$, let $\mu_{i,\theta} = \Pr(F(X) = \cdot|s_i, \theta)$ be the distribution of $F(X)$ conditioned on s_i when $X \sim \theta$. Similarly, define $\mu_{j,\theta}$.
3. Let $W = \sup_{(s_i,s_j)\in\mathcal{Q},\theta\in\Theta} W_\infty(\mu_{i,\theta}, \mu_{j,\theta})$.
4. Output $F(D) + Z$, where $Z \sim \text{Lap}(\frac{W}{\epsilon})$.

3.2. Performance Guarantees

We first establish that the Wasserstein Mechanism satisfies ϵ-Pufferfish privacy.

Theorem 3.2 (Privacy of the Wasserstein Mechanism)**.** *The Wasserstein Mechanism satisfies ϵ-Pufferfish privacy in the framework $(\mathcal{S}, \mathcal{Q}, \Theta)$.*

The proof is presented in Appendix A.1.

Recall that Pufferfish reduces to differential privacy in the special case when (a) $X = \{X_1, \ldots, X_n\}$ with each X_i being a single individual's value, (b) Θ is the set of all product distributions over the domains of each X_i, and (c) $\mathcal{Q} = \{(X_i \text{ is in the dataset with value } x, X_i \text{ is not in the dataset}) : i \in [n], x \in \mathcal{X}\}$. It can be shown that in this case, the Wasserstein mechanism also reduces to the familiar global

sensitivity mechanism. Thus the Wasserstein mechanism is a generalization of the global sensitivity method for Pufferfish.

4. Mechanisms for Bayesian Networks

The Wasserstein mechanism we provide in the last section is very general and applies to any Pufferfish framework; however, perhaps because of its extreme generality, it is extremely computationally expensive. Thus a natural question to ask is whether we can come up with Pufferfish mechanisms which are less computationally challenging, and which apply to some cases of practical interest. In this section, we provide such a mechanism, called the Markov Quilt Mechanism, when the dependence among the variables in X is described by a Bayesian network, Θ has some special structure, and the goal is to keep the value of any individual node in the network private.

We illustrate our mechanism on a Markov Chain, and show that it runs in time polynomial in the length of the chain. This is an important case that models our physical activity measurement example in Section 2. We derive privacy mechanisms for this case, and provide simulations that demonstrate the associated privacy-accuracy tradeoffs.

4.1. The Setting

This section considers a more restricted but more practical setting than the fully general setting of Section 3. We assume that data X can be written as $X = \{X_1, \ldots, X_n\}$, where each X_i lies in a bounded domain $\mathcal{X}$. Let s_a^i denote the event that X_i takes value a. The set of secrets is $\mathcal{S} = \{s_a^i : a \in \mathcal{X}, i \in [n]\}$, and the set of secret pairs is $\mathcal{Q} = \{(s_a^i, s_b^i) : a, b \in \mathcal{X}, i \in [n]\}$. In other words, we would like to hide the value of each individual variable X_i from the adversary.

We assume that there is an underlying known Bayesian network $G = (X, E)$ connecting the variables X that is known to us. Each $\theta \in \Theta$ that describes the distribution of the variables is based on this Bayesian network G, but may have different model parameters. An application of this setting is the physical activity monitoring example in Section 2. See Section 4.3 for more details.

Notations. We use X with a lowercase subscript, for example, X_i, to denote a single node in G, and X with an uppercase subscript, for example, X_A, to denote a set of nodes in G. For a set of nodes X_A we use the notation $\texttt{card}(X_A)$ to denote the number of nodes contained in X_A.

In this section, we will consider functions F that are 1-Lipschitz, as in Definition 4.1. Since any Lipschitz function can be scaled to be 1-Lipschitz, this is not a significant restriction.

Definition 4.1 (1-Lipschitz)**.** *A function $F(X_1, \ldots, X_n)$ is said to be 1-Lipschitz if for all i, and for all pairs X_i and X_i', we have $|F(X_1, \ldots, X_i, \ldots, X_n) - F(X_1, \ldots, X_i', \ldots, X_n)| \leq 1$.*

4.2. The Markov Quilt Mechanism

The main insight behind the Markov Quilt mechanism is that if nodes X_i and X_j are "far apart" in the graph G, then, X_j is largely independent from X_i. Thus, to obscure the effect of X_i on F, it is sufficient to add noise that is proportional to the number of nodes that are "local" to X_i plus a small correction term that accounts for the effect of the distant nodes. The rest of the section will explain how to calculate this correction term, and how to determine the number of local nodes.

We begin with some definitions. First, we quantify how much changing the value of a variable $X_i \in X$ can affect a set of variables $X_R \subset X$, where $X_i \notin X_R$. To this end, we define the *max-influence* of a variable X_i on a set of variables X_R as follows.

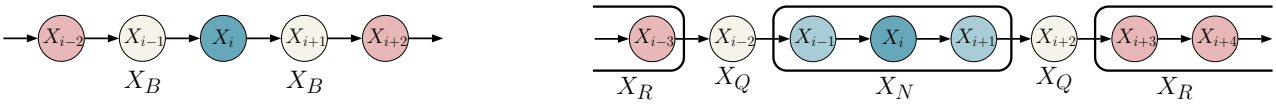
Definition 4.2. *The max-influence of a variable X_i on a set of variables X_R is defined as*

$$e(X_R|X_i) = \max_{a,b \in \mathcal{X}} \sup_{\theta \in \Theta} \max_{x_R \in \mathcal{X}^{\texttt{card}(X_R)}} \log \frac{\Pr(X_R = x_R | X_i = a, \theta)}{\Pr(X_R = x_R | X_i = b, \theta)}.$$

Here $\mathcal{X}$ is the range of any X_j and Θ is the set of allowable distributions. The max-influence is thus the maximum max-divergence (Dwork et al., 2010b) between the distributions $X_R|X_i = a, \theta$ and $X_R|X_i = b, \theta$ where the maximum is taken over any pair $(a, b) \in \mathcal{X} \times \mathcal{X}$ and any $\theta \in \Theta$. If X_R and X_i are independent, then the max-influence of X_i on X_R is 0, and a large max-influence means that changing X_i can have a large impact on the distribution of X_R.

Our goal is to design a mechanism that protects the value of each variable X_i while still adding a small amount of noise to F; to do so, we find large sets of nodes X_R such that X_i has low max-influence on X_R. The naive way to find such sets is brute force search, which takes time exponential in the size of G. We now show how properties of the Bayesian network G can be exploited to perform this search more efficiently.

We next provide a second definition that generalizes the familiar notion of a Markov Blanket. Recall that the Markov Blanket of a node u in a Bayesian network consists of its parents, its children and the other parents of its children, and the rest of the nodes in the network are independent of u conditioned on its Markov Blanket. Its generalization, the Markov Quilt, is defined as follows.



(a) The Markov Blanket of X_i is $X_B = \{X_{i-1}, X_{i+1}\}$

(b) One Markov Quilt of X_i is $X_Q = \{X_{i-2}, X_{i+2}\}$ with corresponding $X_N = \{X_{i-1}, X_i, X_{i+1}\}$, $X_R = \{\dots, X_{i-3}, X_{i+3}, \dots\}$

Figure 1. Illustration of Markov Blanket and Markov Quilt

Definition 4.3 (Markov Quilt)**.** *A set of nodes $X_Q, Q \subset [n]$ in a Bayesian network $G = (X, E)$ is a Markov Quilt for a node X_i if the following conditions hold:*

1. *Deleting X_Q partitions G into parts X_N and X_R such that (a). $X = X_N \cup X_Q \cup X_R$ (b). $X_i \in X_N$ and (c). there is no edge in G that connects a node in X_R to a node in X_N.*
2. *For all $x_R \in \mathcal{X}^{\texttt{card}(X_R)}$, all $x_Q \in \mathcal{X}^{\texttt{card}(X_Q)}$ and for all $a \in \mathcal{X}$, we have:*

$$\begin{aligned} &P(X_R = x_R | X_Q = x_Q, X_i = a) \\ =&P(X_R = x_R | X_Q = x_Q) \end{aligned}$$

In other words, X_R is independent of X_i conditioned on X_Q.

