Imperial College London

Department of Electrical and Electronic Engineering

Privacy and Security in Cyber-Physical Systems

by

Ecenaz Erdemir

February 2022

Supervised by

Prof. Deniz Gündüz and Prof. Pier Luigi Dragotti

A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy in Electrical and Electronic Engineering of Imperial College London and the Diploma of Imperial College London

Abstract

Data privacy has attracted increasing attention in the past decade due to the emerging technologies that require our data to provide utility. Service providers (SPs) encourage users to share their personal data in return for a better user experience. However, users' raw data usually contains implicit sensitive information that can be inferred by a third party. This raises great concern about users' privacy.

In this dissertation, we develop novel techniques to achieve a better privacy-utility trade-off (PUT) in various applications. We first consider smart meter (SM) privacy and employ physical resources to minimize the information leakage to the SP through SM readings. We measure privacy using information-theoretic metrics and find private data release policies (PDRPs) by formulating the problem as a Markov decision process (MDP). We also propose noise injection techniques for time-series data privacy. We characterize optimal PDRPs measuring privacy via mutual information (MI) and utility loss via added distortion. Reformulating the problem as an MDP, we solve it using deep reinforcement learning (DRL) for real location trace data. We also consider a scenario for hiding an underlying "sensitive" variable and revealing a "useful" variable for utility by periodically selecting from among sensors to share the measurements with an SP. We formulate this as an optimal stopping problem and solve using DRL. We then consider privacy-aware communication over a wiretap channel. We maximize the information delivered to the legitimate receiver, while minimizing the information leakage from the sensitive attribute to the eavesdropper. We propose using a variational-autoencoder (VAE) and validate our approach with colored and annotated MNIST dataset. Finally, we consider defenses against active adversaries in the context of security-critical applications. We propose an adversarial example (AE) generation method exploiting the data distribution. We perform adversarial training using the proposed AEs and evaluate the performance against real-world adversarial attacks.

Acknowledgements

First of all, I would like to sincerely thank my supervisors, Prof. Deniz Gündüz and Prof. Pier Luigi Dragotti, for their invaluable academic guidance, immense knowledge, encouragement and continuous support, without which this work would not have been possible. I am grateful to them for providing me with such an excellent environment to grow as a research scientist, and passing their passion and motivation for research on to me, which has definitely affected my decisions for my research career and will continue to affect my future life. I would also like to thank my examiners, Dr. Hamed Haddadi and Prof. Tobias Oechtering, for their thorough reading of this thesis, and the interesting and enjoyable discussions we had during the viva examination, which incented me to widen my research perspective. I would like to acknowledge the financial support provided by the Republic of Turkey, Ministry of National Education for funding my Ph.D. studies, and for enabling me to study with great researchers at one of the most distinguished universities.

I must also thank my manager Barış Coşkun and mentor Sergül Aydöre for their endless support during and after my internship at AWS. It has been a real privilege for me to share their tremendous scientific knowledge and brilliant insights.

I am very thankful to my dearest friends Roy, Zaid and Rui, who shared with me the ups and downs of Ph.D., as well as coffee breaks, dinners, and our lockdown hangouts in the past two years. I appreciate every single moment that we share together. I have to thank all the people of the Information Processing and Communications Lab and some precious friends: Alma, Mohammad, Hakan, Burak, Kate, David, Giulio, Morteza, Mihajlo, Murat, Yasin, Kiarash, Waleed, Szymon, Sam, Emre, Can, Mikolaj, Yuxuan, Mahdi, Nitish, Ezgi, for the warm and friendly atmosphere they created, and lovely experiences we shared throughout the past years. I would also like to thank Mohammad Malekzadeh for his valuable comments and our discussions on this thesis.

Son olarak, beni her konuda destekleyen, her zorlukta yanımda olan aileme teşekkür etmek istiyorum. Geçtiğimiz dört yıl içinde ailemizin yaşadığı zorlukların arasında her zaman bana ayıracak vaktiniz, sabrınız ve sevginiz oldu. Hayatımda tanıdığım en güçlü kadınlar olan annem Neşe Erdemir, ablalarım Miraç Tamer ve Kıvanç Çakıcı, hiçbir zorlukta pes etmeyerek, kendime ve yapabileceklerime olan inancımı körükleyerek ilham kaynağım olduğunuz için size teşekkür ederim. Babam Rıdvan Erdemir, doktora süresince benden hiçbir desteğini esirgemediğin için teşekkür ederim. Ayrıca ağabeylerim Nail Çakıcı ve Bora Tamer, karşılaştığım birçok zorlukta ablalarımla beraber yanımda olduğunuz için size de teşekkür etmek istiyorum. Doktorayı başarılı bir şekilde geçirmemde sizlerin bu süreci benim için ne kadar kolaylaştırdığınız su götürmez bir gerçektir. Ve son olarak yeğenlerim Elif Deniz Çakıcı ve Balkan Tamer, mutluluk ve geleceğe dair umut kaynağım olduğunuz için sizlere de teşekkür etmek istiyorum. Sevgili ailem, ayrı ayrı her birinizin desteği sayesinde bugün hayatta bulunmaktan gurur duyduğum bir yerdeyim ve tüm bu fırsatları bana sağladığınız için sizlere sonsuz teşekkür ederim.

©

Copyright

Declaration of Originality

I, Ecenaz Erdemir, declare that this thesis titled, 'Privacy and Security in Cyber-Physical Systems' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at Imperial College of Science, Technology and Medicine.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.

Signed: Ecenaz Erdemir

Date: 3 February 2022

Contents

List of Figures

List of Tables

Abbreviations

AE	Adversarial Example
A2C	Advantage Actor-Critic
AT	Adversarial Training
CV	Computer Vision
DNN	Deep Neural Networks
DP	Differential Privacy
DRL	Deep Reinforcement Learning
DyP	Dynamic Programming
LSTM	Long short-term memory
MDP	Markov Decision Process
MI	Mutual Information
ITP	Information-theoretic Privacy
IoT	Internet of Things
POMDP	Partially Observable Markov Decision Process
PUT	Privacy-utility Trade-off
RB	Rechargeable Battery
RES	Renewable Energy Source
SG	Smart Grid
SM	Smart Meter
SP	Service Provider
UP	Utility Provider

To my family.

Chapter 1

Introduction

1.1 Motivation

Data sharing has become a common practice in the past decade due to the emerging technologies that utilize personal data to provide better services. In particular, the recent advances in Internet of things (IoT) devices have increased the variety of services they provide, such as health and activity monitoring, financial analysis, weather analysis, location-based services, smart speakers and smart metering. Moreover, the integration of some IoT devices with social networks has encouraged the users to share their personal data in return of *utility* that provides them with a better user experience on these social platforms. While the users can receive hotel, restaurant and product recommendations from Facebook, Twitter or YouTube when they share their location information, they can also benefit from the personalized dietary tips as a result of sharing their Fitbit activity. However, in most of these applications, data collected by IoT devices contain sensitive personal information about the users. The concerning fact is that as soon as the user's raw data is sent to the service provider's cloud, the sensitive information can be inferred, misused or leaked through security vulnerabilities even if the service provider (SP), or utility provider (UP) in energy consumption context, and/or the communication link are trusted third parties. This causes the violation of the user's *privacy*.

Account balances, biomedical measurements, location trace, smart assistant search history, metadata of uploaded pictures and smart meter readings are typical examples of data which carry sensitive personal information. For instance, a malicious third party can derive an individual's frequently visited destinations, financial situation or social relationships using the shared location information [4]. The information containing the camera model and the location where a picture is taken is embedded in its metadata. This information is preserved even when the pictures are inserted in another document, e.g., Microsoft office documents, and anyone receiving that document does not only see the picture but also finds out where and when it was taken, and which camera was used. Sharing a picture on social media or an item on online marketing platforms can unwittingly disclose the user's home address [5]. Using non-intrusive load monitoring techniques on smart meter data, an eavesdropper can deduce the user's presence at home, disabilities and even political views due to the TV channel the user is watching [6]. Besides all, the most sensitive private information, such as patient history, chronic diseases, disabilities, psychological state and daily habits can be revealed by health monitoring systems [7, 8]. Therefore, privacy is an important concern when using IoT services, and there is a growing demand from consumers to keep their personal information private against malicious attackers or untrusted service providers, while preserving the utility obtained from these IoT services.

The need for better data privacy was recently put high on the global cybersecurity agenda by the EU General Data Protection Regulation (GDPR) that took effect on May 25, 2018 [9]. Thanks to the emerging privacy legislation worldwide, such as California Consumer Privacy Act (CCPA), Personal Information Protection and Electronic Documents Act (PIPEDA), companies increasingly recognize that data privacy is mission critical and an essential expenditure. Therefore, there has been a surge of interest in privacy measures and privacy preserving techniques in the literature [10–17].

Besides differential privacy (DP) which is the most widely adopted one [10], various alternative privacy metrics can be used including mutual information (MI) [13–15], total variation distance [18], maximal leakage [19, 20], and guessing leakage [21]. DP assumes a worst-case adversary and requires large amount of noise injection to the data to preserve privacy. Since the applied noise is limited to have a certain form, e.g., Gaussian, rather than an arbitrary distribution optimized by an objective function, DP faces loss of utility as the data size increases. In addition to DP, total variation distance and maximal leakage also focus on the privacy of a single data point. On the other hand, information theoretic measures focus on preserving the privacy in an average sense against an adversary who is interested in the statistics of the sensitive information, and these measures provide guaranteed bounds on the information leakage which can also be characterized for large size or time-dependent data. Unlike the measures preserving single data privacy, MI privacy allows arbitrary noise distributions on the sensitive data, which provides a better privacy-utility trade-off (PUT). Moreover, information theoretic guarantees enable privacy by design, which describes data protection at the design phase of any system, since information theoretic measures do not require making assumptions on a potential adversary's capabilities. However, there is still a need for advanced methods to cover the shortcomings of different methods.

Most privacy preserving techniques require solving difficult optimization problems for the best PUT. However, it is usually intractable to solve these problems using traditional statistical methods. With the emerging technology and computational power that enables artificial intelligence (AI)-powered tools, such as machine learning (ML) or deep neural networks (DNN), data privacy and private data sharing are carried to a more practical level. For instance, large organizations such as the U.S. Census Bureau, Google, Facebook, Uber, Amazon, and Microsoft leverage AI-powered DP techniques to protect their user's sensitive data against potential privacy attacks [22]. A study published in 2019 by Gartner predicts that 40% of the privacy compliance technology will utilize AI in edge and IoT environments by 2023 [23]. Although there is an increasing demand for ML-based privacy protection, there is a lack of ML-based information theoretic privacy (ITP) works in the literature.

Besides privacy-preserving applications, DNNs are commonly used in a wide-variety of security-critical applications such as self-driving cars, spam detection, malware detection and medical diagnosis [24]. Apart from all their benefits, robustness and trustworthiness of neural network models are critical for these applications. In addition to the context of a passive adversary in privacy applications, i.e., the SP which tries to infer sensitive information about the user, there are also active adversaries which try to evade detection in DNN-based security-critical applications. DNNs have been shown to be vulnerable and can be deliberately fooled, evaded or misled by adversarial examples (AEs), which are perturbed inputs designed by real-world adversaries [25–27]. To mitigate this problem, a line of research has focused on adversarial robustness of DNNs as well as the certification of these methods [24, 28–33]. While any ML model can be vulnerable to attacks, e.g., RL agents [34, 35], most defenses in the literature focus on the evasion of classifiers. Moreover, defense mechanisms in the literature mainly consider computer vision (CV) domain applications; however, other domains, such as malware, finance, and social networks, show different characteristics, and the robustness techniques proposed for CV are not effective in these domains. Therefore, there is a need for exploration of adversarial robustness techniques out of the CV domain.

1.2 Objectives

This dissertation analyzes private data sharing techniques that protect a user's privacy in the presence of a third party, which tries to infer the user's sensitive information from the released data. This untrusted third party might be an honest-but-curious legitimate receiver of the released data, e.g., the SP/UP. The goal of the data sharing mechanism is to protect the privacy of the sensitive information by reporting a modified version of the user's data to the SP, while preserving the utility received from the service as much as possible. We focus on a relatively under-explored area by utilizing information theoretic metrics for privacy and/or utility, compared to widely explored DP. The advantages that information theoretic PUT offer include guaranteed information theoretic bounds under statistical assumptions on the data, capability of hiding underlying sensitive information, capability of revealing different levels of sensitive information to different users and enabling the usage of prior information and time dependency. Besides the information content of the user data, physical characteristics of the communication channels between the user and third parties are also exploited.

We utilize various tools for numerically solving these PUT problems, specifically dynamic programming (DyP), deep reinforcement learning (DRL) and generative networks. DyP and DRL enable tractable solutions for online private data sharing due to their sequential nature. Similarly, the end-to-end structure of generative networks, e.g., autoencoders, is an effective representation of communication systems, and enables learning encoding and decoding simultaneously. Although DNNs are highly effective tools in solving optimization problems, they have vulnerabilities against adversarial manipulations. For instance, in decision making, adversarially perturbed input samples can cause evasion of the DNN model. This vulnerability is highly risky for security-critical applications, such as malware, fraud or bot detection. This dissertation analyzes the robustness of neural networks against such adversarial perturbations in security applications, and provides empirical defenses and their provable certification.

1.2.1 Contributions

Firstly, we establish theoretical guarantees on the privacy and utility level achieved by our proposed data sharing mechanisms, regardless of the computational capability of the attacker, by using information-theoretic tools. We specifically consider MI privacy and its SM, location and activity monitoring privacy applications. Secondly, we reformulate the time-series data sharing problem as a Markov decision process (MDP) to take the temporal correlations into account, and solve it numerically by powerful optimization tools, such as DyP for SM and DRL for location and activity monitoring privacy applications. Considering temporal correlations is of significant importance, as current privacy-preserving techniques often ignore the prior information and time dependencies due to computational complexity, whereas integration of DNNs reinforces optimal MDP solutions. Furthermore, we provide an understanding of privacy-aware communications between the user and the SP, which is the legitimate receiver, in the presence of imperfect communication channels. Exploiting the physical characteristics of the SP's channel over an eavesdropper's, we allow communication with privacy guarantees. Deep learning in wireless communications and physical layer security has only recently become popular, and hence there is a need in the literature for exploration of real-world limitations in privacy-aware communications. Despite all its benefits, finally, we also investigate the vulnerabilities of DNNs in security critical applications. In addition to passive adversaries that we consider, we also provide defenses against active adversaries which target these vulnerabilities. We provide empirical and provable guarantees for robustness against malicious adversaries in various domains.

1.3 Outline and Related Publications

In this dissertation, we first present preliminary materials for privacy measures, information theoretic metrics and MDPs in Chapter 2. Then, an overview of the SM privacy problem and privacy enabling techniques using physical resources are presented in Chapter 3, followed by time-series data obfuscation and data release mechanism selection techniques for PUT using DRL in Chapters 4 and 5, respectively. In Chapter 6, private data sharing is investigated over a wiretap channel scenario using generative networks, while Chapter 7 is dedicated to robustness of neural networks in security critical applications. In the following sections, we briefly present the content, results and the corresponding publications of each chapter.

Chapter 3

In Chapter 3, we present an overview of SM privacy-preserving techniques. While the SM data is modified before being reported to the UP in *data manipulation*, *demand shaping* requires direct manipulation of the real energy consumption by exploiting physical resources, such as a renewable energy source (RES) or a rechargeable battery (RB). Privacy-preserving techniques that we present in this chapter contain a data manipulation and three different demand shaping techniques that consider SM with a RES and an RB, SM with only an RB and SM with only a RES. Information theoretic measures are used to quantify SM privacy. Optimal energy management strategies and bounds which are obtained using control theory, specifically MDPs, and rate distortion theory are analyzed. The content of this chapter has been published as a book chapter and a conference paper in:

- Ecenaz Erdemir, Deniz Gündüz, and Pier Luigi Dragotti, "Smart Meter Privacy," *in Privacy in Dynamical Systems*, F. Farokhi (editor), Ed. Singapore: Springer, 2020, pp. 19-41,
- Ecenaz Erdemir, Pier Luigi Dragotti, and Deniz Gündüz, "Privacy-cost trade-off in a smart meter system with a renewable energy source and a rechargeable battery," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Brighton, UK, May 2019.

Chapter 4

In Chapter 4, we consider a user that measures time-series data generated by an IoT device, e.g., GPS readings, and periodically reports a modified version of her true data to an untrusted SP to gain utility. Unlike the demand shaping techniques in Chapter 3, here the measurements are obfuscated with noise up to a certain level for PUT before sharing with the SP. We use the mutual information between the true and distorted data sequences as a measure of privacy loss, and measure the utility by a distortion metric between the true and distorted samples. For the PUT, we introduce an online private data release policy (PDRP) that minimizes the mutual information while keeping the distortion below a certain threshold. We consider data release policies which take the entire release history into account, and show its information theoretic optimality. We recast the information theoretic time-series data PUT problem as an MDP and evaluate the optimal PDRP numerically using advantage actor-critic deep reinforcement learning (A2C-DRL). We apply our PDRP on the location trace privacy scenario, and evaluate its performance using both synthetic and real trajectory datasets. The content of this chapter has been published as a conference paper and a journal paper in:

- Ecenaz Erdemir, Pier Luigi Dragotti, and Deniz Gündüz, "Privacy-aware location sharing with deep reinforcement learning," *IEEE Workshop on Information Forensics and Security (WIFS)*, Delft, Netherlands, Dec. 2019,
- Ecenaz Erdemir, Pier Luigi Dragotti, and Deniz Gündüz, "Privacy-aware time-series data sharing with deep reinforcement learning," *in IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 389-401, 2021.

Chapter 5

In Chapter 5, we consider an active learning scenario for PUT against an honest-but-curious SP. Unlike in the previous scenarios, in this setting, the IoT measurements contain two correlated underlying information, namely the *useful variable*, to be disclosed for utility, and the *secret variable*, to be kept private. We assume that a user wants to share these measurements with the SP by periodically choosing a different data release mechanism with different statistics at each time instance, and stop data release before the SP is confident about the true value of the secret. The user's goal is to determine the best selection mechanism to prevent the secret from being accurately detected by the SP while revealing the useful data accurately for utility. While the first scenario presented in this chapter focuses only on the PUT, the latter one takes the time aspect into account and targets the quickest detection. Both active learning problems are reformulated as an MDP, and numerically solved by utilizing DRL for both for synthetic and real data in human activity privacy scenario. The results in this chapter have been published as a conference paper which has also received the third place in the *ICICS-CAIDA Best Poster Prize* competition in the 2021 North American School of Information Theory (NASIT), and also submitted for a journal publication in:

- Ecenaz Erdemir, Pier Luigi Dragotti, and Deniz Gündüz, "Active privacy-utility trade-off against a hypothesis testing adversary," *IEEE International Conference on Acoustics,Speech and Signal Processing (ICASSP)*, June 2021,

- Ecenaz Erdemir, Pier Luigi Dragotti, and Deniz Gündüz, "Active privacy-utility trade-off against inference in time-series data sharing," *submitted.*

Chapter 6

In Chapter 6, we take into account the physical characteristics of the communication channel between the user and the SP for the first time. Similarly to the previous scenarios, the user wants to reliably transfer her data which contains a latent sensitive information, i.e., the secret, to the SP. However, in this setting, the communication is performed over noisy channels and a passive eavesdropper wants to infer the secret over his channel. For example, the user data may be an image or a video while the secret may be the presence of a particular object or an activity within the scene. In this wiretap channel scenario, we assume binary symmetric channels (BSCs) from the user to both the SP and the eavesdropper. We optimize the trade-off between the reconstruction distortion of the data by the SP and the privacy leakage of the secret to the eavesdropper, which is measured by the MI between the secret and the noisy user data observed by the eavesdropper. Moreover, we propose a data-driven approach using variational autoencoder (VAE)-based joint source channel coding (JSCC), and show through simulations with the colored MNIST dataset that our approach provides high reconstruction quality at the receiver while confusing the eavesdropper about the secret, which consists of the color and thickness of the digits. Finally, we consider a parallel-channel scenario, and show that our approach arranges the information transmission such that the channels with higher noise levels at the eavesdropper carry the sensitive information, while the non-sensitive information is transmitted over more vulnerable channels. The results of this chapter have been submitted for publication in:

- Ecenaz Erdemir, Pier Luigi Dragotti, and Deniz Gündüz, "Privacy-aware communication over a wiretap channel with generative networks", against a hypothesis testing adversary," *IEEE International Conference on Acoustics,Speech and Signal Processing (ICASSP)*, May 2022.

Chapter 7

Complementary to passive adversaries that we have mention in the previous chapter, in Chapter 7, we consider the trustworthiness of DNNs in the presence of active adversaries in security critical applications. Neural network robustness against potential adversaries is significant, since DNN models have been shown to be vulnerable against small modifications in the samples and can be fooled. Adversarial robustness has widely been studied in the literature to mitigate these weaknesses of DNNs both empirically and provably. Prior work, which mostly contains CV domain applications, mainly focus on crafting AEs with small uniform norm-bounded perturbations across features to maintain the requirement of imperceptibility [24, 28–33]. However, uniform perturbations do not result in realistic AEs in domains such as malware, finance, and social networks. For these types of applications, features typically have some semantically meaningful dependencies. The key idea of the proposed approach in this chapter is to enable non-uniform perturbations that can adequately represent these feature dependencies during adversarial training. We propose using characteristics of the empirical data distribution, both on correlations between the features and the importance of the features themselves. Using experimental datasets for malware classification, credit risk prediction, and spam detection, we show that our approach is more robust to real-world attacks. Finally, we present robustness certification utilizing non-uniform perturbation bounds, and show that non-uniform bounds achieve better certification. This chapter contains work done during the remote internship with Amazon Web Services, New York City, US, and published in:

- Ecenaz Erdemir and Jeffrey Bickford and Luca Melis and Sergül Aydöre, "Adversarial Robustness with Non-uniform Perturbations", *Thirty-Fifth Conference on Neural Information Processing Systems (NeurIPS)*, Dec. 2021

Chapter 8

Finally, in Chapter 8, we conclude our research on privacy and security in cyber-physical systems, and discuss potential future directions, as well as open questions and challenges that need to be addressed.

Chapter 2

Preliminaries

In this chapter, fundamental measures and methods which are used throughout the dissertation are introduced, and a brief literature review about each topic is provided. We first introduce the most commonly used privacy measures and mention seminal works that utilize these measures. We give a detailed background specifically for ITP, since it is the main focus of the proposed approaches in Chapters 3, 4, 5 and 6. Moreover, we explain MDPs which have extensively been used throughout Chapters 4 and 5, as we reformulate our time-series data release problems as MDPs. A2C-DRL algorithm will be introduced as a tool for solving MDPs numerically, and used to find approximations for optimal policies of private data release in the following chapters. Finally, we give a brief introduction to adversarial attacks and neural network robustness to provide a background for Chapter 7.

2.1 Privacy Measures

Data privacy has been widely studied in the literature [10, 12–17, 36–49], and numerous privacy measures have been introduced, including differential privacy [10, 12], k-anonymity [38, 40], mutual information (MI) [13–15], total variation distance [18], maximal leakage [19, 20], and guessing leakage [21]. Previous work has mostly focused on protecting the privacy of a single data point, e.g., an individual's identity among multiple user's, or the current measurement in a sequence [40–42, 45, 46], whereas only few works have investigated sequential data privacy, such as electrocardiogram (ECG), body temperature, physical activity, location, weather forecast, account balance and SM readings [50, 51].

2.1.1 Differential Privacy (DP)

DP, which was first introduced for querying databases, has emerged as a widely adopted privacy measure [10]. DP provides guarantee that the changes in one record of the input database do not significantly affect the query output changes in the database. Since this guarantee is required for all adjacent inputs uniformly, DP requires high level of noise and it is considered as a worst-case measure. The formal definition of DP is as follows:

Definition 2.1. [10] For a positive real number ϵ and a randomized algorithm $\mathcal{A}$, the algorithm $\mathcal{A}$ is said to provide ϵ-differential privacy if, for all datasets D_1 and D_2 that differ on a single element, and all subsets S of range $\mathcal{A}$,

$$Pr(\mathcal{A}(D_1) \in S) \leq e^{\epsilon} \cdot Pr(\mathcal{A}(D_2) \in S) \tag{2.1}$$

where the probability is over the random algorithm.

In a scenario where a data sequence or multi-dimensional data of a single user is to be kept private instead of the identity of an individual among multiple users, DP suffers from high utility loss due to the noise injection for every data point. This is because DP is meant to ensure the privacy of a single data point in time. In [52], it is stated that DP and k-anonymity [38, 40], which also guarantees a sensitive data to be indistinguishable from at least $k-1$ other data points, are not appropriate measures for sequential data privacy since temporal correlations are not taken into account.

2.1.2 Pufferfish Privacy

As an intermediate framework between DP, which assumes complete independence, and group-DP which assumes complete correlation, *pufferfish privacy* considers low average temporal correlations in time-series data [53]. In [53], continuous aggregate location sharing is considered in a pufferfish privacy framework under temporal correlations modeled as a Markov chain. This approach takes into account a certain number of steps forward and backward, while minimizing the DP loss of the current location. Hence, the accumulating privacy loss of DP mechanism is limited to a level determined by the number of forward and backward steps.

2.1.3 Information-Theoretic Privacy (ITP)

ITP usually refers to MI privacy since MI is a measure of information flow which suits well for quantifying privacy.

In information theory, entropy is a measure of the uncertainty of a random variable (r.v.). Let X be a discrete r.v. with probability mass function $p(x) = Pr\{X = x\}$ over alphabet $\mathcal{X}$. The entropy of X is denoted by

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x). \tag{2.2}$$

The MI between two r.v.'s X and Y is the relative entropy between the joint probability mass function, $p(x, y)$, and the product of their marginal probability mass functions, $p(x)p(y)$, and is given by

$$I(X;Y) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \tag{2.3}$$

$$= E_{p(x,y)} \log \frac{p(X, Y)}{p(X)p(Y)}. \tag{2.4}$$

MI can also be written as the reduction in the uncertainty of X due to the knowledge of Y, i.e., $I(X;Y) = H(X) - H(X|Y)$, where $H(X|Y)$ is the conditional entropy.

ITP can often be related with *information theoretic secrecy* which dates back to 1949 when *perfect secrecy* was first introduced by Shannon [54]. Both can be considered under an umbrella term *information security* which emerged from communication applications. Certain levels of information leakage exist in all data sharing and communications applications, which poses a privacy risk through unwanted inferences. Quantifying this leakage using information theoretic measures is the first step towards ITP.

While secrecy focuses on negligible or zero information leakage, privacy relaxes this condition in return of the PUT. ITP offers guaranteed information theoretic bounds with statistical assumptions on the data, capability of hiding underlying sensitive information, capability of revealing different levels of sensitive information to different users and enabling the usage of prior information and time dependency. On the other hand, while DP tries to hide the true value of a sensitive information which targets the worst-case adversaries, ITP covers a wide range of privacy measures that can hide the information in an average sense, e.g., MI [55], f-divergences [56], average total variation distance [18], and maximal leakage [19, 20].

It is proved in [54] that there exists an encoding scheme such that an adversary having full access to the communication channel between a transmitter and a receiver has no information about the transmitted message. The perfect secrecy system is impractical since it requires a key of the same size as the message to fully hide the message from the adversary while the receiver can correctly decode the message using the key. [57], and later [58], proposed *the wiretap channel* which eliminated the need for a key by exploiting the uncertainties of the physical medium such as channel noise and fading fluctuations. However, practicality of [57] comes at the cost of zero information leakage, i.e., Wyner's *weak secrecy* tolerates a small amount of information leakage rate and achieves perfect secrecy asymptotically.

One of the biggest advantage of ITP over DP and pufferfish privacy is that it enables the usage of prior information and temporal correlations, which allows arbitrary stochastic transformations of data samples rather than being limited to addition of noise of a specific form. This due to the fact that the MI between two time-series data sequences can be written, by the chain rule, as

$$I(X^n; Y^n) = \sum_{t=1}^{n} I(X^n; Y_t | Y^{t-1}), \tag{2.5}$$

where $X^n = \{X_1, X_2 \ldots, X_n\}$ and $Y^n = \{Y_1, Y_2 \ldots, Y_n\}$ are two sequences of r.v.'s. As a result of introducing the memory in (2.5), the analysis becomes computationally complex as the horizon increases as the MI involves an increasing number of r.v.'s. In the literature, single-letter expressions for the information leakage in time-series data privacy problems have been obtained considering independent and identically distributed (i.i.d.) or Markov relation between the r.v.'s. Single letter expressions guarantee that, no matter how long the problem horizon is, the minimal leakage can be written as a function of the joint distribution of the involved r.v.'s single realization.

2.1.3.1 Applications of ITP

Techniques that would allow controllable amount of information leakage have attracted a growing interest over the past decades. One of the earliest works using source coding for ITP of a sensitive variable that is correlated with the source data is studied in [59], in which a PUT is proposed by associating it with Shannon's rate-distortion theory. Given the publicly revealed encoding of the source, equivocation rate of the sensitive variable is used as a privacy measure. Similarly, in [55], PUT is proposed as a rate-distortion optimization problem, in which privacy leakage is measured by the MI between the source and the legitimate receiver's reconstruction.

An early work on MI privacy proposed in [60] provides foundations for measurement of the effectiveness of privacy-preserving data mining algorithms. Being the first MI privacy paper for privacy-preserving data collection and data mining, [60] proposes perturbing the data and reconstructing the distributions at an aggregate level. The performed expectation maximization algorithm is proved to converge to the maximum likelihood estimate of the original distribution based on the perturbed data. Privacy is measured by the MI between the original and the perturbed records.

As an example of time-series data privacy, in [50], an SM system is considered assuming Markovian energy demands. Privacy is measured by the MI between the demand-side measurements and the SM readings, and achieved by filtering the energy demand with the help of a rechargeable battery. ITP problem is formulated as an MDP, and the minimum leakage is obtained numerically through DyP, while a single-letter expression is obtained for an i.i.d. demand. This approach is extended to the scenario with a renewable energy source in [36]. In [61], PUT is examined with a rechargeable battery. Due to Markovian demand and price processes, the problem is formulated as a partially observable MDP with belief-dependent rewards (ρ-POMDP), and solved by DyP for infinite-horizon.

In [62], PUT of time-series data is considered in both online and offline settings. A user continuously releases data samples which are correlated with its private information, and in return obtains utility from an SP. The proposed schemes are cast as convex optimization problems and solved under hidden Markov model assumption. The simulation results are provided for binary time-series data for a finite time horizon. However, the dimensions of the optimization problems in both schemes grow exponentially with time and the number of sample states. Therefore, in a setting when fine-grained sensor data is considered for a long time horizon, computational complexity of the proposed schemes is very high.

2.1.4 Error Probability

Privacy metrics based on the SP's error probability focus on concealing the true realization of the sensitive information. In [16], the goal is to increase the fidelity of the shared data quantified through an additive distortion measure, while guaranteeing privacy in an online manner. Privacy leakage is measured by the error probability of the SP in estimating the true value of the underlying variables.

In [63], a r.v. containing latent sensitive variable $S \in \mathcal{S}$ and non-sensitive variable $U \in \mathcal{U}$ is considered to go through a sensor, and a third party can infer these variables from the noisy sensor measurement Z. The objective is to design an estimator for the non-sensitive r.v. which minimizes a loss function while the information leakage about the sensitive variable is kept below a certain level. The conditional discrete entropy is used as the privacy metric, since the error probability of estimating the sensitive r.v. after observing the noisy measurements can be lower bounded in terms of this privacy metric using Fano's inequality [64], i.e.,

$$P_{err}(S) = P(S \neq \hat{S}) \geq \frac{H\big[S|Z\big] - 1}{\log|\mathcal{S}|}. \tag{2.6}$$

2.2 MDP Formulation

Throughout the dissertation, we consider privacy-preserving scenarios in which the data of interest is time-series measurements. It is usually intractable to solve these optimization problems while taking the data release history and temporal correlations into account. Therefore, we reformulate them as sequential decision making problems by exploiting the Markov property of the time-series data, and solve by using classical MDP solution methods. Next, we will briefly define MDPs and provide a specific RL solution.

Consider a sequential decision making problem under uncertainty. At each time instance, an external decision maker (agent, controller, etc.) observes the state of the system and takes an action accordingly. As a result of the action taken at a particular state, a reward is received by the decision maker. The goal of such a problem is to find the decision rules that specify the best actions to take at each system state, such that the maximum total reward is accrued by the decision maker under the system constraints [65].

Markov property is based on the idea that the future is independent of the past given the present. Since considering the effect of entire time horizon in a decision problem is computationally complex, these problems are modeled for Markovian state space. Hence, MDPs are discrete time stochastic control processes which are used to model sequential decision problems with uncertainty. MDPs take into account both the short-term outcomes of current decisions and the possible future gain. An MDP is formally defined as a 4-tuple $<\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}>$, which represent the *state space* $\mathcal{S}$, *action space* $\mathcal{A}$, *transition probabilities* reflecting the system dynamics $\mathcal{T}$, and *reward* (or, inversely, *cost*) $\mathcal{R}$ of taking a certain action at a certain state [66]. The state is Markov if

$$P(S_{t+1}|S_t) = P(S_{t+1}|S_t, S_{t-1}, \dots, S_1), \tag{2.7}$$

where $S_t \in \mathcal{S}$. For deterministic policies, transition probabilities are the mappings from each state-action pair to the next state,

$$\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}, \tag{2.8}$$

whereas for stochastic policies, each state-action pair is mapped to a probability distribution over the next states,

$$\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow P(\mathcal{S}). \tag{2.9}$$

The goal of an MDP is to obtain a set of decision rules, so called *policy*, that performs optimally with respect to a certain performance criterion. A policy $\boldsymbol{q}$ is a function specifying the action that the decision maker takes in a particular state, i.e., $\boldsymbol{q} : \mathcal{S} \rightarrow \mathcal{A}$. The objective functions of MDPs map infinite or finite sequences of rewards (or costs) to a single real number. MDPs can have objectives, such as *discounted*, *expected-total* or *average* costs (rewards) to minimize (maximize) over a specified duration, i.e., *finite-horizon* setting, or an indefinite time, i.e., *infinite-horizon* setting [65]. To solve MDPs optimally, Bellman optimality equations are used. Value function of the decision problem in state s is denoted by $V_{\boldsymbol{q}}(s)$ which represents the expected reward/cost obtained following the policy $\boldsymbol{q}$ in state s. Bellman optimality equation for a Markov reward process is denoted by

$$V_{\boldsymbol{q}}(s) = \max_{\boldsymbol{q}(s) \in \mathcal{A}} \Big\{ r(s, \boldsymbol{q}(s)) + \sum_{s' \in \mathcal{S}} T(s, \boldsymbol{q}(s), s') V_{\boldsymbol{q}}(s') \Big\}, \tag{2.10}$$

where the maximization is over all the possible actions induced by the policy $\boldsymbol{q}$ for each state s. The optimal value can be achieved by maximizing/minimizing the right hand side of (2.10) using dynamic programming, which is an optimization method used to avoid redundant calculations in recursive problems with an additive objective function [67].

A POMDP is a generalization of an MDP when the decision maker does not have complete information about the system state. Instead, she can maintain a belief which is a conditional probability distribution over the possible states given the past observations from the environment. POMDPs can be modeled as belief MDPs by inducing a continuous belief state. In the literature, there are various approaches to solve belief MDPs using finite-state MDP solution methods, e.g. value iteration, policy iteration and gradient-based methods. These are based on the discretization of continuous belief states to obtain a finite state MDP [68].

2.2.1 Advantage Actor-Critic DRL

DRL is a combination of DNNs and RL training methods based on rewarding desired actions and/or punishing unwanted ones. It is a very broad topic that has received significant attention in recent years [69]. In this section, we will briefly mention RL and introduce a specific algorithm A2C-DRL.

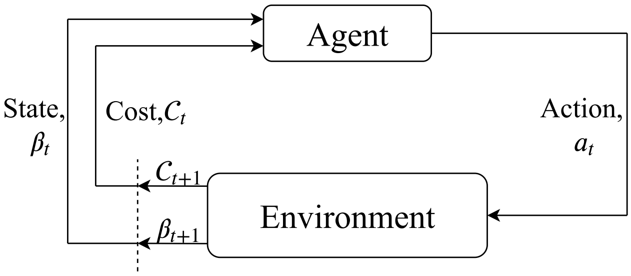


FIGURE 2.1: RL for a known model.

In RL, an agent discovers the best action to take in a particular state by receiving instantaneous rewards/costs from the environment [69]. RL methods can be divided into three groups: value-based, policy-based, and actor-critic [70]. Actor-critic methods combine the advantages of value-based (critic-only) and policy-based (actor-only) methods, such as low variance and continuous action producing capability. The actor represents the policy structure, while the critic estimates the value function [69]. In our settings that will be presented in the following chapters, we parameterize the value function by the parameter vector $\theta \in \Theta$ as $V_\theta(\beta)$, and the stochastic policy by $\xi \in \Xi$ as $q_{\boldsymbol{\xi}}$. The difference between the right and the left hand side of (2.10) is called temporal difference (TD) error, which represents the error between the critic's estimate and the target differing by one-step in time [71]. The TD error for the experience tuple $(\beta_t, a_t, y_t, \beta_{t+1}, \mathcal{C}_t)$ is estimated as

$$\delta_t = \mathcal{C}_t(\beta_t, a_t) + \gamma V_{\theta_t}(\beta_{t+1}) - V_{\theta_t}(\beta_t), \tag{2.11}$$

where $\mathcal{C}_t(\beta_t, a_t) + \gamma V_{\theta_t}(\beta_{t+1})$ is called the TD target, and γ is a discount factor that we choose very close to 1 to approximate the Bellman equation in (2.10) for our infinite-horizon average cost MDP. To implement RL in the infinite-horizon problem, we take sample averages over independent and finite data sequences, which are generated by experience tuples at each time t via Monte-Carlo roll-outs.

Instead of using value functions in actor and critic updates, we use advantage function to reduce the variance in policy gradient methods. The advantage can be approximated by TD error. Hence, the critic is updated by gradient descent as:

$$\theta_{t+1} = \theta_t + \eta_t^c \nabla_\theta \ell_c(\theta_t), \tag{2.12}$$

where $\ell_c(\theta_t) = \delta_t^2$ is the critic loss and η_t^c is the learning rate of the critic at time t. The actor is updated similarly as,

$$\xi_{t+1} = \xi_t - \eta_t^a \nabla_\xi \ell_a(\xi_t), \tag{2.13}$$

where $\ell_a(\xi_t) = \ln(q_s(y_t|\beta_t, \xi_t))\delta_t$ is the actor loss and η_t^a is the actor's learning rate. This method is called *advantage actor-critic RL*.

2.3 Adversarial Robustness

There is a large variety of adversarial attacks that target ML systems, such as evasion, poisoning and exploratory attacks [72]. The most common attack type in the literature is the evasion attack, which adjust malicious samples in testing time for evading classification. On the other hand, poisoning attacks contaminate a percentage of the training data with carefully crafted malicious samples to either reduce the classification accuracy or create a backdoor to exploit during test time. Unlike the previous two methods, exploratory attacks try to gain as much knowledge about the learning algorithm as possible instead of modifying the training or testing data. In this section, we will focus specifically on evasion attacks and adversarial examples (AEs) crafted by them.

AEs are intentionally crafted samples by attackers that aim to cause ML models to make mistakes. Although any ML model can be fooled, e.g., RL agents [34, 35], most adversarial attacks in the literature focus on the evasion of classifiers. The adversary's objective is to maximize the error or loss function of the classifier by adding perturbations to the samples to cause misclassification. Given a dataset $\{x_i, y_i\}_{i=1}^n$ with input $x_i \in \mathbb{R}^d$ and classes $y_i \in \mathcal{Y}$, we can formalize the AE generation as a solution to the following optimization:

$$x_i^* = x_i + \arg\min_{\delta \in \Delta}\{\|\delta\|_p : f_\theta(x_i + \delta) \neq y_i\}, \tag{2.14}$$

where x_i^* is the AE, $f_\theta : \mathbb{R}^d \to \mathcal{Y}$ is DNN function, and Δ is the set of possible adversarial perturbations around the original samples. Various solutions to (2.14) in the literature are called adversarial attacks. Some of the most common state-of-the-art adversarial attacks, such as fast gradient sign method (FGSM) [73] and projected gradient descent (PGD) [24], perturb training samples under a norm-ball constraint to maximize the loss of the network.

2.3.1 Fast Gradient Sign Method (FGSM)

A solution to the optimization (2.14) is proposed in [73], and AEs are crafted as follows:

$$x_i^* = x_i + \epsilon * sign(\nabla_{x_i}\ell(f_\theta(x_i), y_i)) \tag{2.15}$$

where $\ell(.)$ is the loss, e.g. cross-entropy, ∇_{x_i} is the gradient of the model with respect to x_i, and ϵ is the parameter which determine the size of the perturbation. That is, (2.15) crafts AEs that are within an ℓ_∞ norm-ball of radius ϵ around the original sample and maximize $\ell(\cdot)$. Other variations of FGSM appear in the literature as targeted-FGSM and Basic Iterative Method (BIM), where the former maximizes the probability of a specific target class in AE generation and the latter is a straightforward extension of FGSM to iteratively finding the optimal AEs [74].

2.3.2 Projected Gradient Descent (PGD)

In Chapter 7, we propose defenses which take PGD [24] as the base AE generation method and build on top of it. PGD is a state-of-the-art perturbation method for AE generation. It is a well-studied extension of FGSM to ℓ_p norm and iterative optimization, and can be formalized as

$$x_i^* = \mathcal{P}(x_i + \alpha * \nabla_{x_i} \ell(f_\theta(x_i), y_i)) \tag{2.16}$$

where the operation is applied at each time-step with step size α, $\mathcal{P}$ is the projection function that applies the norm-ball constraint $\|\delta\|_p \leq \epsilon$. A more detailed explanation for PGD is provided in Chapter 7.

Adversarial training (AT) is one of the most effective empirical defenses against these adversarial attacks [24,73,74]. The goal of the AT is to minimize the loss of the DNN when perturbed samples are used during training. This way, the model becomes robust to real-world adversarial attacks. Though these empirical defenses do not provide theoretically provable guarantees, they have been shown to be robust against the strongest known attacks [73]. AT can formally be represented as a *min-max optimization* minimizing the DNN loss which is maximized by adversarial perturbations δ. Given $\{x_i, y_i\}_{i=1}^n$ as before, the objective of AT is denoted by

$$\min_\theta \frac{1}{n} \sum_{i=1}^{n} \max_{\delta \in \Delta} \ell(f_\theta(x_i + \delta), y_i). \tag{2.17}$$

The adversary's objective is the inner maximization term in (2.17), and the perturbed samples found as a solution to the norm-constrained inner maximization are the AEs, which is exemplified in Sections 2.3.1 and 2.3.2.

Empirical defenses are effective against many real-world attacks, however, their robustness is not certifiable. The goal of certification, on the other hand, is to report whether an AE exists within an ℓ_p norm centered at a given sample with a fixed radius. Certified defense approaches introduce theoretical robustness guarantees against norm-bounded perturbations [31, 32, 75, 76].

Chapter 3

Smart Meter Privacy

3.1 Introduction

An electrical grid is a network that distributes electricity to consumers. The foundations of the current electrical grid were laid out in the late 19^{th} century as a centralized unidirectional transmission and distribution system. However, the current grid has reached its capacity and is not fit to manage the growing energy demand [77].

Developing technology has led to an increasing number of electronic appliances, electrical vehicles and integration of renewable energy sources. In order to handle load imbalance, inefficient usage of energy, and blackouts with domino effect a new energy grid is currently being introduced. Smart grid (SG) is an energy grid which controls energy generation, distribution, transmission and consumption using advanced communication and sensing technologies. SGs are developed to increase the efficiency of energy infrastructure, reliability against attacks, flexibility with bidirectional energy flows and the load balancing against variations [78]. For instance, thanks to SG's ability to support customer energy generation, farms that produce electricity using methane generators, consumers with solar panels or wind turbines can sell excess generated energy back to the UP.

One of the main enablers of SGs are the SMs, computerized replacement of the traditional analog electrical meters attached to the exterior of households [79]. Unlike traditional electrical meters, which measure only the total consumption, SMs can monitor fine grained electricity usage of a household and report it to the UP. This provides efficient use of energy resources since the SM owners can track and control their consumption almost real-time. SM data can also be used for time-of-usage pricing, which can reduce peak electricity demands by controlling customer behavior. Moreover, SMs also facilitate detecting energy theft, trading user-generated energy to increase grid efficiency, and mitigating effects of load variations [78].