Intuitively, X_R is a set of "remote" nodes that are far from X_i, and X_N is the set of "nearby" nodes; X_N and X_R are separated by the Markov Quilt X_Q. Observe that unlike Markov Blankets, a node can have many Markov Quilts. Figure 1 shows an example.

Suppose our goal is to *protect* the private value of a node X_i. It turns out that if we can find an X_Q such that (a). the max-influence of X_i on X_Q is at most δ and (b). X_Q is a Markov Quilt of X_i, deleting which splits up X into X_N and X_R, then, adding Laplace noise of scale parameter $\texttt{card}(X_N)/(\epsilon - \delta)$ to F is sufficient to preserve ϵ Pufferfish privacy.

This motivates the following mechanism, which we call the Markov Quilt Mechanism. To protect the private value of a specific X_i, we search over a set $S_{Q,i}$ of Markov Quilts X_Q for X_i and pick the one which requires adding the least amount of noise to guarantee privacy. We then iterate this process over all X_i and add to F the maximum amount of noise needed to protect any X_i; this ensures that the private values of *all nodes* are protected.

1. **Inputs.** Dataset D, function F, Pufferfish framework $(\mathcal{S}, \mathcal{Q}, \Theta)$, privacy parameter ϵ.
2. For all X_i:
3. Iterate over all Markov Quilts X_Q for X_i in $S_{Q,i}$:
   4. (a) Suppose deleting X_Q breaks up the underlying Bayesian network into a "nearby" set X_N and a "remote" set X_R with $X_i \in X_N$.
      (b) If the max-influence $e(X_Q | X_i) < \epsilon$, define the score of X_Q as: $\sigma(X_Q) = \frac{\texttt{card}(X_N)}{\epsilon - e(X_Q|X_i)}$; otherwise $\sigma(X_Q) = \infty$.
      (c) Let $\sigma_i = \min_{X_Q \in S_{Q,i}} \sigma(X_Q)$
5. Let $\sigma_{\max} = \max_i \sigma_i$.
6. Output: $F(D) + \sigma_{\max} \cdot Z$, where $Z \sim \text{Lap}(1)$.

First, we note that the algorithm makes an underlying assumption that the max-influence of a node X_i on a set of nodes X_Q may be computed relatively easily; this assumption holds when Θ has some nice structure; see Section 4.3 for a concrete example. Second, the set $S_{Q,i}$ of Markov Quilts that the mechanism searches over can be restricted in specific ways to ensure computational efficiency; however, a potential cost is statistical efficiency – we may end up adding more noise than necessary to preserve privacy because we may exclude some Markov Quilts with very low scores.

Performance Guarantees. The privacy of the Markov Quilt mechanism follows from Theorem 4.4.

Theorem 4.4 (Privacy of Markov Quilt Mechanism)**.** *Let F be a 1-Lipschitz function, and suppose data is represented by a random variable $X = \{X_1, \dots, X_n\}$ where $X_i \in \mathcal{X}$. Suppose we are given a Pufferfish framework $(\mathcal{S}, \mathcal{Q}, \Theta)$ where $\mathcal{S} = \{s_a^i : a \in \mathcal{X}, i \in [n]\}$, the set of secret pairs $\mathcal{Q} = \{(s_a^i, s_b^i) : a, b \in \mathcal{X}, i \in [n]\}$, and each $\theta \in \Theta$ is based on a fixed Bayesian network $G = (X, E)$.*

If each $S_{Q,i}$ is non-empty, then the Markov Quilt Mechanism preserves ϵ-Pufferfish privacy in the framework $(\mathcal{S}, \mathcal{Q}, \Theta)$.

The proof is presented in the Appendix A.2.

4.3. Case Study: Markov Chain

We now show that the Markov Quilt mechanism can be simplified even further when the Bayesian network underlying X is a Markov Chain, which models the physical activity measurement example.

The Setting. We consider data described by a random vector $X = \{X_1, X_2, \ldots, X_T\}$ over time $t \in [T]$ where each X_t represents a state that lies in a finite discrete set $\mathcal{X} = [k]$. In the physical activity monitoring example, $\mathcal{X}$ is a set of activities, and each element in $[k]$ represents a particular activity, such as *Walking*, *Running*, *Sitting* and so on. We would like to hide from an adversary the evidence that X_t takes a specific value; using s_a^t represents the event that X_t takes value a, our secret set is $\mathcal{S} = \{s_a^t : a \in \mathcal{X}, t \in [T]\}$, and our set of secret pairs is $\mathcal{Q} = \{(s_a^t, s_b^t) : a, b \in \mathcal{X}, t \in [T]\}$.

The underlying Bayesian network G is a Markov chain $X_1 \to X_2 \to \ldots \to X_T$, where for each t, X_{t+1} is independent of X_τ, $\tau < t$, conditioned on X_t. Each $\theta \in \Theta$ corresponds to a $k \times k$ transition matrix P_θ for this Markov chain. For simplicity, we focus on Markov Chains whose transition matrices do not change with time; a similar analysis may also be applied to time-varying Markov Chains with some additional complexity.

Properties. We begin by showing that this graphical model has Markov Quilts with particularly simple structures.

Lemma 4.5. *Let a, b and i be positive integers such that $i - a \geq 1$ and $i + b \leq T$. Then, $X_Q = \{X_{i-a}, X_{i+b}\}$ is a Markov Quilt for X_i with $X_N = \{X_{i-a+1}, \ldots, X_{i+b-1}\}$, and $X_R = \{X_1, \ldots, X_{i-a-1}\} \cup \{X_{i+b+1}, \ldots, X_T\}$.*

Moreover, $X_Q = \{X_{i+b}\}$ is a Markov Quilt for X_i with $X_N = \{X_1, \ldots, X_{i+b-1}\}$ and $X_R = \{X_{i+b+1}, \ldots, X_T\}$. Similarly, $X_Q = \{X_{i-a}\}$ is a Markov Quilt for X_i with $X_N = \{X_{i-a+1}, \ldots, X_T\}$ and $X_R = \{X_1, \ldots, X_{i-a-1}\}$.

We next quantify the max-influence $e(X_Q|X_i)$ of a Markov Quilt X_Q for X_i in terms of properties of Θ. In general, this is a complex task, but it is considerably simplified for the model class we consider. We first define two pieces of notation.

For any $\theta \in \Theta$, let π_θ be the stationary distribution of the Markov Chain whose transition matrix is P_θ. We define:

$$\pi_\Theta = \min_{x \in \mathcal{X}, \theta \in \Theta} \pi_\theta(x)$$

as the probability of the least probable state in the stationary distribution corresponding to any $\theta \in \Theta$. Additionally, we define:

$$g_\Theta = \min_{\theta \in \Theta} \min\{1 - |\lambda| : P_\theta x = \lambda x, \lambda < 1\}$$

as the least eigengap of any P_θ.

We make the following assumption about Θ.