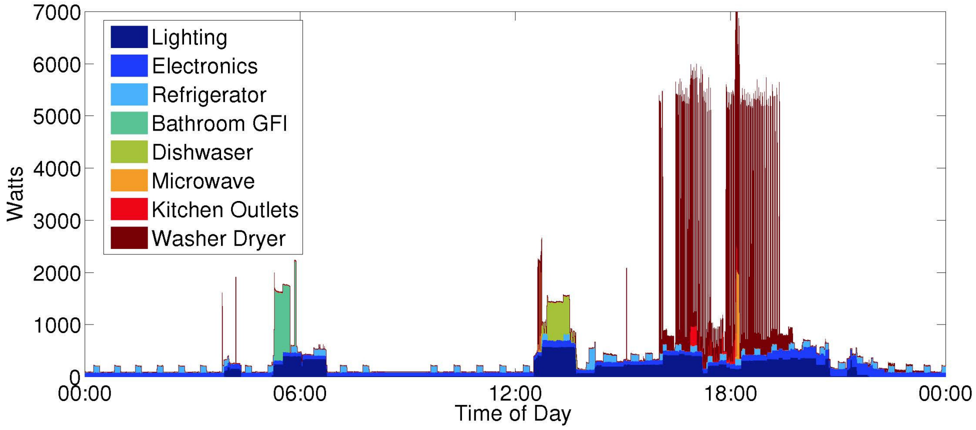


FIGURE 3.1: Electricity consumption profile of a household for 24 hour period [1].

3.1.1 Privacy and Security Concerns

SM measurements contain detailed information related to the real-time state of the customers. The UP or a third party can deduce power signatures of specific home appliances by using non-intrusive load monitoring (NILM) techniques [80]. NILM systems identify appliances by using a series of changes in their power draw. For instance, appliances such as kitchen ovens, tumble dryers and dishwashers go through a number of states, where heaters and fans are turned on and off in various combinations. Such appliances are modelled as finite state machines. On the other hand, when on, a light bulb draws power continuously.

In Figure 3.1, an example of the 24 hour period of SM measurements for a household is illustrated. Specific appliances with distinguishable power signatures are highlighted with different colors. As in Figure 3.1, the high resolution consumption data reveals details about private activities of the user. This real-time data might enable a malicious eavesdropper to learn user's presence at home, illnesses, disabilities and even political views due to the TV channel the user is watching [81]. SM privacy becomes even more critical when we consider businesses, since their power consumption might reveal the state of their business to competitors. The controversy about SM roll-out plans due to privacy concerns have attracted public and political attention across the world. In 2009, a court in Netherlands decided that mandatory installation of SMs would be a violation to the customer's right to privacy, and would be in breach of the European Convention of Human Rights [82]. In 2018, in the case of Naperville Smart Meter Awareness v. City of Naperville, a court in the United States has agreed that the Fourth Amendment protects user's energy consumption data collected by SMs. That is, user's expectation for SMs data privacy is reasonable and the government's access to this private information

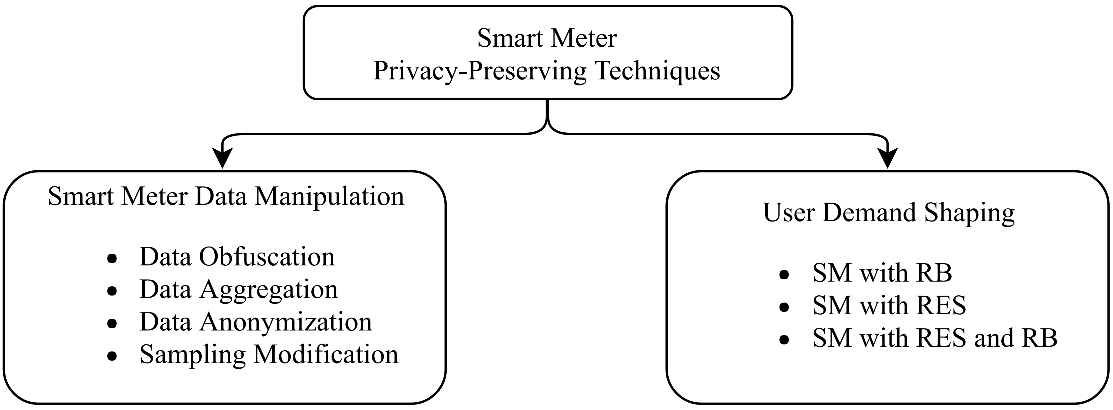


FIGURE 3.2: SM privacy enabling techniques.

constitutes a search [83]. SM privacy concerns can be a major roadblock on the path of achieving worldwide SM usage.

3.2 SM Privacy Techniques

Various SM privacy enabling techniques have been proposed in the literature [51, 84–94], which can be categorized as into two groups (see Figure 3.2): those based on SM data manipulation and those based on demand shaping. While the techniques in the former group focus on modifying SM measurements [84, 85], there in the latter group directly manipulate user's energy consumption exploiting physical resources, such as a RB [86–90] or a RES [51, 91–94]. Representative works for each group are briefly explained in the following sections.

3.2.1 Data Manipulation

Data manipulation techniques modify SM measurements before sending them to the UP. There are many different approaches to SM data manipulation in the literature, such as data obfuscation, data aggregation, data anonymization and down-sampling.

SM data obfuscation can be performed by corrupting the SM measurements with additive noise. A cooperative state estimation technique is proposed in [85] to preserve privacy by obfuscating the power consumption measurement. As the amount of noise added increases, information leaked to the UP decreases. However, such a modification makes SM measurements less relevant to the UP for prediction and control purposes, which contradicts the purpose of installing SMs. In [95], a general theoretical framework is proposed for both utility and privacy requirements of data release mechanisms using information theoretic tools. In this context, SM measurements are perturbed before being reported to the UP. The goal is to minimize the information leakage rate between the perturbed data and the private data of user's choice while keeping the distortion between the real and perturbed meter measurements below a certain level. For a stationary Gaussian Markov model of the electricity load, the optimal utility-privacy trade-off is characterized in [96] using the framework proposed in [95].

Data aggregation, on the other hand, proposes sending aggregated SM readings, instead of individual readings, to the UP. In [84], data aggregation is used in combination with homomorphic encryption and secret sharing techniques. The UP has access only to encrypted SM readings and the total consumption. Moreover, the users send their random shares to the UP after encrypting with each others public keys and aggregating. Hence, the UP does not have access to individual consumption information. However, encryption methods increase the computational complexity substantially [97].

Data anonymization approach [98], instead, considers utilizing pseudonyms rather than the real identities of consumers such that information gleaned from the SM cannot be easily associated with an identified person.

In [99], two data manipulation techniques are combined, namely down-sampling and noise addition. The SM data is first down-sampled by summing up n consecutive samples, then noise is added to the down-sampled data. Similarly to [85], perturbation of SM readings can cause undesired data loss.

3.2.2 Demand Shaping

Manipulating SM readings reduces the relevance of the reported values for grid management and load prediction, limiting the benefits of SMs. Moreover, the grid operator can place sensors outside a household or a business, and obtain the real consumption data, since they own and control the infrastructure. Therefore, data manipulation cannot provide strict privacy against UPs. Demand shaping tackles these issues by manipulating the real energy consumption. Unlike in data manipulation, UP receives accurate measurements of the energy taken from the grid. However, these measurements do not belong to the actual energy consumption of the household; and therefore, only provide very limited information about user behavior. Instead, the energy demand of the appliances are supplied either by alternative energy sources, such as renewable energy sources, or from rechargeable energy storage devices. Hence, the instantaneous energy demand of an appliance is supplied only partially by the power grid, if at all, and the rest can be provided by the RB or RES. This effectively filters the real energy consumption time series, and creates a new time series for the energy received from the grid, which has only limited correlation with the original time series, and consequently reduces the information leakage to the UP. Note that, in the extreme cases of unlimited RES or RB capacity, the two time series can be made completely independent, leading to zero information leakage, i.e., perfect privacy [100]. The objective of the SM privacy demand shaping problem is to determine the optimal energy management strategy between the grid, RB, RES and the appliances, under given physical limitations, such as RB capacity, RES generation rate, peak power constraints, etc. which provide the maximum privacy.

In [50], information theoretic privacy in an SM system with an RB is formulated as an MDP. Markovian energy demand is considered, and the minimum leakage is obtained numerically through DyP, while a single-letter expression is obtained for an i.i.d demand. This approach is extended to the scenario with a RES in [51], which considers both cases where the energy generation process is private and known to the UP. When the energy generation process is known by the UP, it is numerically shown in [2] that the infinite-horizon MDP performance can be achieved by a low complexity algorithm under the assumption of a special energy generation process.

Privacy-cost trade-off is examined in an SM system with an RB in [90]. Due to Markovian demand and price processes, the problem is formulated as a POMDP with belief-dependent rewards. Bellman equation for stationary strategies is provided. However, due to the non-linear and belief-dependent reward, the Bellman equation corresponds to a continuous state, continuous action, continuous reward MDP. Obtaining optimal policies using continuous action Bellman equation is computationally complex. Therefore, the authors provided upper and lower bounds, and presented numerical results using classical rate-distortion theory in [90].

Information theoretic SM privacy with RB and RES is studied in [91] with average and peak power constraints on RES. While closed-form expressions are obtained for the scenarios with zero and infinite capacity RB, low complexity energy management policies are proposed for finite capacity. For a zero-capacity RB, rate-distortion theory is used to obtain a single letter expression under the assumption of i.i.d. demand process. The SM privacy problem in the existence of an alternative energy source, e.g. RES, is also studied in [92] exploiting rate-distortion theory, and numerical results are obtained by Blahut Arimoto algorithm. This approach is extended to multiple-user scenario in [101].

3.3 Information Theoretic SM Privacy

In this section, SM privacy problem is examined from information theory perspective. SM privacy enabling techniques with data manipulation and demand shaping are presented in detail in Sections 3.3.1 and 3.3.2, respectively.

3.3.1 SM privacy with data perturbation

Privacy aware data release mechanism was studied in [95] for the first time, in which a theoretical framework of privacy-utility trade-off for data manipulation is proposed. This framework was later applied to SMs in [96]. In SM systems, load measurements are complex valued including real and reactive components. The empirical load measurements are shown to be approximately Gaussian in [102]; hence the continuous valued discrete-time SM data can be modeled as a sequence $\{Y_t\}$ of r.v.'s $Y_t \in \mathcal{Y}$, $t = \{\dots, -1, 0, 1, \dots\}$, generated by a stationary continuous Gaussian source with memory.

SM data sequence $\{Y_t\}$ contains private information $\{X_t\}$ of the data collector's choice, such as the energy consumption of a particular home appliance. This private information $X_t \in \mathcal{X}$ is correlated with and can be inferred from Y_t. Formally, the encoding function on SM side is a mapping from the meter reading sequence $Y^n = (Y_1, Y_2, \dots, Y_n)$, where $Y_t \in \mathbb{R}$, to an index $Z_n \in \mathcal{Z}_n = \{1, 2, \dots, Z_{max}\}$ given by

$$F_{enc} : \mathcal{Y}^n \to \mathcal{Z}_n, \tag{3.1}$$

where each index is a quantized sequence. The decoder at the UP side computes a distorted output sequence $\hat{Y}^n = (\hat{Y}_1, \hat{Y}_2, \dots, \hat{Y}_n)$, $\hat{Y}_t \in \mathbb{R}$, using the decoding function,

$$F_{dec} : \mathcal{Z} \to \hat{Y}^n. \tag{3.2}$$

To obtain a certain level of privacy in the SM problem, the encoding function F_{enc} is chosen such that the private information X_t cannot be inferred from the distorted output $\hat{Y}_t$. However, the distortion level must be kept limited such that the UP can still achieve utility from the distorted SM readings. The utility is measured by the mean-square error (MSE) distortion function,

$$D_n = \frac{1}{n} \sum_{t=1}^{n} \mathbb{E}\Big[\big(Y_t - \hat{Y}_t\big)^2\Big], \tag{3.3}$$

where the expectation in is over the joint distribution $p(y^n, \hat{y}^n) = P(y^n) p_t(\hat{y}^n | y^n)$. Privacy leakage is measured by the mutual information rate between the SM measurements received by the UP $\{\hat{Y}_t\}$ and the private information sequence $\{X_t\}$,

$$L_n = \frac{1}{n} I(X^n; \hat{Y}^n). \tag{3.4}$$

For a coding scheme given by (3.1) and (3.2) which satisfies (3.3) and (3.4), the SM utility-privacy trade-off region is a set of all (D, L) pairs, where D and L are the limit values of D_n and L_n as $n \to \infty$, respectively. However, this utility-privacy trade-off region does not bound the number of encoded sequences. The rate-distortion-leakage (RDL) trade-off region is the set of all (R, D, L) triplets for which there exists a sequence of coding schemes with (3.1), (3.2), each with a bounded number of encoded sequences

$$Z_{max} \leq 2^{n(R_n+\epsilon)}, \tag{3.5}$$

where $\epsilon > 0$, $R_n = (\log Z_{max})/n$ and $D_n \leq D + \epsilon$ while we have $R = \lim_{n\to\infty} R_n$. Under the constraints (3.3) and (3.4), SM utility-privacy trade-off can be quantified by the RDL trade-off region, where the rate-distortion and minimal leakage functions are denoted by

$$R(D, L) = \lim_{n\to\infty} \inf_{p(y^n,x^n)p(\hat{y}^n|y^n)} \frac{1}{n} I(Y^n; \hat{Y}^n), \tag{3.6}$$

$$\lambda(D) = \lim_{n\to\infty} \inf_{p(y^n,x^n)p(\hat{y}^n|y^n)} \frac{1}{n} I(X^n; \hat{Y}^n). \tag{3.7}$$

Rate-distortion function for Gaussian sources is well known [64] and can be obtained from the covariance matrix which is obtained by transforming the correlated source sequence into its eigen-space where the MSE function and the mutual information leakage are invariant. For example, the optimal encoding strategy for independent Gaussian r.v.'s can be obtained using the reverse water-filling algorithm [96].

3.3.2 SM privacy with demand shaping

In this section, SM privacy problem is examined in the presence of a renewable energy source and a rechargeable battery, which allow the user to physically manipulate its consumption [51,91,93]. A discrete time model of the SM system is illustrated in Figure 3.3, in which the energy demand of the user and energy requested from the grid at time slot t are denoted by $X_t \in \mathcal{X}$ and $Y_t \in \mathcal{Y}$, respectively, where $(|\mathcal{X}|, |\mathcal{Y}| < \infty)$. The RB state of charge at the beginning of time slot t is denoted by $B_t \in \mathcal{B} := \{0, \dots, B_m\}$, in which the initial state B_1 is distributed with probability p_{B_1}. The battery charging and discharging process is assumed ideal without any losses (see [88,103] for a model with energy losses). $E_t \in \mathcal{E} := \{0, \dots, E_m\}$ units of energy are generated by the RES at the beginning of each time slot t, and these can be used by the appliances only through the RB. The E_t process is assumed to be independent of X_t, and as the most general case, the realizations of E_t are not known by the UP. X_t and E_t are assumed to be first-order time-homogeneous Markov chains with transition probabilities q_X and q_E, and initial state distributions p_{X_1} and p_{E_1} for their initial states X_1 and E_1, respectively. Time-homogeneity and Markov

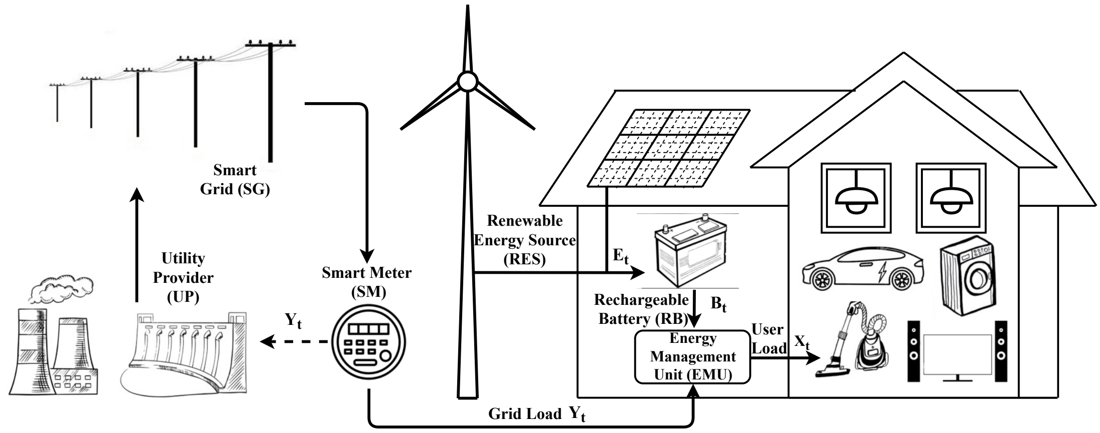


FIGURE 3.3: Illustration of the SM system model with RB and RES.

chain assumptions imply that the transition probabilities between two time instances depend only on the difference between those times, and the stochastic process is memoryless given the previous time, respectively. The E_t process is assumed to be independent of X_t. These assumptions are realistic in stationary environments, and stationarity can be approximated by choosing appropriate time-horizons for the process.

The appliances' energy demand is always satisfied by assuming $E_t + B_t + Y_t \geq X_t$, $\forall t$. In addition, intentional energy waste to provide privacy, or selling energy to the grid are not allowed.

3.3.2.1 SM privacy with a RES

First, we consider the special case where the RB capacity of the SM system illustrated in Figure 3.3 is zero, i.e., $B_m = 0$. Here the energy from the grid or RES cannot be stored in an RB to provide additional privacy. Since the UP cannot access the amount of energy generated by the RES at a particular time instant, users can achieve a certain level of privacy depending on the amount of energy they can receive from the RES. Assume that the RES is limited in terms of the average and peak power it can provide. Therefore, the objective of the SM privacy problem with RES is to obtain the optimal policy providing the best privacy under the average and peak power constraints of the RES. Information leakage rate to be minimized can be written as the mutual information rate between the user demand and the grid energy, i.e.,

$$I_T = \frac{1}{T} I(X^T; Y^T). \tag{3.8}$$

The maximum power which can be received from the RES in a time slot t is denoted by $\hat{P}$, and must satisfy $0 \leq X_t - Y_t \leq \hat{P}$. Moreover, the average power, $\bar{P}_T$, that the RES can provide over a finite horizon T is defined by,

$$\bar{P_T} = \mathbb{E}\Big[\frac{1}{T}\sum_{t=1}^{T}(X_t - Y_t)\Big], \tag{3.9}$$

where the expectation is taken over the joint probability distribution of the user demand and the grid power. Under these constraints, the asymptotic performance limit of the n-letter problem becomes an infinite dimensional optimization problem. On the other hand, using single-letter r.v.'s allows achieving the optimal solution solving a finite-dimensional optimization problem. A single letter expression for the minimum information leakage rate can be obtained under the assumption of i.i.d. demand, and it can be characterized by the *privacy-power function* defined as,

$$\mathcal{I}(\bar{P}, \hat{P}) = \inf_{P_{Y|X} \in \mathcal{F}} I(X;Y), \tag{3.10}$$

where $\mathcal{F} := \{P_{Y|X} : y \in \mathcal{Y}, \mathbb{E}[(X-Y)] \leq \bar{P}, 0 \leq X - Y \leq \hat{P}\}$. Here, the energy constraints are not affected by the past, since there is no battery, and thus no memory in the system. The optimal energy management policy minimizing (3.10) is stochastic and memoryless, and depends only on the current demand.

We note that the objective function (3.10) is similar to *rate-distortion function* $R(D)$ in information theory, which describes the minimum required compression rate R, in bits per sample, for an i.i.d. source sequence X^T with distribution p_X such that the receiver can reconstruct the source sequence achieving a particular expected distortion level D [64]. Average distortion between sequences X^T and $\hat{X}^T$ is denoted by $D = \frac{1}{T}\sum_{t=1}^{T} d(x_t, \hat{x}_t)$, where $d(x, \hat{x})$ and $\hat{X}$ represent the distortion measure used, and the reconstruction alphabet, respectively. The *information rate-distortion function* $R^{(I)}(D)$ for a source X with distortion measure $d(x, \hat{x})$ is defined by Shannon as [64],

$$R^{(I)}(D) = \min_{p(x|\hat{x}) \in \tilde{\mathcal{F}}} I(X;\hat{X}), \tag{3.11}$$

where $\tilde{\mathcal{F}} := \{P(x|\hat{x}) : \sum_{(x,\hat{x})} P(x)P(\hat{x}|x)d(x,\hat{x}) \leq D\}$. The analogy between the privacy-power function (3.10) and the rate-distortion function (3.11) can be made assuming the distortion measure:

$$d(x,y) = \begin{cases} x - y, & \text{if } 0 \leq x - y \leq \hat{P}, \\ D_{max}, & \text{otherwise}, \end{cases} \tag{3.12}$$

where $D_{max} < \infty$ is a very large scalar. Hence, this enables us to use tools from rate-distortion theory to examine SM privacy problems with RES [91,92,101]. However, there are two major differences between the rate-distortion and SM privacy problems, namely

i) grid energy Y^T is the direct output of the "encoder", which is represented by the EMU in the SM problem, rather than the reconstruction of the decoder, and ii) unlike the lossy encoder, EMU determines the output load Y_t instantaneously after receiving the demand. Since the mutual information is a convex function of the distribution $P_{Y|X}$, the privacy-power function can be written as a constrained convex optimization problem and solved numerically using the Blahut Arimoto algorithm [64].

3.3.2.2 SM privacy with an RB

Here, we consider another special case where the SM system illustrated in Figure 3.3 has an RB, and no RES, i.e., $E_t = 0$ for all t. The energy demand of the user is supplied by the grid energy through the RB, and charging of the RB can only be performed by the energy grid. This scenario is studied in [50] and [90], where both recast the problem as an MDP.

The battery state of charge is updated by,

$$B_{t+1} = B_t + Y_t - X_t, \tag{3.13}$$

where Y_t is chosen such that $B_{t+1} \leq B_m$.

The amount of energy requested from the grid is determined by a randomized battery charging policy $\boldsymbol{q} = \{q_t\}_{t=1}^{\infty}$, where q_t is a conditional probability distribution $q_t(Y_t|X^t, B^t, Y^{t-1})$ which randomly decides on the amount of energy received from the grid at time t given the histories of demand $X^t{:=}\{X_1, \dots, X_t\}$, battery charge B^t and grid energy Y^{t-1}, i.e.,

$$q_t : \mathcal{X}^t \times \mathcal{B}^t \times \mathcal{Y}^{t-1} \rightarrow \mathcal{Y}. \tag{3.14}$$

The goal of the SM privacy problem is to find an energy management policy, $\{q_t^*\}_{t=1}^{\infty}$, which provides the best privacy.

Privacy of an energy management policy over a time period T can be measured by the *information leakage rate*, which is defined as the average mutual information between the demand side load (X^T, B^T), and SM readings Y^T:

$$L_{\boldsymbol{q}}(T) := \frac{1}{T} I(X^T, B^T; Y^T). \tag{3.15}$$

The RB state of charge, i.e., B^T, is included in the privacy measure, since a potential adversary having access to B^T can deduce the SM measurements due to the deterministic relationship between X_t, Y_t, B_t and B_{t+1} in (3.13). In [50], it is proved that there is no loss of optimality in considering policies of the form $q_t(Y_t|X_t, B_t, Y^{t-1})$; that is, it is sufficient to consider only the current demand and battery state. Hence, (3.15) can be rewritten in an additive form

$$L_{\boldsymbol{q}}(T) = \frac{1}{T}\sum_{t=1}^{T} I(X_t, B_t; Y_t|Y^{t-1}). \tag{3.16}$$

Markovity of optimal actions and the additive objective function of information leakage rate enable this problem to be cast as a stochastic control problem, which can be formulated as an MDP.

SM privacy problem in the existence of RB can be cast as an average cost, infinite-horizon MDP with state $S_t{=}\{X_t, B_t\} \in \mathcal{S}$. However, the leakage at time t depends on Y^{t-1}, which leads to a growing state space in time. Therefore, the problem is formulated as a belief MDP and belief state $\beta_t(s_t)$ is defined as the causal posterior probability distribution over the state space of (X_t, B_t) given Y^{t-1}:

$$\beta_t(s_t) = P^{\boldsymbol{q}}(S_t = s_t|Y^{t-1} = y^{t-1}). \tag{3.17}$$

The control actions chosen by randomized policies are the conditional probabilities of energy received from the grid given the state and belief, denoted by $a_t(y_t|s_t) = P^{\boldsymbol{q}}(Y_t = y_t|S_t{=}s_t, \beta_t)$, where $a_t \in \mathcal{A}$ [50]. As a result of the action taken at time t, belief is updated for the next time interval as follows:

$$\beta(s_{t+1}) = p(s_{t+1}|y^t) = \frac{\sum\limits_{s_t} p(s_{t+1}, s_t, y_t|y^{t-1})}{p(y_t|y^{t-1})} \tag{3.18a}$$

$$= \frac{\sum\limits_{s_t} p(s_t|y^{t-1})p(y_t|s_t, y^{t-1})p(s_{t+1}|y_t, s_t)}{\sum\limits_{s_t, s_{t+1}} p(s_t|y^{t-1})p(y_t|s_t, y^{t-1})p(s_{t+1}|y_t, s_t)} \tag{3.18b}$$

$$= \frac{\sum\limits_{s_t} \beta(s_t)a_t(y_t|s_t)q_X(x_{t+1}|x_t)}{\sum\limits_{s_t, s_{t+1}} \beta(s_t)a_t(y_t|s_t)q_X(x_{t+1}|x_t)} \times \frac{1_{b_{t+1}}\{b_t + y_t - x_t\}}{1_{b_{t+1}}\{b_t + y_t - x_t\}}. \tag{3.18c}$$

where (3.18b) follows from the Bayes rule and the Markov chain $Y^{t-1} \to (S_t, Y_t) \to S_{t+1}$; and (3.18c) from the definitions of β and a_t. Given Y^{t-1}, per-step leakage of taking action $a_t(y_t|s_t)$ due to policy $\boldsymbol{q}$ is,

$$l_t(s_t, a_t, y^t; \boldsymbol{q}) := \log \frac{a_t(y_t|s_t)}{P^{\boldsymbol{q}}(y_t|y^{t-1})}. \tag{3.19}$$

Taking the expectation of the per-step leakage over a finite-horizon T, $\frac{1}{T}\mathbb{E}_{\boldsymbol{q}}[\sum_{t=1}^{T} l_t(s_t, a_t, y^t)]$, results in an objective function equivalent to the original formulation in (3.16). Given belief and action probabilities, average information leakage at time t is formulated as,

$$\begin{aligned}\mathbb{E}_{\boldsymbol{q}}[l_t(s_t, a_t, y^t)] &= I(S_t; Y_t | Y^{t-1} = y^{t-1}) \\ &= \sum_{s_t \in \mathcal{S} y_t \in \mathcal{Y}} \beta_t(s_t) a_t(y_t|s_t) \log \frac{a_t(y_t|s_t)}{\sum_{\hat{s}_t \in \mathcal{S}} \beta_t(\hat{s}_t) a_t(y_t|\hat{s}_t)}. \\ &= I(S_t; Y_t | \beta_t, a_t). \end{aligned} \tag{3.20}$$

SM privacy problem which is cast as an average cost belief-MDP can be solved by DyP. While an exact DyP solution cannot be achieved due to the continuous belief state, approximate numerical solutions can be obtained by using belief quantization methods [68]. To formulate the corresponding Bellman equation, which is a necessary condition for the optimality of DyP [104], Bellman operator T is written as,

$$[T_a v](\beta) = l(s, q(\beta), \beta) + \sum_{s \in \mathcal{S}, y \in \mathcal{Y}} \beta(s) a(y|s) v(\phi(\beta, y, a)), \tag{3.21}$$

where v is the value function and the updated belief state is represented by $\beta_{t+1} = \phi(\beta_t, y_t, a_t)$. Implementation of DyP for the finite-horizon and infinite-horizon settings is as follows:

Finite horizon DyP

- For $v_{n+1}(\beta) = 0$ and $t \in \{n, \dots, 1\}$, value functions, v_t, are recursively defined [65]:

$$v_t(\beta) = \min_{a \in A} [T_a v_{t+1}](\beta). \tag{3.22}$$

  Optimal leakage rate is given by $v_1(\beta_1)/n$, where $\beta_1(s) = p_{X_1} p_{B_1}$.
- The optimal policy minimizing the right hand side of (3.22) is denoted by $\mathbf{q}^* = (q_1^*, \dots, q_n^*)$:

$$q_t^*(y_t|s_t, \beta) = a_t(y_t|s_t). \tag{3.23}$$

Infinite horizon DyP

- For λ constant [65], the value function v is time-homogeneous and defined iteratively:

$$\lambda + v(\beta) = \min_{a \in A} [T_a v](\beta). \tag{3.24}$$

  Optimal leakage rate is given by λ.

- Time-homogeneous optimal policy, $\mathbf{q}^* = (q^*, q^*, \dots)$,

$$q^*(y_t|s_t, \beta) = a(y_t|s_t). \tag{3.25}$$

3.3.2.2.1 Single letter expression for i.i.d. demand Under the assumption that X_t is i.i.d. with probability distribution p_X, it is possible to achieve the optimal policy by solving a cost function in a single letter form. Consider an auxiliary state variable $W_t = B_t - X_t$, where $w \in \{b - x : b \in \mathcal{B}, x \in \mathcal{X}\}$. Then, the single letter minimum information leakage rate is given by [50],

$$J^* = \min_{\theta \in \mathcal{P}_\mathcal{B}} I(B - X; X) = \min_{\theta \in \mathcal{P}_\mathcal{B}} \{H(B - X) - H(B)\}, \tag{3.26}$$

where r.v.'s X and B are independent; θ is the probability distribution over B given the past observations and actions, i.e., $\theta := p(b_t|y^{t-1}, a^{t-1})$; and actions a_t are the conditional probabilities of grid load given the current demand, battery charge and the belief. Contrary to the Markovian demand case, here belief states are on W_t. Since the objective function (3.26) is convex over θ, the optimal policy can be obtained by Blahut-Arimoto algorithm [64]. The resulting grid load is i.i.d., and the optimal charging policy is memoryless and time-invariant.

3.3.2.2.2 Privacy-Cost Trade-Off In practice, in addition to privacy, energy cost is an important concern. Indeed, home energy storage devices are mainly installed to reduce energy consumption by storing energy during off-peak price periods [105, 106]. It is possible to maximize privacy by constantly purchasing high amount of energy from the grid and wasting the extra energy. However, this is against the purpose of SM from both the user and the UP point of view.

The same as minimizing the mutual information to maximize the achievable privacy, the conditional entropy of the demand process given the observations of UP can also be used as a privacy measure to maximize. In [90], the authors take both privacy and cost into account and recast the problem as an MDP. The privacy is formulated as,

$$\mathcal{P}(\boldsymbol{q}) := \frac{1}{T} H(X^T | Y^T, P^T), \tag{3.27}$$

where $P^T = (P_1, \dots, P_T)$ is the price of the energy purchased from the grid for $t = \{1, \dots, T\}$. Unlike privacy, energy cost has an additive formulation and can be easily incorporated into the MDP formulation. Following policy $\boldsymbol{q}$, the average cost savings per time slot are defined by,

$$\mathcal{C}(\boldsymbol{q}) := \frac{1}{T}\sum_{t=1}^{T} c(X_t, B_{t+1}, Y_t, P_t), \tag{3.28}$$

where $c(X_t, B_{t+1}, Y_t, P_t) = (X_t - Y_t)P_t$, $\forall B_{t+1} \in \mathcal{B}$. The objective of the SM privacy-cost trade-off problem with RB is considered as the weighted sum of privacy, $\mathcal{P}$, and average cost savings per time slot, $\mathcal{C}$. That is, the weighted reward function to be maximized is given by $\mathcal{R}(\boldsymbol{q}, \lambda) = \lambda\mathcal{P}(\boldsymbol{q}) + (1-\lambda)\mathcal{C}(\boldsymbol{q})$, where $\lambda \in [0, 1]$ denotes user's choice regarding the balance between privacy and cost. If $\lambda = 0$, only the cost savings are maximized, whereas if $\lambda = 1$, only the privacy is maximized. The problem in [90] is reformulated as a belief MDP. A Bellman equation which corresponds to a continuous state, continuous action, continuous reward MDP is written for stationary policies. However, due to the high computational complexity, only the privacy of cost-optimal, deterministic and greedy policies are studied in [90]. Optimal privacy-cost trade-off bounds are also obtained using rate distortion theory.

3.3.2.3 SM privacy with a RES and an RB

In this section, we consider a more general case in which the SM system is equipped with a finite capacity RB and a RES with non-zero energy generation (see Figure 3.3) [2, 51]. While the RB provides demand shifting, the RES supplies alternative energy to mask the energy consumption of the appliances. However, the memory introduced by the RB and the additional randomness due to the energy generation process of the RES, the SM privacy problem becomes more complicated than the previous cases with only RB or RES.

Here, the battery state of charge is updated by,

$$B_{t+1} = \min(E_t + B_t - X_t, B_m) + Y_t, \qquad \forall t, \tag{3.29}$$

where Y_t is chosen such that $B_{t+1} \leq B_m$. When the realizations of the energy generation process E_t are not known by the UP, information leakage rate of the SM system with RB and RES is defined by

$$L_{\boldsymbol{q}}(T) := \frac{1}{T} I(X^T, B^T, E^T; Y^T). \tag{3.30}$$

Randomized battery charging policies in the existence of an RB and a RES are defined such that $q_t : \mathcal{X}^t \times \mathcal{E}^t \times \mathcal{B}^t \times \mathcal{Y}^{t-1} \to \mathcal{Y}$. Similarly to Section 3.3.2.2, there is no loss of optimality in considering battery charging policies of the form $q_t(Y_t | X_t, B_t, E_t, Y^{t-1})$.

Therefore, (3.30) can be rewritten in an additive form

$$L_{\boldsymbol{q}}(T) = \frac{1}{T}\sum_{t=1}^{T} I(X_t, B_t, E_t; Y_t|Y^{t-1}). \tag{3.31}$$

Employing Markovian actions and additive objective function, SM privacy problem with RB and RES can be cast as an average cost MDP with states $S_t = \{X_t, B_t, E_t\} \in \mathcal{S}$. As before, the history dependence of the information leakage due to RB causes a growing state space in time. Hence, the problem is formulated as a belief MDP and belief state $\beta_t(s_t)$ is defined as the causal posterior probability distribution over the state space of (X_t, B_t, E_t) given Y^{t-1}. As a result of the action $a_t(y_t|s_t) \in \mathcal{A}$ taken at time t, belief is updated for the next time interval as follows:

$$\beta(s_{t+1}) = \\ \frac{\sum\limits_{s_t} \beta(s_t)a_t(y_t|s_t)q_E(e_{t+1}|e_t)q_X(x_{t+1}|x_t)1_{b_{t+1}}\{\min(e_t + b_t - x_t, B_m) + y_t\}}{\sum\limits_{s_t, s_{t+1}} \beta(s_t)a_t(y_t|s_t)q_E(e_{t+1}|e_t)q_X(x_{t+1}|x_t)1_{b_{t+1}}\{\min(e_t + b_t - x_t, B_m) + y_t\}}. \tag{3.32}$$

The derivation of the intermediate steps can be performed following (3.18) with the corresponding modifications. Given $Y^{t-1} = y^{t-1}$, the average information leakage in (3.31) can be written in terms of belief and actions by averaging the per-step leakage in (3.19) over the belief and action probabilities, when $S_t = \{X_t, B_t, E_t\}$. With the integration of renewable energy generation, the resulting objective (3.20) can be minimized by following the DyP steps (3.22)-(3.25).

3.3.2.3.1 Renewable Energy Known by the UP Here, we consider a special case of the SM privacy problem with an RB and a RES, in which the UP knows the realizations of E_t. In this scenario, energy management policies of the form $q_t(Y_t|X_t, B_t, E^t, Y^{t-1})$ are taken into account, and the information leakage rate induced by policy $\mathbf{q}$ is denoted by,

$$L_{\boldsymbol{q}}(T) := \frac{1}{T} I(X^T, B^T; Y^T|E^T) = \frac{1}{T}\sum_{t=1}^{T} I(X_t, B_t; E_t, Y_t|Y^{t-1}, E^{t-1}). \tag{3.33}$$

Similarly to the E_t unknown case, the problem can be reformulated as a belief MDP. The belief state is defined as the conditional probability on the system state $S_t := (X_t, B_t)$, given the observation history (Y^{t-1}, E^{t-1}), i.e., $\beta(s_t) := p(s_t|y^{t-1}, e^{t-1})$. As a result of the action $a_t(y_t|s_t, e_t) = P^{\boldsymbol{q}}(Y_t = y_t|S_t{=}s_t, E_t = e_t, \beta_t)$, belief is updated as follows,

$$\beta(s_{t+1}) = \frac{\sum\limits_{s_t} \beta(s_t)a_t(y_t|s_t, e_t)q_E(e_t|e_{t-1})q_X(x_{t+1}|x_t)1_{b_{t+1}}\{\min(e_t + b_t - x_t, B_m) + y_t\}}{\sum\limits_{s_t, s_{t+1}} \beta(s_t)a_t(y_t|s_t, e_t)q_E(e_t|e_{t-1})q_X(x_{t+1}|x_t)1_{b_{t+1}}\{\min(e_t + b_t - x_t, B_m) + y_t\}}, \tag{3.34}$$

where the intermediate steps can be derived from the Bayes rule, Markovity of E_t, and the Markov chain $(Y^{t-1}, E^{t-1}) \to (S_t, Y_t, E_t) \to S_{t+1}$. Unlike the E_t unknown scenario, energy generation process is not included in belief since the UP has the exact information about E_t realizations. Given (Y^{t-1}, E^{t-1}), per-step information leakage of taking action $a_t(y_t|s_t, e_t)$ incurred by policy $\boldsymbol{q}$ is,

$$l_t(s_t, e^t, a_t, y^t; \boldsymbol{q}) := \log \frac{a_t(y_t|s_t, e_t)q_E(e_t|e_{t-1})}{P^{\boldsymbol{q}}(y_t, e_t|y^{t-1}, e^{t-1})}. \tag{3.35}$$

Taking average leakage over a finite-horizon T, $\frac{1}{T}\mathbb{E}_{\boldsymbol{q}}[\sum_{t=1}^{T} l_t(s_t, e^t, a_t, y^t)]$, is equal to the original formulation in (3.33). Given belief and action probabilities, average information leakage at time t is denoted by:

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{q}}[l_t(s_t, e^t, a_t, y^t)] = I(S_t; E_t, Y_t|Y^{t-1} = y^{t-1}, E^{t-1} = e^{t-1}) \\
&\quad = \sum_{\substack{s_t \in \mathcal{S} \\ e_t \in \mathcal{E} y_t \in \mathcal{Y}}} \beta_t(s_t)a_t(y_t|s_t, e_t)q_E(e_t|e_{t-1}) \log \frac{a_t(y_t|s_t, e_t)q_E(e_t|e_{t-1})}{\sum\limits_{\hat{s}_t \in \mathcal{S}} \beta_t(\hat{s}_t)a_t(y_t|\hat{s}_t, \hat{e}_t)q_E(e_t|e_{t-1})} \\
&\quad = I(S_t; E_t, Y_t|\beta_t, q_E, a_t).
\end{aligned} \tag{3.36}$$

The problem is recast as a belief MDP, and the Bellman equation to be used in DyP is modified with the integration of observed energy generation process,

$$[T_a v](\beta) = l(s, q(\beta), \beta, q_E) + \sum_{\substack{s \in \mathcal{S} \\ e \in \mathcal{E} y \in \mathcal{Y}}} \beta(s)a(y|s, e)q_E(e|\hat{e})v(\phi(\beta, y, a, e)), \tag{3.37}$$

where $\hat{e}$ is the energy generated in the previous step and the updated belief state is represented by $\beta_{t+1} = \phi(\beta_t, y_t, a_t, e_t)$. Finite-horizon and infinite-horizon MDP steps can be followed from (3.22)-(3.25).

3.3.2.3.2 Special Renewable Energy Generation Process Here, we propose low complexity policies and numerical solutions for SM privacy-cost trade-off in the existence of both RES and RB by exploiting a special energy arrival process that fully recharges the battery at random time instances, i.e, $E_t \in \{0, B_m\}$. The realizations of the renewable energy generation process E_t is assumed to be known by the UP. Due to the special energy arrival process, the problem is an episodic MDP, which resets to an initial state

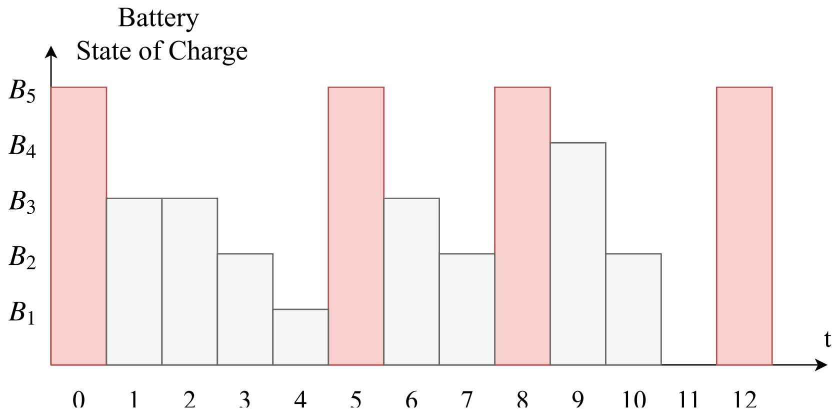


FIGURE 3.4: Illustration of the RB state of charge under the special energy generation process assumption in [2].

of full RB at every renewable energy instant. Between two consecutive energy arrivals, energy transitions occur only between the grid, the battery and the home appliances. An example for the RB state of charge for $B_m = 5$ under the special energy generation process assumption is given in Figure 3.4. Red bars express the fully charged battery state at time instances $t = 0, 5, 8, 12$, when the renewable energy is generated. Between two consecutive energy arrivals, the RB state of charge is represented by grey bars. Hence, for each time period between two RES charging instants, the system can be modeled as an SM with only an RB and no RES. Accordingly, a finite-horizon privacy-cost trade-off problem is formulated for an SM system with an initially full RB, which is used to propose a low-complexity policy as well as a lower bound for the original problem. Between two RES charging instants, battery update is performed according to (3.13) and the finite-horizon average information leakage is formulated as in (3.16). Energy cost has an additive formulation and can be incorporated into the MDP formulation. Price process of the energy purchased from the grid at time t is defined as P_t. Following policy $\boldsymbol{q}$, the average energy cost per time slot is defined by,

$$C_{\boldsymbol{q}}(T) := \frac{1}{T}\sum_{t=1}^{T} Y_t P_t. \tag{3.38}$$

Due to the growing space of observations of the UP, belief states are defined and the problem is recast as a belief-MDP. The weighted objective function is given by $U_{\boldsymbol{q}}(\lambda, T) = \lambda L_{\boldsymbol{q}}(T) + (1-\lambda) C_{\boldsymbol{q}}(T)$, where $\lambda \in [0,1]$ denotes user's choice regarding the privacy-cost balance, is represented in terms of belief states and actions, and minimized over the action space. The optimal policy for each episode is obtained by applying finite-horizon DyP via the Bellman operator given in (3.21).

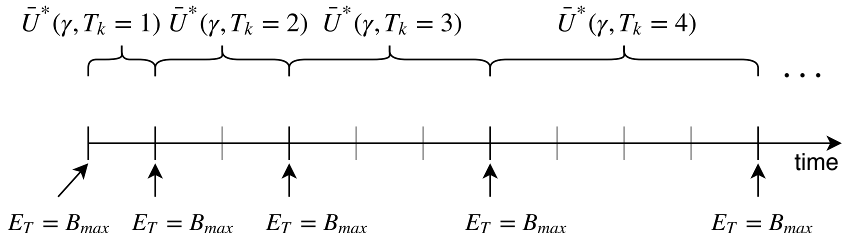

$\bar{U}^*(\gamma, T_k = 1)$
$\bar{U}^*(\gamma, T_k = 2)$
$\bar{U}^*(\gamma, T_k = 3)$
$\bar{U}^*(\gamma, T_k = 4)$
...
time
$E_T = B_{max}$
$E_T = B_{max}$
$E_T = B_{max}$
$E_T = B_{max}$
$E_T = B_{max}$

FIGURE 3.5: Renewable energy generation instances and privacy-cost rate for the corresponding intervals.

Threshold Policy (TP)

According to the low complexity proposed in [2], after each RB recharge instance, the optimal policy obtained for a fixed finite-horizon n is employed. The optimal policy for horizon n is followed until either the battery is recharged again, in which case the algorithm restarts with the same policy, or the time horizon n is reached. If the RB is not recharged at time $(n+1)$, it is assumed that all the energy demand is directly supplied by the grid, resulting in full information leakage. The intuition behind this scheme follows from the law of large numbers, which suggests that, with high probability, the RB will be charged after $n = \frac{1}{P_E}$ time slots, where P_E is the energy generation probability at any t. We consider policies with a fixed time horizon of $n = \frac{1}{P_E}$, as well as those with an optimized time horizon.

Battery Conditioned Policy (BCP)

We propose another low-complexity policy which depends only on the current input load. In BCP, when there is no demand, we allow the RB to be recharged by the grid with a probability P_{C_i} for each battery state $B_t{=}i$, for $i{=}\{0, \ldots, B_{max}\}$. On the other hand, when there is energy demand, the RB is discharged with a probability P_{D_i} for each battery state. As before, intentional energy waste is not allowed. When there is demand in the case of an empty RB, it is entirely supplied from the grid. We choose (P_{C_i}, P_{D_i}) values that minimize (3.16) by an exhaustive grid search on $[0,1]^2$.

Lower Bound

Next, we provide a lower bound on the privacy-cost trade-off by assuming that the user non-causally knows the times at which the RES recharges the RB. In Figure 3.5, these time instances are represented by consecutive arrows. The weighted sum of finite-horizon leakage rate and average energy cost, minimized over policy $\boldsymbol{q}$, is denoted by $\bar{U}^*(\gamma, T_k)$ in Figure 3.5. Given i.i.d. P_E, the probability that the RB is recharged after T_k time slots is given by

$$f(T_k; P_E) = P_E(1 - P_E)^{T_k}. \tag{3.39}$$

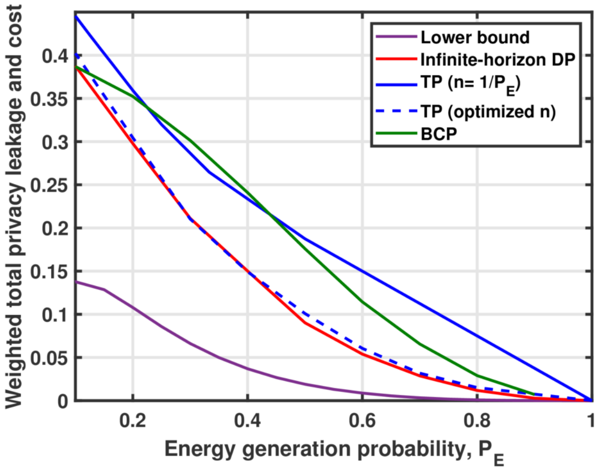


FIGURE 3.6: Privacy-cost trade-off of the lower bound, TP, BCP and infinite-horizon MDP w.r.t. P_E for γ=0.5 and P_X=0.5.

If the RB recharge instances are known in advance, the problem reduces to the finite-horizon MDP for each inter-arrival period.

Once the optimal performance is evaluated for all $T_k > 0$, the lower bound can be derived by taking their average using the probability mass function in (3.39):

$$F_\gamma(P_E) = \sum_{k=1}^{\infty} f(T_k; P_E)\bar{U}^*(\gamma, T_k), \tag{3.40}$$

where the coefficient $f(T_k; P_E)$ approaches zero as $T \to \infty$, while $\bar{U}^*(\gamma, T_k)$ approaches the infinite-horizon privacy-cost trade-off. For the numerical solution of the infinite-sum indicated in (3.40), we perform the summation for finite k=$\{1, \dots, K\}$ such that $\sum_{k=K+1}^{\infty}\{f(T_k; P_E)\bar{U}^*(\gamma, T_k)\} < \epsilon$. To obtain the minimum K satisfying this inequality, we first consider the worst case information leakage rate and average energy cost, where all the demand is supplied by the grid, $Y_t = X_t$, and denote the lower bound by

$$F_\gamma(P_E) \leq \sum_{k=1}^{K} f(T_k; P_E)\bar{U}^*(\gamma, T_k) + \sum_{k=K+1}^{\infty} f(T_k; P_E)\bar{U}_w(\gamma), \tag{3.41}$$

where $\bar{U}_w(\gamma) := [\gamma H(X) + (1-\gamma)\mathbb{E}[X]]$ represents the worst case privacy-cost trade-off, in which $H(X)$ and $\mathbb{E}[X]$ are the entropy and expected value of the demand, respectively. Hence, we choose the minimum K value that satisfies $\sum_{k=K+1}^{\infty} f(T_k; P_E)\bar{U}_w(\gamma) = (1 - P_E)^{T_K}\bar{U}_w(\gamma) < \epsilon$. We can find a finite T_K satisfying this inequality for any $\epsilon > 0$.

A simple binary example

We consider a simple scenario with $(\mathcal{X},\mathcal{Y})$=$\{0, 1\}$, $\mathcal{E}$=$\{0, 2\}$ and $\mathcal{B}$=$\{0, 1, 2\}$. We emphasize that obtaining numerical results for larger alphabets is challenging as the belief grows with the state space, and so does the computational complexity, also due to the quantization of the belief. For simplicity, demand and energy generation processes are assumed to be i.i.d. with Bernoulli P_X=0.5 and $P_E \in [0, 1]$, respectively. Extensions to Markovian E_t process is straightforward for TP and BCP; however, the MDP formulation requires including E_t in the state, and updating the belief accordingly. We consider a privacy-cost trade-off weight of γ=0.5.

The weighted total privacy leakage and energy cost for TP, BCP and infinite-horizon MDP are depicted in Figure 3.6, together with the lower bound. The average weighted cost decreases with P_E, since the demand can be mostly supplied by the RES, decreasing both the cost and leakage. The lower bound is obtained from (3.40) evaluated over a sufficiently long T. While the lower bound is not tight in general, it also shows us the value of predicting the energy generation instances for optimizing the privacy and cost. Two plots of TP are obtained corresponding to different horizons. For the first TP plot, the finite-horizon is set to be $n=\frac{1}{P_E}$. Since TP leads to full information leakage when energy arrives later than the set horizon, this approach has a higher privacy-cost trade-off compared to the infinite-horizon DyP solution of the original problem. For the second TP plot, for each P_E value, the best horizon value is selected by searching over the set $n = [1 : 15]$. We observed that, the optimal fixed horizon is typically longer than $\frac{1}{P_E}$, which reduces the probability of full leakage. Interestingly, the performance of TP with optimized yet fixed horizon follows that of the infinite-horizon MDP solution very closely. We remark here that the curve obtained for the infinite-horizon MDP solution is an approximation as well, due to the quantization of the belief. Finally, we observe that the performance of the BCP scheme can outperform that of fixed horizon TP policy for high P_E values.

3.4 Conclusions

SMs are end user interfaces that monitor the energy consumption of users. SMs provide accurate, high frequency consumption data to the UPs, and they are being widely deployed around the world. The adoption of SM s has created a multi-billion dollar business. However, private information about user's personal lives can be inferred from detailed SM readings by the UP, which has led to significant consumer outrage, creating a serious roadblock in front of the widespread deployment of SMs. Therefore, enabling privacy-aware SM technology has an undeniable importance both for consumers and for other stakeholders in this multi-billion dollar industry.

In this chapter, SM privacy-preserving techniques have been discussed. They are classified into two: data manipulation methods which modify SM measurements and, demand shaping methods which manipulate the energy received from the grid physically. The second group of methods, which use physical resources, such as RB and RES, have been examined in detail as they provide privacy without compromising the role of SM in providing timely and accurate energy consumption information. Unlike SM data manipulation, demand shaping methods report accurate and real consumption measurements to the UP, which maintains the benefits of the SG concept. We have mainly focused on information theoretic privacy measures, in particular the mutual information between the real energy consumption and the energy received from the grid, which is also what the SMs report to the UP. Other measures have also been considered in the literature, see for example [107, 108]. Rate-distortion theory and MDPs have been used as mathematical tools to study the fundamental information theoretic privacy measures.

Although there are a vast number of solutions which have been proposed in the literature, SM privacy problem still has many challenges to be addressed. Among the various privacy metrics defined, there is still lack of a privacy measure which is generic, device-independent and well suited to various privacy-preserving methods. Information theoretic privacy metrics provide solutions independent of the attacker behavior, such as the particular detection technology employed by the attacker; however, they depend on an underlying statistical model governing the various processes involved. The assumed statistical models may not be valid in practice, or more involved models might be needed, under which clean optimal solutions may not be possible, requiring computationally limited sub-optimal solution that can provide reasonable privacy guarantees. Moreover, the cost of privacy-preserving techniques and installation of RB or RES is still considerably high compared to cost savings due to SM usage. However, this cost may reduce as renewable energy becomes more widespread making RES and RBs more commonly available to households.

Chapter 4

Time-Series Data Privacy

In this chapter, we study PUT in time-series data sharing. In the previous chapter, we mainly focused on demand shaping techniques which preserve the privacy by physically modifying the data to be shared. Here, we focus on PUT by obfuscating the data with noise before sending it to the SP. Existing approaches to PUT with data obfuscation mainly focus on a single data point; however, temporal correlations in time-series data introduce new challenges. Methods that preserve the privacy for the current time may leak significant amount of information at the trace level as the adversary can exploit temporal correlations in a trace. In this chapter, a distorted version of a user's true data sequence is shared with the SP, and the privacy leakage is measured by the MI between the user's true data sequence and its shared version. Both the instantaneous and average distortion between the two sequences, under a given distortion measure, are considered as the utility loss metric. To tackle the history-dependent MI minimization, we reformulate the problem as an MDP, and solve it using A2C-DRL. The performance of the proposed solution in location trace privacy are evaluated on both synthetic and real GPS trajectory datasets. For the latter, the validity of the proposed solution is shown by testing the privacy of the released location trajectory against an adversary network.

4.1 Introduction

In this chapter, we study the fundamental PUT when sharing sensitive time-series data. We consider the scenario in which the user measures time-series data (e.g., location, heartbeat, temperature or energy consumption) generated by a first-order Markov process through an IoT device, and periodically reports a distorted version of her true data to an untrusted SP to gain utility. We assume that the true data becomes available to the user in an online manner. We use the MI between the true and distorted data sequences as a measure of privacy loss, and measure the utility of the reported data by a specific distortion metric between the true and distorted samples. For the PUT, we introduce an online private data release policy (PDRP) that minimizes the MI while keeping the distortion below a certain threshold. We consider both instantaneous and average distortion constraints. We consider data release policies which take the entire released data history into account, and show its information theoretic optimality. To tackle the complexity, we exploit the Markovity of the user's true data sequence, and recast the problem as an MDP. After identifying the structure of the optimal policy, we use A2C-DRL framework as a tool to evaluate our continuous state and action space MDP numerically. To the best of our knowledge, this is the first time DRL tools are used to optimize information theoretic time-series data privacy.

The performances of the proposed PDRPs are examined in two specific scenarios: In the first scenario, synthetic location traces are generated considering a user moving in a grid-world with a known Markov mobility pattern. In the second scenario, we use GPS traces of a user from GeoLife dataset [109,110]. For the average distortion constrained case, the proposed PDRP is compared with a myopic location data release mechanism [47]. While the privacy leakage of the considered PDRPs can be evaluated for the synthetic dataset, this cannot be done for the GeoLife trace since we do not know the true statistics of this dataset. Instead, we compare the privacy achieved by the proposed and myopic policies using an adversary which predicts the current location of the user from the past released locations. The adversary is represented by a long short-term memory (LSTM) predictor. The performances of the proposed policies are tested under various adversary memory sizes.

This chapter contains our previous work on PUT for location sharing [111], and its extension to generic time-series data sharing. Our contributions are summarized as follows:

- We propose a simplified PDRP by exploiting the Markov property of the user's true data sequences. Then, we prove the information theoretic optimality of the simplified strategy.
- We recast the information theoretic time-series data PUT problem as an MDP and evaluate the optimal PDRP numerically using A2C-DRL.
- We apply the obtained information-theoretically optimal PDRP on the location trace privacy problem, and evaluate its performance under instantaneous and average distortion constraints using both synthetic and GeoLife [109] trajectory datasets.

The remainder of this chapter is organized as follows. We present the problem statement in Section 4.2 where we also introduce the privacy and utility metrics used in this chapter.

Table 4.1: Notation Summary

Notation	Definition
$\mathcal{W}$	Time-series data set
n	Time-series data length
X_t, Y_t	Random variables representing the user's true and distorted data at time t
p_{x_1}	Probability distribution of the true data at $t = 1$
$q_x(.\|.)$	Markov transition of user data
$\mathcal{Q}_x$	Markov transition matrix of transition probabilities
$q(.\|.)$	Conditional probability distribution, (policy)
$\mathcal{Q}_H$	Probability space of history dependent policies
$\mathcal{Q}_S, \mathcal{Q}'$	Probability space of simplified policies under first-order and m-th order Markov assumptions

In Section 4.3, we introduce simplified data release mechanisms for the time-series data PUT problem. In Section 4.4, we reformulate the problem as an MDP and propose a numerical evaluation approach utilizing advantage actor-critic deep RL. In Section 4.5, we apply the proposed solution to the location trace privacy problem, and compare the performance of the proposed location release strategy with a myopic policy numerically. Finally, we conclude our work in Section 4.6.

4.2 System Model

We consider a time-series $\{X_t\}_{t\geq 1}$, taking values from a finite discrete set $\mathcal{W}$. The user shares $\{X_t\}$ with an SP to gain utility through some online service. We assume that the user's true data sequence $\{X_t\}_{t\geq 1}$ follows a first-order time-homogeneous Markov chain with transition probabilities $q_x(x_{t+1}|x_t)$, and initial probability distribution p_{x_1}. While the first-order Markov structure assumed for the true data may seem restrictive, we will show that our solution techniques generalize to higher-order Markov chains, albeit with increased complexity in the numerical solutions. In the literature, Markov structure is a common assumption for time-series data, and it is proved to be a reasonable assumption for location trajectories [112], smart meter readings [113] and financial data [114] due to the history dependent behavior of these time-series.

Instead of sharing its true data at time t, the user shares a distorted version of her current data, denoted by $Y_t \in \mathcal{W}$. The released data at time t, Y_t, does not depend on future data samples; i.e., for any $1 < t < n$, $Y_t \rightarrow (X^t, Y^{t-1}) \rightarrow (X^n_{t+1}, Y^n_{t+1})$ form a Markov chain, where we have denoted the sequence $(X_{t+1}, \ldots, X_n)$ by X^n_{t+1}, and the sequence $(X_1, \ldots, X_t)$ by X^t. The notations which have been used throughout the chapter are listed in Table 4.1.

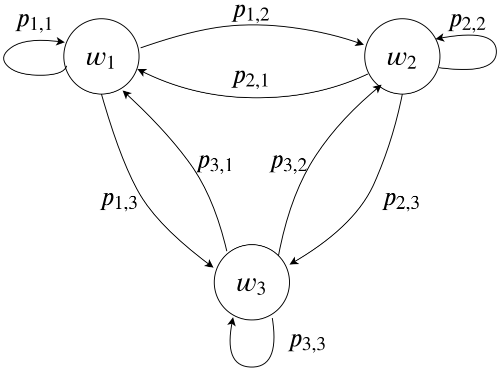


Figure 4.1: Markov chain example for the true data generation.

For a better understanding of the user's private time-series data generation process, a simple Markov chain with state space $\mathcal{W} = \{w_1, w_2, w_3\}$ and state transition probabilities $p_{i,j}$ for $(i, j) \in \{1, 2, 3\}$ are presented in Figure 4.1. The sensitive data X_t takes the values $\{w_1, w_2, w_3\}$ according to the state transition probabilities. The user becomes aware of X_t in an online manner and releases a distorted version $Y_t \in \{w_1, w_2, w_3\}$, following her privacy-preserving strategy.

4.2.1 Privacy and Utility Measures

Drawing from Section 2.1, we quantify the privacy by the information leaked to the untrusted SP measured by the MI between the true and released data sequences. Accordingly, the information leakage of the user's release strategy for a time period n is given by

$$I(X^n; Y^n) = \sum_{t=1}^{n} I(X^n; Y_t|Y^{t-1}) = \sum_{t=1}^{n} I(X^t; Y_t|Y^{t-1}), \tag{4.1}$$

where the first equality follows from the chain rule of MI, while the second from the Markov chain $Y^t \to (X_t, Y^{t-1}) \to X_{t+1}^n$. Even though a malicious third party can obtain the statistics of the user's data release strategy over an infinite time horizon, i.e., $n \to \infty$, he cannot infer the realizations of the private information due to the privacy measure based on uncertainty. Since information theoretic metrics are independent of the attack's behavior and computational capabilities, they are preferable as privacy measures.

In the time-series data privacy problem, we want to minimize the information leakage to the SP. However, as we apply more distortion to the true data sequence for privacy, the more utility is lost due to increased deviation from the original sequence. That is, releasing distorted data reduces the utility received from the SP, and the distortion applied by the user should be limited to a certain level. Therefore, our main purpose is to characterize the trade-off between the privacy and utility. The distortion between the true data sample X_t and the released version Y_t is measured by a distortion measure $d(X_t, Y_t)$ specified based on the underlying application (e.g., Manhattan distance or Euclidean distance), where $d(X_t, Y_t) < \infty, \forall X_t, Y_t \in \mathcal{W}$.

Our main goal is to minimize the information leakage rate to the SP while satisfying the distortion constraint for utility. Throughout this chapter, we consider two different constraints on the distortion introduced by PDRP, namely an *instantaneous distortion constraint* and an *average distortion constraint*. The infinite-horizon optimization problem can be written as:

$$\lim_{n\to\infty} \min_{\substack{\{q_t(y_t|x^t,y^{t-1}):\\ d(X_t,Y_t)\leq\hat{D}\}_{t=1}^{n}}} \frac{1}{n}\sum_{t=1}^{n} I^{\boldsymbol{q}}(X^t;Y_t|Y^{t-1}) \tag{4.2}$$

under the instantaneous distortion constraint $\hat{D}$, and as

$$\lim_{n\to\infty} \min_{\substack{q_t(y_t|x^t,y^{t-1}):\\ \mathbb{E}\left[\frac{1}{n}\sum\limits_{t=1}^{n} d(X_t,Y_t)\right]\leq\bar{D}}} \frac{1}{n}\sum_{t=1}^{n} I^{\boldsymbol{q}}(X^t;Y_t|Y^{t-1}) \tag{4.3}$$

under the average distortion constraint $\bar{D}$, where x_t and y_t represent the realizations of X_t and Y_t, $\boldsymbol{q} = \{q_t(y_t|x^t, y^{t-1})\}_{t=1}^{n}$ is a conditional probability distribution which represents the user's randomized *data release policy* at time t. The randomness stems from both the Markov process generating the true data sequence, and the random release mechanism $q_t(y_t|x^t, y^{t-1})$. The MI induced by policy $q_t(y_t|x^t, y^{t-1}) \in \boldsymbol{q}$ is calculated using the joint distribution

$$P^{\boldsymbol{q}}(X^n = x^n, Y^n = y^n) = p_{x_1} q_1(y_1|x_1) \prod_{t=2}^{n} \left[q_x(x_t|x_{t-1}) q_t(y_t|x^t, y^{t-1})\right]. \tag{4.4}$$

In the next section, we characterize the structure of the optimal data release policy, and using this structure we recast the problem as an MDP, and finally evaluate the optimal trade-off numerically using A2C-DRL.

4.3 PUT for Time-Series Data Sharing

In this section, we analyze the optimal PUT achievable by a privacy-aware time-series data release mechanism under the notion of MI minimization with both instantaneous and average distortion constraints. Moreover, we propose simplified PDRPs that still preserve optimality. By the definition of MI, the objectives (4.2) and (4.3) depend on the entire history of X and Y. Therefore, the user must follow a history-dependent PDRP $q_t^h(y_t|x^t, y^{t-1})$, where the feasible set $\mathcal{Q}_H$ consists of policies that satisfy $\sum_{y_t \in \mathcal{W}} q_t^h(y_t|x^t, y^{t-1}) = 1$. As a result of strong history dependence, computational complexity of the minimization problem increases exponentially with the length of the data sequence. To tackle this problem, we introduce a class of simplified policies, and prove that they do not cause any loss of optimality in the PUT.

4.3.1 Simplified PDRPs

In this section we introduce a set of policies $\mathcal{Q}_S \subseteq \mathcal{Q}_H$ of the form $q_t^s(y_t|x_t, x_{t-1}, y^{t-1})$, which samples the distorted data only by considering the true data in the last two time instances and the entire released data history. Hence, the joint distribution (4.4) induced by $\boldsymbol{q}_s \in \mathcal{Q}_S$, where $\boldsymbol{q}_s = \{q_t^s(y_t|x_t, x_{t-1}, y^{t-1})\}_{t=1}^n$ can be written as

$$P^{\boldsymbol{q}_s}(X^n = x^n, Y^n = y^n) = p_{x_1} q_1^s(y_1|x_1) \prod_{t=2}^{n} \left[q_x(x_t|x_{t-1}) q_t^s(y_t|x_t, x_{t-1}, y^{t-1}) \right]. \tag{4.5}$$

Next, we show that considering PDRPs in set $\mathcal{Q}_S$ is without loss of optimality.

Theorem 4.1. *In both minimization problems (4.2) and (4.3), there is no loss of optimality in restricting the PDRPs to the set of policies $\boldsymbol{q}_s \in \mathcal{Q}_S$. Furthermore, information leakage induced by any $\boldsymbol{q}_s \in \mathcal{Q}_S$ can be written as:*

$$\begin{aligned} I^{\boldsymbol{q}_s}(X^n, Y^n) &= \sum_{t=1}^{n} I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t|Y^{t-1}) && (4.6)\\ &= \sum_{t=1}^{n} \sum_{\substack{y^t \in \mathcal{W}^t \\ x_t, x_{t-1} \in \mathcal{W}}} P^{\boldsymbol{q}_s}(x_t, x_{t-1}, y^t) \log \frac{q_t^s(y_t|x_t, x_{t-1}, y^{t-1})}{P^{\boldsymbol{q}_s}(y_t|y^{t-1})}, && (4.7) \end{aligned}$$

and the average distortion induced by any $\boldsymbol{q}_s \in \mathcal{Q}_S$ can be written as:

$$\begin{aligned} \mathbb{E}^{\boldsymbol{q}_s}\left[\frac{1}{n}\sum_{t=1}^{n} d(X_t, Y_t)\right] &= \frac{1}{n}\sum_{t=1}^{n} \mathbb{E}^{\boldsymbol{q}_s}[d(X_t, Y_t)] && (4.8)\\ &= \frac{1}{n}\sum_{t=1}^{n} \sum_{y_t, x_t \in \mathcal{W}} P^{\boldsymbol{q}_s}(x_t, y_t) d(x_t, y_t), && (4.9) \end{aligned}$$

where the first equation comes from the linearity of expectation.

The proof of Theorem 4.1 relies on the following lemmas and will be presented later.

Lemma 4.2. *For any $\boldsymbol{q} \in \mathcal{Q}_H$,*

$$I^{\boldsymbol{q}}(X^n; Y^n) \geq \sum_{t=1}^{n} I^{\boldsymbol{q}}(X_t, X_{t-1}; Y_t | Y^{t-1}) \tag{4.10}$$

with equality if and only if $\boldsymbol{q} \in \mathcal{Q}_S$.

Proof: For any $\boldsymbol{q} \in \mathcal{Q}_H$,

$$\begin{aligned} I^{\boldsymbol{q}}(X^n; Y^n) &= \sum_{t=1}^{n} I^{\boldsymbol{q}}(X^t; Y_t | Y^{t-1}) && (4.11) \\ &\geq \sum_{t=1}^{n} I^{\boldsymbol{q}}(X_t, X_{t-1}; Y_t | Y^{t-1}), && (4.12) \end{aligned}$$

where (4.11) follows from (4.1), and (4.12) from the fact that MI cannot be negative. ■

Lemma 4.3. *For any $\boldsymbol{q}_h \in \mathcal{Q}_H$, there exists a policy $\boldsymbol{q}_s \in \mathcal{Q}_S$ such that*

$$\sum_{t=1}^{n} I^{\boldsymbol{q}_h}(X_t, X_{t-1}; Y_t | Y^{t-1}) = \sum_{t=1}^{n} I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t | Y^{t-1}), \tag{4.13}$$

for both cases where $\boldsymbol{q}_h$ and $\boldsymbol{q}_s$ satisfy an instantaneous distortion constraint $d(X_t, Y_t) \leq \hat{D}$, and average distortion constraints $\mathbb{E}^{\boldsymbol{q}_h}\left[\frac{1}{n}\sum_{t=1}^{n} d(X_t, Y_t)\right] \leq \bar{D}$ and $\mathbb{E}^{\boldsymbol{q}_s}\left[\frac{1}{n}\sum_{t=1}^{n} d(X_t, Y_t)\right] \leq \bar{D}$, respectively.

Proof: For any $\boldsymbol{q}_h \in \mathcal{Q}_H$, we choose the policy $\boldsymbol{q}_s \in \mathcal{Q}_S$ such that

$$q_t^s(y_t | x_t, x_{t-1}, y^{t-1}) = P^{\boldsymbol{q}_h}_{Y_t | X_t, X_{t-1}, Y^{t-1}}(y_t | x_t, x_{t-1}, y^{t-1}), \tag{4.14}$$

and we show that $P^{\boldsymbol{q}_h}_{X_t, X_{t-1}, Y^t} = P^{\boldsymbol{q}_s}_{X_t, X_{t-1}, Y^t}$. Then, $I^{\boldsymbol{q}_h}(X_t, X_{t-1}; Y_t | Y^{t-1}) = I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t | Y^{t-1})$ holds, which proves the statement in Lemma 4.3. The proof of the equality $P^{\boldsymbol{q}_h}_{X_t, X_{t-1}, Y^t} = P^{\boldsymbol{q}_s}_{X_t, X_{t-1}, Y^t}$ requires the proof of $P^{\boldsymbol{q}_h}_{X_t, X_{t-1}, Y^{t-1}} = P^{\boldsymbol{q}_s}_{X_t, X_{t-1}, Y^{t-1}}$ which is derived by induction as follows,

$$\begin{aligned} P^{\boldsymbol{q}_h}(x_{t+1}, x_t, y^t) &= \sum_{x_{t-1} \in \mathcal{W}} q_x(x_{t+1} | x_t) q_t^h(y_t | x_t, x_{t-1}, y^{t-1}) P^{\boldsymbol{q}_h}(x_t, x_{t-1}, y^{t-1}) \\ &= \sum_{x_{t-1} \in \mathcal{W}} q_x(x_{t+1} | x_t) q_t^s(y_t | x_t, x_{t-1}, y^{t-1}) P^{\boldsymbol{q}_s}(x_t, x_{t-1}, y^{t-1}) \\ &= P^{\boldsymbol{q}_s}(x_{t+1}, x_t, y^t), && (4.15) \end{aligned}$$

where (4.14) holds, and $P^{\boldsymbol{q}_h}_{X_1}(x) = p_{x_1}(x) = P^{\boldsymbol{q}_s}_{X_1}(x)$ is used for the initialization of the induction.

Having shown that the equality $P^{\boldsymbol{q}_h}_{X_t,X_{t-1},Y^{t-1}} = P^{\boldsymbol{q}_s}_{X_t,X_{t-1},Y^{t-1}}$ and (4.14) hold, the proof of $P^{\boldsymbol{q}_h}_{X_t,X_{t-1},Y^t} = P^{\boldsymbol{q}_s}_{X_t,X_{t-1},Y^t}$ is straightforward:

$$\begin{aligned} P^{\boldsymbol{q}_h}(x_t, x_{t-1}, y^t) &= q_t^h(y_t|x_t, x_{t-1}, y^{t-1}) P^{\boldsymbol{q}_h}(x_t, x_{t-1}, y^{t-1}) \\ &= q_t^s(y_t|x_t, x_{t-1}, y^{t-1}) P^{\boldsymbol{q}_s}(x_t, x_{t-1}, y^{t-1}) \\ &= P^{\boldsymbol{q}_s}(x_t, x_{t-1}, y^t). \end{aligned} \tag{4.16}$$

Following (4.16), the equality $I^{\boldsymbol{q}_h}(X_t, X_{t-1}; Y_t|Y^{t-1}) = I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t|Y^{t-1})$ holds, and the integration of the instantaneous distortion constraint into the additive MI is straightforward and does not affect the optimality, and hence, (4.13) holds.

Furthermore, we show that there is no loss of optimality in including the average distortion constraint into the MI optimization when the policy is chosen according to (4.14), as follows:

$$\mathbb{E}^{\boldsymbol{q}_h}[d(X_t, Y_t)] = \sum_{\substack{y^t \in \mathcal{W}^t, \\ x_t, x_{t-1} \in \mathcal{W}}} P^{\boldsymbol{q}_h}(x_t, x_{t-1}, y^t) d(x_t, y_t) \tag{4.17}$$

$$= \sum_{\substack{y^t \in \mathcal{W}^t, \\ x_t, x_{t-1} \in \mathcal{W}}} P^{\boldsymbol{q}_s}(x_t, x_{t-1}, y^t) d(x_t, y_t), \tag{4.18}$$

$$= \sum_{y_t, x_t \in \mathcal{W}} P^{\boldsymbol{q}_s}(x_t, y_t) d(x_t, y_t), \tag{4.19}$$

$$= \mathbb{E}^{\boldsymbol{q}_s}[d(X_t, Y_t)] \tag{4.20}$$

where (4.17) follows from the history independence of $d(X_t, Y_t)$, (4.18) follows from (4.16), and (4.19) from history-independence of $d(x_t, y_t)$. Following the linearity of expectation, the average distortion constraint can be written in an additive form, and hence, (4.13) holds. ■

Proof of Theorem 4.1: Following Lemmas 4.2 and 4.3, for any $\boldsymbol{q}_h \in \mathcal{Q}_H$, there exists a $\boldsymbol{q}_s \in \mathcal{Q}_S$ such that

$$I^{\boldsymbol{q}_h}(X^n; Y^n) \geq I^{\boldsymbol{q}_s}(X^n; Y^n). \tag{4.21}$$

Hence, there is no loss of optimality in using the time-series data release policies of the form $q_t^s(y_t, |x_t, x_{t-1}, y^{t-1})$, and information leakage and the average distortion constraint reduce to (4.7) and (4.9), respectively. ■

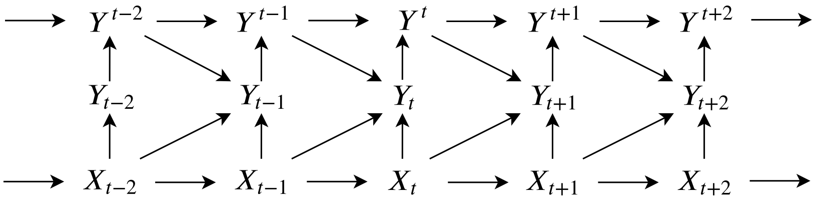


Figure 4.2: Markov chain induced by the simplified PDRP.

4.3.1.1 m^{th} Order Markov Chain

Although the proof of Theorem 4.1 assumes that the true data sequence is a first-order Markov chain, it is possible to generalize it to higher-order Markov chains, i.e., $q_x(X_t|X^{t-1}) = q_x(X_t|X_{t-m}^{t-1})$ for order m. Let $\boldsymbol{Q}_S^m \subseteq \boldsymbol{Q}_H$ denote the set of policies $\boldsymbol{q}'$

$$q_t'(y_t|x_{t-m}^t, y^{t-1}) = P^{\boldsymbol{q}'}_{Y_t|X_{t-m}^t, Y^{t-1}}(y_t|x_{t-m}^t, y^{t-1}). \tag{4.22}$$

Then the following theorem holds.

Theorem 4.4. *If the true data sequence $\{X_t\}$ is a Markov chain of order m, then there is no loss of optimally in using a PDRP from the set $\boldsymbol{Q}_S^m$. Moreover, information leakage induced by $\boldsymbol{q}' \in \boldsymbol{Q}_S^m$ can be written as:*

$$I^{\boldsymbol{q}'}(X^n, Y^n) = \sum_{t=1}^{n} I^{\boldsymbol{q}'}(X_{t-m+1}^t; Y_t|Y^{t-1}), \tag{4.23}$$

and the average distortion induced by any $\boldsymbol{q}' \in \mathcal{Q}_S^m$ can be written as:

$$\mathbb{E}^{\boldsymbol{q}'}\left[\frac{1}{n}\sum_{t=1}^{n} d(X_t, Y_t)\right] = \sum_{t=1}^{n} \sum_{y_t, x_t \in \mathcal{W}} P^{\boldsymbol{q}'}(x_t, y_t) d(x_t, y_t). \tag{4.24}$$

Then the simplified PDRP followed by the user is illustrated by the Markov chain in Figure 4.2, where Y^t denotes the released data history, i.e., $\{Y_1, \ldots, Y_t\}$. That is, the user samples the distorted data, Y_t, at time t following $q_t^s(y_t|x_t, x_{t-1}, y^{t-1})$ by considering the current and previous true data, (X_t, X_{t-1}), and the released data history, Y^{t-1}.

4.3.2 Online PDRP with an Instantaneous Distortion Constraint

As we have stated earlier, we are assuming that the utility gained by the user by sharing its private data diminishes as the distortion between the true data sequence and the released version increases, under the specified distortion measure. Therefore, the utility requirements of the user imposes distortion constraints on the PDPR. Here, we assume that the user would like to guarantee a minimum utility level at each time instant, which, in turn, imposes an instantaneous constraint on the distortion between the true data sample X_t and the released version Y_t at each time instance, i.e., $d(X_t, Y_t) \leq \hat{D}$, $\forall t$.

Accordingly, given $(X_t, X_{t-1}, Y^{t-1}) = (x_t, x_{t-1}, y^{t-1})$, the set of feasible simplified PDRPs satisfying an instantaneous distortion constraint is $\boldsymbol{q}_s^I \in \boldsymbol{Q}_S^I$, and the set of the released data samples induced by $\boldsymbol{q}_s^I$ is given by

$$\mathcal{Y}^{\boldsymbol{q}_s^I}(x_{t-1}^t, y^{t-1}) := \Big\{ y_t \in \mathcal{W} : d(x_t, y_t) \leq \hat{D} \Big\}. \tag{4.25}$$

Furthermore, we require $\boldsymbol{q}_s^I$ to satisfy

$$\sum_{y_t \in \mathcal{Y}^{\boldsymbol{q}_s^I}(x_{t-1}^t, y^{t-1})} q_s^I(y_t | x_{t-1}^t, y^{t-1}) = 1. \tag{4.26}$$

The objective of the PUT for online PDRP with an instantaneous distortion constraints (PDRP-IDC) can be rewritten as

$$\min_{q_s^I(y_t | x_{t-1}^t, y^{t-1})} \frac{1}{n} \sum_{t=1}^{n} I^{\boldsymbol{q}_s^I}(X_t, X_{t-1}; Y_t | Y^{t-1}). \tag{4.27}$$

4.3.3 Online PDRP with an Average Distortion Constraint

Alternatively, the user may want to limit only the average distortion applied to the true-data sequence. That is, the utility loss averaged over the time horizon n is denoted by $D(X^n; Y^n) = \mathbb{E}^{q_s^A}[\frac{1}{n} \sum_{t=1}^n d(X_t, Y_t)]$. The feasible set of simplified PDRPs with an average distortion constraint is $\boldsymbol{q}_s^A \in \boldsymbol{Q}_S^A$, and the feasible set of the released Y_t induced by $\boldsymbol{q}_s^A$ is given by

$$\mathcal{Y}^{\boldsymbol{q}_s^A}(x_{t-1}^t, y^{t-1}) := \Big\{ y_t \in \mathcal{W} : D(x^n, y^n) \leq \bar{D} \Big\}, \tag{4.28}$$

where the constraint follows from the linearity of expectation, i.e., $D(X^n; Y^n) = \frac{1}{n} \sum_{t=1}^n \mathbb{E}^{\boldsymbol{q}_s^A}[d(X_t, Y_t)]$, and the expectation is taken over the joint probabilities of x_t and y_t. Similarly to (4.25), $\boldsymbol{q}_s^A$ is required to satisfy

$$\sum_{y_t \in \mathcal{Y}^{\boldsymbol{q}_s^A}(x_{t-1}^t, y^{t-1})} q_s^A(y_t | x_{t-1}^t, y^{t-1}) = 1. \tag{4.29}$$

Hence, the objective of the problem for online PDRP with an average distortion constraint (PDRP-ADC) can be written as:

$$\min_{q_s^A(y_t|x_{t-1}^t, y^{t-1})} \frac{1}{n} \sum_{t=1}^{n} I^{\boldsymbol{q}_s^A}(X_t, X_{t-1}; Y_t | Y^{t-1}). \tag{4.30}$$

Minimization of the MI subject to a distortion constraint can be converted into an unconstrained minimization problem using Lagrange multipliers. Since the distortion constraint induced by the simplified PDRP is memoryless, we can integrate it into the additive MI objective easily. Hence, the unconstrained minimization problem for time-series data release PUT can be rewritten as

$$\min_{\boldsymbol{q}_s \in \boldsymbol{Q}_s} \frac{1}{n} \sum_{t=1}^{n} \left[I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t | Y^{t-1}) + \lambda(\mathbb{E}^{\boldsymbol{q}_s}[d(X_t, Y_t)] - \bar{D}) \right], \tag{4.31}$$

where λ is the Lagrangian multiplier, and determines the operating point on the trade-off curve, i.e., it represents where the gradients of the MI and the distortion constraint point in the same direction. When $\lambda = 0$, the user releases data samples which only minimize the information leakage. On the other hand, as $\lambda \to \infty$, the released data minimizes only distortion constraint rather than information leakage, which results in full information leakage.

In the following section, we present the MDP formulation of the problem for both PDRPs and the evaluation method utilized by advantage actor-critic RL.

4.4 MDP Formulation

Markovity of the user's true data sequence and the additive objective functions in both (4.27) and (4.31) allow us to represent the problem as an MDP with state X_t. However, the information leakage at time t depends on Y^{t-1}, resulting in a growing state space in time. Therefore, for a given policy $\boldsymbol{q}_s$ and any realization y^{t-1} of Y^{t-1}, we define a belief state $\beta_t \in \mathcal{P}_X$ as a probability distribution over the state space:

$$\beta_t(x_{t-1}) = P^{\boldsymbol{q}_s}(X_{t-1} = x_{t-1} | Y^{t-1} = y^{t-1}). \tag{4.32}$$

This represents the SP's belief on the true data sample at the beginning of time instance t, i.e., after receiving the distorted-data y_{t-1}. The actions are defined as probability distributions with which the user samples the released value Y_t at time t and determined by the randomized PDRPs. The user's action induced by a policy $\boldsymbol{q}_s$ can be denoted by $a_t(y_t|x_t, x_{t-1}) = P^{\boldsymbol{q}_s}(Y_t = y_t|X_t = x_t, X_{t-1}, \beta_t)$. At each time t, the SP updates its belief on the true data sample $\beta_{t+1}(x_t)$, after observing its distorted version y_t by

$$\begin{aligned}\beta_{t+1}(x_t) &= \frac{p(x_t, y_t|y^{t-1})}{p(y_t|y^{t-1})} = \frac{\sum_{x_{t-1}} p(x_t, x_{t-1}, y_t|y^{t-1})}{\sum_{x_t, x_{t-1}} p(x_t, x_{t-1}, y_t|y^{t-1})} \\ &= \frac{\sum_{x_{t-1}} p(x_t|x_{t-1}) q_t^s(y_t|x_t, x_{t-1}, y^{t-1}) p(x_{t-1}|y^{t-1})}{\sum_{x_t, x_{t-1}} p(x_t|x_{t-1}) q_t^s(y_t|x_t, x_{t-1}, y^{t-1}) p(x_{t-1}|y^{t-1})} \\ &= \frac{\sum_{x_{t-1}} q_x(x_t|x_{t-1}) a(y_t|x_t, x_{t-1}) \beta_t(x_{t-1})}{\sum_{x_t, x_{t-1}} q_x(x_t|x_{t-1}) a(y_t|x_t, x_{t-1}) \beta_t(x_{t-1})}. \end{aligned} \tag{4.33}$$

We define the per-step information leakage of the user due to taking the action $a_t(y_t|x_t, x_{t-1})$ at time t as,

$$l_t(x_t, x_{t-1}, a_t, y^t; \boldsymbol{q}_s) := \log \frac{a_t(y_t|x_t, x_{t-1})}{P^{\boldsymbol{q}_s}(y_t|y^{t-1})}. \tag{4.34}$$

The expectation of n-step sum of (4.34) over the joint probability $P^{\boldsymbol{q}_s}(X_t, X_{t-1}, Y^t)$ is equal to the MI expression in the original problem (4.6). Therefore, given the belief and action probabilities, average information leakage at time t can be formulated as,

$$\begin{aligned}\mathbb{E}^{\boldsymbol{q}_s}[l_t(x_{t-1}^t, a_t, y^t)] = &\sum_{x_t, x_{t-1}, y_t \in \mathcal{W}} \beta_t(x_{t-1}) a_t(y_t|x_t, x_{t-1}) q_x(x_t|x_{t-1}) \\ &\times \log \frac{a_t(y_t|x_t, x_{t-1})}{\sum\limits_{\hat{x}_t, \hat{x}_{t-1} \in \mathcal{W}} \beta_t(\hat{x}_{t-1}) a_t(y_t|\hat{x}_t, \hat{x}_{t-1}) q_x(\hat{x}_t|\hat{x}_{t-1})} \\ &:= \mathcal{L}(\beta_t, a_t). \end{aligned} \tag{4.35}$$

We can recast the PDRP-IDC problem in (4.27) as a continuous state and action space MDP. The actions satisfying the instantaneous distortion constraint are denoted by $a_t^{\text{IDC}}(y_t|x_t, x_{t-1})$ and induced by the simplified PDRP $q_s^I(y_t|x_{t-1}^t, y^{t-1})$. The solution of the MDP for PDRP-IDC problem relies on minimizing the objective

$$\mathcal{C}_{\text{IDC}}(\beta_t, a_t^{\text{IDC}}) := \mathcal{L}(\beta_t, a_t^{\text{IDC}}), \tag{4.36}$$

where $\mathcal{L}(\beta_t, a_t^{\text{IDC}})$ is the average information leakage obtained by taking the actions $a_t^{\text{IDC}}(y_t|x_t, x_{t-1})$, at each time step t.

We remark that the representation of average distortion in terms of belief and action probabilities is straightforward due to its additive form. Similarly to (4.35), average distortion for PDRP-ADC at time t can be written as,

$$\begin{aligned}\mathbb{E}^{\boldsymbol{q}_s}[d(x_t, y_t)] &= \sum_{x_t, x_{t-1}, y_t \in \mathcal{W}} \beta_t(x_{t-1}) a_t(y_t|x_t, x_{t-1}) q_x(x_t|x_{t-1}) d(x_t, y_t) \\ &:= \mathcal{D}(\beta_t, a_t), \end{aligned} \tag{4.37}$$

where there is no restriction on how the actions are chosen, i.e., $y_t \in \mathcal{W}$. Hence, we can recast the PDRP-ADC problem in (4.31) as a continuous state and action space MDP with a per-step cost function given by

$$\mathcal{C}_{\text{ADC}}(\beta_t, a_t) := \mathcal{L}(\beta_t, a_t) + \lambda(\mathcal{D}(\beta_t, a_t) - \bar{D}). \tag{4.38}$$

Finding optimal policies for continuous state and action space MDPs is a PSPACE-hard problem [115]. In practice, they can be solved by various finite-state MDP evaluation methods, e.g., value iteration, policy iteration and gradient-based methods. These are based on the discretization of the continuous belief states to obtain a finite state MDP [68]. While finer discretization of the belief reduces the loss from the optimal solution, it causes an increase in the dimension of the state space; hence, in the complexity of the problem. To overcome the complexity limitation, we will employ a deep learning based method as a tool to numerically solve our continuous state and action space MDP problem.

4.4.1 A2C-DRL Solution

In this section, we simply use $\mathcal{C}(\beta_t, a_t)$ and $a_t(y_t|x_t, x_{t-1})$ to represent the MDP cost and action pair of both PDRP-IDC and PDRP-ADC, respectively. Integration of the solution into the instantaneous and average distortion constrained cases is straightforward.