Condition 4.6. *We assume that the Markov Chain induced by each $P_\theta \in \Theta$ is irreducible and aperiodic. This implies that each P_θ has a unique stationary distribution π_θ.*

Additionally, we assume that each P_θ is a reversible Markov Chain, $\pi_\Theta > 0$ and that $g_\Theta > 0$.

Condition 4.6 is quite reasonable; if P_θ does not have a stationary distribution, then the influence of the initial state may not decay over time, and we may need to add $\mathcal{O}(T)$ noise for privacy. The second part ensures that the decay in the influence of the initial state is fast enough, and may be relaxed somewhat.

Provided Condition 4.6 holds, we can provide upper bounds on the max-influence of a Markov Quilt X_Q on a node X_i.

Lemma 4.7. *Let X_i be any node in the Markov Chain, and suppose a and b are integers such that $\min(a, b) \geq \frac{\log(1/\pi_\Theta)}{g_\Theta}$. If $X_Q = \{X_{i-a}, X_{i+b}\}$ is a Markov Quilt for X_i, then:*

$$e(X_Q|X_i) \leq \log\left(\frac{\pi_\Theta + \exp(-g_\Theta b)}{\pi_\Theta - \exp(-g_\Theta b)}\right) + 2\log\left(\frac{\pi_\Theta + \exp(-g_\Theta a)}{\pi_\Theta - \exp(-g_\Theta a)}\right).$$

Moreover, if $X_Q = \{X_{i-a}\}$ is a Markov Quilt for X_i, then:

$$e(X_Q|X_i) \leq 2\log\left(\frac{\pi_\Theta + \exp(-g_\Theta a)}{\pi_\Theta - \exp(-g_\Theta a)}\right).$$

and if $X_Q = \{X_{i+b}\}$ is a Markov Quilt for X_i, then:

$$e(X_Q|X_i) \leq \log\left(\frac{\pi_\Theta + \exp(-g_\Theta b)}{\pi_\Theta - \exp(-g_\Theta b)}\right).$$

and if $X_Q = \emptyset$ is a trivial Markov Quilt for X_i, then $e(X_Q|X_i) = 0$.

As a consequence of Lemmas 4.5 and 4.7, we can assign $S_{Q,i}$, the set of Markov Quilts for node X_i, to:

$$S_{Q,i} = \Big\{\{X_{i-a}, X_{i+b}\}, \{X_{i-a}\}, \{X_{i+b}\}, \emptyset \\ \forall a \in \Big\{\frac{\log(1/\pi_\Theta)}{g_\Theta}, \ldots, i-1\Big\}, \\ b \in \Big\{\frac{\log(1/\pi_\Theta)}{g_\Theta}, \ldots, T-i\Big\}\Big\}.$$

When g_Θ and π_Θ are known, (an upper bound on) the score of each X_Q may be calculated quite easily in $\mathcal{O}(1)$ time based on Lemma 4.7. Thus, the computational complexity of the Markov Quilt mechanism in this case is $\mathcal{O}(T^3)$. A more optimized version which runs in $\mathcal{O}(T^2)$ time in some good cases is derived in Appendix B.2.

Generalization to Vector-valued Functions. The Markov Quilt Mechanism can be easily generalized to vector-valued functions F. If F is 1-Lipschitz with respect to L_1 norm, then from Proposition 1 of (Dwork et al., 2006), adding noise drawn from $\sigma_{\max} \cdot \text{Lap}(1)$ to each output dimension of F guarantees ϵ-Pufferfish privacy, where $\sigma_{\max}$ is the quantity in the Markov Quilt Mechanism.

5. Simulations

We now illustrate the performance of the Markov Quilt Mechanism through simulations. We consider the setting in Section 4.3 where the underlying Bayesian network is an *aperiodic*, *irreducible* and *reversible* discrete time Markov chain with k states. The function F we use is the histogram of states over time $t = 1, \ldots, T$. The output of F is a k-dimensional vector $(f_1, \ldots, f_k)$ where f_i is the frequency of state i in times 1 to T, and the desired output is noisy frequencies $(f'_1, \ldots, f'_k)$. This histogram function is $2/T$-Lipschitz with respect to the L_1 norm, and thus can be scaled to fit our mechanism.

Synthetic data is generated as follows. Firstly, we choose two generating parameters $\pi_{\text{gen}} \in (0, 1/k)$, $g_{\text{gen}} \in (0, 1)$ and generate an *aperiodic*, *irreducible* and *reversible* transition matrix P such that $\pi_{\min}(P) \geq \pi_{\text{gen}}$ and $g(P) \geq g_{\text{gen}}$. Then we generate a random vector $q \in \mathbb{R}^k$ uniformly from the probability simplex as the initial distribution. A sequence $X = \{X_i\}_{i=1}^T, X_i \in [k]$ is then generated from the Markov Chain induced by P and q as our dataset.

Notice that the transition matrix P or the generating parameters $\pi_{\text{gen}}, g_{\text{gen}}$ are not known to the private mechanism; the mechanism only knows conservative estimates of π_{gen} and g_{gen}, denoted by π_{est} and g_{est}. We enforce that $\pi_{\text{est}} \leq \pi_{\text{gen}}$ and $g_{\text{est}} \leq g_{\text{gen}}$. Therefore, $\pi_{\text{est}}, g_{\text{est}}$ together with ϵ can be viewed as a set of "privacy parameters". As ϵ gets smaller, the mechanism guarantees more privacy; as π_{est} or g_{est} gets smaller, the mechanism is able to guarantee protection against a larger set of distributions.

Utility of the mechanism is measured by the L_1 distance between the output $(f'_1, \ldots, f'_k)$ with $(f_1, \ldots, f_k)$, the exact frequencies. A smaller distance implies higher utility. In our simulations, we investigate the effect on utility of the three privacy parameters, as well as T, the total length of the sequence. In each simulation, we keep two of the parameters $\{\pi_{\text{est}}, g_{\text{est}}, T\}$ fixed, and vary the remaining one and ϵ. We plot L_1 distance on the y-axis as a function of $\log \epsilon$. P, q are generated with $k = 5$ and generating parameters $\pi_{\text{gen}} = 0.05, g_{\text{gen}} = 0.1$, and are fixed throughout the simulations. We repeat each simulation 100 times and plot the averaged error in the figures. A new dataset $X = \{X_i\}_{i=1}^T$ is generated at each repeated run.

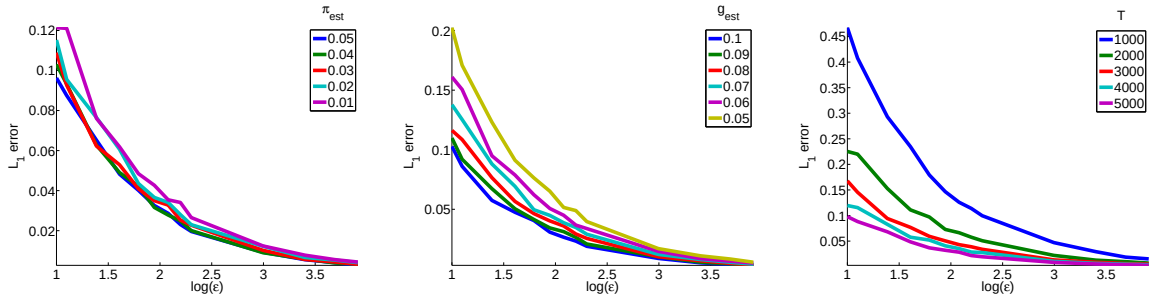
Figure 2 shows the results. We observe that as expected, when the remaining parameters are fixed, utility improves with larger ϵ as well as larger π_{est} and larger g_{est}. An interesting observation is that π_{est} is less influential to utility than g_{est}; this agrees with the dependence of max-influence on π_{est} in Lemma A.2. Finally, Figure 2c shows that as the total sequence length T gets larger, our Markov Quilt Mechanism outputs more accurate results.

6. Related Work

There is a great deal of work on differential privacy (Dwork et al., 2006) and differentially private mechanisms, especially for machine learning. For more details, see (Sarwate and Chaudhuri, 2013; Dwork and Roth, 2013).

Our work uses Pufferfish, a recent generalization of differential privacy introduced by (Kifer and Machanavajjhala, 2014). (Kifer and Machanavajjhala, 2014) provides a number of elegant examples of Pufferfish with associated privacy mechanisms. (He et al., 2014) provides practical and efficient privacy algorithms for specific Pufferfish frameworks. However, none of these works provide a fully general Pufferfish mechanism, nor do they provide algorithms for time series and social network applications. (Kessler et al., 2015) apply Pufferfish to smart-meter data; instead of directly modeling correlation through Markov models, they add noise to the wavelet coefficients of the time series that correspond to different frequencies. Finally, (Yang et al., 2015) consider privacy issues in correlated data with privacy definition that is similar to Pufferfish; unlike us, they look at correlation that can be modeled by a Gaussian Markov Random Field, and their privacy definition accounts for the adversary's prior knowledge more explicitly.

Finally, coupled-worlds privacy (Bassily et al., 2013) is an elegant privacy framework alternative to Pufferfish, which is also capable of accounting for correlation in data. How to model complex kinds of correlated data in this framework, and how to design privacy mechanisms for these models is left as an avenue for future work.



(a) Vary π_{est}. ($g_{\text{est}} = 0.1, T = 5000$). (b) Vary g_{est}. ($\pi_{\text{est}} = 0.05, T = 5000$). (c) Vary T. ($g_{\text{est}} = 0.1, \pi_{\text{est}} = 0.05$).

Figure 2. Averaged L_1 distance between private and non-private histograms vs. $\log(\epsilon)$. Different lines represent different values of (a).π_{est} (b).g_{est} (c). T.

Acknowledgements.

We thank Mani Srivastava for inspiring us to work on the physical activity measurement problem and for early discussions, and Supriyo Chakravarty for helpful discussions. We thank NSF for supporting this work under IIS 1253942 and NIH for partially supporting it under U54 HL108460.

References

R. Bassily, A. Groce, J. Katz, and A. Smith. Coupled-worlds privacy: Exploiting adversarial uncertainty in statistical data privacy. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 439–448. IEEE, 2013.

R. Bassily, A. Smith, and A. Thakurta. Private empirical risk minimization, revisited. arXiv:1405.7085, 2014.

K. Chaudhuri, C. Monteleoni, and A. D. Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12:1069–1109, 2011.

K. Chaudhuri, A. D. Sarwate, and K. Sinha. Near-optimal differentially private principal components. In *Advances in Neural Information Processing Systems*, pages 998–1006, 2012.

K. Chaudhuri, D. Hsu, and S. Song. The large margin mechanism for differentially private maximization. In *NIPS*, 2014.

J. Duchi, M. J. Wainwright, and M. Jordan. Local privacy and minimax bounds: Sharp rates for probability estimation. In *Advances in Neural Information Processing Systems 26*, pages 1529–1537, 2013.

C. Dwork and J. Lei. Differential privacy and robust statistics. In *Proceedings of the 41st annual ACM symposium on Theory of computing*, pages 371–380. ACM, 2009.

C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Theoretical Computer Science*, 9(3-4):211–407, 2013.

C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography*, 2006.

C. Dwork, M. Naor, T. Pitassi, and G. N. Rothblum. Differential privacy under continual observation. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 715–724. ACM, 2010a.

C. Dwork, G. Rothblum, and S. Vadhan. Boosting and differential privacy. In *FOCS*, 2010b.

M. Hardt and A. Roth. Beyond worst-case analysis in private singular vector computation. In *STOC*, 2013.

X. He, A. Machanavajjhala, and B. Ding. Blowfish privacy: tuning privacy-utility trade-offs using policies. In *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*, pages 1447–1458, 2014.

H. Imtiaz and A. D. Sarwate. Symmetric matrix perturbation for differentially private pca. In *ICASSP*, 2015.

P. Jain, P. Kothari, and A. Thakurta. Differentially private online learning. In *COLT*, 2012.

S. Kessler, E. Buchmann, and K. Böhm. Deploying and evaluating pufferfish privacy for smart meter data. *Karlsruhe Reports in Informatics*, 1, 2015.

D. Kifer and A. Machanavajjhala. No free lunch in data privacy. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2011, Athens, Greece, June 12-16, 2011*, pages 193–204, 2011.

D. Kifer and A. Machanavajjhala. Pufferfish: A framework for mathematical privacy definitions. *ACM Trans. Database Syst.*, 39(1):3, 2014.

D. Kifer, A. Smith, and A. Thakurta. Private convex optimization for empirical risk minimization with applications to high-dimensional regression. In *COLT*, 2012.

D. Levin, Y. Peres, and E. Wilmer. *Markov Chains and Mixing Time*. American Mathematical Society, 2009.

F. McSherry and K. Talwar. Mechanism design via differential privacy. In *FOCS*, 2007.

K. Nissim, S. Raskhodnikova, and A. Smith. Smooth sensitivity and sampling in private data analysis. In *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing (STOC '07)*, pages 75–84, New York, NY, USA, 2007. ACM. doi: 10.1145/1250790.1250803. URL http://dx.doi.org/10.1145/1250790.1250803.

A. Sarwate and K. Chaudhuri. Signal processing and machine learning with differential privacy: Algorithms and challenges for continuous data. *Signal Processing Magazine, IEEE*, 30(5):86–94, Sept 2013. ISSN 1053-5888. doi: 10.1109/MSP.2013.2259911.

Y. Wang, S. E. Fienberg, and A. J. Smola. Privacy for free: Posterior sampling and stochastic gradient monte carlo. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 2493–2502, 2015.

B. Yang, I. Sato, and H. Nakagawa. Bayesian differential privacy on correlated data. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 747–762. ACM, 2015.

A. Proofs

A.1. Proof to Theorem 3.2 (Privacy of the Wasserstein Mechanism)

Proof. Let (s_i, s_j) be a pair of secrets in $\mathcal{Q}$ and let $\theta \in \Theta$ such that $p(s_i|\theta) > 0$ and $p(s_j|\theta) > 0$. Recall that $\mu_{i,\theta} = p(F(X) = \cdot|s_i, \theta)$, and $\mu_{j,\theta}$ is defined similarly. Let $\gamma^* = \gamma^*(\mu_{i,\theta}, \mu_{j,\theta})$ be the coupling between $\mu_{i,\theta}$ and $\mu_{j,\theta}$ that achieves the ∞-Wasserstein distance.