A2C-DRL is explained in Section 2.2.1 in detail. In this chapter, we have the knowledge of the state transition probabilities and the cost for every state-action pair without the need for interacting with the environment. We use A2C-DRL as a computational tool to numerically evaluate the optimal PDRP for our continuous state and action space MDP. To integrate RL framework into our problem, we create an artificial environment which inputs the user's current action, $a_t(y_t|x_t, x_{t-1})$, samples an observation y_t, and calculates the next state, β_{t+1}, using Bayesian belief update (4.33). Instantaneous cost revealed by the environment is calculated by (4.38). The user receives the experience tuple $(\beta_t, a_t, y_t, \beta_{t+1}, \mathcal{C}_t)$ from the environment, and refines her policy accordingly. An illustration of the interaction between the artificial environment and the user, which is represented by the RL agent, is presented in Figure 2.1. The corresponding Bellman

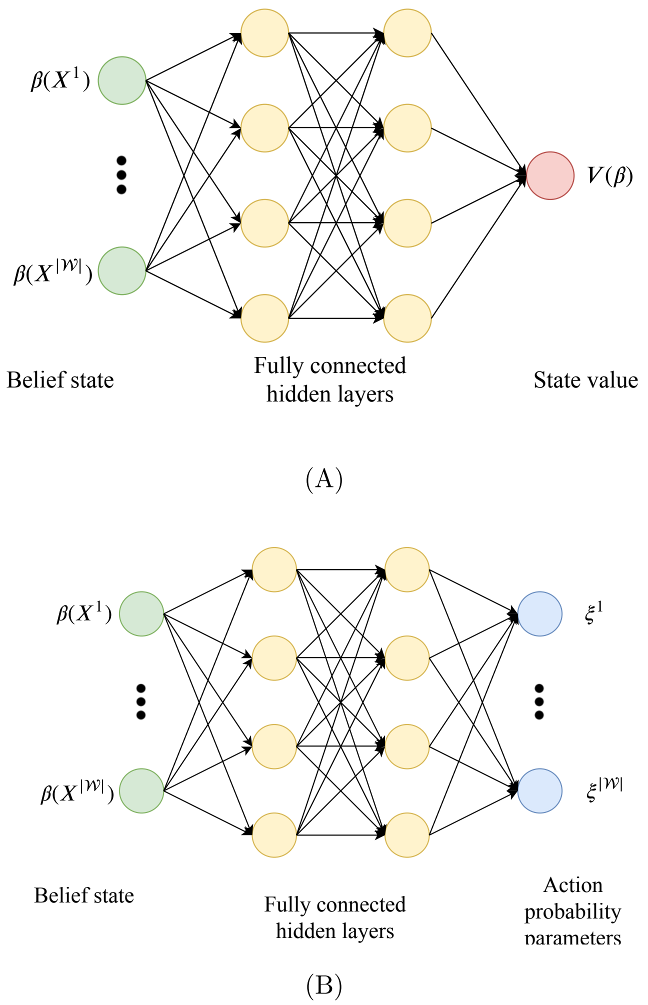

$\beta(X^1)$
$\beta(X^{|\mathcal{W}|})$
$V(\beta)$
Belief state
Fully connected hidden layers
State value
(A)
$\beta(X^1)$
$\beta(X^{|\mathcal{W}|})$
ξ^1
$\xi^{|\mathcal{W}|}$
Belief state
Fully connected hidden layers
Action probability parameters
(B)

FIGURE 4.3: Critic (A) and actor (B) DNN structures.

equation induced by policy $\boldsymbol{q}_s$

$$V^{\boldsymbol{q}_s}(\beta) + J(\boldsymbol{q}_s) = \min_{a} \Big\{ \mathcal{C}(\beta, a) + V^{\boldsymbol{q}_s}(\beta') \Big\}, \tag{4.39}$$

where $V^{\boldsymbol{q}_s}(\beta)$ is the state-value function, β' is the updated belief state according to (4.33), a represents action probability distributions, and $J(\boldsymbol{q}_s)$ is the cost-to-go function, i.e., the expected future cost induced by policy $\boldsymbol{q}_s$ [67].

We solve the MDP using A2C-DRL as described in 2.2.1. In our implementation, we represent the actor and critic mechanisms by fully connected feed-forward DNNs with two hidden layers as illustrated in Figure 4.3. The critic DNN takes the current belief state $\beta(\boldsymbol{X})$ of size $|\mathcal{W}|$ as input, where $\boldsymbol{X}$ is the true data sequence vector, and outputs the value of the belief state for the current action probabilities $V_\theta^\xi(\beta)$. The actor DNN also takes the current belief state $\beta(\boldsymbol{X})$ as input, and outputs the parameters used for determining the action probabilities of the corresponding belief. Hence, the input/output

Algorithm 1 A2C-DRL algorithm for PDRP

```
Initialize DNNs with random weights $\xi$ and $\theta$
Initialize environment $E$
for $episode$=$1, N$ do
    Initialize belief state $\beta_0$ for $t = 0, n$ do
        Sample action $a_t \sim Dirichlet(a|\xi_t)$ according to current policy;
        Perform action $a_t$ and calculate cost $\mathcal{C}_{\xi_t}$ in $E$;
        Sample observation $y_t$ and calculate next belief state $\beta_{t+1}$ in $E$;
        Set TD target $\mathcal{C}_{\xi_t} + \gamma V^{\xi}_{\theta_t}(\beta_{t+1})$;
        Minimize loss $\ell_c(\theta) = \delta^2 = (\mathcal{C}_{\xi_t} + \gamma V^{\xi}_{\theta_t}(\beta_{t+1}) - V^{\xi}_{\theta_t}(\beta_t))^2$;
        Update critic $\theta \leftarrow \theta + \eta^c \nabla_\theta \delta^2$;
        Minimize loss $\ell_a(\xi_t) = \ln(Dirichlet(a|\xi_t))\delta_t$;
        Update actor $\xi \leftarrow \xi - \eta^a \nabla_\xi \ell_a(\xi_t)$;
        Update belief state $\beta_{t+1} \leftarrow \beta_t$
    end
end
```

sizes of the critic and actor DNNs are $|\mathcal{W}| \times 1$ and $|\mathcal{W}| \times |\mathcal{W}|$, respectively. Here, the actor DNN output parameters $\{\xi^1, \ldots, \xi^{|\mathcal{W}|}\}$ are used to generate a Dirichlet distribution, which represents the action probabilities. The overall A2C-DRL algorithm for online PDRP is described in Algorithm 1. In the next section, we apply the proposed DRL solution to a location trace privacy problem.

4.5 Numerical Results

In this section, we consider an application of the theoretical framework we have introduced to the location trace privacy problem. We focus on location trace as an example of time-series data. In this scenario, the user shares a distorted version of her trajectory with the SP due to privacy concerns. An example for the user trajectory of length $n = 5$ in a grid area is illustrated in Figure 4.4. While the user's location at time $t = 0$ is depicted with a grey circle, the true and released user trajectories over the next 5 time steps are represented by black and grey arrows, respectively.

4.5.1 Numerical Results for Synthetic Data

In this section, we evaluate the PUT of the proposed PDRP-ADC and PDRP-IDC methods for synthetic user mobility data. We also compare the PDRP-ADC results with the myopic Markovian location release mechanism proposed in [47]. For the simulation results we train two fully connected feed-forward DNNs, representing the actor and critic networks, respectively, by utilizing ADAM optimizer [116]. Both networks contain two

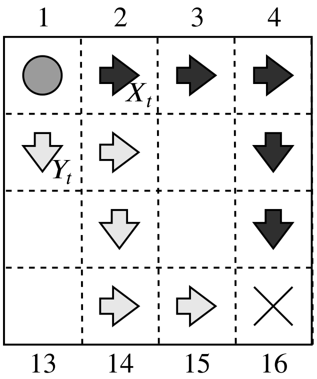


FIGURE 4.4: True and released user trajectory example for $n = 5$.

hidden layers of sizes 3000 with leaky-ReLU activation [117]. We obtain the corresponding PUT by averaging the total information leakage for the specified distortion constraint over a time horizon of $n = 300$.

4.5.1.1 PDRP-IDC Results

We first consider a simple 4×4 grid-world, where $|\mathcal{W}| = 16$ as in Figure 4.4. The cells are numbered such that the first and the last rows of the grid-world are represented by $\{1, 2, 3, 4\}$ and $\{13, 14, 15, 16\}$, respectively. The user's trajectory forms a first-order Markov chain with a transition probability matrix $\boldsymbol{Q}_x$ of size $|\mathcal{W}| \times |\mathcal{W}|$, whose index $Q_x(i, j)$, $i, j \in \{1, \ldots, |\mathcal{W}|\}$, represents the transition probability $q_x(x_t = i|x_{t-1} = j)$ from the state j to i. The user can start its movement at any square with equal probability, i.e., $p_{x_1} = \frac{1}{16}$. Our goal is to obtain the PUT under instantaneous distortion constraints $\hat{D} \in \{1, \ldots, 4\}$ with Manhattan distance on the distortion measure between the true position and the reported one.

In Figure 4.5, PUT curves are obtained for transition probability matrices $\boldsymbol{Q}_x^0$, $\boldsymbol{Q}_x^1$ and $\boldsymbol{Q}_x^2$, each corresponding to a different temporal correlation level. In all the cases, the user can move from any square to any other square in the grid at each step, i.e., $Q_x^m(i, j) > 0$, $\forall m, i, j$. While all the transition probabilities are equal to $\frac{1}{|\mathcal{W}|}$ for $\boldsymbol{Q}_x^0$, the probability of the user moving to a nearby square is greater than taking a larger step to a more distant one for $\boldsymbol{Q}_x^1$ and $\boldsymbol{Q}_x^2$. Moreover, $\boldsymbol{Q}_x^1$ represents a more uniform trajectory, where the agent moves to equidistant cells with equal probability, while with $\boldsymbol{Q}_x^2$ the agent is more likely to follow a certain path, i.e., the random trajectory generated by $\boldsymbol{Q}_x^2$ has lower entropy. The transition probabilities for $\boldsymbol{Q}_x^1$ are given by:

$$q_x^1(x_t|x_{t+1}) = \frac{r_{d(x_t,x_{t+1})}/d(x_t, x_{t+1})}{\sum_{x_{t+1} \in \mathcal{W}} r_{d(x_t,x_{t+1})}/d(x_t, x_{t+1})}, \tag{4.40}$$

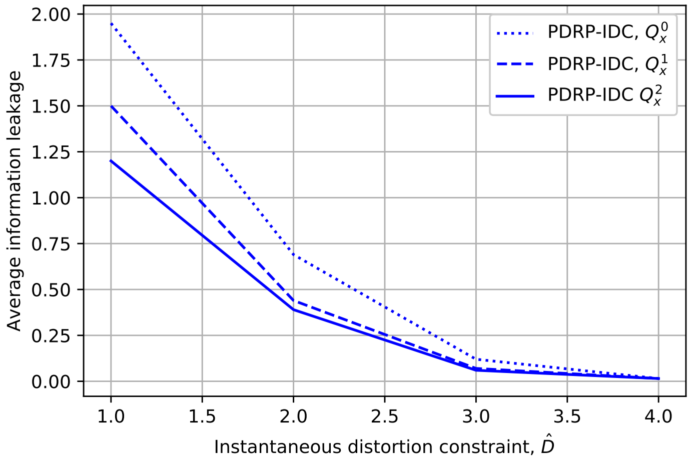


FIGURE 4.5: Average information leakage as a function of the allowed instantaneous distortion under Manhattan distance as the distortion measure.

where $d(x_t, x_{t+1})$ is the Manhattan distance between positions x_t and x_{t+1}; $r_{d(x_t,x_{t+1})}$ is a scalar which determines the probability of the user moving from one square to any of the equidistant squares in the next step. Figure 4.5 is obtained by setting $r_0 = 1$ and $r_i = 7 - i$, $i = 1, \dots, 6$.

For $\boldsymbol{Q}_x^2$, we set

$$q_x^2(x_t|x_{t+1}) = \frac{u(x_t, x_{t+1})/d(x_t, x_{t+1})}{\sum_{x_{t+1} \in \mathcal{W}} u(x_t, x_{t+1})/d(x_t, x_{t+1})}, \tag{4.41}$$

where, for $x_t \in \{1, 2, \dots, 15\}$, we have

$$u(x_t, x_{t+1}) = \begin{cases} r_1, & \text{for } \mathrm{mod}(x_t, 4) \neq 0,\ x_{t+1} = x_t + 1, \\ r_1, & \text{for } \mathrm{mod}(x_t, 4) = 0,\ x_{t+1} = x_t + 4, \\ r_0, & \text{otherwise}, \end{cases}$$

where mod(.) is the modulo operator which finds the remainder after division of x_t by 4, and $u(16, x_{t+1}) = r_0$ for $x_{t+1} \in \{1, \dots, 15\}$, and $u(16, 16) = r_1$. As a result, temporal correlations in the location history increase in the order $\boldsymbol{Q}_x^0$, $\boldsymbol{Q}_x^1$, $\boldsymbol{Q}_x^2$.

We train our DNNs for a time horizon of $n = 300$ in each episode, and over 5000 Monte Carlo roll-outs. Figure 4.5 shows that, information leakage increase in the order $\boldsymbol{Q}_x^2$, $\boldsymbol{Q}_x^1$, $\boldsymbol{Q}_x^0$. As the temporal correlations between the locations on a trace increases, the proposed PDRP-IDC leaks less information since it takes the entire released location history into account.

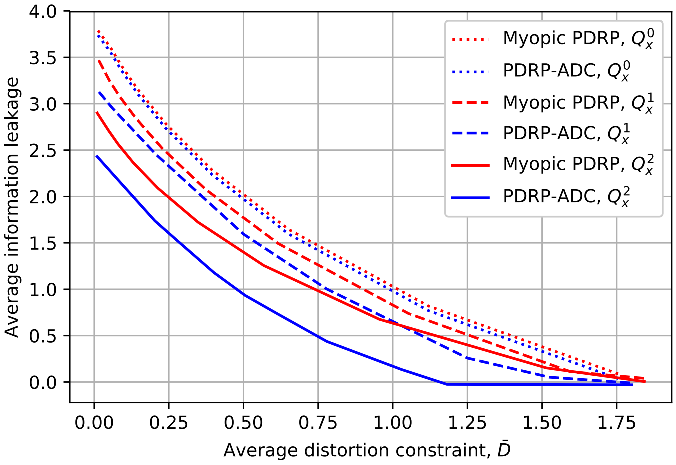


FIGURE 4.6: Average information leakage as a function of the allowed average distortion under Manhattan distance as the distortion measure.

4.5.1.2 PDRP-ADC Results

Next, we consider the same scenario as before, but evaluate the PUT under an average distortion constraint. We evaluate the performance of PDRP-ADC and compare the results with the myopic Markovian location release mechanism proposed in [47], where an upper bound on the PUT is given by a myopic policy as follows:

$$\sum_{t=1}^{n} \min_{\substack{q(y_t|x_t,x_{t-1},y_{t-1}): \\ \mathbb{E}^q[d(x_t,y_t)]\leq \bar{D}}} I^q(X_t, X_{t-1}; Y_t|Y_{t-1}). \tag{4.42}$$

Exploiting the fact that (4.42) is similar to the rate-distortion function, Blahut-Arimoto algorithm is used in [47] to minimize the conditional MI at each time step. Finite-horizon solution of the objective function (4.42) is obtained by applying alternating minimization sequentially. In our simulations, we obtained the average information leakage and distortion for this approach by normalizing for $n = 300$.

In Figure 4.6, PUT curves of the proposed PDRP-ADC and the myopic location release mechanism are obtained for the same environment defined in Section 4.5.1.1. The same transition matrices are used, i.e., $\boldsymbol{Q}_x^0$, $\boldsymbol{Q}_x^1$ and $\boldsymbol{Q}_x^2$ represent increasing temporal correlations in the user's trajectory. The Lagrangian multiplier $\lambda \in [0, 20]$ denotes the user's choice for the operating point on the PUT curve. Distortion is again measured by the Manhattan distance. Similarly to Section 4.5.1.1, we train our DNNs for $n = 300$ in each episode, and over 5000 Monte Carlo roll-outs. Figure 4.6 shows that, for $\boldsymbol{Q}_x^2$ the proposed PDRP-ADC obtained through deep RL leaks much less information than the myopic location release mechanism for the same distortion level, indicating the benefits of considering all the history when taking actions at each time instant. The gain is less

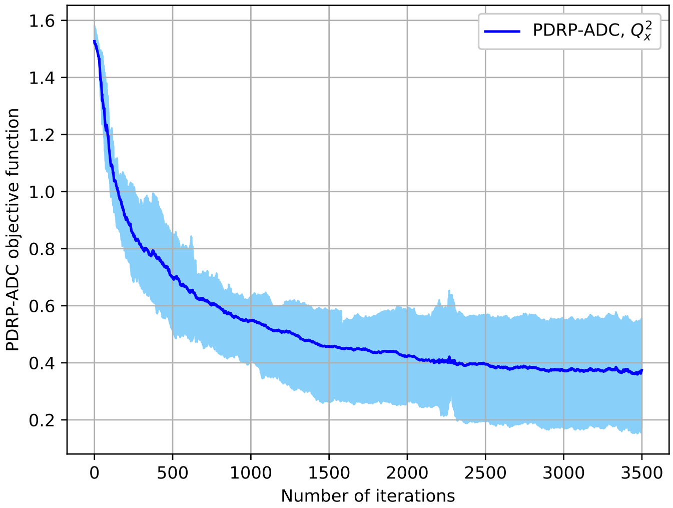


FIGURE 4.7: Convergence of PDRP-ADC for $\lambda = 1$, $\bar{D} = 0.8$ and $\boldsymbol{Q}_x^2$.

TABLE 4.2: The Transition Probability Matrix Q_x of Toy Example for PDRP-ADC, when $|\mathcal{W}| = 6$.

x_{t-1} \ x_t	1	2	3	4	5	6
1	0.11	0.64	0.05	0.11	0.05	0.04
2	0.1	0.1	0.6	0.05	0.1	0.05
3	0.05	0.11	0.11	0.04	0.05	0.64
4	0.11	0.05	0.04	0.11	0.64	0.05
5	0.05	0.1	0.05	0.1	0.1	0.6
6	0.04	0.05	0.11	0.05	0.11	0.64

for $\boldsymbol{Q}_x^1$, since there is less temporal correlations in the location history compared to $\boldsymbol{Q}_x^2$; and hence, there is less to gain from considering all the history when taking actions. Finally, for $\boldsymbol{Q}_x^0$ the proposed scheme and the myopic policy perform the same, since the user movement with uniform distribution does not have temporal memory; and therefore, taking the history into account does not help.

Figure 4.7 shows the convergence behaviour of the A2C-DRL algorithm when evaluating PDRP-ADC's objective function (4.31) for $\boldsymbol{Q}_x^2$, $\lambda = 1$, $\bar{D} = 0.8$. Various realizations of the convergence curve lie in the light blue area, and the dark blue curve represents the average value of these realizations. We observe that the convergence typically occurs after about 2500 iterations. On the other hand, we remark that the optimal policy for a stationary environment can be obtained in an offline manner using the available dataset; therefore the convergence time and the number of iterations has no impact on the real-time application of this solution in practice.

TABLE 4.3: Best Action Probabilities $a_t(y_t|x_t, x_{t-1})$ for $\boldsymbol{Q}_x$ in Table 4.2, $\beta = [\frac{1}{6}, \ldots, \frac{1}{6}]$ and $\lambda = 3$.

y_t x_t, x_{t-1}	1	2	3	4	5	6
(1,1)	0.19	0.06	0.22	0.18	0.23	0.12
(1,2)	0.21	0.19	0.28	0.09	0.06	0.17
(1,3)	0.19	0.13	0.18	0.19	0.28	0.03
(1,4)	0.3	0.24	0.17	0.07	0.07	0.15
(1,5)	0.03	0.05	0.51	0.01	0.25	0.15
(1,6)	0.22	0.14	0.13	0.16	0.21	0.14
⋮	⋮	⋮	⋮	⋮	⋮	⋮
(6,1)	0.03	0.07	0.21	0.21	0.32	0.16
(6,2)	0.18	0.13	0.35	0.1	0.16	0.08
(6,3)	0.21	0.08	0.18	0.12	0.13	0.28
(6,4)	0.18	0.05	0.19	0.36	0.14	0.08
(6,5)	0.31	0.14	0.3	0.07	0.16	0.02
(6,6)	0.09	0.29	0.21	0.16	0.01	0.24

We next consider a toy example for PDRP-ADC to visualize the location release strategy for a better understanding. We consider a 2×3 grid-world, where the user's trajectory forms a first-order Markov chain with the transition probability matrix $\boldsymbol{Q}_x$, given in Table 4.2. We assume that the user can start its movement at any square with equal probability, i.e., $p_{x_1} = \frac{1}{6}$. The Lagrange multiplier is chosen as $\lambda = 3$, and the distortion constraint is $\bar{D} = 0.6$. After training the actor and critic DNNs, we obtain the best action probabilities that minimize the objective function $\mathcal{C}_{\text{ADC}}$ in (4.38). Given the user's pattern in Table 4.2, $\beta = [\frac{1}{6}, \ldots, \frac{1}{6}]$ and $\lambda = 3$, the action distribution matrix induced by PDRP-ADC is obtained as in Table 4.3. It is clear from the table that Y_t is not released according to a deterministic pattern.

4.5.2 Numerical Results for GeoLife Dataset

Next, we present the simulation results on the GeoLife dataset [109, 110], which contains 182 user's GPS trajectories collected by Microsoft Research Asia. GeoLife trajectories are recorded densely, e.g., every $1 \sim 5$ seconds or every $5 \sim 10$ meters per point [110]. In our experiments, we focus on the high-density areas which represent the important stops for the users. Hence, we use a density-based data mining algorithm, namely DBSCAN (density-based spatial clustering of applications with noise) [118] to cluster the raw GPS data into the important stops of the user trajectory. We obtain a 16-cluster representation of the user-016's data, i.e., $\mathcal{W} = 16$, by applying DBSCAN algorithm to the 51 trajectories of user-016 provided in GeoLife dataset. For the implementation of

Table 4.4: Cross-entropy Loss of the Predictor for Certain PUT Levels of PDRP-IDC.

Instantaneous Distortion Constraint:			15 km	5 km	3 km
PDRP-IDC	Avg. Info. Leakage		0.18	0.39	0.53
	Cross-entropy Loss	m=1	1.05	0.66	0.52
		m=5	0.46	0.40	0.35

our MDP approach in the clustered dataset, center-points of the clusters represent user locations $X_t \in \mathcal{W}$, and the trajectories through the clusters represent user's state transitions. We use Euclidean distance between the true and released user cluster centers as the distortion measure.

Assuming that the user mobility forms a first-order Markov chain, we generate a transition probability matrix $\boldsymbol{\mathcal{Q}}_x^{016}$ from the user-016's trajectories. That is, we assume the user location X_t at time t depends only on the previous location X_{t-1}, and we find the empirical probabilities of transitions between locations. After the generation of $\boldsymbol{\mathcal{Q}}_x^{016}$, implementation of PDRP-IDC, PDRP-ADC or the myopic policy is the same as in the synthetic data case. To obtain the optimal policies, we train two fully connected feed-forward DNNs, representing the actor and critic networks, respectively, by using ADAM optimizer. Both networks contain two hidden layers each with 3000 nodes. While all the hidden layers have ReLU activation, the output layers of the actor and critic networks have tanh and Softmax activations, respectively. We obtain the PUT curves by averaging the total information leakage for the corresponding distortion constraint over a time horizon of $n = 600$ for 1000 Monte Carlo roll-outs.

Note that the MI computed based on the first-order Markov assumption, used by our approach to obtain the PDRP, may not correspond to the real information leakage. Since we do not know the underlying "true" statistics of the data, we examine the effectiveness of the proposed algorithms using an adversary which tries to predict the user's current true location from past released locations in an online manner. The predictor consists of an LSTM recurrent neural network layer with 200 nodes and a dropout of 0.5, which is followed by a fully connected hidden layer of 200 nodes with ReLu activation, and a fully connected output layer with Softmax activation. We train the predictor on the released distorted locations with the goal of minimizing the categorical cross-entropy between the estimated and true current locations by utilizing ADAM optimizer.

In Table 4.4, we show the adversary's cross-entropy loss for predicting user-016's true locations from their distorted versions released by PDRP-IDC at various PUT points. Here, m is the LSTM based adversary's look-back memory. For both $m = 1$ and $m = 5$, Table 4.4 shows that the cross-entropy loss decreases as the average information leakage increases. In Table 4.4, there is a decrease in the adversarial loss for $m = 5$

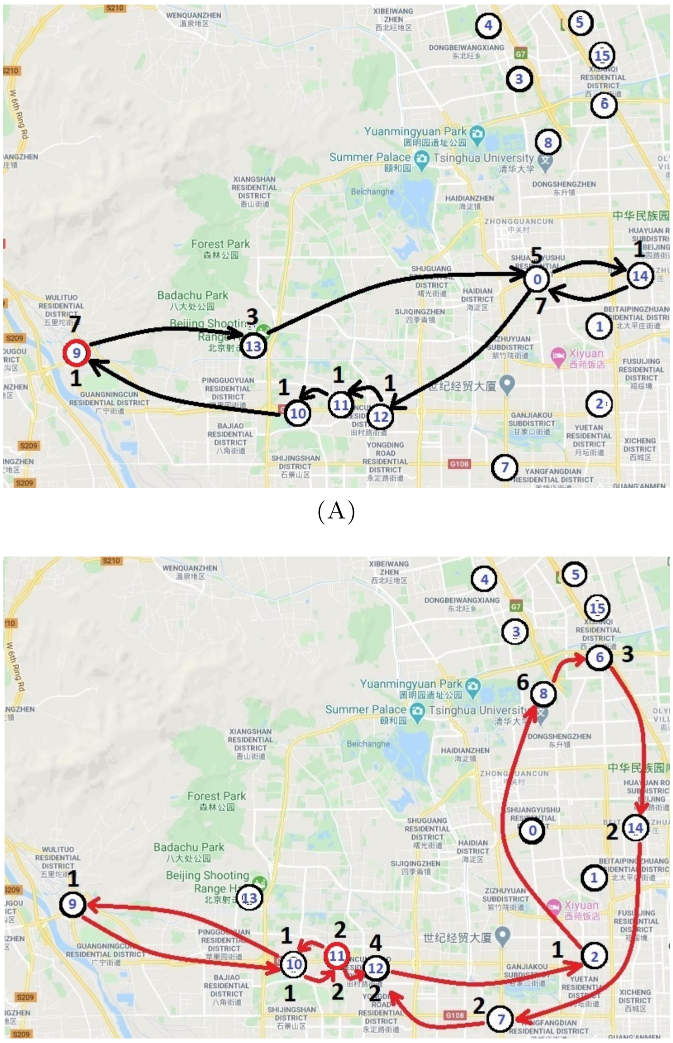
Table 4.5: Cross-entropy Loss of the Predictor for Certain PUT Levels of PDRP-ADC and Myopic Policy.

Average Distortion Constraint:			9 km	5.7 km	1.7 km
PDRP-ADC	Avg. Info. Leakage		0.11	0.20	0.35
	Cross-entropy Loss	m=1	1.30	1.25	0.90
		m=5	0.78	0.73	0.67
Myopic PDRP	Avg. Info. Leakage		0.27	0.33	0.50
	Cross-entropy Loss	m=1	1.10	0.99	0.82
		m=5	0.52	0.48	0.45

compared to $m = 1$, which means that the first-order Markov assumption may not be valid for the data as the adversary benefits from considering information further in the past. To understand the benefit of releasing distorted data better, we also obtained the cross-entropy loss of the adversary when it predicts the current location by observing the past true locations. When the privacy is not preserved, the adversary's cross-entropy loss is 0.36 for $m = 1$ and 0.28 for $m = 5$, which is much lower than the privacy preserved case as expected.

In Table 4.5, we show the adversary's prediction performance against PDRP-ADC and the myopic policy at various PUT points. For the same average distortion constraints, the adversary has higher cross-entropy loss of predicting true locations when they are distorted by PDRP-ADC rather than the myopic policy for both $m = 1$ and $m = 5$. Hence, considering the temporal correlations in the trajectory preserves PDRP-ADC's advantage over the myopic policy even when the adversary has a less strict Markov assumption on the true location distribution than both policies.

To understand the true and released location trajectories better, we provide a toy example in which we apply PDRP-ADC to previously clustered user-016 trajectories for $\mathcal{W} = 16$, $\lambda = 1$ and $\bar{D} = 5$km. An example for the true trajectory of the user is shown in Figure 4.8A, where the numbered circles are the cluster center-points with the corresponding cluster numbers in blue, black numbers represent how many steps the user takes in that cluster, the black arrows show the direction of the movement and the movement starts from the red circled cluster 9. For instance, Figure 4.8A represents the true trajectory $\{9, 9, 9, 9, 9, 9, 9, 13, 13, 13, 0, 0, 0, 0, 0, 14, 0, 0, 0, \ldots\}$. The distorted version of the trajectory in Figure 4.8A is depicted in Figure 4.8B, where the movement starts from the red circled cluster 11 and the red arrows show the direction of movement. The released trajectory can be deduced from the map in Figure 4.8B as $\{11, 11, 10, 9, 10, 11, 11, 12, 12, 12, 12, 2, 8, 8, 8, 8, 8, 8, 8, 6, \ldots\}$. These figures show that the released locations by PDRP-ADC follow a different path from the true locations for privacy concerns, while the distortion constraint is satisfied.



(A)

(B)

FIGURE 4.8: True (A) and the distorted (B) trajectory of user-016 by PDRP-ADC for $\mathcal{W} = 16$, $\lambda = 1$ and $\bar{D} = 5$km.

4.6 Conclusions

In this chapter, we have studied the PUT of time-series data using MI as a privacy measure. Having identified some properties of the optimal policy, we proposed information theoretically optimal online PDRPs under instantaneous and average distortion constraints, which represent utility constraints, and solved the PUT problem as an MDP. Due to continuous state and action spaces, it is challenging to characterize or even numerically compute the optimal policy. We overcome this difficulty by employing advantage actor-critic deep RL as a computational tool. Then, we applied the theoretical approach which we introduced for time-series data privacy into the location trace privacy problem. Utilizing DNNs, we numerically evaluated the PUT curve of the proposed PDRPs under both instantaneous and average distortion constraints for both synthetic data and GeoLife GPS trajectory dataset. We compared the results with the myopic location release policy introduced recently in [47], and observed the effect of considering temporal correlations on information leakage-distortion performance. We also examined the effectiveness of our Markov assumption by testing the proposed policies using an LSTM-based predictor network which represents the adversary with adjustable memory. According to the simulation results, we have seen that the proposed data release policies provide significant privacy advantage, especially when the user trajectory has higher temporal correlations. Even though higher privacy leakage was observed for larger adversary memory, proposed policies outperformed myopic policy.

Chapter 5

Active Privacy Against Inference

In this chapter, we consider a scenario in which a user releases her data containing personal information in return of a service from an honest-but-curious SP. In the previous chapters, we focused on the privacy leakage between the user data and its modified version. Here, instead, we model user's personal information as a time-series containing two correlated latent r.v.'s, one of them, called the *secret variable*, is to be kept private, while the other, called the *useful variable*, is to be disclosed in return of utility. We consider active sequential data release, where at each time step the user chooses from among a finite set of release mechanisms, each revealing some information about the user's personal information, i.e., the true values of the r.v.'s, albeit with different statistics. This differs from the scenarios in the previous chapters where the measurements were either physically modified or obfuscated with noise. In this chapter, the user's goal is to manage data release in an online fashion such that maximum amount of information is revealed about the latent useful variable, while the confidence for the sensitive variable is kept below a predefined level. For privacy measure, we consider both the probability of correctly detecting the true value of the secret and the MI between the secret and the released data. We formulate both problems as POMDPs, and numerically solve them by advantage A2C-DRL. We evaluate the PUT of the proposed policies on both the synthetic data and *smoking activity dataset* [119], and show their validity by testing the activity detection accuracy of the SP modeled by an LSTM neural network.

5.1 Introduction

In this chapter, we consider the PUT for time-series data sharing using active learning. We take into account the causal relations in time-series data for the privacy of the entire sequence. Among the limited number of works that consider temporal correlations in the literature, most existing works focus on the privacy of the time-series data itself rather than hiding latent sensitive attributes [45–48, 111, 120]. For instance, in the SM privacy scenario in Chapter 3 or location sharing in Chapter 4, sensitive information is the time-series data itself and the utility loss can be measured by data distortion, whereas in other applications, the user might be interested in hiding an underlying sensitive hypothesis. For instance, the user's presence at home or favorite TV channel can be inferred from SM readings, while her sensitive daily habits can be revealed to the SP through the sensors of a wearable device.

Inference privacy protects the user's data from an adversary's attempt to deduce sensitive information from an underlying distribution [15, 21, 62, 121–124]. These techniques perform well against inference attacks, in which the adversary targets detecting the user's underlying private information with high confidence [48]. PUT between correlated sensitive and useful r.v.'s has also been studied under the *privacy funnel* framework [121], which is closely related to the *information bottleneck* concept introduced in [125]. In privacy funnel approaches [15, 21, 62, 121–123], the goal is to conceal the sensitive information from SP's inference while gaining enough utility from the useful information, where both the utility and the privacy leakage are measured by MI. However, [15,121–123] consider independent data without temporal correlations, hence, these approaches are not suitable for temporally correlated time-series data.

In this chapter, we assume that a user wants to share the "useful" part of her data with the SP. However, the SP might also try to deduce user's "secret" information from the shared time-series data (e.g., location, heartbeat, temperature or energy consumption). We model the user's secret and useful data as correlated discrete r.v.'s. The user's goal is to prevent the secret from being accurately detected by the SP while revealing the useful data accurately for utility.

Differently from the existing works [13, 14, 18, 19, 21, 121, 123], which typically consider a time-independent data release problem, we consider a discrete time system, and assume that the user can actively choose from among a finite number of data release mechanisms (DRMs) at each time. While each measurement reveals some information about user's latent states, we assume that each DRM has different measurement characteristics, i.e., conditional probability distributions. User's objective is to choose a DRM at each time in an online fashion to reveal the value of the useful r.v. for maximum utility while keeping the leakage of the sensitive information below a prescribed value. Our proposed privacy measures are based on the SP's confidence in the secret and the MI between the secret r.v. and the observations. These measures are similar to those proposed in [16], [124], and [126]. However, [16] considers the PUT of a binary secret r.v. in an asymptotic regime, while [126] considers binary as well as M-ary r.v.'s for an offline scenario using semi-definite programming, which has high computational complexity when fine-grained data is considered. [124] takes the data release history into account for M-ary r.v.'s, however, it does not consider the time aspect in the PUT objective.

In this chapter, we introduce sequential private data release policies (PDRP's) for two different problems: Problem A in Section 5.2 aims to maximize the SP's confidence in the true value of the useful r.v. and stops data sharing right before the confidence in secret r.v. is exceeded, and Problem B in Section 5.3 aims to minimize the SP's error probability on the useful r.v. as quick as possible subject to a constraint on the SP's confidence in the true secret. Besides confidence-based utility, we also consider MI-based utility for Problem A. In Problem B, on the other hand, we investigate MI-based privacy in addition to confidence-based privacy. Note that MI-based privacy, which keeps the total MI between the secret r.v. and the shared data below a certain level, does not necessarily prevent the detection of the true secret value; instead, it limits the information leakage in an average sense. While confidence-based privacy is strong against worst case adversaries, MI-based privacy is useful when average-case adversaries try to infer the sensitive data. We validate this in our simulation results.

We consider data release policies which take the entire release history into account, and recast both Problems A and B as POMDPs. POMDPs can be represented as continuous state belief-MDPs; however, finding optimal policies for continuous state and action spaces is a PSPACE-hard problem [115]. Therefore, after identifying the structure of the optimal policy, we use A2C-DRL to evaluate our continuous state and action probability space MDP numerically. Besides assuming known distributions for MI calculation with synthetic data, we also use variational representations for MI estimation through neural networks [127] with real data. Finally, we examine the performances of the proposed policies in human activity privacy scenario, in which we use both synthetic data and smartwatch sensor readings from *smoking activity dataset* [119]. We compare the privacy levels achieved by the proposed policies using an SP that predicts the true values for useful data and secret from the shared observation history. The SP is represented by a long short-term memory (LSTM) neural network.

Our contributions are summarized as follows:

- We propose two active learning frameworks in which one takes only PUT, i.e., Problem A, and the other takes both PUT and the stopping time into account, i.e., Problem B, in online sharing of time-series data.
- We propose PDRP's that consider confidence and MI-based utility for optimal PUT against an SP performing sequential Bayesian inference for Problem A.

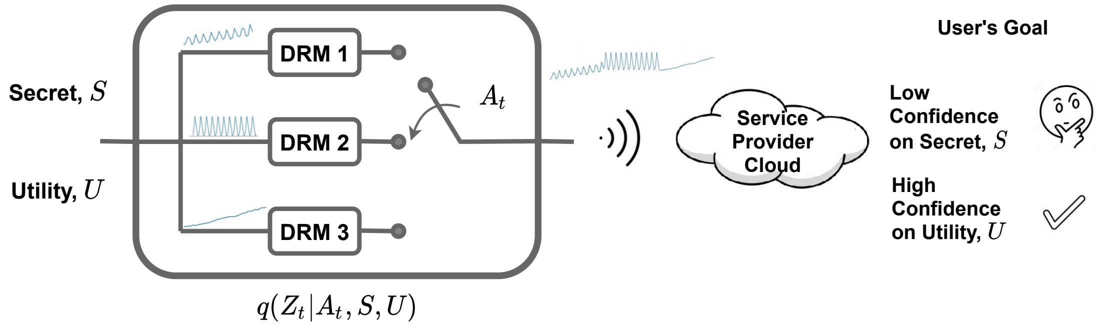


FIGURE 5.1: System model for active PUT against the SP.

- We propose PDRP's based on privacy measured by the error probability of the SP on the secret, and MI between the secret and the released data history against average-case adversaries for Problem B.
- We recast the active time-series data release problems for PUT as POMDPs, and evaluate the proposed PDRP's numerically using A2C-DRL.

The remainder of the chapter is organized as follows. We present the problem formulations in Sections 5.2 and 5.3 for Problem A and Problem B, respectively. Synthetic and real data evaluations for human activity privacy are presented in Section 5.4. Finally, we conclude this chapter in Section 5.5.

5.2 Active Private Data Sharing

We consider a user that wants to share her data with the SP in return of utility. The data reveals information about two underlying latent variables; a *secret* variable and a non-sensitive *useful* variable. The user's goal is to maximize the SP's confidence for the non-sensitive useful information to gain utility, while keeping his confidence in the secret variable below a predefined level.

Let $\mathcal{S} = \{0, 1, \ldots, N-1\}$ and $\mathcal{U} = \{0, 1, \ldots, M-1\}$ be the finite sets of the hypotheses represented by the r.v.'s $S \in \mathcal{S}$ for the secret and $U \in \mathcal{U}$ for the non-sensitive useful information, respectively. Consider a finite set $\mathcal{A}$ of different data release mechanisms (DRMs) available to the user, each modeled with a different statistical relation with the underlying hypotheses. For example, in the case of a user sharing activity data, e.g., Fitbit records, set $\mathcal{A}$ may correspond to different types of sensor measurements the user may share. Useful information the user wants to share may be the exercise type, while the sensitive information can be various daily habits. Similarly, in the case of smart meter readings, the useful information might be ON/OFF state of home appliances for smart power scheduling whereas the sensitive information might be the types of TV channels the user watches. We assume that the data revealed at time t, Z_t, is generated by an independent realization of a conditional probability distribution that depends on the true hypotheses and the chosen DRM $A_t \in \mathcal{A}$, denoted by $q(Z_t|A_t, S, U)$. Figure 5.1 shows an illustration of the system model with three DRMs.

The user's goal is to disclose U through the released data Z_t, as long as the SP's confidence in S is below a certain threshold. We assume that the observation statistics $q(Z_t|A_t, S, U)$ and the employed DRM A_t are known both by the user and the SP. To maximally confuse the SP, the user selects action A_t with a probability distribution $\pi(A_t|Z^{t-1}, A^{t-1})$ conditioned on the SP's observation history up to that time, $\{Z^{t-1}, A^{t-1}\}$. If the user has the knowledge of the true hypotheses, she can select the actions depending on both the observation history and the true hypotheses. However, our assumption is that the true hypotheses are unknown to all the parties involved.

The optimal strategy for the SP is to employ classical *sequential HT*, i.e., he observes the data samples released by the user and updates its belief on the true hypotheses accordingly. Here, we quantify the confidence of SP as his belief on hypotheses S and U after observing $\{Z^{t-1}, A^{t-1}\}$, which is shown by

$$\beta_t(s, u) = P(S = s, U = u|Z^{t-1} = z^{t-1}, A^{t-1} = a^{t-1}), \tag{5.1}$$

where s, u, z_t and a_t are the realizations of S, U, Z_t, and A_t, respectively. The SP's belief on the secret is $\beta_t(s) = \sum_{u\in\mathcal{U}} \beta_t(s, u)$. We assume that the SP becomes more confident about a hypothesis being correct as the belief on the corresponding hypothesis becomes larger than the belief on the others. This is motivated by a worst-case adversary model which is interested in the value of the true hypothesis.

Let τ be the time that we believe the SP reaches the prescribed confidence threshold on the secret. The user stops releasing data at this point. The main objective of this section is to obtain a policy $\boldsymbol{\pi}$, which generates the best action probabilities, such that the SP's belief on the true U at time τ is maximized. Therefore, our goal is to solve the Problem A:

$$\underset{A_0, A_1, \ldots, A_{\tau-1}}{\text{maximize}} \quad \beta_\tau(u) \tag{5.2}$$

$$\text{subject to} \quad \beta_t(s) \leq L_s, \forall t \leq \tau, \forall s \in \mathcal{S} \tag{5.3}$$

where L_s is a predetermined scalar of the user's choice. Note that PUT will be obtained by considering a range of L_s values.

5.2.1 POMDP Formulation

The above PUT can be recast as a POMDP with partially observable static states $\{S, U\}$, actions A_t, and observations Z_t. POMDPs can be reformulated as belief-MDPs and solved using classical MDP solution methods. Hence, we define the state of the belief-MDP as the SP's belief on hypotheses $\{S, U\}$ after observing $\{Z^{t-1}, A^{t-1}\}$, i.e., $\beta_t(s, u)$. After defining the states as the belief, the user's action probabilities become conditioned on the belief distribution, i.e., $\pi(A_t = a_t|\beta_t)$, while the observation probabilities are the same as before.

The user stops sharing data when the SP's belief on any secret $s \in \mathcal{S}$ exceeds a threshold. Therefore, the problem is an episodic MDP, which ends when a final state is reached. We define a new state space $\mathcal{X} = P(\mathcal{S},\mathcal{U}) \cup \{F\}$ of size $N \times M$, where $P(\mathcal{S},\mathcal{U})$ is the belief space, and F is a recurrent final state reached when the SP's confidence on $\mathcal{S}$ surpasses the prescribed maximum value. After a single observation $\{z_t, a_t\}$, the SP updates its belief by Bayes' rule as follows:

$$\phi^{\boldsymbol{\pi}}(\beta_t, z_t, a_t) = \frac{q(z_t|a_t, s, u)\beta_t(s, u)}{\sum\limits_{\hat{s},\hat{u}} q(z_t|a_t, \hat{s}, \hat{u})\beta_t(\hat{s}, \hat{u})}, \tag{5.4}$$

where $\phi^{\boldsymbol{\pi}}(\beta_t, z_t, a_t)$ represents the next belief state $\beta_{t+1}(s, u)$, and it can also be denoted by $\phi^{\boldsymbol{\pi}}(\beta, z, a)$ in time-independent notation. Hence, the state transitions of the belief-MDP are governed by the observation probabilities of different actions, $q(z_t|a_t, s, u)$. If $\beta_t(s) \geq L_s$ holds for any secret $s \in \mathcal{S}$, we transition to the final state F. The overall strategy for belief update is represented by the Bayes' operator as follows:

$$\phi^{\boldsymbol{\pi}}(x, z, a) = \begin{cases} \phi^{\boldsymbol{\pi}}(\beta, z, a), & \text{if } x = \beta(s, u) \text{ for } \beta(s) < L_s \\ F, & \text{if } x = \beta(s, u) \text{ for } \beta(s) \geq L_s \\ F, & \text{if } x = F. \end{cases}$$

We define an instantaneous reward function for the current state, which induces policy $\boldsymbol{\pi}$ when maximized:

$$r_B(x) = \begin{cases} 0, & \text{if } x = \beta(s, u) \text{ for } \beta(s) < L_s \\ \max\limits_{u} \beta(u), & \text{if } x = \beta(s, u) \text{ for } \beta(s) \geq L_s \\ 0, & \text{if } x = F. \end{cases}$$

Due to the belief-based utility, we call this approach belief-reward policy. According to her strategy, the user checks if the SP's belief on any secret exceeds a threshold L_s, if not, she believes that the SP updates his belief as in (5.4) in the next time step. If the threshold is reached, the user stops data sharing, updates the state $x = \beta(s, u)$ to the final state $x = F$ and the episode ends.

We assume that the SP follows the optimal sequential HT strategy. Since the user has access to all the information that the SP has, it can perfectly track his beliefs. Hence, the user decides her own policy facilitating the SP's strategy, episodic behavior and belief. Accordingly, reward function $r_B(x)$ is defined such that the user receives no reward until the SP's belief on the secret reaches the prescribed threshold, at which point she receives a reward measured by the SP's current belief on the true useful hypothesis, and the episode ends by reaching the final state.