Using M to denote the Wasserstein mechanism, we can write for any w,

$$\begin{aligned}
&\frac{p(M(X) = w|s_i, \theta)}{p(M(X) = w|s_j, \theta)} \\
=\ &\frac{\int_t p(F(X) = t|s_i, \theta)p(Z = w - t)dt}{\int_s p(F(X) = s|s_j, \theta)p(Z = w - s)ds} \\
=\ &\frac{\int_t p(F(X) = t|s_i, \theta)e^{-\epsilon|w-t|/W}dt}{\int_s p(F(X) = s|s_j, \theta)e^{-\epsilon|w-s|/W}ds} \\
=\ &\frac{\int_t \int_{s=t-W}^{t+W} \gamma^*(t, s)e^{-\epsilon|w-t|/W}dsdt}{\int_s \int_{t=s-W}^{s+W} \gamma^*(t, s)e^{-\epsilon|w-s|/W}dtds}
\end{aligned} \tag{5}$$

Here the first step follows from the definition of the Wasserstein mechanism, the second step from properties of the Laplace distribution, and the third step from the following property of ∞-Wasserstein distance:

$$\begin{aligned}
p(F(X) = t|s_i, \theta) &= \int_{s=t-W}^{t+W} \gamma^*(t, s)ds, \\
p(F(X) = s|s_j, \theta) &= \int_{t=s-W}^{s+W} \gamma^*(t, s)dt.
\end{aligned}$$

Observe that in the last step of (5), $|s - t| \leq W$ in both the numerator and denominator integrals; therefore we have the following inequality

$$\frac{\int_t \int_{s=t-W}^{t+W} \gamma^*(t, s)e^{-\epsilon|w-t|/W}dsdt}{\int_s \int_{t=s-W}^{s+W} \gamma^*(t, s)e^{-\epsilon|w-s|/W}dtds} \leq e^{\epsilon} \frac{\int_t \int_{s=t-W}^{t+W} \gamma^*(t, s)e^{-\epsilon|w-s|/W}dsdt}{\int_s \int_{t=s-W}^{s+W} \gamma^*(t, s)e^{-\epsilon|w-s|/W}dtds}.$$

By definition of ∞-Wasserstein distance, $\gamma^*(t, s) = 0$ when $|s - t| > W$. Therefore, the right hand side of the above inequality is equal to

$$e^{\epsilon} \frac{\int_t \int_s \gamma^*(t, s)e^{-\epsilon|w-s|/W}dsdt}{\int_s \int_t \gamma^*(t, s)e^{-\epsilon|w-s|/W}dtds} \leq e^{\epsilon}.$$

Similar argument can be used to show that $\frac{p(M(X)=w|s_i,\theta)}{p(M(X)=w|s_j,\theta)} \geq e^{-\epsilon}$. Combining these two facts concludes the proof of the theorem. □

A.2. Proof to Theorem 4.4 (Privacy of the Markov Quilt Mechanism)

Proof. Pick a specific secret pair $(s_a^i, s_b^i) \in \mathcal{Q}$ and a fixed $\theta \in \Theta$. Let X_Q be the Markov Quilt for X_i which has the minimum score $\sigma(X_Q)$, and suppose that deleting X_Q breaks up the underlying Bayesian network into X_N and X_R where $X_i \in X_N$.

For any w we have

$$\begin{aligned}
&\frac{p(F(X) + \sigma_{\max} \cdot Z = w|X_i = a, \theta)}{p(F(X) + \sigma_{\max} \cdot Z = w|X_i = b, \theta)} \\
=&\frac{\int_{x_{R\cup Q}} p(F(X) + \sigma_{\max} Z = w|X_i = a, X_{R\cup Q} = x_{R\cup Q}, \theta)p(X_{R\cup Q} = x_{R\cup Q}|X_i = a, \theta)dx_{R\cup Q}}{\int_{x_{R\cup Q}} p(F(X) + \sigma_{\max} Z = w|X_i = b, X_{R\cup Q} = x_{R\cup Q}, \theta)p(X_{R\cup Q} = x_{R\cup Q}|X_i = b, \theta)dx_{R\cup Q}}.
\end{aligned} \tag{6}$$

Recall that for any value $x_{R\cup Q}$ of $X_{R\cup Q}$,

$$\frac{p(X_{R\cup Q}=x_{R\cup Q}|X_i=a,\theta)}{p(X_{R\cup Q}=x_{R\cup Q}|X_i=b,\theta)}=\frac{p(X_R=x_R|X_Q=x_Q,X_i=a,\theta)p(X_Q=x_Q|X_i=a,\theta)}{p(X_R=x_R|X_Q=x_Q,X_i=b,\theta)p(X_Q=x_Q|X_i=b,\theta)}.$$

Since X_Q is a Markov Quilt for X_i and $X_i \notin X_R$, we have $p(X_R|X_Q,X_i=a,\theta)=p(X_R|X_Q,X_i=b,\theta)$; moreover, by definition of max-influence, $\frac{p(X_Q=x_Q|X_i=a,\theta)}{p(X_Q=x_Q|X_i=b,\theta)} \leq e^{e(X_Q|X_i)}$.

Therefore, the right hand side of (6) is at most:

$$e^{e(X_Q|X_i)} \cdot \frac{\int_{x_{R\cup Q}} p(F(X)+\sigma_{\max}Z=w|X_i=a,X_{R\cup Q}=x_{R\cup Q},\theta)dx_{R\cup Q}}{\int_{x_{R\cup Q}} p(F(X)+\sigma_{\max}Z=w|X_i=b,X_{R\cup Q}=x_{R\cup Q},\theta)dx_{R\cup Q}}. \tag{7}$$

Since F is 1-Lipschitz, for any fixed value of $X_{R\cup Q}$, $F(X)$ can vary by at most $\texttt{card}(X_N)$ (potentially when all the variables in X_N change values). Since $\sigma_{\max} \geq \frac{\texttt{card}(X_N)}{\epsilon - e(X_Q|X_i)}$ and $Z \sim \text{Lap}(1)$, for any $x_{R\cup Q}$, we have:

$$\frac{p(F(X)+\sigma_{\max}Z=w|X_i=a,X_{R\cup Q}=x_{R\cup Q},\theta)}{p(F(X)+\sigma_{\max}Z=w|X_i=b,X_{R\cup Q}=x_{R\cup Q},\theta)} \leq e^{\epsilon - e(X_Q|X_i)}.$$

Combining this with (7), for any w and any secret pair (s_a^i, s_b^i) we have

$$\frac{p(F(X)+\sigma_{\max}Z=w|\theta,s_a^i)}{p(F(X)+\sigma_{\max}Z=w|\theta,s_b^i)} \leq e^{\epsilon},$$

which completes the proof of the theorem. □

A.3. Proof of Lemma 4.5

Proof. It is easy to check that X_R, X_Q, X_N partitions the node set X, thus meets the first criterion of 4.3.

It remains to check that X_R is conditional independent of X_i given X_Q. This can be verified by the d-separation criteria. If the any path from X_i to nodes in X_R is d-separated by X_Q, then the conditional independence claim will hold. In a discrete Markov chain, there exists only one path between X_i and any node $X_j \in X_R$. Since the path connects the nodes in ascending index order, there must be a node in X_Q that blocks the path. Hence X_i is conditionally independent of set X_R given X_Q. The second criterion of 4.3 is met.

Therefore, the choice of X_Q in Lemma 4.5 is indeed an Markov Quilt. □

A.4. Proof of Lemma 4.7

The main ingredient in the proof of Lemma 4.7 is the following (fairly standard) result in Markov Chain theory:

Lemma A.1. *Consider a k-state discrete time Markov Chain with a transition matrix P with eigengap g_*. Let π be the stationary distribution of the chain and let $\pi_{\min} = \min_x \pi(x)$ be the minimum probability of any state in the stationary distribution. If P^t is the t-th power of P such that $P^t(x,y) = \Pr(X_{i+t}=y|X_i=x)$, then,*

$$\left|\frac{P^t(x,y)}{\pi(y)} - 1\right| \leq \frac{\exp(-tg_*)}{\pi_{\min}}.$$

Or equivalently,

$$(1-\Delta_t)\pi(y) \leq P^t(x,y) \leq (1+\Delta_t)\pi(y), \quad \Delta_t = \frac{\exp(-tg_*)}{\pi_{\min}}.$$

This lemma, along with some algebra, suffices to show the following two facts:

Lemma A.2. *Suppose $t > \frac{\log(1/\pi_\Theta)}{g_\Theta}$, and $\Delta_t = \frac{\exp(-tg_\Theta)}{\pi_\Theta}$. Then, for any j, any $\theta \in \Theta$, and any x, x' and y,*

$$\frac{1-\Delta_t}{1+\Delta_t} \leq \frac{\Pr(X_{t+j}=y|X_j=x,\theta)}{\Pr(X_{t+j}=y|X_j=x',\theta)} \leq \frac{1+\Delta_t}{1-\Delta_t}.$$

Lemma A.3. *Suppose* $t > \frac{\log(1/\pi_\Theta)}{g_\Theta}$ *and* $\Delta_t = \frac{\exp(-tg_\Theta)}{\pi_\Theta}$. *Then, for any* $j > 0$, *any* $\theta \in \Theta$, *and any* x, x' *and* y,

$$\left(\frac{1-\Delta_t}{1+\Delta_t}\right)^2 \leq \frac{\Pr(X_j = y | X_{j+t} = x, \theta)}{\Pr(X_j = y | X_{j+t} = x', \theta)} \leq \left(\frac{1+\Delta_t}{1-\Delta_t}\right)^2.$$

Proof. (of Lemma 4.7) First, we can express $e(X_Q|X_i)$ as

$$e(X_Q|X_i) = \max_{x,x' \in \mathcal{X}} \sup_{\theta \in \Theta} \max_{x_Q \in \mathcal{X}^{|Q|}} \log \frac{\Pr(X_Q = x_Q | X_i = x, \theta)}{\Pr(X_Q = x_Q | X_i = x', \theta)}.$$

For any x, x' and any x_Q, by conditional independence,

$$\begin{aligned}\frac{\Pr(X_Q = x_Q | X_i = x, \theta)}{\Pr(X_Q = x_Q | X_i = x', \theta)} =& \frac{\Pr(X_{i+b} = x_{i+b}, X_{i-a} = x_{i-a} | X_i = x, \theta)}{\Pr(X_{i+b} = x_{i+b}, X_{i-a} = x_{i-a} | X_i = x', \theta)} \\ =& \frac{\Pr(X_{i+b} = x_{i+b} | X_i = x, \theta)}{\Pr(X_{i+b} = x_{i+b} | X_i = x', \theta)} \frac{\Pr(X_{i-a} = x_{i-a} | X_i = x, \theta)}{\Pr(X_{i-a} = x_{i-a} | X_i = x', \theta)}.\end{aligned}$$

Taking $\log$ on both sides, we have

$$\log \frac{\Pr(X_Q = x_Q | X_i = x, \theta)}{\Pr(X_Q = x_Q | X_i = x', \theta)} = \log \frac{\Pr(X_{i+b} = x_{i+b} | X_i = x, \theta)}{\Pr(X_{i+b} = x_{i+b} | X_i = x', \theta)} + \log \frac{\Pr(X_{i-a} = x_{i-a} | X_i = x, \theta)}{\Pr(X_{i-a} = x_{i-a} | X_i = x', \theta)}.$$

For any $\theta \in \Theta$, let P be the corresponding transition matrix, and π, g_* be the stationary distribution and eigengap of P. Let $\Delta_t^\theta = \frac{\exp(-tg_*)}{\pi_{\min}}$. Then by Lemma A.2 and Lemma A.3, we have

$$\log \frac{\Pr(X_{i+b} = x_{i+b} | X_i = x, \theta)}{\Pr(X_{i+b} = x_{i+b} | X_i = x', \theta)} \leq \log \frac{1+\Delta_t^\theta}{1-\Delta_t^\theta} \leq \log \frac{1+\Delta_t}{1-\Delta_t} = \log \left(\frac{\pi_\Theta + \exp(-g_\Theta b)}{\pi_\Theta - \exp(-g_\Theta b)}\right)$$

and

$$\log \frac{\Pr(X_{i-a} = x_{i-a} | X_i = x, \theta)}{\Pr(X_{i-a} = x_{i-a} | X_i = x', \theta)} \leq \log \left(\frac{1+\Delta_t^\theta}{1-\Delta_t^\theta}\right)^2 \leq \log \left(\frac{1+\Delta_t}{1-\Delta_t}\right)^2 = 2 \log \left(\frac{\pi_\Theta + \exp(-g_\Theta a)}{\pi_\Theta - \exp(-g_\Theta a)}\right).$$

Combining the two inequalities together, when $X_Q = \{X_{i-a}, X_{i+b}\}$, we have

$$\log \frac{\Pr(X_Q = x_Q | X_i = x, \theta)}{\Pr(X_Q = x_Q | X_i = x', \theta)} \leq \log \left(\frac{\pi_\Theta + \exp(-g_\Theta b)}{\pi_\Theta - \exp(-g_\Theta b)}\right) + 2 \log \left(\frac{\pi_\Theta + \exp(-g_\Theta a)}{\pi_\Theta - \exp(-g_\Theta a)}\right).$$

Moreover, when $X_Q = \{X_{i-a}\}$, the $\frac{\Pr(X_{i+b}=x_{i+b}|X_i=x,\theta)}{\Pr(X_{i+b}=x_{i+b}|X_i=x',\theta)}$ term will degenerate to 1, thus

$$\log \frac{\Pr(X_Q = x_Q | X_i = x, \theta)}{\Pr(X_Q = x_Q | X_i = x', \theta)} \leq 2 \log \left(\frac{\pi_\Theta + \exp(-g_\Theta a)}{\pi_\Theta - \exp(-g_\Theta a)}\right).$$

Similarly, when $X_Q = \{X_{i+b}\}$, the $\frac{\Pr(X_{i-a}=x_{i-a}|X_i=x,\theta)}{\Pr(X_{i-a}=x_{i-a}|X_i=x',\theta)}$ term will degenerate to 1, thus

$$\log \frac{\Pr(X_Q = x_Q | X_i = x, \theta)}{\Pr(X_Q = x_Q | X_i = x', \theta)} \leq \log \left(\frac{\pi_\Theta + \exp(-g_\Theta b)}{\pi_\Theta - \exp(-g_\Theta b)}\right).$$

Since the above results hold for any x, x', x_Q and θ, we can conclude the three statements of Lemma 4.7. □

A.5. Proof to Lemma A.1

The full standard proof can be found at (Levin et al., 2009).

Proof. Let P be the transition matrix. Construct a matrix A s.t. $A(x,y) = \pi(x)^{1/2}\pi(y)^{-1/2}P(x,y)$. A is symmetric since the chain is reversible, therefore A can be written as

$$A = V\Lambda V^{-1} = V\Lambda V^T$$

, in which V is orthonormal.

Define another inner product

$$\langle \cdot, \cdot \rangle_\pi := \sum_{x\in\Omega} f(x)g(x)\pi(x)$$

We call a matrix π-orthonormal if its basis are orthonormal with respect to $\langle \cdot, \cdot \rangle_\pi$.

Let $D = diag\pi$. Since $A = D^{1/2}PD^{-1/2}$ by the construction of A and $A = V\Lambda V^T$, we have

$$P[D^{-1/2}V] = D^{-1/2}V\Lambda$$

Let $U = D^{-1/2}V$ and let u_i, v_i represent the i-th column vector of U and V respectively, we observe

$$\begin{aligned}\langle u_i, u_j \rangle_\pi =& \sum_{x\in\Omega} u_i(x)u_j(x)\pi(x) \\ =& \sum_{x\in\Omega} u_i(x)\pi(x)^{1/2}u_j(x)\pi(x)^{1/2} \\ =& \langle v_i, v_j \rangle\end{aligned}$$

Thus P is π-orthonormal with $\{u_i\}$ as its basis.