The corresponding Bellman equation induced by the optimal policy $\boldsymbol{\pi}$ can be written as [65],

$$\mathcal{V}^{\boldsymbol{\pi}}(\beta) = \max_{\pi(a|\beta) \in P(\mathcal{A})} \Big\{ r(\beta, \pi(a|\beta)) + \mathbb{E}_{z,a} \mathcal{V}^{\boldsymbol{\pi}}(\phi^{\boldsymbol{\pi}}(\beta, z, a)) \Big\},$$

where $\mathcal{V}^{\boldsymbol{\pi}}(\beta)$ is the state-value function, and $P(\mathcal{A})$ is the action probability space. The objective is to find a policy $\boldsymbol{\pi}$ that optimizes the reward function. Since, finding optimal policies for continuous state and action MDPs is PSPACE-hard as mentioned before, we will use A2C-DRL as a computational tool to numerically solve the continuous state and action space MDP.

5.2.2 MI as Utility

In this section, we consider a scenario where the SP is more interested in the statistics of the public information rather than its true value. Accordingly, we consider MI as a utility measure; that is, the user wants to maximize the MI between the useful hypothesis and the observations by the time the SP reaches the prescribed confidence level on the secret. MI is commonly used both as a privacy and a utility measure in the literature [120, 121, 128]

The MI between U and (Z^T, A^T) over time T is given by

$$I(U; Z^T, A^T) = \sum_{t=1}^{T} I(U; Z_t, A_t | Z^{t-1}, A^{t-1}). \tag{5.5}$$

The MI between the useful hypothesis and the observations at time t can be written in terms of the belief, action and observation probabilities as follows:

$$I(U;Z_t,A_t|\beta) = -\sum_{s,u,z_t,a_t} q(z_t|a_t,s,u)\pi(a_t|\beta)\beta(s,u) \\ \times \log \frac{\sum_{\hat{s}} q(z_t|a_t,\hat{s},u)\pi(a_t|\beta)\beta(\hat{s},u)}{\beta(u)\sum_{\bar{s},\bar{u}} q(z_t|a_t,\bar{s},\bar{u})\pi(a_t|\beta)\beta(\bar{s},\bar{u})}. \tag{5.6}$$

Accordingly, the information reward gained in the current time step after taking action a_t, and releasing the corresponding observation z_t is defined as

$$r_{MI}(x) = \begin{cases} I(u;z_t,a_t|\beta), & \text{if } x = \beta(s,u) \text{ for } \beta(s) \leq L_s \\ 0, & \text{if } x = F. \end{cases}$$

SP's belief is updated by $\phi^{\boldsymbol{\pi}}(x,z,a)$ as before. This policy maximizes the leakage not only for the true hypothesis for u but all possible hypotheses for U. For example, a policy may disclose a lot of information even if the SP is confused between two out of many hypotheses, as he learns that the true state is none of the other possibilities.

Numerical evaluation of this section is presented in Section 5.4. In the next section, we introduce another scenario in which the user aims for quickest stopping while optimizing the PUT.

5.3 Active Quickest Private Data Sharing

In this section we consider the same setting as Section 5.2 as shown in Figure 5.1. On the other hand, the user's goal is to release her data such that the intended SP can detect the non-sensitive information with minimum error as quickly as possible, while keeping his confidence in the secret part below a predefined level. In other words, the user wants to disclose the true value of the r.v. U through the released data Z_t, while keeping the SP's confidence in S below a certain threshold. Let τ be the time that the SP is confident enough about the true useful variable and makes a declaration. This is also the time at which the user stops releasing data, since U is already disclosed to the SP. The objective of the problem is to find a sequence of actions $\{A_0,\ldots,A_{\tau-1}\}$, a stochastic stopping time τ and a declaration rule $d:\mathcal{A}^{\tau-1}\times\mathcal{Z}^{\tau-1}\rightarrow\mathcal{U}$ that collectively solve the following optimization, Problem B:

$$\begin{aligned} \underset{A_0,\dots,A_{\tau-1},d}{\text{minimize}} \quad & \mathbb{E}[\tau]+\lambda P_{err}(u) \\ \text{subject to} \quad & \mathcal{C}_t(s) < L_B, \forall t \leq \tau, \forall s \in \mathcal{S} \end{aligned} \tag{5.7}$$

where $P_{err}(u) = P(d(A^{\tau-1}, Z^{\tau-1}) \neq u)$ is the error probability of making wrong declaration for the true value $u \in \mathcal{U}$; $\mathcal{C}_t(s)$ is the SP's instantaneous confidence in the true sensitive value $s \in \mathcal{S}$, which is quantified by the SP's belief on s given the observation history, i.e., $P(S = s|A^{\tau-1}, Z^{\tau-1})$; L_B is a scalar of user's choice; B will later represent the name of the policy, e.g., (B)elief privacy-data release policy; and the expectation is taken over the action and observation distributions as well as the initial distributions of the r.v.'s. The main difference between the Problem A in Section 5.2 and Problem B is the declaration rule d. While the stopping action in Problem A is directly determined by whether the privacy constraint is violated or not, declaration rule d determines the stopping time according to the optimization in Problem B.

For our theoretical results, we assume that the observation statistics $q(Z_t|A_t, S, U)$ and the employed DRM A_t are known by both the user and the SP. Later, we will also consider real datasets with unknown data distributions in our simulations. To maximally confuse the SP, the user selects action A_t with a probability distribution $\pi(A_t|Z^{t-1}, A^{t-1})$ conditioned on the SP's observation history up to that time, $\{Z^{t-1}, A^{t-1}\}$. In this work, we assume that the true values s and u are unknown to all the parties involved.

5.3.1 POMDP Formulation

The above PUT can be recast as a POMDP with partially observable static states $\{S, U\} \in \mathcal{S} \times \mathcal{U}$, actions $A_t \in \mathcal{A} \cup \{d\}$, and noisy observations $Z_t \in \mathcal{Z}$, and solved using classical MDP solution methods. We will follow this approach, and introduce SP's belief to determine the state variable in three steps. Firstly, we define the belief of the SP on S and U after he observes $\{Z^{t-1}, A^{t-1}\}$ by $\beta_t(s, u)$ as in (5.4) for belief space $\mathbb{P}(\mathcal{B}) := \{\beta_t \in [0, 1]^{M\times N} : \sum_{s\in\mathcal{S},u\in\mathcal{U}} \beta_t(s, u) = 1\}$, where the marginal beliefs are represented by $\beta_t(u) := \sum_{s\in\mathcal{S}} \beta_t(s, u)$ and $\beta_t(s) := \sum_{u\in\mathcal{U}} \beta_t(s, u)$, respectively. The SP's confidence that $S = s$ at time t is represented by $\mathcal{C}_t(s) := \beta_t(s)$. The user's action probabilities become conditioned on the belief distribution, i.e., $\pi(A_t = a_t|\beta_t)$, while the observation probabilities are the same as before. Secondly, we introduce a new state $F_B := \{\beta_t(s) \geq L_B : \sum_{s\in\mathcal{S}} \beta_t(s) = 1\}$ for $F_B \subseteq \mathbb{P}(\mathcal{B})$, called the *forbidden-state*, which represents the condition where the constraint in (5.7) is violated. F_B is ideally an infinite cost state; however, in practice, we assume it has a large-cost. As the third step of defining the state space, we include a terminal state to fully characterize the state in which the user stops sharing her data with the SP. We assume that after the user takes the stopping action, the system goes to a terminal state, denoted by F, and remains there forever. This makes the problem an episodic MDP. Consequently, the state space becomes $\mathcal{X} = \mathbb{P}(\mathcal{B}) \cup \{F\}$.

We always refer the time independent expression of belief, i.e., β, as the current belief state. The optimal expected total cost of our problem is defined as follows:

Definition 5.1. For all $\beta \in \mathbb{P}(\mathcal{B})$, let the optimal value function $V^*(\beta)$ represent the optimal expected cost of problem (5.7), given the initial belief β. That is,

$$V^*(\beta) := \min\{\mathbb{E}[\tau] + \lambda P_{err}(u)\}, \tag{5.8}$$

where the minimization is with respect to τ, action and observation sequences, and the declaration rule d.

Optimal expected total cost for active PUT against an SP can be obtained by evaluating V^* at the initial belief. This can be done by solving a DyP problem. After a single observation $\{z_t, a_t\}$, the SP updates its belief by Bayes' rule as in (5.4). We define a Markov operator $\mathbb{T}^a$ for action a, such that for any measurable function $V : \mathbb{P}(\mathcal{B}) \to \mathbb{R}$,

$$(\mathbb{T}^a V)(\beta) := \int V(\Phi(\beta, z, a)) \sum_{s,u} q(z|a, s, u)\beta(s, u)dz. \tag{5.9}$$

For any state $\beta \in \mathbb{P}(\mathcal{B})$, the user's data release action a under the optimal policy results in an expected total cost of $1 + (\mathbb{T}^a V^*)(\beta)$, where time spent by the user for data release is represented by cost 1, and $(\mathbb{T}^a V^*)(\beta)$ is the expected future value of V^*. On the other hand, the user's stopping decision d results in error probability of the declaration of true useful value u with penalty λ, i.e., $\lambda P_{err}(u) := \lambda(1 - \beta(u))$. Solution for the optimal V^* is formalized by the following theorem.

Theorem 5.2. *[129] The optimal V^* for $\beta \in \mathbb{P}(\mathcal{B})$ satisfies the fixed point equation:*

$$V^*(\beta) = \min\{1 + \min_{a \in \mathcal{A}} (\mathbb{T}^a V^*)(\beta), \min_{u \in \mathcal{U}} \lambda(1 - \beta(u))\}. \tag{5.10}$$

Definition 5.3. Let a Markov stationary policy π be a stochastic kernel from the state space to the action space, including the stopping action which determines the stopping time τ, i.e., $\mathbf{\Pi} := \mathbb{P}(\mathcal{B}) \to \mathcal{A} \cup \{d\}$. That is, the probability of choosing DRM a under policy π at state β is denoted by $\pi(a|\beta)$.

Following from Corollary 9.12.1 in [129], DyP equation (5.10) characterizes the optimal deterministic stationary policy π^* for $\beta \in \mathbb{P}(\mathcal{B})$. The intuition behind Theorem 5.2

is that the user's data release action $a^* = \arg\min_{a\in\mathcal{A}} T^a(V^*)(\beta)$ is the least costly action with cost $1+\min_{a\in\mathcal{A}} T^a(V^*)(\beta)$, unless choosing the stopping action d and letting the SP make a decision for u is less costly, i.e., $\lambda(1-\beta(u))$. We also ensure that for any two hypotheses $u, u' \in \mathcal{U}$, $u \neq u'$, there exists an action $a \in \mathcal{A}$, such that $D(q(z|a,s,u)||q(z|a,s,u')) > 0, \forall s \in \mathcal{S}$, where $\mathrm{D}(\cdot||\cdot)$ denotes the Kullback-Leibler (KL) divergence. That is, hypotheses u and u' are distinguishable all the time, such that (5.7) has a meaningful solution.

Theorem 5.4. *Suppose there exists a parameter $C_T > 0$, e.g., time cost, and a functional $V : \mathbb{P}(\mathcal{B}) \to \mathbb{R}_+$ such that for all belief states $\beta \in \mathbb{P}(\mathcal{B})$,*

$$V(\beta) \leq \min\{C_T + \min_{a\in\mathcal{A}}(\mathbb{T}^a V^*)(\beta), \min_{u\in\mathcal{U}} \lambda C_T(1-\beta(u))\}. \tag{5.11}$$

Then $V^*(\beta) \geq \frac{1}{C_T} V(\beta)$ for all $\beta \in \mathbb{P}(\mathcal{B})$.

Proof. For the proof of Theorem 5.4, we include a termination state in our state space. We assume that after the user takes the stopping action for data release, the system goes to a recursive termination state, denoted by F, and remains there forever. Hence, the new state space is $\mathcal{X} = \mathbb{P}(\mathcal{B}) \cup \{F\}$. Let instantaneous cost of taking action $a \in \mathcal{A} \cup \{d\}$

$$c^{\pi_B}(x,a) = \begin{cases} 1, & \text{if } x = \beta \in \mathbb{P}(\mathcal{B}) \setminus \{F_B\}, a \in \mathcal{A} \\ \min_{u\in\mathcal{U}}(1-\beta(u))\lambda, & \text{if } x = \beta \in \mathbb{P}(\mathcal{B}) \setminus \{F_B\}, a = d \\ C_B, & \text{if } x = \beta = F_B, a \in \mathcal{A} \\ 0, & \text{if } x = F. \end{cases} \tag{5.12}$$

The constraint on the adversary's confidence in s is enforced with an instantaneous cost C_B for state F_B, which is ideally infinity but can be applied as a very large scalar in practice. Assuming that the system follows the optimal policy, transition to F_B with a very large cost C_B would not be chosen by the minimization problem. The overall strategy for belief update is represented by the Bayes' operator as follows:

$$\Phi^{\pi_B}(x,z,a) = \begin{cases} \Phi^{\pi_B}(\beta,z,a), & \text{if } x = \beta \in \mathbb{P}(\mathcal{B}), a \in \mathcal{A} \\ F, & \text{if } x = \beta \in \mathbb{P}(\mathcal{B}), a = d \\ F, & \text{if } x = F. \end{cases} \tag{5.13}$$

Using the instantaneous cost and state update, the condition $V(\beta) \leq \min\{C_T + \min_{a\in\mathcal{A}}(\mathbb{T}^a V^*)(\beta), \min_{u\in\mathcal{U}} \lambda C_T(1-\beta(u))\}$ is rewritten as

$$\begin{aligned}
&V(F) = 0,\\
&V(x) \leq \min_{a\in\mathcal{A}\cup\{d\}} \{T_c c(x,a) + \mathbb{E}[V(\Phi(x,z,a))]\}, \forall x \in \mathbb{P}(\mathcal{B}),
\end{aligned}$$

as well as the state sequence at times $t = 0, 1, 2, \ldots$ is denoted by

$$\begin{aligned}
&X_0 = x,\\
&X_n = \Phi(X_{n-1}, Z, A(n)), \forall n, n > 0.
\end{aligned}$$

When the condition is written in terms of the state sequence of duration N for the optimal policy π^*, we obtain

$$V(x) \leq C_T \mathbb{E}_{\pi^*}[\sum_{n=0}^{N-1} c(X_n, A_n)] + \mathbb{E}_{\pi^*}[V(X_N)]. \tag{5.14}$$

Taking the limit as $N \to \infty$, we get

$$\begin{aligned}
V(x) &\leq C_T \mathbb{E}_{\pi^*}[\sum_{n=0}^{\infty} c(X_n, A_n)] + \lim_{N\to\infty} \mathbb{E}_{\pi^*}[V(X_N)] && (5.15)\\
&= C_T V^*(x) + \lim_{N\to\infty} \mathbb{E}_{\pi^*}[X_N] && (5.16)\\
&= C_T V^*(x) + \lim_{N\to\infty} \mathbb{E}_{\pi^*}\Big[V(F)\mathbf{1}_{\{X_N=F\}} + V(F_B)\mathbf{1}_{\{X_N=F_B\}}\\
&+ V(X_N)\mathbf{1}_{\{X_N\neq F, X_N\neq F_B\}}\Big]\\
&= C_T V^*(x) + \lim_{N\to\infty} \mathbb{E}_{\pi^*}\Big[V(X_N)\mathbf{1}_{\{X_N\neq F\}} + V(F_B)\mathbf{1}_{\{X_N=F_B\}}\Big]\\
&\leq C_T V^*(x) + \lambda \lim_{N\to\infty} \Big(\mathbb{P}_{\pi^*}[X_N \neq F] + \mathbb{P}_{\pi^*}[X_N = F_B]\Big) && (5.17)\\
&= C_T V^*(x), && (5.18)
\end{aligned}$$

where (5.15) holds due to the monotone convergence theorem; (5.16) follows from the definition of V^*; (5.17) is due to the fact that for any $\beta \in \mathbb{P}(\mathcal{B})$, $V(\beta) \leq \min_{u\in\mathcal{U}} \lambda(1-\beta(u)) \leq \lambda$; and (5.18) holds since $\lambda \geq V^*(x) \geq \mathbb{E}_{\pi^*}[\tau] = \sum_{n=0}^{\infty} P_{\pi^*}(\tau > n) = \sum_{n=0}^{\infty} P_{\pi^*}(X_n \neq F)$, and the probability of the system following the optimal policy π^* to transition to highest-cost state F_B at N is zero, i.e, $\lim_{N\to\infty} \mathbb{P}_{\pi^*}[X_N = F_B] = 0$. □

Theorem 5.4 provides a lower bound for a fixed-point expression of V^*. However, it is difficult to calculate the real value of V^* and solve DyP equation with continuous belief space. Hence, we solve (5.7) using an RL approach to obtain a good approximation. Due to the belief-based privacy constraint, we call our policy *belief-privacy data release policy* (belief-PDRP), π_B. In our RL approach, the optimal policy π_B^* is induced as a result of the minimization of the instantaneous cost $c^{\pi_B}(x, a)$ that we introduced in (5.12) for current state x and action $a \in \mathcal{A} \cup \{d\}$. The constraint on the SP's confidence in s is enforced with a large instantaneous cost C_B for state F_B, which is ideally infinite. Assuming that the system follows the optimal policy, data release actions resulting in a transition to F_B with a large-cost C_B would not be selected by the minimization problem, as shown in the proof of the Theorem 5.4. The overall strategy for belief update becomes (5.13). Since the user has access to all the information that the SP has, it can perfectly track his beliefs. Hence, the user decides her own policy facilitating the SP's detection strategy, episodic behavior and belief.

According to her strategy, the user checks whether the selected optimal action is the stopping action d. If so, she receives a cost determined by the current error probability of u with penalty λ, then transitions to the terminal state and ends the episode. If not, she checks whether the SP's belief on any secret exceeds L_B. If the user is in the *forbidden-state* she receives a large-cost C_B; otherwise, either she receives a time cost 1 or terminal state cost 0 depending on her state. If the terminal state has not already been reached and stopping action has not been taken at the moment, the user updates the SP's belief as in (5.4); otherwise she updates the state to the final state $x = F$. Using the condition (5.11) in Theorem 5.4, we write a lower bound for the Bellman equation induced by the optimal policy π_B^* as [65],

$$V(x) = \min_{a \in \mathcal{A} \cup \{d\}} \{c^{\pi_B}(x, a) + \mathbb{E}[V(\Phi^{\pi_B}(x, z, a))]\}, \forall x \in \mathbb{P}(\mathcal{B}). \tag{5.19}$$

The objective is to find a policy π_B^* that optimizes the cost function. Since the proposed POMDP has a continuous state space and action probabilities, as mentioned earlier, finding optimal policies is PSPACE-hard. Hence, we use A2C-DRL to numerically solve the continuous state and action space MDP in Section 5.4.3. In addition to the confidence-based privacy, we also consider an MI privacy policy in Section 5.3.2.

5.3.2 MI as Privacy Constraint

In this section, we consider a scenario, in which the user is interested in hiding the sensitive information in an average sense, rather than hiding its true value. For instance, the SP might be confused about the true secret, however, he might still have an idea about which secret values are unlikely. More concretely, consider a secret r.v. with alphabet size of three, e.g., $\mathcal{U} = \{1, 2, 3\}$. From the perspective of confidence, the belief of $\beta(U = 1) = 1/2$, $\beta(U = 2) = 1/4$, $\beta(U = 3) = 1/4$ would be the same as $\beta(U = 1) = 1/2$, $\beta(U = 2) = 1/2$, $\beta(U = 3) = 0$. While the latter clearly has additional information about the secret resulting in reduced uncertainty. We tackle this issue by measuring the privacy by the MI between the secret variable S and the observation history $\{Z^t, A^t\}$ for $t \leq \tau$. According to her policy, the user wants to minimize the error on useful information as quickly as possible while keeping the total MI between the secret and the observations below a prescribed level, i.e., $\forall Z \in \mathcal{Z}$ and $\forall A \in \mathcal{A}$,

$$\begin{aligned} \underset{A_0,\dots,A_{\tau-1},d}{\text{minimize}} \quad & \mathbb{E}[\tau] + \lambda P_{err}(u) \\ \text{subject to} \quad & I(S; Z^t, A^t) < L_{MI}, \forall t \leq \tau, \forall S \in \mathcal{S} \end{aligned} \tag{5.20}$$

where L_{MI} is a scalar of the user's choice.

MI is commonly used both as a privacy and a utility measure in the literature [120, 121, 128]. As opposed to Section 5.2.2, here, it is used as a privacy measure to control PUT between the useful variable and the secret. Due to the MI-based privacy constraint in (5.20), we call this policy *MI-privacy data release policy* (MI-PDRP), π_{MI}. MI between S and (Z^T, A^T) over time T is given by

$$I(S; Z^T, A^T) = \sum_{t=1}^{T} I(S; Z_t, A_t | Z^{t-1}, A^{t-1}). \tag{5.21}$$

Theorem 5.5. *The instantaneous MI cost between the secret and the observations induced by policy π_{MI} at time t can be written as:*

$$\begin{aligned} I^{\pi_{MI}}(S; Z_t, A_t | \beta) = - & \sum_{s,u,z_t,a_t} q(z_t|a_t, s, u)\pi(a_t|\beta)\beta(s, u) \\ & \times \log \frac{\sum_{\tilde{u}} q(z_t|a_t, s, \tilde{u})\pi(a_t|\beta)\beta(s, \tilde{u})}{\beta(s) \sum_{\bar{s},\bar{u}} q(z_t|a_t, \bar{s}, \bar{u})\pi(a_t|\beta)\beta(\bar{s}, \bar{u})}. \end{aligned} \tag{5.22}$$

Proof. Consider a POMDP with the belief state $\mathcal{X} = \mathbb{P}(\mathcal{B}) \cup \{F\}$. At time t, a decision maker observes Z^{t-1}, A^{t-1} and chooses an action $A_t \in \mathcal{A} \cup \{d\}$ as follows:

$$A_t = f_t(Z^{t-1}, A^{t-1}), \tag{5.23}$$

where $\boldsymbol{f} = (f_1, f_2, \dots)$ is called the policy. Based on the conditional probability $\pi(A_t|A^{t-1}, Z^{t-1})$ of taking this action, $Z_t \in \mathcal{Z}$ is observed and revealed by the sensor distribution $q(Z_t|A_t, S, U)$, and the state evolves to the next belief state. At each step, the system incurs a per-step cost

$$c(s, u, z^t, a^t; \boldsymbol{f}) := \log \frac{P^{\boldsymbol{f}}(Z_t = z_t, A_t = a_t | S = s, Z^{t-1} = z^{t-1}, A^{t-1} = a^{t-1})}{P^{\boldsymbol{f}}(Z_t = z_t, A_t = a_t | Z^{t-1} = z^{t-1}, A^{t-1} = a^{t-1})}. \tag{5.24}$$

The objective is to find a policy $\boldsymbol{f} = (f_1, \dots, f_T)$ that minimizes the total cost given by $\frac{1}{T}\mathbb{E}^{\boldsymbol{f}}\Big[\sum_{t=1}^{T} c(S, U, Z^t, A^t; \boldsymbol{f})\Big]$, where the expectation is taken with respect to the distributions induced by the policy $\boldsymbol{f}$.

Let $\boldsymbol{f} = (f_1, \dots, f_T)$ be $f_t(z^{t-1}, a^{t-1}) = \pi(\cdot|z^{t-1}, a^{t-1})$. Then the following holds:

$$\begin{aligned} I^{\pi_{MI}}(S; Z_t, A_t|Z^{t-1}, A^{t-1}) &= \sum_{s,u,z^t,a^t} P^{\pi_{MI}}(S, U, Z^t, A^t) \\ &\times \log \frac{P^{\pi_{MI}}(Z_t = z_t, A_t = a_t|S = s, Z^{t-1} = z^{t-1}, A^{t-1} = a^{t-1})}{P^{\pi_{MI}}(Z_t = z_t, A_t = a_t|Z^{t-1} = z^{t-1}, A^{t-1} = a^{t-1})} \\ &= \mathbb{E}^{\boldsymbol{f}}\left[\sum_{t=1}^{T} c(S, U, Z^t, A^t; \boldsymbol{f})\right] \end{aligned} \tag{5.25}$$

The probability distribution on (S, U, Z^T, A^T) induced by the decision policy $\boldsymbol{f}$ is given by

$$\begin{aligned} P^{\boldsymbol{f}}(S = s, U = u, Z^T = z^T, A^T = a^T) &= P(s, u)q(z_1|a_1, s, u)\pi(a_1) \\ &\times \prod_{t=2}^{T}\Big[q(z_t|a_t, s, u)\pi(a_t|z^{t-1}, a^{t-1})\Big], \end{aligned} \tag{5.26}$$

where $\pi(\cdot|z^{t-1}, a^{t-1}) = f(z^{t-1}, a^{t-1})$. Under the transformations described above, $P^{\boldsymbol{f}}$ and $P^{\pi_{MI}}$ are identical probability distributions. As a result, $\mathbb{E}^{\boldsymbol{f}}\left[\sum_{t=1}^{T} c(S, U, Z^t, A^t; \boldsymbol{f})\right] = I^{\pi_{MI}}(S; Z_t, A_t|Z^{t-1}, A^{t-1})$. Hence, Theorem 5.5 holds. □

Similarly to the previous section, we define the state in three stages, i.e., the belief, the *forbidden-MI-state* as $F_{MI} := \{\beta_t(s) : I^{\pi_{MI}}(S; Z^t, A^t) \geq L_{MI}, \forall t \leq \tau\}$ for $F_{MI} \subseteq \mathbb{P}(\mathcal{B})$, where the constraint in (5.20) is violated, and the final state F in which the episode terminates.

We define an instantaneous cost function, $c^{\pi_{MI}}(x, a)$, for current state $x \in \mathcal{X} = \mathbb{P}(\mathcal{B}) \cup \{F\}$ and action $a \in \mathcal{A} \cup \{d\}$, which induces the optimal MI-PDRP π^*_{MI} when minimized:

$$c^{\pi_{MI}}(x, a) = \begin{cases} 1, & \text{if } x = \beta \in \mathbb{P}(\mathcal{B}) \setminus \{F_{MI}\}, a \in \mathcal{A} \\ \min\limits_{u \in \mathcal{U}}(1 - \beta(u))\lambda, & \text{if } x = \beta \in \mathbb{P}(\mathcal{B}) \setminus \{F_{MI}\}, a = d \\ C_{MI}, & \text{if } x = \beta = F_{MI}, a \in \mathcal{A} \\ 0, & \text{if } x = F. \end{cases} \tag{5.27}$$

The constraint on the total MI leakage from S is enforced with a large-cost C_{MI} for state F_{MI}. Assuming that the system follows the optimal MI-PDRP π^*_{MI}, F_{MI} would not be visited at all. The overall strategy for belief update is represented by the Bayes' operator as follows:

$$\Phi^{\pi_{MI}}(x,z,a) = \begin{cases} \Phi^{\pi_{MI}}(\beta,z,a), & \text{if } x=\beta \in \mathbb{P}(\mathcal{B}), a \in \mathcal{A} \\ F, & \text{if } x=\beta \in \mathbb{P}(\mathcal{B}), a=d \\ F_{MI}, & \text{if } x=\beta(s,u) \in \mathbb{P}(\mathcal{B}), \\ & \sum_{i=1}^{t} I^{\pi}(S;Z_i,A_i|\beta_i) \geq L_{MI} \\ F, & \text{if } x=F. \end{cases} \tag{5.28}$$

Theorem 5.4 holds for (5.20) when we replace $\{c^{\pi_B}, \Phi^{\pi_B}, F_B\}$ with $\{c^{\pi_{MI}}, \Phi^{\pi_{MI}}, F_{MI}\}$, and provides a lower bound for the value function V^* for all $\beta \in \mathbb{P}(\mathcal{B})$. Hence, to find the policy π^*_{MI}, we solve the Bellman equation (5.19) using RL for c^{π_B} and Φ^{π_B}. This policy minimizes the SP's error on the true value of u in the quickest way while constraining the MI leakage from not only true secret s but all possible values for S.

5.3.3 Estimating MI

Exact computation of MI is possible when the data distribution is known. However, in most practical scenarios, the user's data distribution is not known or it is inaccurate. Hence, we approximate $I(S;Z^T,A^T)$ via a variational representation which is inspired by Barber-Agakov MI estimation for single letter MI [127]. Since (5.21) is history-dependent, we modify this variational bound to a history dependent expression as follows:

$$\begin{aligned} I(S;&Z_t,A_t|Z^{t-1},A^{t-1}) \\ &= H(S|Z^{t-1},A^{t-1}) - H(S|Z^t,A^t) \qquad (5.29) \\ &= H(S|Z^{t-1},A^{t-1}) + \mathrm{D}(P(S|Z^t,A^t)||Q(S|Z^t,A^t)) + \mathbb{E}[\log Q(S|Z^t,A^t)] \qquad (5.30) \\ &= H(S|Z^{t-1},A^{t-1}) + \max_{Q(S|Z^t,A^t)} \mathbb{E}[\log Q(S|Z^t,A^t)] \qquad (5.31) \end{aligned}$$

where (5.29) follows from the definition of MI, (5.30) holds for any distribution $Q(S|Z^t,A^t)$ over $\mathcal{S}$ given the values in $\mathcal{Z}^t \times \mathcal{A}^t$, which represents what the belief would be after observing (A_t,Z_t), and (5.31) follows from the fact that maximum is attained when $Q(S|Z^t,A^t) = P(S|Z^t,A^t)$.

Given $(Z^{t-1},A^{t-1}) = (z^{t-1},a^{t-1})$, we can rewrite the variational representation for the MI conditioned on the neural estimation of the current belief $\hat{\beta}(S) = Q(S|Z^{t-1},A^{t-1})$

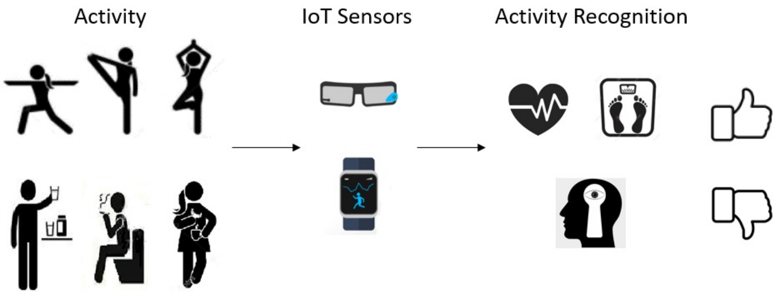


Figure 5.2: Activity recognition with wearable IoT devices does not only infer physical exercise but also sensitive daily habits.

as

$$I(S; Z_t, A_t|\hat{\beta}) = H(\hat{\beta}(S)) + \max_{Q(S|Z_t,A_t,\hat{\beta})} \mathbb{E}[\log Q(S|Z_t, A_t, \hat{\beta})], \tag{5.32}$$

where $H(\hat{\beta}(S)) = -\sum_{s\in\mathcal{S}} \hat{\beta}(s)\log\hat{\beta}(s)$, and the expectation is with respect to $(S, Z_t, A_t) \sim \hat{\beta}(S), \pi(A_t|\hat{\beta}), q(Z_t|A_t, S, U)$. Since the current belief realization is known to both the user and the SP, $H(\hat{\beta}(S))$ is a constant. Numerical estimation of the MI via neural networks is explained in Section 5.4.3.2.

5.4 Numerical Results

In this section, we present our results for both synthetic data and human activity privacy use-cases for Sections 5.2 and 5.3. In the synthetic data case we assume that all the distributions of the DRM are known by both the user and the SP, while in the latter, these distributions are learnt from a real dataset. In human activity privacy use-case, we focus on the sensors in wearable devices as an example of DRMs, and their measurements as time-series data. In this scenario, the user shares sensor readings of her wearable device with the SP, while performing physical activities, with the goal of tracking the type and duration of her activities. However, as in Figure 5.2, not only useful activities, such as exercise type, but also sensitive activities, such as smoking, drinking or eating habits, can be inferred from these readings, which the user may not want to share with the SP as the SP can exploit such information for commercial benefit at the detriment of the user. Hence, the user shares a single sensor reading from among multiple sensors at a time such that the useful activity is revealed to the SP while his confidence in the sensitive activity is kept hidden at a pre-defined level.

The POMDP formulation in Sections 5.2.1 and 5.3.1 enable us to numerically approximate the proposed policies using RL. We use A2C-DRL as described in Section 2.2.1 for the numerical evaluation of our problems.

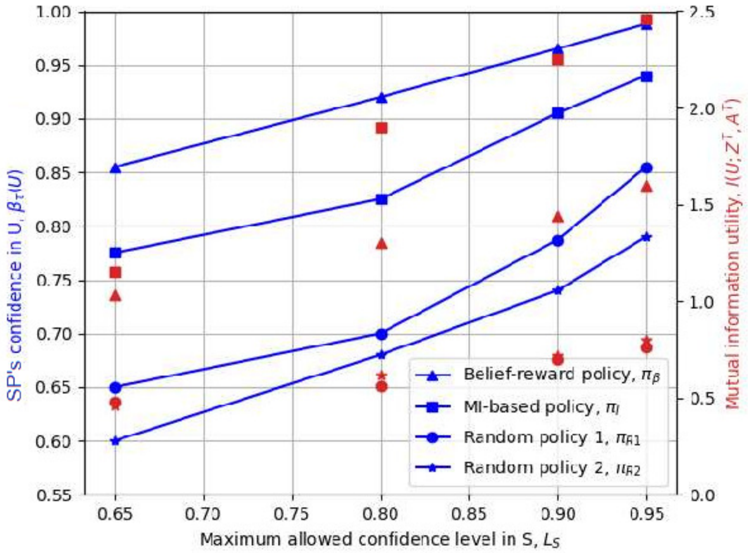


FIGURE 5.3: The confidence on U and MI utility w.r.t the maximum allowed confidence level on S for the proposed policies.

5.4.1 Active Private Data Sharing: Synthetic Data Use-Case

In this section, the results of Section 5.2 are presented for synthetically generated probability densities and $N = 3$, $M = 3$, $|\mathcal{A}| = 3$ and $|\mathcal{Z}| = 21$, and uniformly distributed S and U. The final state is reached when the SP's belief on any $s \in \mathcal{S}$ exceeds the threshold L_s for $L_s \in \{0.65, 0.8, 0.9, 0.95\}$. Observation probabilities are selected such that each action distinguishes a different pair of hypotheses well for both S and U. For example, we created a matrix with each row representing the conditional distribution of z for different (a, s, u) realizations. For sensor $a = 0$, we used $\mathcal{N}(0, \sigma_j)$ for $(s, u) = \{(0, 0), (0, 1), (0, 2), (1, 0), (1, 1), (1, 2)\}$, $\mathcal{N}(1, \sigma_j)$ for $(s, u) = (2, 0)$, $\mathcal{N}(2, \sigma_j)$ for $(s, u) = (2, 1)$, and $\mathcal{N}(3, \sigma_j)$ for $(s, u) = (2, 2)$, and we normalized through the columns representing z. Here, σ_j's are chosen randomly from the interval $[0.5, 1.5]$ for each (a, s, u) with index j=$\{1, .., N\times M\times|A|\}$. This sensor discloses s=2 case more than the other secrets. Moreover, a=1 and a=2 reveal more information for s=1 and s=0 cases, respectively. In this model, there is no perfect sensor which reveals only the useful hypothesis while giving no information about the secret. As a benchmark, we also consider a random policy taking the actions independently of the SP's observations and belief. We choose two random policies with action probabilities $\pi_{R1}(a)$=$[0.3, 0.6, 0.1]$ and $\pi_{R2}(a)$=$[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$. When the belief on the secret exceeds the threshold, episode ends as before.

In Figure 5.3, we show the SP's confidence about U at the decision time on the left axis and MI between U and observations on the right axis as a function of the allowed confidence level on S. While blue lines and red markers are scaled by the left and right axes, respectively, same markers in both colors represent the same particular policy.

Table 5.1: Stopping time τ of each policy's data release for the threshold values $L_s = \{0.65, 0.8, 0.9, 0.95\}$.

Policies:	$\tau(L_s = 0.65)$	$\tau(L_s = 0.8)$	$\tau(L_s = 0.9)$	$\tau(L_s = 0.95)$
π_B	105 ± 18	250 ± 46	470 ± 65	780 ± 120
π_{MI}	95 ± 15	180 ± 38	420 ± 60	485 ± 92
π_{R1}	4.2 ± 0.8	5.5 ± 1	8.2 ± 1.3	9 ± 2
π_{R2}	3.1 ± 0.5	4.5 ± 0.6	6 ± 1.3	8 ± 1.4

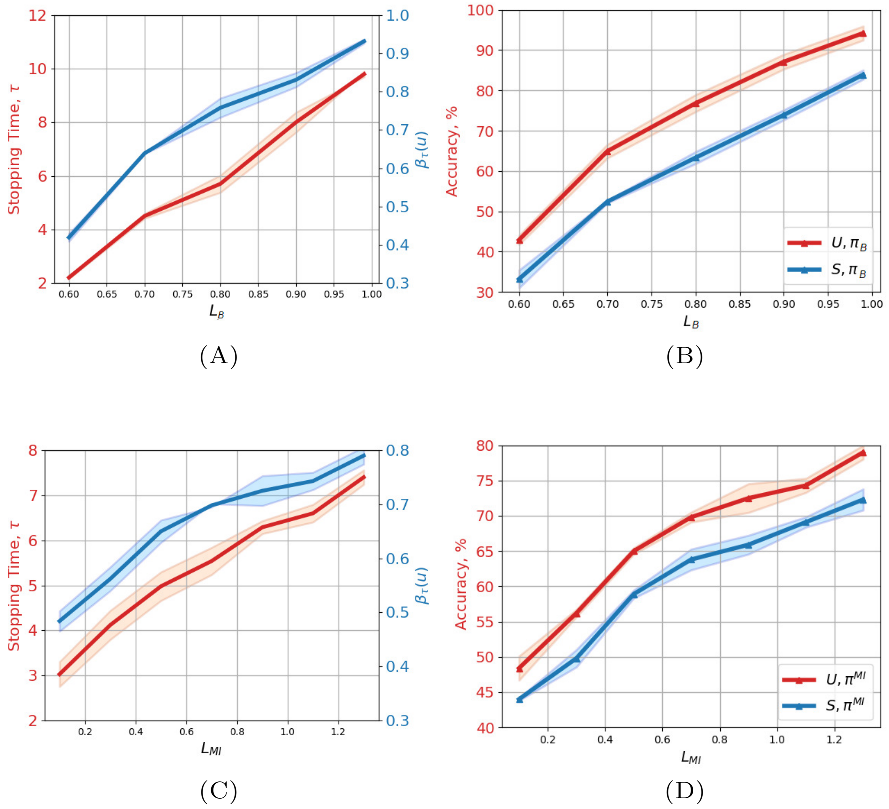
We represent the belief-reward and MI utility policies by, π_β and π_I, respectively. We observe that through the proposed active release mechanism, the useful information can be shared with high confidence while keeping the SP relatively confused about the secret. We conclude from the results that maximizing MI provides more information about the set of hypothesis U than maximizing $\beta_\tau(U)$; however, it does not directly reveal the true hypothesis as much as π_β reveals. However, π_I still performs relatively close to the belief-reward policy π_β for $\beta_\tau(U)$ at higher L_s. Although the random policy provides simplicity for action selection, it has no control on the UP's confidence on the useful hypothesis. Hence, π_{R1} and π_{R2} perform poorly for both $\beta_\tau(U)$ and MI as expected since they do not use the observations to determine the best actions.

Note that we have not explicitly considered τ as part of our optimization. In theory, we allow unlimited time steps as long as the confidence bound on the secret is not violated. On the other hand, since the confidence level on $\mathcal{S}$ monotonically increases with time, the user stops revealing data after a finite number of steps. In Table 5.1, we observed that τ follows an increasing trend as the constraint on the secret is relaxed. For π_{MI}, we observed shorter decision times, which means that MI-maximizing actions also reveal more about the secret. For π_{R1} and π_{R2}, on the other hand, we observed much shorter decision times. Random policies end up choosing actions that leak significant amount of information about the secret without providing much utility.

5.4.2 Active Quickest Private Data Sharing: Synthetic Data Use-Case

This section presents the synthetic data scenario of the policies proposed in Section 5.3. This scenario represents the situations where the probability distributions of DRMs and belief update rules are known by both the user and the SP, while only the actions are learned by the privacy mechanism.

We create a dataset for $|\mathcal{A} \cup \{d\}|$=4, $|\mathcal{S}|$=3, $|\mathcal{U}|$=3, $|\mathcal{Z}|$=50 and uniformly distributed S and U, and $L_B \in \{0.6, 0.7, 0.8, 0.9, 0.99\}$. Observation probabilities are selected such that each action distinguishes a different pair of hypotheses well for both S and U. For example, we created a matrix with each row representing the conditional distribution



(A) (B)

(C) (D)

FIGURE 5.4: Belief-PDRP's, π_B, (A) stopping time τ and $\beta(u)$, and (B) SP's accuracy for the secret and the useful information with respect to L_B, and MI-PDRP's, π_{MI}, (C) stopping time τ and $\beta(u)$, and (D) SP's accuracy for the secret and the useful information with respect to L_{MI}.

of z for different (a, s, u) realizations. For sensor $a = 0$, we used $\mathcal{N}(0, \sigma_j)$ for $(s, u) = \{(0,0), (0,1), (0,2), (1,0), (1,1), (1,2)\}$, $\mathcal{N}(1, \sigma_j)$ for $(s, u) = (2, 0)$, $\mathcal{N}(2, \sigma_j)$ for $(s, u) = (2, 1)$, and $\mathcal{N}(3, \sigma_j)$ for $(s, u) = (2, 2)$, and we normalized through the columns representing z. Here, σ_j's are chosen randomly from the interval $[0.5, 1.5]$ for each (a, s, u) with index j=$\{1, .., N{\times}M{\times}|A|\}$. This sensor discloses s=2 case more than the other secrets. Moreover, a=1 and a=2 reveal more information for s=1 and s=0 cases, respectively.

Figure 5.4A shows the average stopping time τ and the maximum belief on u, $\beta(u)$, with respect to L_B for the belief-PDRP, π_B. As the constraint on $\beta(s)$ is relaxed, the stopping time increases as well as the maximum $\beta(u)$. In Figure 5.4B, on the other hand, we present the prediction accuracy of the true-useful activity u from the belief calculation. Red lines in Figure 5.4B represent accuracy on u, and blue lines show the accuracy on s. The gap between the accuracy shows the effectiveness of the proposed policy π_B in minimizing the SP's error probability of u in the quickest way while keeping his confidence in s below the threshold for the synthetic data.

Figure 5.4C shows the average stopping time τ and the maximum confidence in u, $\hat{\beta}(u)$,

Table 5.2: Selected Activities and Smartwatch Sensors from Smoking Activity Dataset.

Sensors:	**A**	**Activities:**	**(S,U)**
Accelerometer	0	Sitting	(0,0)
Gyroscope	1	Standing	(0,1)
Magnetometer	2	Walking	(0,2)
Linear-accelerometer	3	Sitting while smoking	(1,0)
		Standing while smoking	(1,1)
		Walking while smoking	(1,2)
		Sitting while drinking	(2,0)
		Standing while drinking	(2,1)

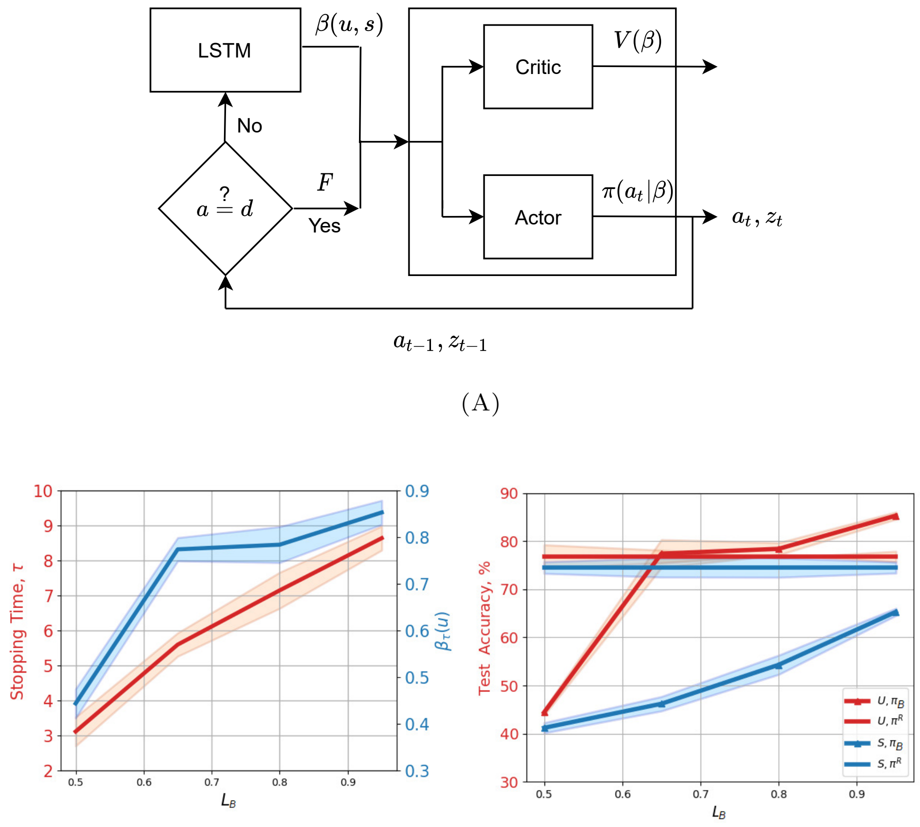
with respect to L_{MI} for the MI-PDRP, π_{MI}. As before, when the constraint on MI is relaxed, the stopping time increases as well as the maximum $\hat{\beta}(u)$. In Figure 5.4D, red lines represent accuracy on u, and blue lines show the accuracy on s. Although π_{MI} shows similar results with π_B, π_B is more effective in hiding the true realization of S. This is because MI-PDRP provides PUT by constraining the statistics of all the realizations of S rather than only the true realization.

5.4.3 Active Quickest Private Data Sharing: Activity Data Use-Case

In this section, we present the numerical evaluation for the policies proposed in Section 5.3 using real time-series measurements. In human activity privacy scenario, we use *smoking activity dataset* [119] which contains more than 40 hours of sensor measurements for activities, such as smoking while walking, drinking while standing, sitting etc. We use measurements from four selected sensors of a smartwatch, i.e., $|\mathcal{A} \cup \{d\}| = 5$. Table 5.2 shows these sensors and sensitive-useful activity pairs from the dataset. We learn the probability distributions together with the actions from the real-world measurements.

5.4.3.1 Numerical Results for Belief-PDRP, π_B

In this section, we evaluate the PUT of the proposed optimal policy π_B for smoking activity dataset. We model the SP by a long short-term memory (LSTM) recurrent neural network with parameters ϕ, which predicts the true useful variable u and secret s. The LSTM-based predictor has 2 layers with 128 nodes and 2 look-backs, and inputs the past observations $\{z^{t-1}, a^{t-1}\}$. The output is a probability distribution representing the belief vector $\hat{\beta}_\phi(S,U)$ obtained by minimizing a cross-entropy loss between $\hat{\beta}_\phi(S,U)$ and true values of $\{S,U\}$. This is equivalent to maximizing the log-likelihood of $\hat{\beta}_\phi(S,U)$,



(A)

(B)

(C)

FIGURE 5.5: (A) A2C-DRL process for belief-PDRP, π_B, (B) stopping time τ and $\hat{\beta}(u)$, and (C) SP's accuracy for the secret and the useful information with respect to L_B.

i.e.,

$$H(\beta, \hat{\beta}) = -\sum_{s,u} \beta(s, u) \log(\hat{\beta}(s, u)) = -\mathbb{E}_{s,u}[\log(\hat{\beta}(s, u))]$$

To train the LSTM SP beforehand, we split the training data into 3 portions. One is for pre-training the LSTM SP, which will be used during A2C-DRL, one is for online A2C-DRL training, and the last portion is to train an SP, i.e., LSTM predictor, for testing the performance of PUT with A2C-DRL. Let π_R be a random policy with uniform action probabilities. We create observation pairs $\{Z_t, A_t\}$ for LSTM training by randomly sampling actions A_t from π_R, and obtaining time-series Z_t from the corresponding portion of the dataset. We also used $C_T = 0.5$ for the time cost.

Figure 5.5A shows A2C-DRL process in which LSTM is used as an online state predictor from the past observations. The user checks if the termination action, i.e., $a_{t-1} = d$, has been taken, then she accordingly terminates the process. Otherwise she predicts the current belief with the LSTM network, and selects an action a_t via the actor. The actor-critic network updates its parameters with the state value $V(\beta)$ and action probability

$\pi(a_t|\beta)$ accordingly. Sensor measurement z_t is observed as per the selected action, and the observation pair z_t, a_t is shared with the SP.

Figure 5.5B shows the average stopping time τ and the predicted maximum belief on u, $\hat{\beta}(u)$, with respect to L_B for the belief-PDRP, π_B. As the constraint on $\hat{\beta}(s)$ is relaxed, the stopping time increases as well as the maximum $\hat{\beta}(u)$. In Figure 5.5C, on the other hand, no-PUT and PUT cases are compared in terms of prediction accuracy of the SP on true-useful activity u and the secret s, where accuracy of the SP for the randomly generated A_t and corresponding Z_t represents no-PUT case, while its accuracy for the A2C-DRL generated actions A_t and Z_t represents the PUT case. Red lines in Figure represent accuracy on u, and blue lines show the accuracy on s. The flat lines show the no-PUT case which does not depend on L_B, and the curved lines represent the PUT case. While the gap between the accuracy of u and s is very low for random policy (no-PUT case), it is very large for π_B (PUT case). This shows the effectiveness of the proposed policy π_B in minimizing the SP's error probability of u in the quickest way while keeping his confidence in s below the threshold. On the other hand, generating random actions from a random policy does not yield a sophisticated strategy to reveal u and hide s. The largest gap, i.e., the best performance of π_B, occurs at $L_B = 0.65$ for π_B.

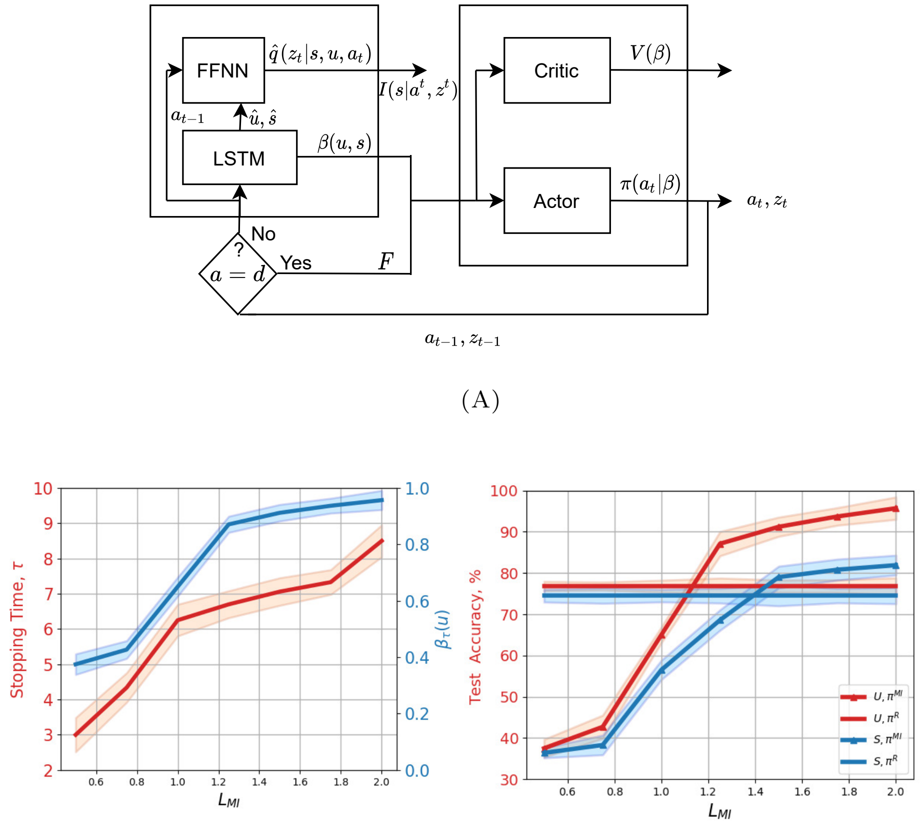
5.4.3.2 Numerical Results for MI-PDRP, π_{MI}

In this section, we model the SP using two components; one is an LSTM-based belief predictor with 2 layers of 128 nodes and 2 look-backs, and the other one is a feed-forward neural network (FFNN)-based observation generator with 3 layers of 256 nodes, where the output determines the mean μ and standard deviation σ of a Gaussian distribution. As before, we use $C_T = 0.5$ for the time cost.

As in Section 5.4.3.1, we train the LSTM network with parameters ϕ by minimizing a cross-entropy loss between the observations $\{Z^{t-1}, A^{t-1}\}$ and $\{S, U\}$, which is equivalent to maximizing the log-likelihood of $\hat{\beta}_\phi(S, U)$. As a result, KL divergence between the real belief distribution β and the predicted distribution $\hat{\beta}_\phi$ goes to zero when the log-likelihood is maximized [127]. In addition, we estimate $q(Z_t|A_t, S, U)$, which is represented by a Gaussian distribution,

$$\hat{q}(Z_t|A_t, S, U) = \mathcal{N}(Z_t|(\mu, \Sigma)) = f_\psi(A_t, S, U), \tag{5.33}$$

where (μ, σ) are determined by an FFNN f_ψ by maximizing its log-likelihood. During A2C-DRL, we sample observations Z_t and A_t to calculate the variational bound for MI using the pre-trained FFNN and LSTM networks which satisfy the maximization in (5.32). We approximate the MI by sampling k observations $\{z_t^i, a_t^i\}_i^k \sim \hat{q}(z_t|a_t, \hat{s}, \hat{u}), \pi_{MI}(a_t|\hat{\beta})$,



(A)

(B)

(C)

FIGURE 5.6: (A) A2C-DRL process for MI-PDRP, π_{MI}, (B) stopping time τ and $\hat{\beta}(u)$ and (C) SP's accuracy for the secret and the useful information with respect to L_{MI}.

and using the predictions for the next k belief states $\{Q(s|z_t^i, a_t^i, \hat{\beta})\}_i^k$ as follows:

$$\hat{I}(S; Z_t, A_t|\phi, \psi) = H(\hat{\beta}_\phi) + \frac{1}{n}\sum_{j=1}^{n}\Big[\frac{1}{k}\sum_{i=1}^{k}\log[Q_\psi((\hat{s}^j|z_t^i, a_t^i, \hat{\beta}_\phi))]\Big], \tag{5.34}$$

where $\hat{s}^j$ is a realization of s sampled from the predicted belief vector $\hat{\beta}_\phi(s)$. Figure 5.6A illustrates A2C-DRL process with belief and MI calculation using pre-trained LSTM and FFNN. The user checks if the termination action, i.e., $a_{t-1} = d$, has been taken. If so, she accordingly terminates the process. Otherwise, she predicts the current belief from the previous observations using the LSTM network, and takes action a_t. The actor-critic network updates its parameters with the state value $V(\beta)$ and action probability $\pi(a_t|\beta)$ accordingly. Sensor measurement is observed as per the selected action, and the observation pair z_t, a_t is shared with the SP. $\hat{I}(\hat{S}|A^t, Z^t|\beta_t)$ is calculated by the SP using previous action a_{t-1} and $(\hat{s}, \hat{u})$ according to (5.34).

Figure 5.6B shows the average stopping time τ and the maximum confidence in u, $\hat{\beta}(u)$, with respect to L_{MI} for the MI-PDRP, π_{MI}. As the constraint on MI is relaxed, the stopping time increases as well as the maximum $\hat{\beta}(u)$. In Figure 5.6C, activity prediction accuracy of the SP for observations (Z_t, A_t) generated by random policy π_R and π_{MI}

TABLE 5.3
Adversary Accuracy for all Activities Under Belief-privacy and MI-privacy Policies.

Policy	**Constraint**	$\tau/\hat{\beta}(U)$	**Acc. U**	**Acc. U=0 / U=1 / U=2**	**Acc. S**	**Acc. S=0 / S=1 / S=2**
π_B	0.5	3.12 / %43	%44.4	%60.67 / %53.44 / %5.65	%41.2	%24.58 / %23.6 / %4.58
	0.65	5.6 / %74	%77.4	%78.9 / %69.15 / %88.21	%46.2	%55.32 / %48.05 / %5.15
	0.8	7.15 / %77	%78.4	%85.07 / %61.99 / %92.52	%54.3	%68.77 / %48.95 / %34.5
	0.95	8.64 / %81	%85.3	%91.58 / %71.23 / %96.3	%65.3	%79.48 / %73.07 / %42.8
π_{MI}	0.5	3 / %38	%37.5	%47.01 / %51.1 / %0	%36.4	%46.53 / %52.01 / %0.74
	0.75	4.34 / %40	%42.7	%63.29 / %49.17 / %0	%38.3	%47.26 / %53.74 / %1.2
	1	6.25 / %62	%65	%87.26 / %55.35 / %46.7	%56.5	%47.48 / %57.42 / %13.83
	1.25	6.7 / %88	%87.1	%94 / %80.8 / %80.86	%68.6	%89.55 / %65.63 / %28.74
	1.5	7.06 / %91	%91.2	%97.62 / %84 / %92.16	%79	%93.02 / %72.22 / %34.67
	1.75	7.8 / %93	%93.7	%98.42 / %88.8 / %96.25	%80.8	%96.9 / %96.38 / %87.3
	2	8.5 / %94	%95.7	%99.45 / %96.19 / %97.78	%81.9	%98,3 / %98.12 / %92.64

are compared. Red lines in Figure 5.6C represent accuracy on u, and blue lines show the accuracy on s. Similarly to Section 5.4.3.1, the gap between the accuracy of u and s is very low for random policy, while it is large for π_{MI}. This shows that the proposed policy π_{MI} minimizes the SP's error probability of u in a speedy manner while keeping the information leakage from s below the threshold. Although π_{MI} shows similar results with π_B, π_B is more effective in hiding the true realization of S. This is because MI-PDRP provides PUT by constraining the statistics of all the realizations of S rather than only the true realization. The largest gap in Figure 5.6C, i.e., the best performance of π_{MI}, occurs at $L_{MI} = 1.2$ for π_{MI}.

In Table 5.3, there is detailed breakdown of performance of π_B and π_{MI} policies, where "Acc." represents accuracy. Individual accuracy for U and S show that all activities are revealed as the constraint is relaxed. On the other hand, $U = 2$ and $S = 2$ are almost completely hidden for low constraint level, but they are revealed faster then the other hypotheses. Moreover, π_B and π_{MI} policies reveal or hide different activities better due to the different characteristics of the activities. We also see the same results that Figures 5.5 and 5.6 show, i.e., π_B outperforms π_{MI} in minimizing the error probability of U in a speedy manner while keeping the secret below the pre-defined level.

5.5 Conclusions

We studied the PUT in time-series data release to an SP. In our model, the goal is to reveal the true value of a latent utility variable, while keeping the secret variable private from the SP. In a sense, the SP is the legitimate receiver for the utility variable, while acting as the adversary for the sensitive variable. In particular, we measured the utility by the confidence of the SP in the latent useful information. For privacy, we considered both the confidence of the SP on the sensitive information and the MI between the sensitive variable and the revealed measurements. We proposed active sequential data release policies to minimize the error probability on the true useful variable in a speedy manner, while constraining the confidence of the SP or the MI leakage for the secret variable. We provided a POMDP formulation of the problem, and used A2C-DRL for numerical evaluations. Utilizing DNNs, we numerically evaluated the PUT curve of the proposed policies for *smoking activity dataset*, where useful and sensitive activities are revealed to the adversary through smartwatch sensors selected by the user. We examined the effectiveness of the optimal belief-PDRP and MI-PDRP using an LSTM-based adversary network. According to the numerical results, we have seen that the proposed data release policies provide significant privacy advantage compared to random sensor selection. We have also seen that constraining the MI does not necessarily hide the true value of the secret at the same level as the belief-PDRP. However, this approach may be more useful when the objective is not necessarily to hide the true value of the secret, but limit the knowledge of the SP in an average sense. We have also shown that decision time gets longer when the constraint on the secret is relaxed.

Chapter 6

Privacy Aware Communication Over a Wiretap Channel

In this chapter, we study privacy-aware communication over a wiretap channel using end-to-end learning. Differently from the previous chapters, here, the noisy channel characteristics are exploited for privacy preserving. For instance, Alice wants to transmit a source signal to Bob over a binary symmetric channel, while passive eavesdropper Eve tries to infer some sensitive attribute of Alice's source based on its overheard signal. Since we usually do not have access to true distributions, we propose a data-driven approach using VAE-based JSCC. We show through simulations with the colored MNIST dataset that our approach provides high reconstruction quality at the receiver while confusing the eavesdropper about the latent sensitive attribute, which consists of the color and thickness of the digits. Finally, we consider a parallel-channel scenario, and show that our approach arranges the information transmission such that the channels with higher noise levels at the eavesdropper carry the sensitive information, while the non-sensitive information is transmitted over more vulnerable channels.

6.1 Introduction

As mentioned in the previous chapters, secrecy and privacy in data communication and data sharing systems have been studied extensively in the literature [13–15, 18, 54, 57, 120, 121, 124, 128, 130, 131]. Although deep learning applications of data transmission have also been well investigated, deep learning in wireless communications and physical layer security has only recently become popular [132, 133]. The similarity between the communication systems and end-to-end learning motivates the use of autoencoder based neural network architectures, which simultaneously learn encoding and decoding [133, 134]. Recently, it has been shown that end-to-end approaches can also be utilized for physical layer secrecy [135–138]. In a wiretap channel setting, these techniques exploit the physical characteristics of the legitimate receiver's channel over the eavesdropper's, and allow communication with secrecy guarantees.

In this chapter, we consider a wiretap channel scenario in which Alice wants to deliver its source, S^m, to Bob over a noisy communication channel, while a passive eavesdropper Eve tries to infer a latent sensitive information T about S^m. For example, S^m may be an image or a video captured by Alice, while T may be the presence of a particular object or an activity within the scene. We assume BSCs from Alice to both Bob and Eve. The aim is to optimize the trade-off between the reconstruction distortion of source S^m at Bob and the privacy leakage of T to Eve, which is measured by the MI between the sensitive information and the noisy codewords observed by Eve. Note that, the wiretap channel model considered here is normally studied in the context of secure communications. Indeed, when $T = S^m$, our problem becomes a special case of the one studied in [139]. We, instead, call this *privacy-aware communications* since secrecy typically focuses on making the information leakage negligible, while privacy tolerates some leakage in return of utility [140]. Hence, we propose a PUT for communication over the wiretap channel by balancing the information leakage to Eve and the distortion at the legitimate receiver, i.e., Bob. We highlight that in the special case of identical channels to Bob and Eve, our problem also reduces to the well-known privacy funnel [121] with a noisy communication channel. In that scenario, Bob and Eve merge into a single receiver, to which we want to send S^m with the highest fidelity while hiding T. Therefore, our problem generalizes both the wiretap channel and the privacy funnel problems. Additionally, unlike in [139] and [121], we follow a data-driven approach by using an encoder-decoder pair, represented by a VAE network and a classifier which represents Eve.

Similar data-driven wiretap channel approaches have recently been proposed for Gaussian channels in [135–138]. However, [135, 137, 138] focus on channel coding, and [137, 138] enforce coding structure to the encoder, while we carry out end-to-end joint learning corresponding to a JSCC approach. In addition, unlike these works, we are interested in hiding an underlying sensitive information that is correlated with, but different from the original signal. The same problem is considered in [136] for an additive white Gaussian channel using a generative adversarial network (GAN), which minimizes the distortion of the reconstructed signal at the legitimate receiver while characterizing the privacy with a constraint on the likelihood of the sensitive information. On the other hand, we propose a PUT for a BSC wiretap channel using a VAE-based neural network architecture.

VAEs provide several advantages in this framework compared to standard autoencoders [134]. They embed the input to a distribution rather than a point, and a random channel input is sampled from the latent distribution rather than being generated by the encoder directly. Hence, VAEs are more aligned with the stochastic encoding approach employed in information theoretic derivation of the wiretap channel capacity [57, 139]. Additionally, VAEs provide significant control over how to model the latent distribution, since the encoder is designed as a generative network. This is difficult to achieve within the autoencoder framework, and also allows a tractable calculation of the variational approximations of our cost function based on MI. Last but not least, it is challenging to optimize autoencoders for communication over discrete channels due to their non-differentiability, whereas sampling discrete codewords from a latent distribution is possible for VAEs.

We apply our approach to privacy-aware image transmission and show that while the receiver can reconstruct high quality images, the eavesdropper is confused about the sensitive information. We also consider a parallel-channel case in which Bob and Eve might experience different noise levels over each channel. We show that our end-to-end approach judiciously adjusts its transmission to exploit the more secure channels to transmit the sensitive information.

6.2 System Model

We consider a communication scenario in which a user wants to reliably transmit data from one point to another over a noisy communication channel, while a passive eavesdropper tries to infer a latent sensitive information through its noisy observation of the transmitted signal. Figure 6.1 illustrates the communication problem via a simple example. Alice wants to reveal her data $S^m \in \mathcal{S}$, e.g., images of the applicants for a certain job position, to Bob over a noisy channel. Eve eavesdrops through her own channel and receives a noisy version of the transmitted signal by Alice. Eve's goal is to extract Alice's sensitive information $T \in \mathcal{T}$, e.g., ethic or socioeconomic background of the applicants, which is correlated with S^m but not explicitly observed by any of the involved parties. Alice's goal, on the other hand, is to encode the source such that it can be reconstructed by Bob with high fidelity, while the sensitive information T cannot be accurately detected by Eve. The source is encoded into codewords $X^n \in \{x_1, \dots, x_n\}$, where $X_i \in \mathcal{X} = \{0, 1\}$, by a stochastic encoding function $f_{enc}(S^m) = X^n$ represented by a conditional distribution $P(X^n|S^m)$. Although the source S^m and the sensitive information T are correlated, encoding function depends only on S^m and not on T since the realizations of T are not available to any of the parties at the inference time. In other words, Alice is aware that Eve is interested in the sensitive information T, however, she cannot utilize T in encoding due to the lack of labels. This setting favors the eavesdropper, and hence presents a more difficult problem.

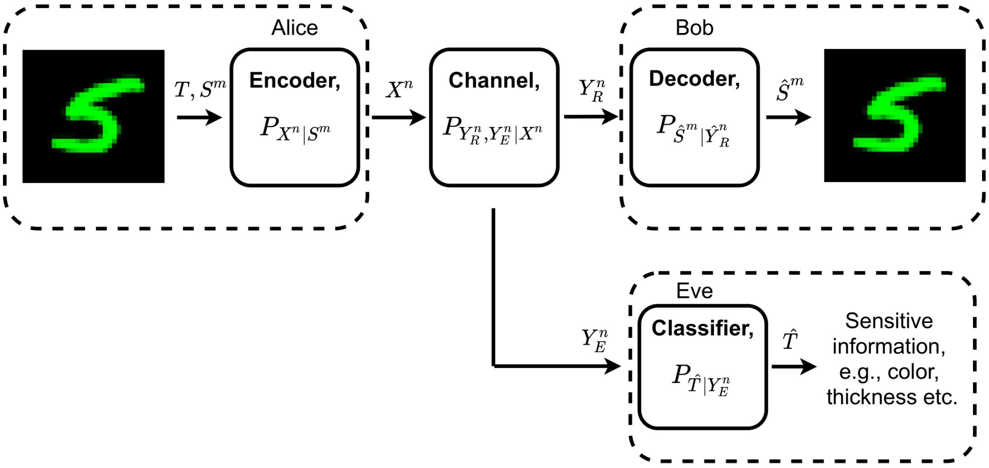


FIGURE 6.1: Communication system with wiretap channel.

We consider a BSC characterized by the joint conditional distribution $P(Y_B^n, Y_E^n|X^n)$, $Y_{B,i}, Y_{E,i} \in \mathcal{X}$. The noisy codeword received by Bob is decoded as $f_{dec}(Y_B^n) = \hat{S}^m$, and Eve receives its own noisy observation Y_E^n.

We model the joint distribution of $T, S^m, X^n, Y_B^n, \hat{S}^m$, i.e., the r.v.'s for the sensitive information, source signal, transmitted codeword, noisy codeword received by Bob, and his reconstruction, respectively, using the following graphical model $T - S^m \to X^n \to Y_B^n \to \hat{S}^m$ as:

$$P(T, S^m, X^n, Y_B^n, \hat{S}^m) = P(T, S^m)P(X^n|S^m)P(Y_B^n, Y_E^n|X^n)P(\hat{S}^m|Y_B^n). \tag{6.1}$$

The two BSCs independently flip each bit in the transmitted codeword with crossover probabilities ϵ_B and ϵ_E at Bob's and Eve's channels, respectively. Hence, the joint probability of the channel can be decomposed as follows:

$$P(Y_B^n|X^n) = \prod_{i=1}^{n} \epsilon_B^{x_i \oplus y_{B,i}} (1 - \epsilon_B)^{x_i \oplus y_{B,i} \oplus 1}, \tag{6.2}$$

$$P(Y_E^n|X^n) = \prod_{i=1}^{n} \epsilon_E^{x_i \oplus y_{E,i}} (1 - \epsilon_E)^{x_i \oplus y_{E,i} \oplus 1}, \tag{6.3}$$

where $\oplus$ represents the exclusive OR operation, and x_i, $y_{B,i}$ and $y_{E,i}$ are the i^{th} bits of X^n, Y_E^n and Y_B^n, respectively.

We formulate the optimization problem as

$$\begin{aligned} \underset{f_{enc}, f_{dec}}{\text{minimize}} \quad & \mathbb{E}[d(S^m, \hat{S}^m)] - I(S^m; Y_B^n) + \lambda I(T; Y_E^n) \\ \text{subject to} \quad & T, S^m \to X^n \to Y_B^n \to \hat{S}^m, \end{aligned} \tag{6.4}$$

where λ is the tuning parameter for the privacy level. Here, in addition to the reconstruction distortion between S^m and $\hat{S}^m$, measured by $d(\cdot,\cdot)$, we also maximize the MI between the user's data S^m and the noisy codewords observed by Bob, i.e., $I(S^m;Y_B^n)$ for improved utility. While minimizing the distortion $\mathbb{E}[d(S^m,\hat{S}^m)]$ improves pixel-wise data reconstruction quality, we have observed in our simulations that maximizing the MI between the source signal and Bob's channel output enhances the information flow and helps with capturing the high level features at the receiver side.

Exact calculation of the MI is difficult when the data distribution is not known. Hence, we approximate $I(S^m;Y_B^n)$ and $I(T;Y_E^n)$ via their variational representations [127]. Due to the intractability of the true posteriors $P(S^m|Y_B^n)$ and $P(T|Y_E^n)$, we use their amortized variational approximations $f_{enc}(Y_B^n) = P(\hat{S}^m|Y_B^n)$ and $f_{eve}(Y_E^n) = P(\hat{T}|Y_E^n)$, respectively. Here, we assume that the eavesdropper tries to predict the sensitive information T as $f_{eve}(Y_E^n) = \hat{T}$. We can write $I(S^m;Y_B^n)$ as follows:

$$\begin{aligned} I(S^m;Y_B^n) &= H(S^m) - H(S^m|Y_B^n) && (6.5)\\ &= H(S^m) + \mathrm{KL}(P(S^m|Y_B^n)||f_{dec}(Y_B^n)) + \mathbb{E}[\log f_{dec}(Y_B^n)] && (6.6)\\ &\geq H(S^m) + \max_{f_{dec}} \mathbb{E}[\log f_{dec}(Y_B^n)], && (6.7)\end{aligned}$$

where $\mathrm{KL}(\cdot\|\cdot)$ denotes the KL divergence, $H(S^m)$ is constant, (6.5) follows from the definition of MI, (6.6) holds for any distribution $f_{dec}(Y_B^n)$ over S^m given the values in Y_B^n. Finally, (6.7) follows from the fact that maximum is attained when the decoder is optimum, i.e., $f_{dec}(Y_B^n) = P(S^m|Y_B^n)$. Likewise, the information leakage to the eavesdropper becomes

$$\begin{aligned} I(T;Y_E^n) &= H(T) - H(T|Y_E^n) && (6.8)\\ &= H(T) + \mathrm{KL}(P(T|Y_E^n)||f_{eve}(Y_E^n)) + \mathbb{E}[\log f_{eve}(Y_E^n)] && (6.9)\\ &\geq H(T) + \max_{f_{eve}} \mathbb{E}[\log f_{eve}(Y_E^n)], && (6.10)\end{aligned}$$

where $H(T)$ is a constant term, (6.8), (6.9) and (6.10) follow similarly to (6.5), (6.6) and (6.7), respectively. Here, (6.7) is attained when the decoder is optimum since we maximize $I(S^m;Y_B^n)$ in our objective. However, (6.10) is not attained even if the classifier representing the eavesdropper is optimum, because we minimize $I(T;Y_E^n)$ in the objective. This is due to intractability of representing $I(T;Y_E^n)$ with an upper-bound [141]. On the other hand, our numerical results indicate that although we do not optimize exact bounds for MI terms, in practice our model still learns an effective PUT.

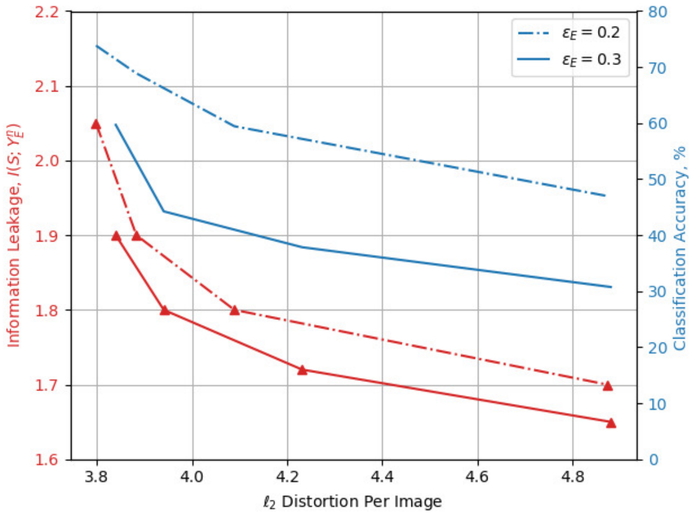


FIGURE 6.2: PUT curve of our privacy-aware JSCC mechanism for $\epsilon_B = 0.1$ and $\epsilon_E = \{0.2, 0.3\}$.

6.2.1 Parallel-Channel Scenario

In this section, we assume the codewords are transmitted over parallel channels with different noise levels, e.g., due to OFDM. Our setting represents the scenario in which the transmitter divides the total available bandwidth into non-overlapping bands carrying separate portions of the data. Each of the parallel bands face a different noise level for both the receiver and the eavesdropper, i.e., $(\epsilon_{B_i}, \epsilon_{E_i})$. For instance, in a three-channel scenario with equal bandwidths $n/3$, crossover probabilities $\epsilon_B = \{\epsilon_{B_1}, \epsilon_{B_2}, \epsilon_{B_3}\}$ and $\epsilon_E = \{\epsilon_{E_1}, \epsilon_{E_2}, \epsilon_{E_3}\}$, channel probabilities can be written as

$$P(Y_B^n|X^n) = \prod_{i=1}^{\frac{n}{3}} \epsilon_{B_1}^{x_i \oplus y_{B,i}} (1-\epsilon_{B_1})^{x_i \oplus y_{B,i} \oplus 1} \prod_{j=\frac{n}{3}+1}^{\frac{2n}{3}} \epsilon_{B_2}^{x_j \oplus y_{B,j}} (1-\epsilon_{B_2})^{x_j \oplus y_{B,j} \oplus 1} \\ \times \prod_{k=\frac{2n}{3}+1}^{n} \epsilon_{B_3}^{x_k \oplus y_{B,k}} (1-\epsilon_{B_3})^{x_k \oplus y_{B,k} \oplus 1} \tag{6.11}$$

for the receiver, and as follows for the eavesdropper:

$$P(Y_E^n|X^n) = \prod_{i=1}^{\frac{n}{3}} \epsilon_{E_1}^{x_i \oplus y_{E,i}} (1-\epsilon_{E_1})^{x_i \oplus y_{E,i} \oplus 1} \prod_{j=\frac{n}{3}+1}^{\frac{2n}{3}} \epsilon_{E_2}^{x_j \oplus y_{E,j}} (1-\epsilon_{E_2})^{x_j \oplus y_{E,j} \oplus 1} \\ \times \prod_{k=\frac{2n}{3}+1}^{n} \epsilon_{E_3}^{x_k \oplus y_{E,k}} (1-\epsilon_{E_3})^{x_k \oplus y_{E,k} \oplus 1}. \tag{6.12}$$

We solve (6.4) using the channel probabilities (6.11) and (6.12). We want our solution for (6.4) to control the transmission through the channels such that the sensitive information

T is transmitted over the channels in which Eve experiences high noise, while the rest of the source is transmitted over the channels Bob experiences low noise, independent of Eve's channel. We numerically verify that the proposed VAE-based encoder indeed satisfies these expectations.

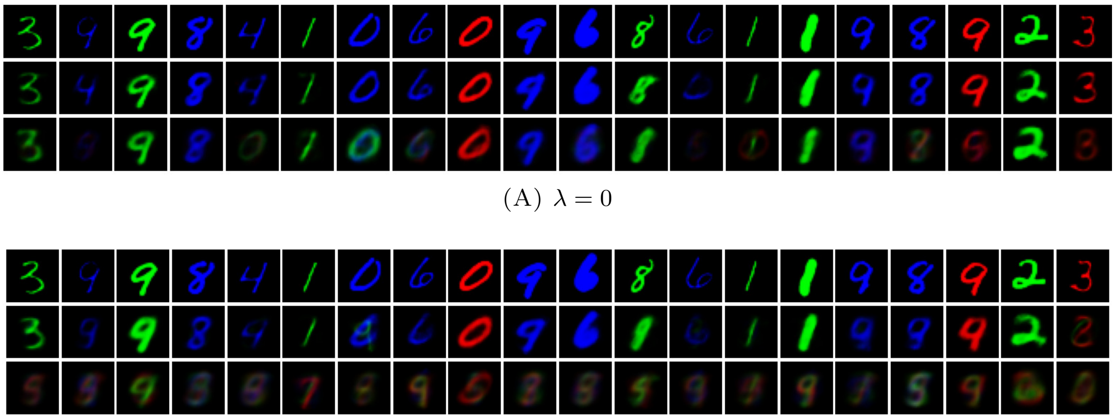
6.3 Numerical Results

We consider the wiretap channel in Figure 6.1, where the encoder and decoder at Alice and Bob are represented by a VAE, while Eve employs a classifier. For the encoder-decoder pair, we employed the network structure "NECST" proposed in [142]. We designed our privacy aware JSCC network by incorporating our classifier based eavesdropper in NECST. We used colored MNIST handwritten digits as S^m for $m = 32 \times 32$ pixels, and color and thickness of the digits as the sensitive information $T \in \mathcal{T} = \{(R, 0), (R, 1), (R, 2), (G, 0), (G, 1), (G, 2), (B, 0), (B, 1), (B, 2)\}$, where R, G and B denote red, green and blue colors, while 0, 1 and 2 represent thin, medium and thick digits, respectively. We set the total channel bandwidth to n=200 bits.

6.3.1 Single Channel

We first consider a single channel scenario. In Figure 6.2, information leakage $I(T; Y_E^n)$ and Eve's classification accuracy are shown with respect to ℓ_2 distortion per image. Dashed and straight lines represent the cases with $\epsilon_E = 0.2$ and $\epsilon_E = 0.3$, respectively, while we have $\epsilon_B = 0.1$ for both cases. Data points are taken at $\lambda = \{0, 5, 10, 20\}$. Figure 6.2 shows that the information leakage about the sensitive information decreases as the image distortion increases, which is expected due to the PUT. Moreover, noisier eavesdropper channel leaks less information at the same level of distortion. Similar trend can be seen for Eve's accuracy. In Figure 6.2, we also observe that a MI gap as small as 0.06 corresponds to 20% accuracy gap between $\epsilon_E = 0.2$ and $\epsilon_E = 0.3$ cases.

For illustration purposes, we trained an additional decoder on the noisy bits received by Eve (Y_E^n) with the same structure as Bob's decoder. Figure 6.3 depicts the original images, reconstructed images by Bob and Eve, respectively, from top to bottom. We can see that in the absence of privacy ($\lambda = 0$), both Bob and Eve can reconstruct the images rather accurately, while, thanks to the employed JSCC approach, Bob's better channel allows it to have better fidelity. On the other hand, when privacy is imposed ($\lambda = 20$), we can see that Eve cannot recover neither the colour nor the thickness information. On the other hand, we can see that this information is available to Bob; and hence, it has been successfully hidden from Eve while being available in the transmitted signal.



(A) $\lambda = 0$

(B) $\lambda = 20$

FIGURE 6.3: Original images and their reconstructions by Bob and Eve from top to bottom, respectively, for $\epsilon_B = 0.1$, $\epsilon_E = 0.3$.

TABLE 6.1: Information leakage and Eve's classification accuracy for the sensitive r.v. $\hat{T}$ and individual sensitive attributes at each channel for $\lambda = 10$

Channels	**Ch1**	**Ch2**	**Ch3**	**Ch4**
$I(T; Y_E^n)$	1.0836	1.5703	0.0689	0.6411
Accuracy, T	13.65%	31.85%	16.15%	19.6%
Accuracy, Color	34.5%	62.35%	41.2%	45.25%
Accuracy, Thickness	35.75%	48.65%	38.05%	38.55%

6.3.2 Parallel Channels

Next, we consider a parallel-channel scenario, where the signal is transmitted over multiple channels with different noise levels. We use 4 parallel channels each with a bandwidth of $n/4 = 50$ bits. Error probability pairs for Bob's and Eve's channels are set as $(\epsilon_B, \epsilon_E) = \{\text{Ch1} : (0.1, 0.1), \text{Ch2} : (0.001, 0.2), \text{Ch3} : (0.2, 0.001), \text{Ch4} : (0.001, 0.001)\}$. Table 6.1 shows the information leakage, Eve's classification accuracy on T, and separately on the sensitive attributes *color* and *thickness* for each channel. *Accuracy, Color* and *Accuracy, Thickness* are calculated as the success of the classifications for only the *color* and only the *thickness*, respectively. Our privacy-aware generative network obtains the PUT by minimizing the information leakage of the sensitive attributes and the distortion. This leads to smaller information leakage at the best quality channel of Eve, i.e., Ch2, and larger at the worst one, i.e., Ch3. Eve's classification accuracy of T, and individual color and thickness attributes, are the highest for Ch2 and lowest for Ch1. We observed that Ch1 accuracy is low because the classifier is confused between *blue* and *green*, as well as the *medium* and *thick*, but still has high accuracy for *red* and *thin* attributes. On the other hand, Ch3 has low accuracy for all the attributes. This leads to the difference between the leakage and accuracy results for Ch1 and Ch3.

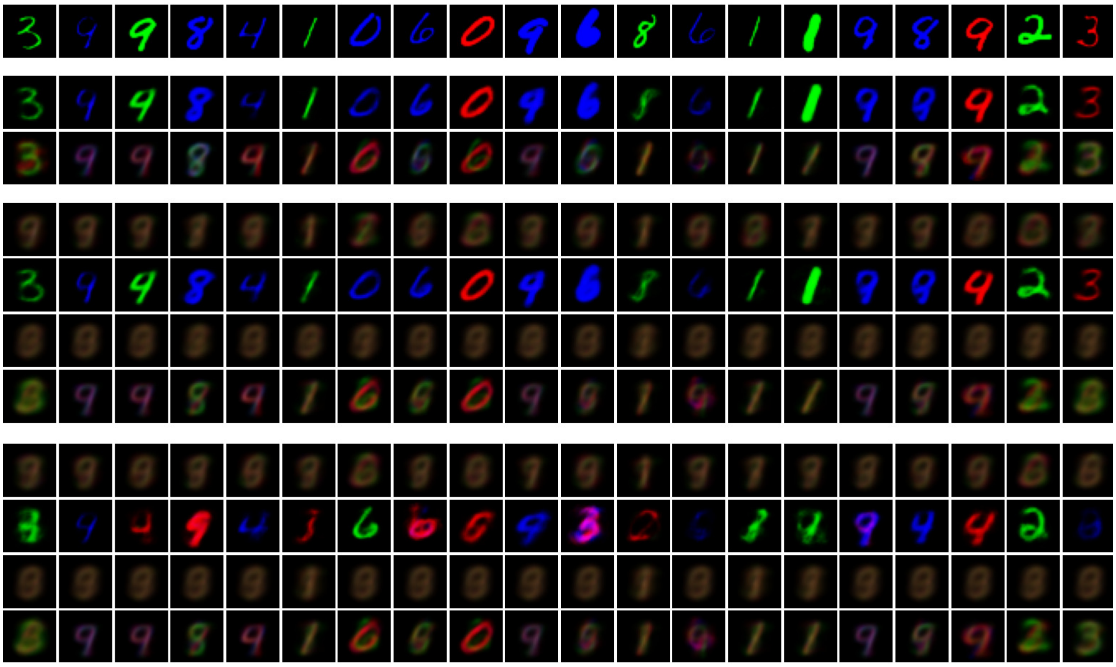


FIGURE 6.4: Original images and reconstructions by Bob, Eve, Bob's individual channels (Ch1-4), and Eve's channels (Ch1-4), respectively, from top to bottom, for $\lambda = 10$.

In Figure 6.4, we show the original and reconstructed images by Bob, Eve, Ch1 to Ch4 of Bob, and Ch1 to Ch4 of Eve, respectively, from top to bottom. First three rows show similar results with the single channel case, i.e., Eve is confused about the color and thickness of the digits, while Bob can reconstruct at high quality. Moreover, Ch1 and Ch3 do not have meaningful reconstructions for either Bob or Eve. This is because Eve faces less noise in these channels, which might lead to larger leakage. Hence, our network minimizes the information flow through these channels. Ch2, on the other hand, carries more information than Ch4 since it can better hide the sensitive attributes from Eve while maximizing the information transmission for Bob.

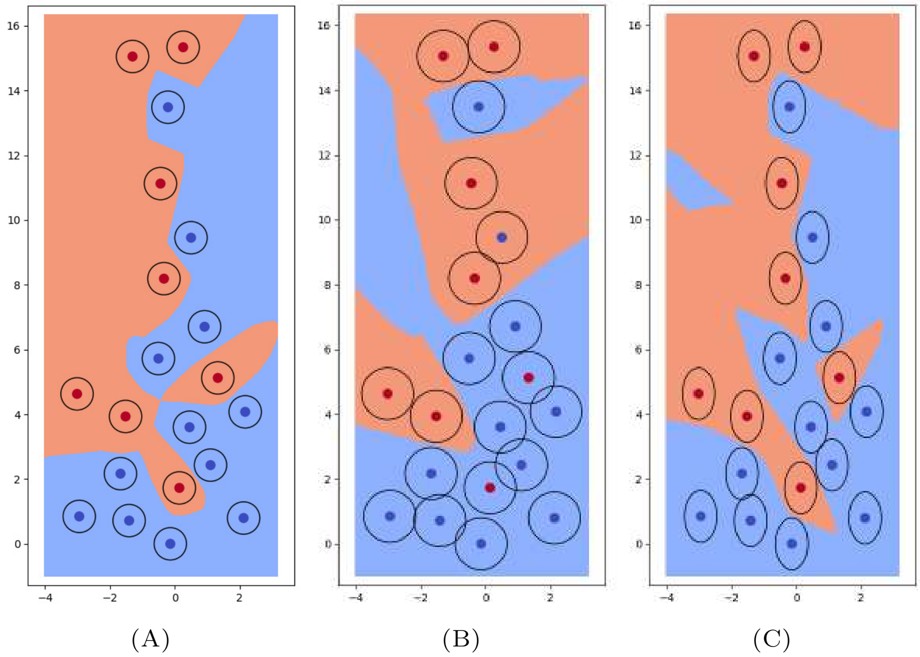
6.4 Conclusions

We proposed a VAE-based privacy-aware communication scheme over a wireless wiretap channel. In our simulation results, we showed that our end-to-end learning approach provides minimally distorted source transmission with maximum channel capacity while minimizing the information leakage about sensitive information to an eavesdropper. We also showed that our approach balances the information flow in a parallel-channel scenario such that the PUT is obtained according to the receiver's and eavesdropper's channel noises.