$\langle u_i, u_j \rangle_\pi = 1$ when $i = j$ and $\langle u_i, u_j \rangle_\pi = 0$ otherwise.

Then express P^t in terms of U,

$$\begin{aligned}P^t =& D^{1/2}A^tD^{1/2} \\ =& D^{-1/2}V\Lambda V^T D^{1/2} \\ =& [D^{-1/2}V]\Lambda[V^T D]D^{-1/2} \\ =& U\Lambda U^T D\end{aligned}$$

Therefore,

$$P^t(x,y) = \sum_i u_i(y)\lambda_i^t u_i(x)\pi(y)$$

Since the chain is aperiodic and irreducible, P has a unique largest eigenvalue $\lambda_1 = 1$, we can rewrite the expression above as

$$P^t(x,y) = 1 + \sum_{i=2}^{|\Omega|} u_i(y)\lambda_i^t u_i(x)\pi(y)$$

Apply Cauchy-Schwarz inequality,

$$\begin{aligned}\left|\frac{P^t(x,y)}{\pi(y)}-1\right| &= \sum_{i=2}^{|\Omega|} u_i(y)\lambda_i^t u_i(x) \\ &\le \sum_{i=2}^{|\Omega|} u_i(y)\lambda_*^t u_i(x) \\ &\le \lambda_*^t \left[\sum_{i=2}^{|\Omega|} u_i(x)^2 \sum_{i=2}^{|\Omega|} u_i(y)^2\right]^{1/2}\end{aligned}$$

, where λ_* is the eigenvalue of P whose absolute value is the second largest. Furthermore,

$$\sum_{i=2}^{|\Omega|} u_i(x)^2 \le \sum_{i=1}^{|\Omega|} u_i(x)^2 \le \pi(x)^{-1}$$

because $\pi(x) = \left\langle \sum_{i=1}^{|\Omega|} u_i(x)\pi(x)u_i, \sum_{i=1}^{|\Omega|} u_i(x)\pi(x)u_i \right\rangle = \pi(x)^2 \sum_{i=1}^{|\Omega|} u_i(x)^2$ We eventually get

$$\left|\frac{P^t(x,y)}{\pi(y)}-1\right| \le \frac{\lambda_*^t}{\sqrt{\pi(x)\pi(y)}} \le \frac{e^{-(1-\lambda_*)^t}}{\pi_{\min}} = \frac{e^{-g^t}}{\pi_{\min}}$$

, where $g = 1-\lambda_*$ denotes the eigengap of P. □

A.6. Proof to Lemma A.2

Proof. Consider any underlying distribution $\theta \in \Theta$ with transition matrix P, and let the stationary distribution be π and the eigengap of P be g_*. We have

$$\frac{\Pr(X_{t+j}=y|X_j=x)}{\Pr(X_{t+j}=y|X_j=x')} = \frac{P^t(x,y)}{P^t(x',y)} = \frac{P^t(x,y)/\pi(y)}{P^t(x',y)/\pi(y)}.$$

Based on Lemma A.1, for $\Delta_t^\theta = \frac{\exp(-tg_*)}{\pi_{\min}}$ with $\pi_{\min} = \min_x \pi(x)$, we have

$$1-\Delta_t^\theta \le P^t(x,y)/\pi(y) \le 1+\Delta_t^\theta.$$

Recall that $0 \le \Delta_t^\theta \le \Delta_t \le 1$, where Δ_t is as defined in Lemma A.2. Therefore

$$1-\Delta_t \le P^t(x,y)/\pi(y) \le 1+\Delta_t$$

and the lemma follows. □

A.7. Proof to Lemma A.3

Proof. Consider any underlying distribution $\theta \in \Theta$ with transition matrix P, and let the stationary distribution be π and the eigengap of P be g_*. By Bayes' rule we have

$$\frac{\Pr(X_j=y|X_{t+j}=x)}{\Pr(X_j=y|X_{t+j}=x')} = \frac{\Pr(X_{t+j}=x|X_j=y)\Pr(X_j=y)/\Pr(X_{t+j}=x)}{\Pr(X_{t+j}=x'|X_j=y)\Pr(X_j=y)/\Pr(X_{t+j}=x')} = \frac{P^t(y,x)\Pr(X_{t+j}=x')}{P^t(y,x')\Pr(X_{t+j}=x)}.$$

Let $\Delta_t^\theta = \frac{\exp(-tg_*)}{\pi_{\min}}$ with $\pi_{\min} = \min_x \pi(x)$. Based on Lemma A.1, we have

$$(1-\Delta_t^\theta)\pi(x) \le P^t(y,x) \le (1+\Delta_t^\theta)\pi(x).$$

We also have

$$\begin{aligned} P(X_{t+j} = x) &= \sum_y P(X_{t+j} = x | X_j = y) P(X_j = y) \\ &\leq \max_y P(X_{t+j} = x | X_j = y) = \max P^t(y, x) \\ &\leq (1 + \Delta_t^\theta)\pi(x), \end{aligned}$$

and similarly,

$$\begin{aligned} P(X_{t+j} = x) &= \sum_y P(X_{t+j} = x | X_j = y) P(X_j = y) \\ &\geq \min_y P(X_{t+j} = x | X_j = y) = \min_y P^t(y, x) \\ &\geq (1 - \Delta_t^\theta)\pi(x). \end{aligned}$$

Therefore

$$\left(\frac{1 - \Delta_t^\theta}{1 + \Delta_t^\theta}\right)^2 \leq \frac{P^t(y, x) \Pr(X_{t+j} = x')}{P^t(y, x') \Pr(X_{t+j} = x)} \leq \left(\frac{1 + \Delta_t^\theta}{1 - \Delta_t^\theta}\right)^2.$$

The lemma follows from the fact that $0 \leq \Delta_t^\theta \leq \Delta_t \leq 1$, where Δ_t is as defined in Lemma A.2. □

B. Implementing the Markov Quilt Mechanism for Markov Chains

In this section, we provide a detailed description of the Markov Quilt Mechanism for Markov chain, along with a more computationally efficient version of the algorithm. Throughout this section we assume that the Pufferfish parameters $(\mathcal{S}, \mathcal{Q}, \Theta)$ and the corresponding g_Θ, π_Θ are known.

B.1. The Markov Quilt Mechanism for Markov Chains

Algorithm 1 presents the Markov Quilt mechanism for a Markov Chain. Recall that in this case, the set $S_{Q,i}$ of plausible Markov Quilts for node X_i that we search over is of the form:

$$S_{Q,i} = \left\{ \{X_{i-a}, X_{i+b}\}, \{X_{i-a}\}, \{X_{i+b}\}, \emptyset, \forall a \in \left\{ \frac{\log(1/\pi_\Theta)}{g_\Theta}, \ldots, i-1 \right\}, b \in \left\{ \frac{\log(1/\pi_\Theta)}{g_\Theta}, \ldots, T-i \right\} \right\}. \quad (8)$$

Algorithm 1 enumerates over all Markov Quilts in $S_{Q,i}$ for all nodes X_i in the chain, computes the requisite scores, and calculates $\sigma_{\max}$, which is (approximately) the minimum standard deviation of the amount of noise needed to keep the values of all nodes private. Finally, it adds Laplace noise with standard deviation proportional to $\sigma_{\max}$ to the exact value of $F(D)$ to ensure privacy.

Algorithm 1 BasicMarkovQuiltMechansim(ϵ, F, D)

```
Initialize σ_max ← 0
for X_i ∈ 𝒳 do
    σ_i ← FindBestMarkovQuilt(i, ε)
    σ_max = max(σ_max, σ_i)
end for
return F(D) + σ_max · Lap(1)
```

Algorithm 2 shows the FindBestMarkovQuilt function that returns the minimal score of any Markov Quilt in $S_{Q,i}$. This is computed by searching over the set $S_{Q,i}$ of candidate Markov quilts for X_i.

Algorithm 2 FindBestMarkovQuilt(i, ϵ)

$\sigma_i \leftarrow \infty$
$(a^*, b^*) \leftarrow (i, n-i+1)$
for $(X_{i-a}, X_{i+b}), a, b \geq \log(1/\pi_\Theta)/g_\Theta$ that is a Markov Quilt for X_i **do**
  if $i - a = 0$ and $i + b = n + 1$ **then**
    $e(a, b) \leftarrow 0$
  else if $i - a = 0$ **then**
    $e(a, b) \leftarrow \log\left(\frac{\pi_\Theta + \exp(-g_\Theta b)}{\pi_\Theta - \exp(-g_\Theta b)}\right)$
  else if $i + a = n + 1$ **then**
    $e(a, b) \leftarrow 2\log\left(\frac{\pi_\Theta + \exp(-g_\Theta a)}{\pi_\Theta - \exp(-g_\Theta a)}\right)$
  else
    $e(a, b) \leftarrow \log\left(\frac{\pi_\Theta + \exp(-g_\Theta b)}{\pi_\Theta - \exp(-g_\Theta b)}\right) + 2\log\left(\frac{\pi_\Theta + \exp(-g_\Theta a)}{\pi_\Theta - \exp(-g_\Theta a)}\right)$
  end if
  if $e(a, b) > \epsilon$ **then**
    Pass
  else
    $\sigma_i \leftarrow \min(\frac{a+b-1}{\epsilon - e(a,b)}, \sigma_i)$
    If $\sigma_i = \frac{a+b-1}{\epsilon - e(a,b)}$ then $(a^*, b^*) \leftarrow (a, b)$.
  end if
end for
return σ_i, a^*, b^*

B.2. A Faster Algorithm

BasicMarkovQuiltMechansim needs to enumerate over all nodes in the chain and compute the best Markov quilt for each one. This can be improved in most cases. The intuition is that the best Markov quilt for a node near the center of the chain should also be a privacy-preserving choice of quilt for a node near the end of the chain. For ease of presentation, we introduce two imaginary nodes X_0 and X_{n+1} at either end of the chain. We prepend X_0 to to X_1 and append X_{n+1} to X_n. Quilt $\{X_0, X_{i+b}\}$ in the algorihm description corresponds to Quilt $\{X_{i+b}\}$ in $S_{Q,i}$, $\{X_{i-a}, X_{n+1}\}$ corresponds to Quilt $\{X_{i-a}\}$, and Quilt $\{X_0, X_{n+1}\}$ corresponds to the trivial quilt $\emptyset$.

We shall show that the noise parameter σ calculated from this quilt is enough to work for all nodes, that is, preserves ϵ-Pufferfish privacy for all nodes. Starting from a node in the middle of the chain offers better chance of finding such a quilt than starting from the end. Algorithm 3 shows the algorithm.

Claim B.1. *The standard deviation of the noise added to $F(D)$ by BasicMarkovQuiltMechansim is larger than or equal to that added by FastMarkovQuiltMechanism.*

Proof. This is obvious. The $\sigma_{\max}$ in BasicMarkovQuiltMechansim is the maximum of FindBestMarkovQuilt(i, ϵ) for all X_is in $\mathcal{X}$, while FastMarkovQuiltMechanism's $\sigma_{\max}$ is the maximum of of FindBestMarkovQuilt(i, ϵ) for only a subset of $\mathcal{X}$. Therefore, the amount of noise added to $F(D)$ by BasicMarkovQuiltMechansim cannot be smaller. □

Claim B.2. *Suppose the best Markov Quilt found by FastMarkovQuiltMechanism is $\{X_{i-a}, X_{i+b}\}$ where $i - a \geq 1$ and $i + b \leq n$. Then, FastMarkovQuiltMechanism ensures ϵ-Pufferfish privacy.*

Proof. First let $j \neq i$ be such that $j - a_i \geq 1$ and $j + b_i \leq n$. Then, $\{X_{j-a}, X_{j+b}\}$ is a Markov Quilt for X_j with score equal to σ_i, and thus $\sigma_{\max} = \sigma_i \geq \sigma_j$. Thus the amount of noise added by FastMarkovQuiltMechanism is sufficient to protect the privacy of X_j.

Next consider a $j \neq i$ such that $j - a_i \geq 1$ but $j + b_i > n$. Then, $\{X_{j-a}\}$ is a Markov Quilt for X_j with score less than σ_i. Thus $\sigma_{\max} = \sigma_i \geq \sigma_j$ in this case as well. The final case, where $j - a_i < 1$ but $j + b_i \leq n$ is analogous. Thus, the amount of noise added by FastMarkovQuiltMechanism is sufficient to protect the privacy of such X_j's, and as such it preserves ϵ-Pufferfish privacy. □

Algorithm 3 FastMarkovQuiltMechanism(ϵ, F, D)

Initialize $\sigma_{\max} \leftarrow 0$
$n = |\mathcal{X}|$
$i = \lfloor \frac{n}{2} \rfloor$
while i has not being explored **do**
  Mark i as explored
  $(a_i, b_i, \sigma_i) \leftarrow$ FindBestMarkovQuilt (i, ϵ)
  $\sigma_{\max} = \max(\sigma_{\max}, \sigma_i)$
  if $i - a_i = 0$ and $i + b_i = n + 1$ **then**
    break
  else if $i - a_i = 0$ **then**
    $i = i + 1$
  else if $i + b_i = n + 1$ **then**
    $i = i - 1$
  else
    break
  end if
end while
return $F(D) + \sigma_{\max} \cdot \text{Lap}(1)$

For the rest of the proof, observe that there are three cases:

1. FastMarkovQuiltMechanism finds a node X_i and a corresponding Markov Quilt $\{X_{i-a}, X_{i+b}\}$ such that $i - a \geq 1$ and $i + b \leq n$. Claim B.2 shows that in this case it ensures ϵ-Pufferfish privacy.

2. FastMarkovQuiltMechanism finds the Markov Quilt $\{X_0, X_{n+1}\}$. In this case, it adds noise with parameter n/ϵ, which trivially ensures ϵ-Pufferfish privacy.

3. FastMarkovQuiltMechanism finds a Markov Quilt of the form $\{X_0, X_m\}$ where $m \leq n$. This case will only happen when (a) there exists some k such that the best Markov Quilts for $X_k, \ldots, X_{n/2}$ are of the form $\{X_{j-a_j}, X_{n+1}\}$ and (b) the best Markov Quilt for X_{k-1} is $\{X_0, X_m\}$. In this case, observe that any X_j for $j \geq k$ is protected as σ_j can only be smaller than the maximum of $\sigma_k, \sigma_{k+1}, \ldots, \sigma_{n/2}$. Moreover, any X_j for $j \leq k - 1$ is also protected, as σ_j is at most σ_{k-1}.

4. FastMarkovQuiltMechanism find a Markov Quilt of the form $\{X_m, X_{n+1}\}$ for $m \geq 1$. This case is analogous to the previous case.