Chapter 7

Adversarial Robustness for Security Applications

In this chapter, we consider the robustness of DNNs in security-critical applications, such as cyber-security, finance and social networks. We move our focus from the passive adversary, e.g., the SP/UP or the eavesdropper, to an active adversary which tries to deliberately fool a neural network. Besides their effectiveness in privacy-security related applications, DNN's vulnerability to adversarial examples (AEs) has recently been an emerging topic in the literature. However, most work mainly focus on computer vision (CV) domain, while security related domains still remain under-explored. This chapter is based on the idea that despite being sufficient for CV domain, crafting AEs using uniform perturbations do not result in realistic AEs in domains such as malware, finance, and social networks. For these types of applications, features typically have some semantically meaningful dependencies. The key idea of our proposed approach is to enable non-uniform perturbations that can adequately represent these feature dependencies during adversarial training. We propose using characteristics of the empirical data distribution, both on correlations between the features and the importance of the features themselves. Using experimental datasets for malware classification, credit risk prediction, and spam detection, we show that the proposed approach is more robust to real-world attacks. Finally, we present robustness certification utilizing non-uniform perturbation bounds, and show that non-uniform bounds achieve better certification. Our code is available at https://github.com/amazon-research/adversarial-robustness-with-nonuniform-perturbations



(A) (B) (C)

Figure 7.1: Classification boundaries from adversarial training with uniform perturbation limits for (A) $\|\delta\|_2 \leq 0.5$, (B) $\|\delta\|_2 \leq 0.8$ and non-uniform perturbation limits for (C) $|\delta_x| \leq 0.5$ and $|\delta_y| \leq 0.8$. The figures are obtained by modifying [3].

7.1 Introduction

In this chapter, we mainly focus on realistic AE generation against real-world attacks in under-explored domains, such as malware, finance and social networks. In the well studied CV domain, the adversary's goal is to generate perturbed images that cause misclassifications by a DNN. It is often assumed that limiting a uniform norm-ball constraint results in perturbations that are imperceptible to the human eye. However in other applications such as fraud detection [143], spam detection [144], credit card default prediction [145, 146] and malware detection [147–149], norm-bounded uniform perturbations may result in unrealistic transformations. Perturbed samples must comply with certain constraints related to the domain, hence preventing us from borrowing these assumptions from CV. These constraints can be on semantically meaningful feature dependencies, expert knowledge of possible attacks, and immutable features [148, 150]. This chapter proposes a methodology to generate non-uniform perturbations that takes into account the characteristics of the empirical data distribution.

AT is a state-of-the-art approach for empirical defenses as mentioned in Section 2.3. Most approaches for optimizing δ perturbations usually assume that all the input features require equal levels of robustness, however, this might not be the case for many applications as mentioned earlier. Consider the 2D toy example of binary classification in Figure 7.1. Figure 7.1 illustrates adversarially robust decision boundaries with red and blue regions, and l_2-norm perturbation limits around the data points with black circles. While Figure 7.1A shows that adversarially trained model with input constraint $\|\delta\|_2 \leq 0.5$ gains complete robustness against input perturbations, in Figure 7.1B there is loss of clean performance due to overlapping regions of increased allowed perturbations. Although the constraint $\|\delta\|_2 \leq 0.5$ might provide sufficient robustness in x-axis, there are still uncovered regions in y-axis in Figure 7.1A. On the other hand, when we fit the allowable perturbations to y-axis by choosing a larger perturbation $\|\delta\|_2 \leq 0.8$, x-axis suffers from unnecessary overlaps. This can be solved by customizing the perturbation constraint such that the perturbation radius in x-axis follows $|\delta_x| \leq 0.5$ and the radius in y-axis follows $|\delta_y| \leq 0.8$, which results in an ellipsoid perturbation region in 2D as shown in Figure 7.1C. This toy example highlights the advantage of a non-uniform constraint across both axes.

Uniformly perturbing all pixels in an image is often imperceptible to the human eye, but uniform perturbations are wholly inappropriate in many tabular datasets, where positive and negative correlations are strong, consistent, and meaningful. For example, in the German dataset used in Section 7.3.2, we find the largest positive correlation (0.62) between the amount of credit and the payment duration, while the largest negative correlation (-0.31) is between the checking account status and the credit risk score. Both relationships are intuitive, and both would be broken by applying uniform perturbations.

The intuition behind the need for non-uniform constraints is apparent across many industrial applications. A common cybersecurity application is malware detection, which identifies if an executable file is benign or malicious. Unlike images, diverse and semantically meaningful features are extracted from the executable file and are passed to a machine learning model. To maintain the functionality of an executable file during an adversarial attack, certain features may be immutable and perturbations may result in an unrealistic scenario. For example in the Android malware space, application permissions, such as permission to access a phone's location service, are required for malicious functionality and cannot be perturbed [147]. In a finance scenario where customer credit card applications are evaluated by machine learning models, a possible set of features include age, gender, income, savings, education level, number of dependents, etc. In this type of dataset there are clear dependencies between features, for example the number of dependents has a meaningful correlation with age. When detecting spammers within social networks, features are extracted from accounts and may include the length of the username, length of user description, number of following and followers as well as the ratio between them, percentage of bidirectional friends, etc. Similar to the previous finance example, there is a meaningful correlation between features such as the percentage of bidirectional friends and the ratio of followers.

In all of these scenarios, non-uniform perturbations can be used to maintain these correlations and semantically meaningful dependencies resulting in more realistic AEs. In this chapter, we propose adversarial training with these more realistic perturbations to increase the robustness against real-world adversarial attacks. Specifically, our contributions are:

- Instead of considering an allowed perturbation region where all the features are treated uniformly, i.e., $\|\delta\|_p \leq \epsilon$, we consider a transformed input perturbation constraint, i.e., $\|\Omega\delta\|_p \leq \epsilon$ where Ω is a transformation matrix, which takes the available information into account, such as feature importance, feature correlations and/or domain knowledge. Hence, the transformation in the norm ball constraint results in non-uniform input perturbations over the features
- For various applications such as malware detection, credit risk prediction and spam detection, we show that robustness using non-uniform perturbations outperforms the commonly-used uniform approach
- To provide provable guarantees for non-uniform robustness, we modify two known certification methods, linear programming and randomized smoothing, to account for non-uniform perturbation constraints.

7.2 Non-uniform Adversarial Perturbations

In adversarial training, the worst case loss for an allowed perturbation region is minimized over parameters of a function representing a DNN. The objective of the adversary can be written as the inner maximization of adversarial training:

$$\underset{\delta \in \Delta_{\epsilon,p}}{\text{maximize}} \quad \ell(f_\theta(x+\delta), y), \tag{7.1}$$

where $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ are dataset inputs and labels, $\Delta_{\epsilon,p} = \{\delta : \|\delta\|_p \leq \epsilon\}$ is an ℓ_p ball of radius ϵ which defines the feasible perturbation region. Standard PGD follows steepest descent which iteratively updates δ in the gradient direction to increase the loss:

$$\delta^{t+1} = \delta^t + \alpha \frac{\nabla_\delta \ell(f_\theta(x+\delta^t), y)}{\|\nabla_\delta \ell(f_\theta(x+\delta^t), y)\|_p} \tag{7.2}$$

at iteration t, and then it projects δ to the closest point onto the ℓ_p ball:

$$\mathcal{P}_{\Delta_{\epsilon,p}}(\delta) := \underset{\delta' \in \Delta_{\epsilon,p}}{\arg\min} \|\delta - \delta'\|_2^2 = \epsilon \frac{\delta}{max\{\epsilon, \|\delta\|_p\}} \tag{7.3}$$

where the distance between δ and δ' is the Euclidean distance, and the projection corresponds to normalizing δ to have a maximum ℓ_p norm which is equal to ϵ. Now, we introduce an adversarial constraint set that non-uniformly limits adversarial variations in different, potentially correlated dimensions, by

$$\tilde{\Delta}_{\epsilon,p} = \{\delta : \|\Omega\delta\|_p \leq \epsilon\} \tag{7.4}$$

where $\Omega \in \mathbb{R}^{d\times d}$. In our approach, δ is updated by equation (7.2) similar to the standard PGD, however, it is projected back to a non-uniform norm ball satisfying $\|\Omega\delta\|_p \leq \epsilon$. The corresponding projection operator will then be:

$$\mathcal{P}_{\tilde{\Delta}_{\epsilon,p}}(\Omega\delta) = \begin{cases} \epsilon\frac{\delta}{\|\Omega\delta\|_p} & if \quad \|\Omega\delta\|_p > \epsilon \\ \delta & \text{otherwise.} \end{cases} \tag{7.5}$$

The choice of Ω depends on how we model the expert knowledge or feature relationships. The following are our choices for the non-uniform perturbation sets.

7.2.1 Mahalanobis Distance (MD)

Euclidean distance between two points in a multi-dimensional space is a useful metric when the vectors have isotropic distribution (i.e. radially symmetric). This is because the Euclidean distance assumes each dimension has same scale (or spread) and are uncorrelated to other dimensions. However, isotropy is usually not the case for real datasets in which different features might have different scales and can be correlated. Fortunately, MD accounts for how the features are scaled and correlated to one another [151]. Hence, it is a more useful metric if the data has non-isotropic distribution.

By formal definition, MD between vectors $z, z' \in \mathbb{R}^d$ is denoted by $d_M(z, z'|M) := \sqrt{(z-z')^T M(z-z')}$, where $M \in \mathbb{R}^{d\times d}$ is a positive semi-definite matrix which can be decomposed as $M = U^T U$, for $U \in \mathbb{R}^{d\times d}$. The dissimilarity between two vectors from a distribution with covariance Σ can be measured by selecting $M = \Sigma^{-1}$. If feature vectors of a dataset are uncorrelated and have unit variances, their covariance matrix is $\Sigma = I$, which reduces their MD to Euclidean distance.

We are interested in the distance between the original and the perturbed sample. Since we assume all perturbations are additive, as common practice, the distance term we consider is $\sqrt{\delta^T M\delta}$. For a generalized MD in ℓ_p norm, selecting $\Omega = U^T$ corresponds to the perturbation set $\tilde{\Delta}_{\epsilon,p} = \{\delta : \|U^T\delta\|_p \leq \epsilon\}$ which generates AEs with feature correlations similar to the original dataset.

Robustness of an adversarially trained model is directly related to how realistic the generated AEs are during training. Now, we explore implications of selecting ℓ_2 MD to define the limits of the perturbation set. To ensure the validity of the AEs, we consider the notion of consistency of the generated sample with real samples. [146] introduced the notion of ϵ-inconsistency to quantify how likely an AE is. With slight change in their notation, we define γ-consistency as follows:

Definition 7.1. For a consistency threshold $\gamma > 0$, an AE is γ-consistent if $f(x \mid y) \geq \gamma$, where $x \in \mathbb{R}^d$, and f is a probability density function of a conditional Gaussian distribution with zero mean and covariance matrix Σ_y.

Theorem 7.2. *If the AEs are generated according to MD constraint, then their γ-consistency has a direct relation to ϵ such that*

$$0 < \sqrt{2C - 2\log\gamma} \leq \epsilon. \tag{7.6}$$

where $C = -\log(2\pi)^{d/2}|\Sigma_y|^{1/2}$, d is the dimension of x, and $\sqrt{\delta^T \Sigma_y^{-1} \delta} \leq \epsilon$.

See Appendix A.1 for the proof. Theorem 7.2 implies that there is a direct relationship between limiting the MD of δ and ensuring consistent samples when the data is Gaussian. In other words, when the ℓ_2 MD of the perturbations gets smaller, AEs become more consistent.

7.2.2 Weighted Norm

When Ω is a diagonal matrix, inner maximization constraint simply becomes the weighted norm of δ limited by ϵ, and the weights are denoted by $\{\Omega_{i,i}\}_{i=1}^d$. Projection of δ under the new constraint corresponds to projection onto an ℓ_p norm ball of radius $\frac{\epsilon}{\Omega_{i,i}}$ for i^{th} feature. These weights can be chosen exploiting domain, attack or model knowledge. For instance, more important features can be allowed to be perturbed more than the other features which have less effect on the output score of the classifier. This knowledge might come from Pearson's correlation coefficients [145] between the features of the training data and the corresponding labels, or Shapley values [152] for each feature.

Using Pearson's correlation coefficient of each feature with the corresponding target variable, i.e., $|\rho_{i,y}|$ for i^{th} feature and output y, we let larger perturbation radii for more correlated features with the output. Due to the inverse relation between $\Omega_{i,i}$ and the radius of the norm ball, for $\bar{\rho}_{i,y} = \frac{1}{|\rho_{i,y}|}$ we select $\Omega = \frac{diag(\{\bar{\rho}_{i,y}\}_{i=1}^d)}{\|\{\bar{\rho}_{i,y}\}_{i=1}^d)\|_2}$. Similarly, using Shapley values to represent feature importance, we define $\bar{s}_i = \frac{1}{|s_i|}$, where s_i is the Shapley value of feature i. Then, we choose $\Omega = \frac{diag(\{\bar{s}_i\}_{i=1}^d)}{\|\{\bar{s}_i\}_{i=1}^d)\|_2}$ by following the intuition that more important features should have larger perturbation radii.

In the malware domain, expert knowledge might help to rule out specific type of attacks crafted on immutable features due to feasibility constraints. This can be modelled by the proposed weighted norm constraint as masking the perturbations on immutable features. Hence, non-uniform perturbation approach enables various transformations on the attack space for robustness against realistic attacks.

7.3 Experimental Results

Here, we present experimental results to evaluate robustness of DNNs against adversarial attacks for binary classification problems on three applications: malware detection, credit risk prediction, and spam detection. We compare PGD with non-uniform perturbations during AT with PGD proposed in [24] based on uniform perturbations. For all applications, we evaluate our defense mechanisms on adversarial attacks proposed by other works. We use a machine with an Intel Xeon E5-2686 v4 @ 2.3 GHz CPU, and 4 Nvidia Tesla V100 GPUs.

In all applications, we use a fully-connected neural network model composed of 4 densely connected layers with the first three using *ReLU* activations followed by a *softmax* activation in the last layer. After each of the first three layers, we apply 20% *Dropout* rate for regularization during training. We use 5 random initialization for malware and 10 for both credit risk and spam detection use-cases to report average results.

For pre-processing, we use standardization as a normalization method, which is a common practice with many machine learning techniques.Min-max scaling transforms all features into the same scale while standardization, which is recommended in presence of outliers [153], only ensures zero mean and unit standard deviation. This approach does not guarantee same range (min and max) for all features. As a result, it is possible that the features have different scales even after normalization.

Note that our goal is not to design the best possible neural network but instead compare the uniform perurbations [24] with various non-uniform perturbations during AT for a given DNN.

Adversarial training (AT): We perform AT in all use-cases by applying ℓ_2-norm PGD for uniform perturbation sets, i.e., $\Delta_{\epsilon_1,2} = \{\delta : \|\delta\|_2 \leq \epsilon_1\}$, and non-uniform perturbation sets, i.e., $\tilde{\Delta}_{\epsilon_2,2} = \{\delta : \|\Omega\delta\|_2 \leq \epsilon_2\}$. Since potential adversaries are not interested in fooling the classifiers with negative class (target class) samples, δ perturbations are only applied to the positive classes during AT as commonly used especially in malware detection [148]. Positive classes are the malicious class in malware detection, bad class in credit risk prediction, and spammer class in spam detection. Moreover, for the sake of clean accuracy within the positive class, adversarial perturbations are applied to 90% of the positive samples during training. Such hybrid approach where a weighted clean adversarial loss are optimized at once is common in literature [154].

To model the expert knowledge with diagonal Ω, we use Pearson's correlation coefficient, Shapley values, and masking to allow perturbation only in mutable features. To compute Shapley values, we use SHAP [155] which utilizes a deep learning explainer. We also consider AT under the MD constraint, and select $\Omega{=}U^T$ such that $U^TU{=}\Sigma_y^{-1}$ considering two cases; Σ_y is the covariance matrix of the entire training data, i.e., $y{=}\{0,1\}$, and Σ_y is only for the negative (target) class $y{=}0$. We call the models after AT with non-uniform perturbations according to their Ω selection, e.g., *NU-δ-Pearson* for Pearson's coefficients, *NU-δ-SHAP* for Shapley values, *NU-δ-Mask* for masking, *NU-δ-MD* for MD using full covariance matrix and *NU-δ-MDtarget* for MD using the covariance for only $y{=}0$. The choice $\Omega{=}I$ corresponds to AT with uniform perturbation constraint, which we call *Uniform-δ*.

7.3.1 Malware Use-case

First, we consider a binary classification problem for malware detection using the EMBER dataset [156]. EMBER is a feature-based public dataset which is considered a benchmark for Windows malware detection. It contains 2381 features extracted from Windows PE files: $600K$ labeled training samples and $200K$ test samples. The EMBER dataset consists of two types of features:

1. **Parsed features** are extracted after parsing the portable executable (PE) file. Parsed features include 5 different groups:
   - *General file information*: virtual size of the file; number of imported/exported functions and symbols; whether the file has a debug section, thread local storage, resources, relocations, or a signature.
   - *Header information*: timestamp in the header; target machine; list of image and DLL characteristics; target subsystem; file magic; image, system and subsystem versions; code, headers and commit sizes (hashing trick).
   - *Imported functions*: functions extracted from the import address table (hashing trick)
   - *Exported functions*: list of exported functions (hashing trick).
   - *Section information*: name, size, entropy, virtual size, and a list of strings representing section characteristics (hashing trick).
2. **Format-agnostic features** do not require parsing the PE file structure and include:

- *Byte histogram*: counts of each byte value within the file (256 integer values).
- *Byte-entropy histogram*: quantized and normalized of the joint distribution $p(H, X)$ of entropy H and byte value X (256 bins).
- *String information*: number of strings and their average length; a histogram of the printable characters within those strings; entropy of characters across all printable strings.

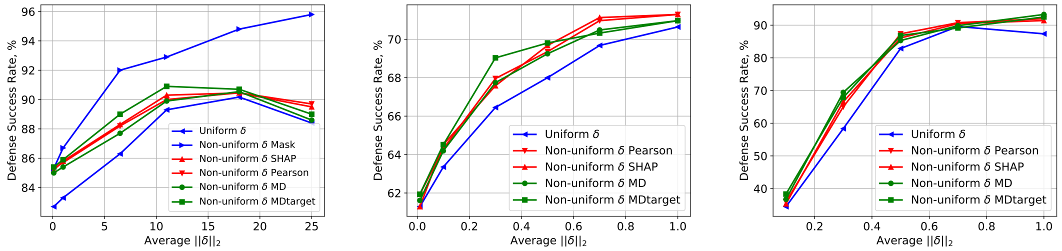
Given a malware sample, an adversary's goal is to make the DNN conclude that a malicious sample is benign. We also consider PDF malware detection. We use the extracted features of the PDF malware classification dataset and its attacked samples provided in [157]. The repository contains 110841 samples with 135 features that are extracted by PDFrate-R [158].

Attacks used for evaluation: In the malware domain, test-time evasion attacks can be classified as *feature-space* and *problem-space* attacks. While the former crafts AEs by modifying the features extracted from binary files, the latter directly modifies malware binaries making sure of the validity and inconspicuousness of the modified object. We evaluate the robustness of our model against evasion attacks which are crafted in problem-space, i.e., on PE files. For Windows malware, we incorporate the most successful attacks [159] from the machine learning static evasion competition [160]. Since the EMBER dataset only contains the extracted features of a file, a subset of malware binaries used for AE generation are obtained from VirusTotal [161] using the SHA-256 hash as identifier.

Below is a detailed explanation about the winner attacks [159] of malware competition [160].

Greedy Attack: Bytes in a range 256 are added iteratively to the malware binaries to make sure the prediction score for a known model lowers and none of the packing, functionality, or anti-tampering checks are affected. Byte addition is stopped when the prediction score gets lower than a threshold value or the file size exceeds 5MB. We generate 1000 adversarial examples from the malicious binaries of EMBER test set for each target model, such as standard trained neural network, adversarially trained model with ℓ_2-PGD for $\epsilon = 5$ and LGBM model which were provided as benchmark together with EMBER dataset [156]; and we call these adversarial example sets GNN, GAdv and GLGBM, respectively.

Constant Padding Attack: A new section is created in the binary file and filled with a constant value of size 10000. This attack is applied to 2000 binaries from EMBER malicious test set for constants "169" and "0", and we call these adversarial example sets C1 Pad. and C2 Pad., respectively.



(A) Malware Use-case

(B) Credit Risk Use-case

(C) Spam Detection Use-case

FIGURE 7.2: Defense success rate of ℓ_2-PGD AT against the problem-space attacks, where all non-uniform perturbation defense approaches outperform the uniform approach for all use-cases.

String Padding Attack: Strings of size 10000 from a benign file, such as Microsoft's End User License Agreement (EULA), are added to a new section created in the malware binary. We generate 2000 adversarial examples, which we call set Str. Pad., by string padding EMBER malicious test set.

We observe that these problem-space attacks, which add various bytes to a file without modifying the core functionality, affect only the feature groups "Byte Histogram", "Byte Entropy Histogram" and "Section Information". Experts aware of these byte padding attacks understand which features can be manipulated by an attacker. In addition to the previous AT methods, we represent this *best case expert knowledge* by $\Omega = I_{mask}$, which is an identity matrix with non-zero diagonal elements only for "Byte Histogram", "Byte Entropy Histogram" and "Section Information" features. That is, the model is trained using PGD perturbations applied only to these features, and we call it *NU-δ-Mask*. Our masking approach for the immutable features is similar to the *conserved features* in [162].

For PDF malware classification, we use a problem space attack called EvadeML [163]. It allows adding, removing and swapping objects, hence it is a stronger attack than most other problem space attacks in the literature, which typically only allow addition to preserve the malicious functionality.

Numeric results: To make a fair comparison between uniform and non-uniform approaches, ϵ for each method is selected such that their average distortion budgets, i.e., $||\delta||_2$, are approximately equal. For Windows malware classification, we test the detection success of adversarially trained models with 9000 AE sets generated by the problem-space attacks described previously. Figure 7.2A illustrates the average defense success rate against various problem-space attacks and shows that non-uniform perturbation approaches outperform the uniform perturbation in all cases. Moreover, *NU-δ-MDtarget* performs closest to the best case expert knowledge *NU-δ-Mask* for all cases except when

Table 7.1: **Malware Use-case:** Average number of successful evasions on standard training, uniform and non-uniform ℓ_2-PGD adversarial trainings by the adversarial example sets out of 1000 samples for approximately equal $\|\delta\|_2$. Defense success rates shown in Table 7.2 and Figure 7.2A are calculated by averaging the success rate over these individual attacks results.

Model	$\|\|\delta\|\|_2$	**GNN**	**GLGBM**	**GAdv**	**C1 Pad.**	**C2 Pad.**	**Str. Pad.**
Std. Training	-	832	217	337	168	35	123
Uniform-δ	0.1	472.6	105	249.6	66.3	37.6	114.9
NU-δ-Mask	0.1	408.5	89.2	241.7	46.2	35.2	74.5
NU-δ-SHAP	0.1	392.8	86.8	206.8	64.9	39	104.7
NU-δ-Pearson	0.1	417.6	92.8	221.4	45.1	38.2	74.5
NU-δ-MD	0.1	413	101	216	56	38.7	81.8
NU-δ-MDtarget	0.1	391.6	84.2	234.6	52.2	38.7	79
Uniform-δ	1	447.2	111.8	273.8	50.1	38.5	83.4
NU-δ-Mask	1	299.4	88.2	223.4	58.3	40	91
NU-δ-SHAP	1	359.7	82.2	244.5	53.8	33.6	81.7
NU-δ-Pearson	1	304	96.2	265	60.5	38.8	99.2
NU-δ-MD	1	373.2	89.5	244	54.7	37	81.8
NU-δ-MDtarget	1	360.8	103.4	246.6	45.1	36.7	72.4
Uniform-δ	6.7	231.5	129	333	37.7	38	58.7
NU-δ-Mask	6.7	104.4	68.4	153.4	43.4	47.3	70.8
NU-δ-SHAP	6.7	170	113	302.5	32.7	41.2	39.7
NU-δ-Pearson	6.7	213	78	304	38	38	48.6
NU-δ-MD	6.7	234	91	314	27.7	31	34.2
NU-δ-MDtarget	6.7	196	61	301	37.5	33.5	36.5
Uniform-δ	11	177	77.6	278	37.8	41.3	44.6
NU-δ-Mask	11	94.4	45.2	160.8	30.8	43.7	50.5
NU-δ-SHAP	11	178	62	296	32	40	35
NU-δ-Pearson	11	142	75.7	273	31.6	40.7	42.8
NU-δ-MD	11	195	46	247	40	32.5	43
NU-δ-MDtarget	11	122.7	44	251.7	34	41	48
Uniform-δ	18	152.2	57.3	234	42.7	51.6	52.3
NU-δ-Mask	18	44.5	20.2	116.5	27.1	48.2	47.2
NU-δ-SHAP	18	159.4	48.6	207.2	53.1	59.6	61.3
NU-δ-Pearson	18	154.2	49	220.4	42.6	61.7	47.2
NU-δ-MD	18	144.2	49.2	204.6	53	54	63.8
NU-δ-MDtarget	18	132.4	52.4	215	50.6	53.4	53.2
Uniform-δ	25	233.2	58	228	59.3	51	68.4
NU-δ-Mask	25	25	14.8	108	21.8	48.9	34.8
NU-δ-SHAP	25	193.8	53.4	226.2	44.7	56.9	58.7
NU-δ-Pearson	25	158.2	59.5	191.2	67.6	65.6	75.8
NU-δ-MD	25	199.7	54	248.5	58.6	59.5	59.7
NU-δ-MDtarget	25	210	55	225.6	57.7	56.4	60.5

$||\delta||_2 = 25$. The advantage of selecting Σ from benign samples versus selecting from the entire dataset is that the direction of perturbations are led towards the target class, i.e. benign samples, for *NU-δ-MDtarget*. We also do not observe a significant performance difference between *NU-δ-Pearson* and *NU-δ-SHAP*, while *NU-δ-MD* only differs from the two for $||\delta||_2 = 25$. We refer to Table 7.1 for detailed attack performances and to Table 7.2 for defense S.R. results with clean accuracy.

TABLE 7.2: **Malware Use-case:** Clean accuracy (Ac.) and defense success rate (S.R.) of standard training, uniform and non-uniform ℓ_2-PGD adversarial trainings with EMBER dataset for approximately equal $\|\delta\|_2$. Non-uniform perturbation defense approaches outperform the uniform perturbation for all cases against adversarial attacks.

Model	$\|\delta\|_2$	**Clean Ac., %**	**Defense S.R., %**
Std. Training	-	96.6	73
Uniform-δ	0.1	96.2	82.7 ± 0.88
NU-δ-Mask	0.1	96.2	85.3 ± 0.25
NU-δ-SHAP	0.1	96.2	85.2 ± 0.39
NU-δ-Pearson	0.1	96.1	85.3 ± 0.94
NU-δ-MD	0.1	96.3	85 ± 0.99
NU-δ-MDtarget	0.1	96.2	$\mathbf{85.4 \pm 0.80}$
Uniform-δ	1	96.1	83.3 ± 0.41
NU-δ-Mask	1	96.1	$\mathbf{86.7 \pm 0.68}$
NU-δ-SHAP	1	96.3	85.5 ± 0.61
NU-δ-Pearson	1	96.2	85.7 ± 0.45
NU-δ-MD	1	96.3	85.4 ± 0.67
NU-δ-MDtarget	1	96.3	85.9 ± 0.19
Uniform-δ	6.7	95.8	86.3 ± 0.15
NU-δ-Mask	6.7	95.7	$\mathbf{92 \pm 0.07}$
NU-δ-SHAP	6.7	95.8	88.3 ± 0.33
NU-δ-Pearson	6.7	95.9	88.2 ± 0.30
NU-δ-MD	6.7	96	87.7 ± 0.32
NU-δ-MDtarget	6.7	95.8	89 ± 0.18
Uniform-δ	11	95.6	89.3 ± 0.54
NU-δ-Mask	11	95.8	$\mathbf{92.9 \pm 0.57}$
NU-δ-SHAP	11	96	90.3 ± 0.36
NU-δ-Pearson	11	95.8	90 ± 0.36
NU-δ-MD	11	95.9	89.9 ± 0.25
NU-δ-MDtarget	11	95.7	90.9 ± 0.29
Uniform-δ	18	95.5	90.17 ± 0.71
NU-δ-Mask	18	95.8	$\mathbf{94.8 \pm 0.51}$
NU-δ-SHAP	18	95.3	90.45 ± 0.30
NU-δ-Pearson	18	95.3	90.46 ± 0.25
NU-δ-MD	18	95.4	90.54 ± 0.46
NU-δ-MDtarget	18	95.4	90.7 ± 0.51
Uniform-δ	25	95.6	88.4 ± 0.39
NU-δ-Mask	25	95.7	$\mathbf{95.8 \pm 0.21}$
NU-δ-SHAP	25	95.5	89.5 ± 0.27
NU-δ-Pearson	25	94.9	89.7 ± 0.40
NU-δ-MD	25	95.2	88.6 ± 0.26
NU-δ-MDtarget	25	95.2	89 ± 0.57

Table 7.1 shows the average number of adversarial examples out of 1000 which successfully evade the corresponding models. While *NU-δ-Mask* and *NU-δ-MDtarget* have better performance against Greedy attacks for most of the time, i.e., sets GNN, GLGBM and GAdv, *NU-δ-Pearson*, *NU-δ-SHAP* and *NU-δ-MD* have better accuracy against padding attacks, i.e, sets C1 Pad., C2 Pad. and Str. Pad.

For PDF malware classification, we compare *NU-δ-MDt* with *Uniform-δ* against

TABLE 7.3: Clean accuracy (Ac.), AUC score and defense success rate (D.S.R.) against EvadeML for standard training, *Uniform-δ* and *NU-δ-MDt*.

Model	$\|\|\delta\|\|_2$	**Clean Ac.**	**AUC score**	**D.S.R.**
Std. training	-	99.52%	0.99912	11.1%
Uniform-δ	1	97.64%	0.99826	87.4%
NU-δ-MDt	1	97.83%	0.99835	**92.9**%

EvadeML. We observe the best performances at $||\delta||_2 = 1$ for both methods. Table 7.3 depicts the clean accuracy (Ac.), AUC score and defense success rate (D.S.R.) against EvadeML for standard training, *Uniform-δ* and *NU-δ-MDt*. Although *NU-δ-MDt* is a feature space defense, the results show that it is highly effective against problem space attacks, and it outperforms *Uniform-δ*. Since our approach does not assume any attack knowledge, it is more generalizable than the problem space defenses.

7.3.2 Credit Risk Use-case

Our second use-case is a credit risk detection problem where the DNN's goal is to make decisions on loan applications for bank customers. For this scenario, we use the well-known German Credit dataset [164], which contains classes "good" and "bad", as well as applicant features such as age, employment status, income, savings, etc. It has 20 features and 1000 samples with 300 in the "bad" class. Similar to [145], we treat discrete features as continuous and drop non-ordinal categorical features.

Attacks used for evaluation: The goal of an adversary in this situation is to make DNN models conclude that they are approved for a loan when they actually may not be eligible. Since modifications to tabular data can be detected by an expert eye, attackers try to fool classifiers with imperceptible attacks. We use German Credit dataset implementation of LowProFool [145] which considers attack imperceptibility and represents expert knowledge using feature correlations. We apply the attack on the "bad" class of the test set and generate 155 AEs. After dropping the non-ordinal categorical features, we treat the remaining 12 features as continuous values.

Numeric results: Similar to the malware use-case, ϵ for each method is selected such that their average $\|\delta\|_2$ are approximately equal. In Figure 7.2B, we report defense success rate of PGD with uniform and non-uniform perturbations in detecting 155 AEs generated by LowProFool. The figure shows that for every given $\|\delta\|_2$, non-uniform perturbations outperform uniform perturbations in PGD. Although LowProFool represents feature importance by Pearson correlation coefficients between features and the output score, surprisingly *NU-δ-Pearson* is the best approach among the other non-uniform approaches for only $\delta = \{0.7, 1\}$. We refer to Table 7.4 for clean accuracy results.

Table 7.4: **Credit Risk Use-case:** Clean accuracy (Ac.) and defense success rate (S.R.) of standard training, uniform and non-uniform ℓ_2-PGD adversarial trainings with German Credit dataset for approximately equal $\|\delta\|_2$. Non-uniform perturbation defense approaches outperform the uniform perturbation for all cases against adversarial attacks.

Model	$\|\delta\|_2$	**Clean Ac., %**	**Defense S.R., %**
Std. Training	-	69.7	60
Uniform-δ	0.01	69	61.3 ± 0.40
NU-δ-SHAP	0.01	68.3	61.3 ± 0.35
NU-δ-Pearson	0.01	68.3	$\mathbf{61.9 \pm 0.37}$
NU-δ-MD	0.01	69.6	61.6 ± 0.32
NU-δ-MDtarget	0.01	69.7	$\mathbf{61.9 \pm 0.30}$
Uniform-δ	0.1	67.7	63.4 ± 0.31
NU-δ-SHAP	0.1	67.1	$\mathbf{64.5 \pm 0.20}$
NU-δ-Pearson	0.1	66.8	64.3 ± 0.56
NU-δ-MD	0.1	66.7	64.2 ± 0.32
NU-δ-MDtarget	0.1	66.7	$\mathbf{64.5 \pm 0.41}$
Uniform-δ	0.3	66.7	66.4 ± 0.22
NU-δ-SHAP	0.3	65.8	67.6 ± 0.30
NU-δ-Pearson	0.3	66	68 ± 0.21
NU-δ-MD	0.3	66.5	67.1 ± 0.64
NU-δ-MDtarget	0.3	66.3	$\mathbf{69 \pm 0.32}$
Uniform-δ	0.5	66.2	68 ± 0.32
NU-δ-SHAP	0.5	66.5	69.7 ± 0.37
NU-δ-Pearson	0.5	65.9	69.4 ± 0.27
NU-δ-MD	0.5	66.3	69.2 ± 0.35
NU-δ-MDtarget	0.5	66	$\mathbf{69.8 \pm 0.13}$
Uniform-δ	0.7	66.1	69.6 ± 0.20
NU-δ-SHAP	0.7	65.8	$\mathbf{71.1 \pm 0.57}$
NU-δ-Pearson	0.7	65.6	71 ± 0.37
NU-δ-MD	0.7	66.4	70.5 ± 0.30
NU-δ-MDtarget	0.7	65.6	70.3 ± 0.30
Uniform-δ	1	65.3	70.6 ± 0.44
NU-δ-SHAP	1	64.5	$\mathbf{71.3 \pm 0.32}$
NU-δ-Pearson	1	64.3	$\mathbf{71.3 \pm 0.32}$
NU-δ-MD	1	64.9	71 ± 0.37
NU-δ-MDtarget	1	65	71 ± 0.21

7.3.3 Spam Detection Use-case

Finally, we evaluate robustness within the context of detecting spam within social networks. We use a dataset from Twitter, where data from legitimate users and spammers is harvested from social honeypots over seven months [165]. This dataset contains profile information and posts of both spammers and legitimate users. After pre-processing [166], we extract 31 numeric features with 14 being integers and the rest being continuous. Some examples of these features are the number of following and followers as well as the ratio between them, percentage of bidirectional friends, number of posted messages per day, etc. We treat all features as continuous values in our experiments. Moreover, we extract 41,354 samples where the training set has 17,744 “bad" and 15,339 “good"

Table 7.5: **Spam Detection Use-case:** Clean accuracy and defense success rate of standard training, uniform and non-uniform ℓ_2-PGD adversarial trainings with Twitter Spam dataset for approximately equal $\|\delta\|_2$.

Model	$\|\delta\|_2$	**Clean Ac., %**	**Defense S.R., %**
Std. Training	-	94.6	17.5
Uniform-δ	0.1	91.1	34.4 ± 0.16
NU-δ-SHAP	0.1	93.9	35.3 ± 0.32
NU-δ-Pearson	0.1	94	$36 \pm 0.50.$
NU-δ-MD	0.1	93.9	36.7 ± 0.48
NU-δ-MDtarget	0.1	93.9	$\mathbf{38.3 \pm 0.50}$
Uniform-δ	0.3	92.6	58.3 ± 0.66
NU-δ-SHAP	0.3	91.9	66.5 ± 0.86
NU-δ-Pearson	0.3	91.8	65 ± 0.21
NU-δ-MD	0.3	91.9	$\mathbf{69.4 \pm 0.25}$
NU-δ-MDtarget	0.3	92	67.9 ± 0.25
Uniform-δ	0.5	91.3	82.8 ± 0.46
NU-δ-SHAP	0.5	90.9	86.1 ± 0.14
NU-δ-Pearson	0.5	91.2	$\mathbf{87.3 \pm 0.20}$
NU-δ-MD	0.5	91.1	85.3 ± 0.30
NU-δ-MDtarget	0.5	91.2	86.8 ± 0.28
Uniform-δ	0.7	91.1	89.6 ± 0.48
NU-δ-SHAP	0.7	90.5	90.5 ± 0.19
NU-δ-Pearson	0.7	90.6	$\mathbf{90.7 \pm 0.11}$
NU-δ-MD	0.7	90.5	89.8 ± 0.35
NU-δ-MDtarget	0.7	90.5	89.1 ± 0.18
Uniform-δ	1	90.5	87.3 ± 0.62
NU-δ-SHAP	1	89.8	91.4 ± 0.62
NU-δ-Pearson	1	89.9	92 ± 0.53
NU-δ-MD	1	89.7	$\mathbf{93.3 \pm 0.30}$
NU-δ-MDtarget	1	89.8	92.5 ± 0.64

samples, and the testing set has 3885 “bad" and 4386 “good" samples. The adversary’s goal is to make the DNN predict that a tweet was posted by a legitimate user when it was written by a spammer.

Attacks used for evaluation: We incorporate the evasion attack [167] from [144] for our Twitter spam detector. The attack strategy is based on minimizing the maliciousness score of an AE which is measured by a local interpretation model LASSO, while satisfying ℓ_2 norm constraint on perturbations. We generate the AEs by constraining the perturbations to $0.5 \times dist^{avg}_{pos-neg}$, where $dist^{avg}_{pos-neg}$ is defined by [144] as the average distance between the spammer samples and the closest non-spammers to these samples. We split the Twitter dataset with ratio 25% for training and testing, and generate the AEs using the spammer class of the entire test set.

Numeric results: Again, we apply perturbations only to the spammer set during AT and report the results for approximately equal average $\|\delta\|_2$ perturbations. Figure 7.2C illustrates defense success rate in detecting AEs of the proposed approaches against the model interpretation based attack [144] for Twitter dataset. The figure shows that

Table 7.6: Clean accuracy (Cl. Ac.) and defense success rates of *NU-δ-MDt* and *Uniform-δ* against FGSM, Carlini-Wagner (CW), JSMA and Deep Fool attacks for Spam Detection Use-case.

Defenses	δ	**Cl. Ac.**,%	**FGSM** ϵ=0.5,%	**FGSM** ϵ=1,%	**CW**,%	**JSMA**,%	**Deep Fool**,%
Uniform-δ	0.1	91.61	24.1 ± 0.25	20.17 ± 0.31	38.93 ± 0.54	41.51 ± 0.24	39.3 ± 0.27
NU-δ-MDt		91.93	$\mathbf{28.7 \pm 0.17}$	$\mathbf{27.1 \pm 0.21}$	$\mathbf{49.62 \pm 0.33}$	$\mathbf{49.59 \pm 0.18}$	$\mathbf{51.73 \pm 0.31}$
Uniform-δ	0.3	90.40	25.22 ± 0.12	21.61 ± 0.43	45.83 ± 0.41	46.38 ± 0.27	47.39 ± 0.25
NU-δ-MDt		91.31	$\mathbf{32.15 \pm 0.34}$	$\mathbf{30.88 \pm 0.36}$	$\mathbf{53.45 \pm 0.22}$	$\mathbf{52.68 \pm 0.13}$	$\mathbf{54.61 \pm 0.19}$
Uniform-δ	0.5	87.12	27.05 ± 0.50	23.08 ± 0.34	50.05 ± 0.32	49.5 ± 0.25	49.75 ± 0.16
NU-δ-MDt		87.78	$\mathbf{43.85 \pm 0.62}$	$\mathbf{36.12 \pm 0.20}$	$\mathbf{55.24 \pm 0.38}$	$\mathbf{53.71 \pm 0.41}$	$\mathbf{64.28 \pm 0.55}$
Uniform-δ	1	86.46	40.20 ± 0.57	32.98 ± 0.31	52.77 ± 0.24	52.94 ± 0.26	53.22 ± 0.10
NU-δ-MDt		87.64	$\mathbf{81.34 \pm 0.83}$	$\mathbf{79.85 \pm 0.75}$	$\mathbf{61.89 \pm 0.37}$	$\mathbf{72.93 \pm 0.77}$	$\mathbf{88.75 \pm 0.78}$
Uniform-δ	1.5	85.98	74.40 ± 0.47	62.36 ± 0.75	59.71 ± 0.28	68.95 ± 0.85	64.5 ± 0.86
NU-δ-MDt		87.03	$\mathbf{94.45 \pm 0.18}$	$\mathbf{91.03 \pm 0.10}$	$\mathbf{72.98 \pm 0.48}$	$\mathbf{86.43 \pm 0.32}$	$\mathbf{98.36 \pm 0.25}$

non-uniform perturbations outperform uniform case in terms of defense S.R. for all given $||\delta||_2$. We refer to Table 7.5 for clean accuracy results.

7.3.4 Performance Against Uniform Attacks

Throughout the experiments, we tested our non-uniform approach against various realistic attacks, such as problem space attacks in Section 7.3.1, feature importance-based attack in Section 7.3.2 and explainability-based attack in Section 7.3.3. So far we emphasized that problem space attacks and non-uniformly norm bounded attacks are more realistic compared to the traditional uniformly norm-bounded attacks which are mostly considered in image domain. Yet, in this section, we also test our non-uniform approach against the well-known uniform attacks to investigate the generalizability of our approach. We use the setting for the spam detection use-case, and compare *NU-δ-MDt* with *Uniform-δ*. We utilize the adversarial robustness toolbox (ART) [168] to craft AEs by using the default parameters for the AE generators of Carlini-Wagner (CW), JSMA and DeepFool Methods. We also use FGSM for $\epsilon = 0.5$ and $\epsilon = 1$. Table 7.6 shows the clean accuracy (Cl. Ac.) and defense S.R.'s of the robust models *NU-δ-MDt* and *Uniform-δ*. We observe in Table 7.6 that the Cl. Ac. decreases as $||\delta||_2$ increases for both *Uniform-δ* and *NU-δ-MDt* but the degradation in non-uniform is less. Defense S.R.'s, on the other hand, improve for both approaches but *NU-δ-MDt* significantly outperforms *Uniform-δ* in all cases.

We further investigate the performance of our non-uniform approach against uniformly norm-bounded attacks for generalizability as in Section 7.3.4. We use the same setting as in spam detection use-case, and craft AEs using standard PGD attack, i.e., the attack in *Uniform-δ*, for $\epsilon = \{0.1, 0.3, 0.5, 0.7\}$. For a fair comparison between the uniform and non-uniform approaches, we set approximately equal $||\delta||_2$ for both models in the average

Table 7.7: Defense success rates of *Uniform-δ*, *NU-δ-MDt* and *MDt-Combo* against PGD attacks for Spam Detection Use-case. Both non-uniform defenses outperform the uniform approach while *NU-δ-MDt* also outperforms *MDt-Combo* for all cases.

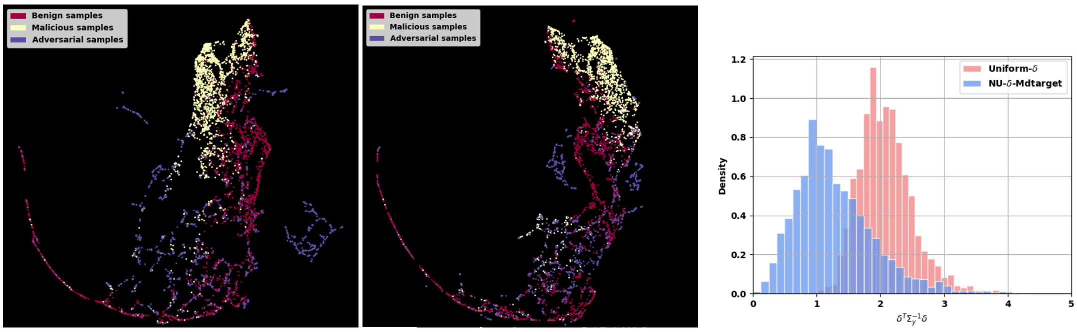
Defenses	$\|\|\delta\|\|_2$	**Attack** ϵ=0.1	**Attack** ϵ=0.3	**Attack** ϵ=0.5	**Attack** ϵ=0.7
Uniform-δ	0.1	90.6 ± 0.18	83.9 ± 0.34	16.6 ± 0.47	14.8 ± 0.65
NU-δ-MDt		**92.8 ± 0.21**	**88.2 ± 0.41**	**34.95 ± 0.5**	**21.4 ± 0.22**
MDt-Combo		91.6 ± 0.24	86.3 ± 0.28	24.4 ± 0.38	19.2 ± 0.32
Uniform-δ	0.3	92.6 ± 0.14	89.05 ± 0.24	30.5 ± 0.42	20.85 ± 0.58
NU-δ-MDt		**93.2 ± 0.10**	**90.95 ± 0.22**	**61.75 ± 0.51**	**46.3 ± 0.41**
MDt-Combo		93 ± 0.11	89.24 ± 0.19	47.88 ± 0.72	31.5 ± 0.39
Uniform-δ	0.5	92.9 ± 0.08	91.4 ± 0.05	88 ± 0.22	86.5 ± 0.25
NU-δ-MDt		**93.3 ± 0.05**	**91.45 ± 0.06**	**89.45 ± 0.17**	**87.7 ± 0.12**
MDt-Combo		93.1 ± 0.10	91.4 ± 0.11	89.30 ± 0.14	87.2 ± 0.18
Uniform-δ	1	93.2 ± 0.14	91.9 ± 0.25	90.18 ± 0.20	88.38 ± 0.22
NU-δ-MDt		**94.5 ± 0.11**	**94.17 ± 0.42**	**93.33 ± 0.15**	**93.14 ± 0.31**
MDt-Combo		94.39 ± 0.10	92.42 ± 0.28	91.44 ± 0.19	91.33 ± 0.26
Uniform-δ	1.5	93.22 ± 0.16	92.01 ± 0.27	90.61 ± 0.24	89.78 ± 0.15
NU-δ-MDt		**94.45 ± 0.12**	**94.38 ± 0.31**	**93.76 ± 0.16**	**93.48 ± 0.11**
MDt-Combo		94.27 ± 0.11	93.1 ± 0.20	92.3 ± 0.27	91.8 ± 0.17

sense. In this section, we also consider a non-uniform robust model which enforces the AT constraint first on $||\Omega\delta||_2$ and then $||\delta||_2$. That is, the non-uniform attack is always a valid uniform attack in the strict sense. We call this defense *Combo* due to using the combination of both projections in (7.3) and (7.5).

Table 7.7 shows the defense success rates of *Uniform-δ*, *NU-δ-MDt* and *MDt-Combo*, which denotes the *Combo* approach for Ω selected as the Mahalanobis matrix for the benign samples, against PGD attacks for Spam Detection Use-case. We observe that our non-uniform approach outperforms the uniform approach for all cases, hence it is also effective against the uniformly norm-bounded attacks which makes it generalizable. Furthermore, Table 7.7 shows that *MDt-Combo* performs in between *Uniform-δ* and *NU-δ-MDt*. This is due to the fact that the strict constraint on $||\delta||_2$ reduces the effect of non-uniform projection.

7.3.5 Quality of Perturbation Sets

In this section, we quantitatively and qualitatively analyze how well non-uniform perturbations capture realistic attacks using γ-consistency property defined in Section 7.2 and lower dimensional space visualization. Our intuition is that a successful attack evades detection since AEs appear benign to the model. That is, AEs have high likelihood according to the distribution of benign samples. Therefore, we measure a perturbed sample's quality by its γ-consistency with the benign set distribution. Definition 7.1 leverages Theorem 7.2, which shows that smaller MD for δ indicates higher γ-consistency and hence higher quality of the perturbed sample. Moreover, we expect



(A) Uniform-δ

(B) NU-δ-MDtarget

(C) Histogram of MD square, $\delta^T\Sigma_{y=0}^{-1}\delta$

Figure 7.3: UMAP visualization of benign, malicious and adversarial samples generated by (A) Uniform-δ and (B) NU-δ-MDtarget, and (C) the density histogram of their $\delta^T\Sigma_{y=0}^{-1}\delta = 2C - 2\log\gamma$.

AEs that evade the model and benign samples to be embedded closer to each other in the lower-dimensional subspace. Figure 7.3 illustrates UMAP visualization [169] of benign, malicious and adversarial samples for the spam detection use-case. AEs generated by NU-δ-MDtarget show better alignment with benign distribution, which shows that NU-δ-MDtarget mimics a more realistic attack. We also show the histogram of MD squares, i.e. $\delta^T\Sigma_{y=0}^{-1}\delta = 2C - 2\log\gamma$, of 1660 AEs from Uniform-δ and NU-δ-MDtarget in Figure 7.3C, where the average values are 2.1 and 1.28, respectively. Following Theorem 7.2 and Figure 7.3C, δ's from NU-δ-MDtarget have higher γ, and hence, are more realistic.

7.4 Certified Robustness with Non-uniform Perturbations

In this section, we present methods for certifying robustness with non-uniform perturbations. We consider two well-known methods; linear programming (LP) [32] and randomized smoothing [170].

7.4.1 LP Formulation

We can provably certify the robustness of deep ReLU networks against non-uniform adversarial perturbations at the input. Our derivation follows an LP formulation of the adversary's problem with ReLU relaxations, then the dual problem of the LP and activation bound calculation. It can be viewed as an extension of [32]. Similar to [32], we consider a k layer feedforward deep ReLU network with

$$\hat{z}_{i+1} = W_i z_i + b_i,\;\; z_i = \max\{\hat{z}_i, 0\},\;\text{for } i = 1, \cdots, k-1 \tag{7.7}$$

We denote $\mathcal{Z}_{\epsilon,\Omega}(x) := \{f_\theta(x+\delta) : ||\Omega\delta||_p \leq \epsilon\}$ as the set of all attainable final-layer activations by input perturbation δ. Since this is a non-convex set for multi-layer networks which is hard to optimize over, we consider a convex outer bound on $\mathcal{Z}_{\epsilon,\Omega}(x)$ and optimize the worst case loss over this bound to guarantee that no AEs within $\mathcal{Z}_{\epsilon,\Omega}(x)$ can evade the network. As done in [32], we relax the ReLU activations by representing $z = \max\{0, \hat{z}\}$ with their upper convex envelopes $z \geq 0, z \geq \hat{z}, -u\hat{z} + (u-l)z \leq -ul$, where l and u are the known lower and upper bounds for the pre-ReLU activations. We denote the new relaxed set of all attainable final-layer activations by $\tilde{\mathcal{Z}}_{\epsilon,\Omega}(x)$. Assuming that an adversary targets a specific class to fool the classifier, we write the LP as

$$\underset{\hat{z}_k}{\text{minimize}} \quad c^T\hat{z}_k \qquad \text{s.t.} \quad \hat{z}_k \in \tilde{\mathcal{Z}}_{\epsilon,\Omega} \tag{7.8}$$

where $c := e_{y^{true}} - e_{y^{target}}$ is the difference between the selection vector of true and the target class.

A positive valued objective for all classes as a solution to equation (7.8) indicates that there is no adversarial perturbation within $\tilde{\Delta}_{\epsilon,p}$ which can evade the classifier. To be able to solve equation (7.8) in a tractable way, we consider its dual whose feasible solution provides a guaranteed lower bound for the LP. It is previously shown by [32] that a feasible set of the dual problem can be formulated similar to a standard backpropagation network and solved efficiently. The dual problem of our LP with ReLU relaxation and non-uniform perturbation constraints is expressed in the following theorem.

Theorem 7.3. *The dual of the linear program (7.8) can be written as*

$$\begin{aligned}
&\underset{\hat{\nu},\nu}{\textit{maximize}} && -\sum_{i=1}^{k-1}\nu_{i+1}^T b_i + \sum_{i=2}^{k-1}\sum_{j\in\mathcal{I}_i} l_{i,j}[\hat{\nu}_{i,j}]_+ - \hat{\nu}_1^T x - \epsilon||\Omega^{-1}\hat{\nu}_1||_q \\
&\textit{s.t.} && \nu_k = -c, \quad \hat{\nu}_i = (W_i^T\nu_{i+1}), \quad \textit{for} \;\; i = k-1,\ldots,1 \\
& && \nu_{i,j} = \begin{cases} 0 & j \in \mathcal{I}_i^- \\ \hat{\nu}_{i,j} & j \in \mathcal{I}_i^+ \\ \frac{u_{i,j}}{u_{i,j}-l_{i,j}}[\hat{\nu}_{i,j}]_+ - \eta_{i,j}[\hat{\nu}_{i,j}]_- & j \in \mathcal{I}_i \end{cases}, \quad \textit{for} \;\; i = k-1,\ldots,2
\end{aligned} \tag{7.9}$$

where $\mathcal{I}_i^-$, $\mathcal{I}_i^+$ and $\mathcal{I}_i$ represent the activation sets in layer i for l and u are both negative, both positive and span zero, respectively.

When $\eta_{i,j} = \frac{u_{i,j}}{u_{i,j}-l_{i,j}}$, Theorem 7.3 shows that the dual problem can be represented as a linear back propagation network, which provides a tractable solution for a lower bound of the primal objective. To solve equation (7.9), we need to calculate lower and upper bounds for each layer incrementally. The proof of the Theorem 7.3 is provided in

Algorithm 2 Activation Bound Calculation

```
**Input:** Network parameters $\{W_i, b_i\}$, input data $x$, input constraint matrix $\Omega$ and ball size $\epsilon$, norm type $q$.
Initialize $\hat{\nu}_1 := W_1^T$, $\zeta_1 := b_1^T$
$l_2 = x^T W_1^T + b_1^T - \epsilon||\Omega^{-1}W_1^T||_q$
$u_2 = x^T W_1^T + b_1^T + \epsilon||\Omega^{-1}W_1^T||_q$
$\nu_{2,\mathcal{I}_2} := (D_2)_{\mathcal{I}_2} W_2^T$
$\zeta_2 = b_2^T$
**for** $i = 2$ **to** $k-1$ **do**
    $l_{i+1} = x^T\hat{\nu}_1 + \sum_{j=1}^{i}\zeta_j - \epsilon||\Omega^{-1}\hat{\nu}_1||_q + \sum_{i=2,i'\in\mathcal{I}_i}^{i} l_{j,i'}[-\nu_{j,i'}]_+$
    $u_{i+1} = x^T\hat{\nu}_1 + \sum_{j=1}^{i}\zeta_j + \epsilon||\Omega^{-1}\hat{\nu}_1||_q - \sum_{i=2,i'\in\mathcal{I}_i}^{i} l_{j,i'}[\nu_{j,i'}]_+$
    $\nu_{j,\mathcal{I}_j} = \nu_{j,\mathcal{I}_j}(D_i)_{\mathcal{I}_i} W_i^T$
    $\zeta_j = \zeta_j D_i W_i^T$
    $\hat{\nu}_1 = \hat{\nu}_1 (D_i)_{\mathcal{I}_i} W_i^T$
**end for**
**Output:** $\{l_i, u_i\}_{i=2}^{k}$
```

Appendix A.2, and lower and upper bound calculations are explained in the following parts.

Activation Bounds: The dual objective function provides a bound on any linear function $c^T\hat{z}_k$. Therefore, we can compute the dual objective for $c = -I$ and $c = I$ to obtain lower and upper bounds. For $c = I$, value of ν_i for all activations simultaneously is given by

$$\hat{\nu}_i = W_i^T D_{i+1} W_{i+1}^T \dots D_n W_n^T \quad \text{and} \quad \nu_i = D_i \hat{\nu}_i, \quad \text{where} \quad (D_i)_{jj} = \begin{cases} 0 & j \in \mathcal{I}_i^- \\ 1 & j \in \mathcal{I}_i^+ \\ \frac{u_{i,j}}{u_{i,j} - l_{i,j}} & j \in \mathcal{I}_i \end{cases} \tag{7.10}$$

Similar to [32], bounds for ν_i and $\hat{\nu}_i$ can be computed for each layer by cumulatively generating bounds for $\hat{z}_2$, then $\hat{z}_3$ and so on. By initializing $\hat{\nu}_1 := W_1^T$, $\zeta_1 := b_1^T$, first bounds are $l_2 := x^T W_1^T + b_1^T - \epsilon||\Omega^{-1}W_1^T||_q$ and $u_2 := x^T W_1^T + b_1^T + \epsilon||\Omega^{-1}W_1^T||_q$, where the norms are taken over the columns. Calculation of the bounds for each layer is given below in Algorithm 2.

For certification of robustness within a non-uniform norm ball around a test sample, we need the objective of the LP to be positive for all classes. Since the solution of the dual problem is a lower bound on the primal LP, it provides a worst case certification guarantee against the AEs within the non-uniform norm ball. We provide certification results for the robustness of *Uniform-δ* and *NU-δ-MDt* (*NU-δ-MDtarget*) for spam detection use-case in Table 7.8. We consider both uniform and non-uniform input constraints in

Table 7.8: Average certification margin and number of successful certified samples out of 1000 spammers for *NU-δ-MDt* and *Uniform-δ* for Spam Detection Use-case.

Model	Defense S.R.	Cert. Method	Margin	Cert. Success
Uniform-δ	$54.87 \pm 1.1\%$	Uniform-Cert	1.07	$34.72 \pm 0.94\%$
		NU-Cert-SHAP	1.84	$72.64 \pm 0.6\%$
		NU-Cert-Pearson	2.04	$76.8 \pm 0.71\%$
		NU-Cert-MD	2.40	$80.2 \pm 0.56\%$
		NU-Cert-MDt	2.40	$80.2 \pm 0.55\%$
NU-δ-MDt	$63.4 \pm 0.74\%$	Uniform-Cert	1.11	$42.95 \pm 0.69\%$
		NU-Cert-SHAP	1.9	$74.65 \pm 0.85\%$
		NU-Cert-Pearson	2.06	$78.38 \pm 0.76\%$
		NU-Cert-MD	2.41	$81.3 \pm 0.68\%$
		NU-Cert-MDt	2.41	$81.3 \pm 0.67\%$

certification, namely *Uniform-Cert* for the standard LP approach for certification with uniform perturbation constraint [32], and *NU-Cert-(.)* for the non-uniform constraint. We implement our non-uniform approach into the LP by modifying [32] with our Ω matrix, and generate various certification methods by non-uniform Ω, e.g. *NU-Cert-SHAP*, *NU-Cert-Pearson*, *NU-Cert-MD* and *NU-Cert-MDt*. Our purpose is not to propose the tightest certification bounds but to show that non-uniform constraints result in larger certification margins compared to the uniform case.

We compare *Uniform-δ* and *NU-δ-MDt* to evaluate certification results. Dropout layers are removed from the model for LP solution, and AT is performed for $\epsilon = 0.3$. Certification is done by solving the LP for $\epsilon = 0.3$ over 1000 spammers. The objective should be positive for all classes to certify the corresponding sample. The margin between the objective and zero gives an idea about how tight the bound is [171]. Table 7.8 demonstrates two main results: (i) the certification success of *NU-δ-MDtarget* over *Uniform-δ* for each certification method supports our claim that non-uniform perturbations provide higher robustness than the uniform approach; and (ii) certification with non-uniform constraints provide larger certification margins and hence tighter bound.

7.4.2 Randomized Smoothing

Robustness certification via *randomized smoothing* [170] is an empirical alternative to the LP. The idea is constructing a "smoothed" classifier g from the base classifier f. In the original formulation in [170], g returns the most likely output returned by f given input x is perturbed by isotropic Gaussian noise. Here, we provide robustness guarantee in binary case for randomized smoothing framework when non-isotropic Gaussian noise is used to allow robustness to non-uniform perturbations:

$$g(x) = \text{argmax}_{y \in \mathcal{Y}} \mathbb{P}(f(x+n) = y) \quad \text{where} \quad n \sim \mathcal{N}(0, \Sigma). \tag{7.11}$$

Table 7.9: Percentage of successfully certified samples for *NU-δ-MDt* and *Uniform-δ* with various certification approaches with randomized smoothing for Spam Detection use-case.

Model	UC	NUC-Pearson	NUC-SHAP	NUC-MD	NUC-MDt
Uniform-δ	50.96%	61.11%	64.24%	65.72%	66.45%
NU-δ-MDt	**61.8%**	**67.14%**	**71.25%**	**85.34%**	**90.11%**

Adapting notation and Theorem 2 from [170], let p_a be the probability of the most probable class $y = a$ when the base classifier f classifies $\mathcal{N}(x, \Sigma)$. Then the below theorem holds.

Theorem 7.4. *In binary classification problem, suppose* $\underline{p_a} \in (\frac{1}{2}, 1]$ *satisfies* $\mathbb{P}(f(x+n) = a) \geq \underline{p_a}$*. Then* $g(x + \delta) = a$ *for all* $\sqrt{\delta^T \Sigma^{-1} \delta} \leq \Phi^{-1}_{r,n}(\underline{p_a}) - q_{50}$*, where* $r := \sqrt{\delta^T \Sigma^{-1} \delta}$*,* $\Phi^{-1}_{r,n}(\underline{p_a})$ *is the quantile function of the* χ *distribution of* d *degrees of freedom, and* q_{50} *is the* 50^{th} *quantile.*

See Appendix A.3 for the proof. In Theorem 7.4, we show that a smoothed classifier g is robust around x within ℓ_2 Mahalanobis distance $\sqrt{\delta^T \Sigma^{-1} \delta} \leq \Phi^{-1}_{r,n}(p_a) - q_{50}$, where $\Phi^{-1}_{r,n}(p_a)$ is the quantile function for probability p_a. The same result holds if we replace p_a with lower bound $\underline{p_a}$.

We implement our non-uniform approach into randomized smoothing by modifying [172] with our non-isotropic noise space. Table 7.9 shows certification S.R. of Uniform-δ and NU-δ-MDt, when they are certified by standard randomized smoothing with $\mathcal{N}(0, \sigma I)$ (UC), and our non-uniform methods with $\mathcal{N}(0, \Sigma_y)$ for corresponding Σ_y. That is, $\Sigma_{y=0}$ for NUC-MDt, $\Sigma_{y=\{0,1\}}$ NUC-MD, $\frac{1}{\rho^2} I$ for NUC-Pearson and $\frac{1}{s^2} I$ for NUC-SHAP are used when the average training distortion budget is $\|\delta\|_2$=5 and the average certification distortion is $\|\delta\|_2$=2.8. Table 7.9 shows that NU-δ-MDt is certifiably robust for more samples than Uniform-δ for all certification methods. Moreover, certification with non-uniform noise, especially with NUC-MDt, provides higher certification S.R. compared to uniform noise.

7.5 Conclusions

In this work, we study adversarial robustness against evasion attacks, with a focus on applications where input features have to comply with certain domain constraints. We assume Gaussian data distribution in our consistency analysis, as well as precomputed covariance matrix and Shapley values. Under these assumptions, our results on three different applications demonstrate that non-uniform perturbation sets in AT improve adversarial robustness, and non-uniform bounds provide better robustness certification.

As an unintended negative social impact, our insights might be used by malicious parties to generate AEs. However, this work provides the necessary defense mechanisms against these potential attacks.

Chapter 8

Conclusions

Right to privacy is a fundamental human right, which has been recognized in the Universal Declaration of Human Rights [173]. Specifically, information privacy is one's right to control how personal information is collected, shared, archived or used. With the increasing number of countries enacting their own privacy regulations, such as the General Data Protection Regulation (GDPR), California Consumer Privacy Act (CCPA), Personal Information Protection and Electronic Documents Act (PIPEDA), service providers' failure to follow applicable data privacy may lead to fines, lawsuits, and even prohibition of a site's use in certain jurisdictions [9]. Hence, addressing private data sharing problem is necessary. Especially when we consider the emerging IoT technologies and growing number of services that ask the users to share their personal data, private data sharing techniques are the key tools bridging between the service providers' and users' demands. One of the most important changes that has been brought to GDPR in 2018 is the need for *privacy by design*. While the *privacy by default* means that when a service is released to the public, the strictest privacy settings should apply by default, *privacy by design* states that any personal data processing action must contain privacy-preserving at every step. The techniques proposed in this dissertation target achieving the *privacy by design*.

In this dissertation, we have exclusively focused on PUT for data sharing using information theoretic tools. We investigated various methods for sharing a modified version of the user data to keep the sensitive information private, such as using RB and RES for SM data, noise injection, data release mechanism selection and exploiting the physical characteristic of the communication channel. The main advantage of our approaches is that the information theoretic metrics provide theoretical guarantees on the achievable level of privacy and utility. Hence, the proposed approaches are provably effective regardless of the computational capability of the attacker. Besides learning the best released data distributions for privacy applications, we have also proposed a method to generate realistic adversarial example distributions for trustworthiness of machine learning models in security-critical applications. This can be considered a complementary topic for privacy-critical applications that we have presented in the earlier chapters. We proposed defenses against adversarial attacks for various cyber-security domains, such as malware, fraud and spam detection, which has not widely been covered in the literature.

In Chapter 3, we have presented an extensive overview for SM privacy enabling techniques containing both data manipulation and demand shaping. We introduced MDP formulations for information theoretic privacy in SM systems and analyzed their solutions. Besides the existing work, we also proposed PUT for SM with an RB and a RES under a special energy generation process and solved it by DyP.

In Chapter 4, we have studied a fundamental PUT when a user is sharing sensitive time-series data with the SP. While Chapter 3 has the main focus on demand shaping techniques, here, data obfuscation techniques have been proposed and information theoretic guarantees are provided for the PUT. We have focused on the information leakage at the trace level. This is due to the fact that prior works mostly preserve the privacy for the current time but may still leak significant amount of information as the adversary can exploit temporal correlations in a trace. We have measured the time-series privacy by the MI between the released and the original trajectories. By characterizing the optimal solution using the Markov property in the time-series, we have proposed a simplified online private data release policy which preserves the optimality. We have reformulated the online data release problem as an MDP, and numerically evaluated it using A2C-DRL on both synthetic data and GPS trajectory dataset.

In Chapter 5, we have considered a scenario in which the data release mechanisms are fixed and the user actively chooses from among them to make sure the utility received from the SP is maximized while his confidence about a sensitive latent information is kept below a threshold. The user stops sharing her data with the SP right before this threshold is exceeded. We consider two different scenarios and various privacy and utility measures. In the first scenario, we assume the user is only concerned about the PUT and does not consider the stopping time for data release. The proposed policy maximizes the confidence of the SP on the non-sensitive information which is represented by the SP's belief on its true value, and stops the data release when the confidence on the sensitive information reaches the threshold. In this scenario, MI between the released data and the non-sensitive information is also considered as a utility measure and compared with the belief utility numerically. In the second scenario, the user aims to minimize the SP's error probability in non-sensitive information as quickly as possible while keeping his belief in the true value of the sensitive variable below a threshold. Besides the belief based privacy constraint, we also consider the MI between the sensitive variable and the released data as the privacy measure. We numerically compare the belief and MI-based privacy constraints that represent the worst-case and the average case privacy policies, respectively. Similarly to Chapter 4, in our numerical evaluations, we have used MDP formulation and A2C-DRL solution both scenarios.

In Chapter 6, we have studied a wiretap channel scenario in which the user wants to share her data with a legitimate receiver over a noisy communication channel, and a passive eavesdropper tries to infer the user's sensitive information through his noisy channel. Similar to previous chapters, we considered privacy-aware data sharing in this scenario, i.e., a certain level of information leakage about the sensitive information to the eavesdropper is allowed in return of utility from the receiver. In addition to the private data sharing techniques presented previously, in this chapter, we have also exploited the physical characteristics of the noisy channel to preserve privacy. We have evaluated the performance of sharing image data in a wiretap channel setting represented by a VAE and a classifier.

In Chapter 7, we have investigated the trustworthiness of neural network models for security-critical applications. So far we had focused on passive adversaries which are curious about the user's sensitive information, and breach the privacy. We have proposed various data modification techniques, such as demand shaping, noise injection, and etc. In this chapter, as a complementary work, we have focused on active adversaries which inject noise in the test samples to create adversarial examples that can evade the DNN. We propose an empirical defense that exploits the input data distribution to generate realistic adversarial examples during training. We have also proposed robustness certification methods with non-uniform certification bounds around the data samples. Robustness of DNNs against realistic attacks is crucial for certain applications, such as malware, fraud and spam detection, since these applications are critical for the user's security. However, in the literature, most work has focused on CV domain, which has distinct properties than other domains. Since the defenses commonly used in the literature for CV domain do not usually provide high robustness for other applications, there is a need for more studies on generating realistic attacks and effective defenses in other domains than CV. This chapter has proposed a complete attack-defense-certification approach especially in these less explored domains.

Research Challenges

In this dissertation, we have studied several problems related to privacy and security in cyber-physical systems, e.g., private data sharing and neural network security. However, there are challenges in addressing certain questions and the literature still lacks solutions for these problems.

Firstly, there are various application-dependent privacy measures in the literature. The lack of a generic privacy measure makes it difficult to compare different privacy-preserving strategies. Hence, information theoretic measures and tools, such as MI that has been used throughout the thesis, are more preferable since they provide theoretical guarantees on the achievable privacy and utility level regardless of the computational capability of the attacker. However, using information theoretic measures leads us to the next challenge, i.e., the concerns about the data-driven real-world applications.

In most real-world applications, data distributions are not available to either the user or the SP. However, computation of correct information theoretic measures relies heavily on accurate estimation of the underlying distributions. Various variational bounds have been proposed for estimating the MI using neural networks. Although these bounds are effective for certain downstream tasks, they are still far from representing the real MI. Hence, it is crucial to further investigate tools for accurate estimation of MI for privacy sensitive applications.

Finally, robustness of DNNs against active adversaries must be investigated further in domains other than CV. For example, every year there is a new wave of cyber-attacks crafted by attackers using AI and new technologies in malware domain. This makes it difficult to keep up with new unseen data for traditional malware detectors. On the other hand, DNN detectors might also fail when the attacks are well-crafted such that they are imperceptible to a domain expert and modified to evade the detection. Defending the detection in such domains is not an easy task since the adversarial examples are crafted on the real malware binaries rather than deferentiable DNN inputs, that we call problem space attacks. Typical CV domain defenses cannot be easily mapped from feature space to problem space, therefore, there is a high demand in the literature for realistic defenses that are robust against problem space attacks.

In conclusion, despite the various works targeting private data sharing with passive adversaries and defenses against active adversaries, the literature still lacks unified solutions for both private data sharing and adversarial defenses. However, we hope that our work presented in this dissertation has contributed towards answering some of these questions and unsolved issues in privacy and security problems, as well as encouraging further developments in the field.

Future Directions

ITP has been widely studied in the past decades, however, only recent works have provided data-driven approaches for real-world applications. For example, similarly to Chapter 3, [174] and [175] propose privacy-cost trade-off for SM systems using ITP and provide MDP solutions via DyP. Moreover, they extend this approach further to real-data and solve an RL problem using Q-learning. While [174] assumes distribution knowledge as in Chapter 3, [175] uses DNNs to estimate the conditional distribution for a lower bound on MI privacy. A potential future direction for SM privacy application can be a fully data-driven ITP approach, which uses tighter lower and upper bounds for accurate MI estimation. This direction also requires considering stationarity assumption of RL approaches while learning MI approximation, which is a big challenge.

In Chapter 6, we have proposed an end-to-end learning for privacy over a wiretap channel. We estimate MI-based terms for both privacy and utility by assuming certain DNN models for both the legitimate receiver and the eavesdropper, since we do not have access to the real distributions and it is intractable to estimate the priors. Another future direction can be considering privacy-aware communications over a wiretap channel without making an assumption on the receiver or eavesdropper network.

Finally, we have proposed an effective adversarial defense in Chapter 7 using non-uniform perturbations during adversarial training. The proposed method is generalizable since it provides robustness against both feature-space and problem-space attacks. A potential extension of this work can be applying non-uniform perturbations in the problem-space, e.g., malware space, by taking challenging domain specific constraints into account.

Bibliography

[1] J. Z. Kolter and M. J. Johnson, “Redd: A public data set for energy disaggregation research,” in *Proceedings of the SustKDD Workshop on Data Mining Applications in Sustainability*, Aug 2011.

[2] E. Erdemir, P. L. Dragotti, and D. Gündüz, “Privacy-cost trade-off in a smart meter system with a renewable energy source and a rechargeable battery,” *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2019.

[3] E. Wong and Z. Kolter, “Provably robust neural networks,” 2018. [Online]. Available: https://github.com/locuslab/convex_adversarial

[4] V. Primault, A. Boutet, S. B. Mokhtar, and L. Brunie., “The long road to computational location privacy: A survey,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2772–2793, Oct 2018.

[5] Office-Watch.com, “Image privacy breach in microsoft office,” Mar 2021. [Online]. Available: https://office-watch.com/2021/image-privacy-breach-in-microsoft-office/

[6] G. Giaconi, D. Gündüz, and H. V. Poor, “Privacy-aware smart metering: Progress and challenges,” *IEEE Signal Processing Magazine*, vol. 35, no. 6, pp. 59–78, Nov 2018.

[7] S. R and V. V, “Ecg-based secure healthcare monitoring system in body area networks,” in *2018 Fourth Int’l Conf. on Biosignals, Images and Instrum. (ICBSII)*, Mar 2018, pp. 206–212.

[8] T. Wearing and N. Dragoni, “Security and privacy issues in health monitoring systems: ecare@home case study,” in *Proc. of the Int’l Conf. on IoT Technol. for HealthCare*, Oct 2016, pp. 165–170.

[9] S. Philips, “Machine learning and data security,” May 2018. [Online]. Available: https://datascience.foundation/datatalk/machine-learning-and-data-security

[10] C. Dwork, “Differential privacy,” in *International Colloquium on Automata, Languages, and Programming.* Springer, 2006, pp. 1–12.

[11] S. Asoodeh, F. Alajaji, and T. Linder, “Notes on information-theoretic privacy,” in *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2014, pp. 1272–1278.

[12] Y. Cao, M. Yoshikawa, Y. Xiao, and L. Xiong, “Quantifying differential privacy under temporal correlations,” in *2017 IEEE 33rd Int’l Conf. Data Eng. (ICDE)*, Apr 2017, pp. 821–832.

[13] F. du Pin Calmon and N. Fawaz, “Privacy against statistical inference,” in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2012, pp. 1401–1408.

[14] A. Zamani, T. Oechtering, and M. Skoglund, “A design framework for epsilon-private data disclosure,” *ArXiv*, vol. abs/2009.01704, 2020.

[15] B. Rassouli and D. Gündüz, “On perfect privacy,” *IEEE Journal on Selected Areas in Information Theory*, pp. 1–1, 2021.

[16] Z. Li, T. J. Oechtering, and D. Gündüz, “Privacy against a hypothesis testing adversary,” *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 6, pp. 1567–1581, Jun 2019.

[17] Y.-X. Wang, J. Lei, and S. E. Fienberg, “On-average kl-privacy and its equivalence to generalization for max-entropy mechanisms,” in *International Conference on Privacy in Statistical Databases*, 2016.

[18] B. Rassouli and D. Gündüz, “Optimal utility-privacy trade-off with total variation distance as a privacy measure,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 594–603, 2020.

[19] J. Liao, O. Kosut, L. Sankar, and F. P. Calmon, “A tunable measure for information leakage,” in *2018 IEEE Int’l Symp. Inf. Theory (ISIT)*, Jun 2018, pp. 701–705.

[20] I. Issa, S. Kamath, and A. B. Wagner, “An operational measure of information leakage,” in *2016 Annual Conference on Information Science and Systems (CISS).* IEEE, 2016, pp. 234–239.

[21] S. A. Osia, B. Rassouli, H. Haddadi, H. R. Rabiee, and D. Gündüz, “Privacy against brute-force inference attacks,” in *2019 IEEE Int’l Symp. Inf. Theory (ISIT)*, Jul 2019, pp. 637–641.

[22] A. Wood, M. Altman, A. Bembenek, M. Bun, M. Gaboardi, J. Honaker, K. Nissim, D. R. O'Brien, T. Steinke, and S. Vadhan, "Differential privacy: A primer for a non-technical audience," *Vand. J. Ent. & Tech. L.*, vol. 21, p. 209, 2018.

[23] "Gartner says over 40% of privacy compliance technology will rely on artificial intelligence in the next three years," Gartner. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases

[24] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[25] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli, "Evasion attacks against machine learning at test time," in *Joint European conference on machine learning and knowledge discovery in databases.* Springer, 2013, pp. 387–402.

[26] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[27] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[28] K. Dvijotham, R. Stanforth, S. Gowal, T. A. Mann, and P. Kohli, "A dual approach to scalable verification of deep networks." in *UAI*, vol. 1, 2018, p. 2.

[29] T. Gehr, M. Mirman, D. Drachsler-Cohen, P. Tsankov, S. Chaudhuri, and M. Vechev, "Ai2: Safety and robustness certification of neural networks with abstract interpretation," in *2018 IEEE Symposium on Security and Privacy (SP).* IEEE, 2018, pp. 3–18.

[30] G. Singh, T. Gehr, M. Mirman, M. Püschel, and M. Vechev, "Fast and effective robustness certification," *Advances in Neural Information Processing Systems*, vol. 31, pp. 10 802–10 813, 2018.

[31] A. Raghunathan, J. Steinhardt, and P. S. Liang, "Semidefinite relaxations for certifying robustness to adversarial examples," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 877–10 887.

[32] E. Wong and Z. Kolter, "Provable defenses against adversarial examples via the convex outer adversarial polytope," in *International Conference on Machine Learning.* PMLR, 2018, pp. 5286–5295.

[33] H. Zhang, T.-W. Weng, P.-Y. Chen, C.-J. Hsieh, and L. Daniel, "Efficient neural network robustness certification with general activation functions," in *Advances in neural information processing systems*, 2018, pp. 4939–4948.

[34] S. Huang, N. Papernot, I. Goodfellow, Y. Duan, and P. Abbeel, "Adversarial attacks on neural network policies," *arXiv preprint arXiv:1702.02284*, 2017.

[35] V. Behzadan and A. Munir, "Vulnerability of deep reinforcement learning to policy induction attacks," in *International Conference on Machine Learning and Data Mining in Pattern Recognition.* Springer, 2017, pp. 262–275.

[36] G. Giaconi and D. Gündüz, "Smart meter privacy with renewable energy and a finite capacity battery," in *IEEE Int. Workshop on Sig. Proc. Advances in Wireless Communications (SPAWC)*, Jul 2016, pp. 1–5.

[37] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography*, S. Halevi and T. Rabin, Eds. Springer Berlin Heidelberg, 2006.

[38] A. Aristodimou, A. Antoniades, and C. S. Pattichis, "Privacy preserving data publishing of categorical data through k-anonymity and feature selection," *Healthcare Technol. Lett.*, vol. 3, pp. 16–21(5), Mar 2016.

[39] N. Saleheen, S. Chakraborty, N. Ali, M. M. Rahman, S. M. Hossain, R. Bari, E. Buder, M. Srivastava, and S. Kumar, "msieve: Differential behavioral privacy in time series of mobile sensor data," in *Proc. of the 2016 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Comput.*, ser. UbiComp '16. New York, NY, USA: ACM, 2016, pp. 706–717.

[40] R. Shokri, C. Troncoso, C. Diaz, J. Freudiger, and J.-P. Hubaux, "Unraveling an old cloak: k-anonymity for location privacy," in *ACM Conference on Computer and Communications Security*, Sep 2010.

[41] R. Shokri, G. Theodorakopoulos, C. Troncoso, J.-P. Hubaux, and J.-Y. Le Boudec, "Protecting location privacy: Optimal strategy against localization attacks," in *ACM Conference on Computer and Communications Security*, Oct 2012, pp. 617–627.

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

[43] Z. Montazeri, A. Houmansadr, and H. Pishro-Nik, "Achieving perfect location privacy in wireless devices using anonymization," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 11, pp. 2683–2698, Nov 2017.

[44] N. Takbiri, A. Houmansadr, D. L. Goeckel, and H. Pishro-Nik, "Matching anonymized and obfuscated time series to users' profiles," *IEEE Trans. Inf. Theory*, vol. 65, no. 2, pp. 724–741, Feb 2019.

[45] Z. Zhang, Z. Qin, L. Zhu, J. Weng, and K. Ren, "Cost-friendly differential privacy for smart meters: Exploiting the dual roles of the noise," *IEEE Trans. Smart Grid*, vol. 8, no. 2, pp. 619–626, Mar 2017.

[46] G. Giaconi, D. Gündüz, and H. V. Poor, "Smart meter privacy with renewable energy and an energy storage device," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 1, pp. 129–142, Jan 2018.

[47] W. Zhang, M. Li, R. Tandon, and H. Li, "Online location trace privacy: An information theoretic approach," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 1, pp. 235–250, Jan 2019.

[48] V. Bindschaedler and R. Shokri, "Synthesizing plausible privacy-preserving location traces," in *IEEE Symposium on Security and Privacy (SP)*, May 2016, pp. 546–563.

[49] J. Hua, W. Tong, F. Xu, and S. Zhong, "A geo-indistinguishable location perturbation mechanism for location-based services supporting frequent queries," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 5, pp. 1155–1168, May 2018.

[50] S. Li, A. Khisti, and A. Mahajan, "Information-theoretic privacy for smart metering systems with a rechargeable battery," *IEEE Transactions on Information Theory*, vol. 64, no. 5, pp. 3679–3695, May 2018.

[51] G. Giaconi and D. Gündüz, "Smart meter privacy with renewable energy and a finite capacity battery," in *2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Jul 2016, pp. 1–5.

[52] R. Shokri, G. Theodorakopoulos, J. Le Boudec, and J. Hubaux, "Quantifying location privacy," in *IEEE Symposium on Security and Privacy*, May 2011, pp. 247–262.

[53] Y. Cao, M. Yoshikawa, Y. Xiao, and L. Xiong, "Quantifying differential privacy in continuous data release under temporal correlations," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 7, p. 1281—1295, Jul 2019.

[54] C. E. Shannon, "Communication theory of secrecy systems," *The Bell system technical journal*, vol. 28, no. 4, pp. 656–715, 1949.

[55] H. Yamamoto, "A rate-distortion problem for a communication system with a secondary decoder to be hindered," *IEEE Transactions on Information Theory*, vol. 34, no. 4, pp. 835–842, 1988.

[56] S. Verdú, "α-mutual information," in *2015 Information Theory and Applications Workshop (ITA)*, 2015, pp. 1–6.

[57] A. D. Wyner, "The wire-tap channel," *Bell system technical journal*, vol. 54, no. 8, pp. 1355–1387, 1975.

[58] I. Csiszár and J. Korner, "Broadcast channels with confidential messages," *IEEE transactions on information theory*, vol. 24, no. 3, pp. 339–348, 1978.

[59] H. Yamamoto, "A source coding problem for sources with additional outputs to keep secret from the receiver or wiretappers," *IEEE Trans. Inf. Theory*, vol. 29, pp. 918–923, 1983.

[60] D. Agrawal and C. C. Aggarwal, "On the design and quantification of privacy preserving data mining algorithms," in *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, 2001, pp. 247–255.

[61] P. Venkitasubramaniam, "Privacy in stochastic control: a Markov decision process perspective," in *2013 51st Annual Allerton Conf. Commun., Control, and Comput. (Allerton)*, Oct 2013, pp. 381–388.

[62] M. A. Erdogdu and N. Fawaz, "Privacy-utility trade-off under continual observation," in *2015 IEEE Int'l Symp. Inf. Theory (ISIT)*, Jun 2015, pp. 1801–1805.

[63] E. Nekouei, H. Sandberg, M. Skoglund, and K. H. Johansson, "Privacy-aware minimum error probability estimation: An entropy constrained approach," *23rd International Symposium on Mathematical Theory of Networks and Systems*, Jul 2018.

[64] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecom. and Signal Proc.)*. New York, NY, USA: Wiley-Interscience, 2006.

[65] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, 1st ed. USA: John Wiley & Sons, Inc., 1994.

[66] L. Kallenberg, "Lecture notes in markov decision processes," 2011.

[67] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. II*, 3rd ed. Athena Scientific, 2007.

[68] N. Saldi, T. Linder, and S. Yüksel, *Approximations for Partially Observed Markov Decision Processes*. Cham: Springer International Publishing, 2018, pp. 99–123.

[69] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. The MIT Press, 2018.

[70] V. R. Konda and J. N. Tsitsiklis, "On actor-critic algorithms," *SIAM J. Control Optim.*, vol. 42, no. 4, pp. 1143–1166, Apr 2003.

[71] I. Grondman, L. Busoniu, G. A. D. Lopes, and R. Babuska, "A survey of actor-critic reinforcement learning: Standard and natural policy gradients," *IEEE Trans. Syst., Man, Cybern., Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1291–1307, Nov 2012.

[72] A. Chakraborty, M. Alam, V. Dey, A. Chattopadhyay, and D. Mukhopadhyay, "Adversarial attacks and defences: A survey," *ArXiv*, vol. abs/1810.00069, 2018.

[73] E. Wong, L. Rice, and J. Z. Kolter, "Fast is better than free: Revisiting adversarial training," *arXiv preprint arXiv:2001.03994*, 2020.

[74] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," *arXiv preprint arXiv:1611.01236*, 2016.

[75] S. Wang, Y. Chen, A. Abdou, and S. Jana, "Mixtrain: Scalable training of formally robust neural networks," *arXiv preprint arXiv:1811.02625*, vol. 14, 2018.

[76] M. Mirman, T. Gehr, and M. Vechev, "Differentiable abstract interpretation for provably robust neural networks," in *International Conference on Machine Learning*, 2018, pp. 3578–3586.

[77] H. Farhangi, "The path of the smart grid," *IEEE Power and Energy Magazine*, vol. 8, no. 1, pp. 18–28, Jan 2010.

[78] G. Giaconi, D. Gündüz, and H. V. Poor, "Privacy-aware smart metering: Progress and challenges," *IEEE Signal Processing Magazine*, vol. 35, no. 6, pp. 59–78, Nov 2018.

[79] P. McDaniel and S. McLaughlin, "Security and privacy challenges in the smart grid," *IEEE Security Privacy*, vol. 7, no. 3, pp. 75–77, May 2009.

[80] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, p. 1870–1891, Dec. 1992.

[81] S. Makonin, "Approaches to non-intrusive load monitoring (nilm) in the home," *Technical report*, 2012.

[82] C. Cuijpers and B.-J. Koops, *Smart Metering and Privacy in Europe: Lessons from the Dutch Case.* Dordrecht: Springer Netherlands, 2013, pp. 269–293.

[83] "Naperville smart meter awareness v. city of naperville," vol. 16, p. 3766, 2018.

[84] F. D. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," *in Proc. 6th Workshop Security and Trust Management*, vol. 6710, p. 226–238, 2017.

[85] Y. Kim, E. C. H. Ngai, and M. B. Srivastava, "Cooperative state estimation for preserving privacy of user behaviors in smart grid," *2011 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 178–183, 2011.

[86] M. Arrieta and I. Esnaola, "Smart meter privacy via the trapdoor channel," *2017 IEEE International Conference on Smart Grid Communications (SmartGrid-Comm)*, pp. 277–282, 2017.

[87] S. Han, U. Topcu, and G. J. Pappas, "Event-based information-theoretic privacy: A case study of smart meters," *American Control Conference (ACC)*, pp. 2074–2079, Jul 2016.

[88] R. R. Avula, T. J. Oechtering, and D. Månsson, "Privacy-preserving smart meter control strategy including energy storage losses," in *2018 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, 2018, pp. 1–6.

[89] S. Li, A. Khisti, and A. Mahajan, "Information-theoretic privacy for smart metering systems with a rechargeable battery," *IEEE Transactions on Information Theory*, vol. 64, no. 5, pp. 3679 – 3695, 2018.

[90] J. Yao and P. Venkitasubramaniam, "On the privacy-cost tradeoff of an in-home power storage mechanism," in *Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Oct 2013, pp. 115–122.

[91] G. Giaconi, D. Gündüz, and H. V. Poor, "Smart meter privacy with renewable energy and an energy storage device," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 1, pp. 129–142, Jan 2018.

[92] J. Gomez-Vilardebo and D. Gündüz, "Smart meter privacy for multiple users in the presence of an alternative energy source," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 132–141, Jan 2015.

[93] O. Tan, D. Gunduz, and H. V. Poor, "Increasing smart meter privacy through energy harvesting and storage devices," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1331–1341, Jul 2013.

[94] J. Chin, T. Tinoco De Rubira, and G. Hug, "Privacy-protecting energy management unit through model-distribution predictive control," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 3084–3093, Nov 2017.

[95] D. Rebollo-Monedero, J. Forne, and J. Domingo-Ferrer, "From t-closeness-like privacy to postrandomization via information theory," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 11, pp. 1623–1636, Nov 2010.

[96] S. R. Rajagopalan, L. Sankar, S. Mohajer, and H. V. Poor, "Smart meter privacy: A utility-privacy framework," in *2011 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Oct 2011, pp. 190–195.

[97] J. Sakuma, S. Kobayashi, and R. N. Wright, "Privacy-preserving reinforcement learning," in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML '08. New York, NY, USA: ACM, 2008, pp. 864–871.

[98] C. Efthymiou and G. Kalogridis, "Smart grid privacy via anonymization of smart metering data," in *2010 First IEEE International Conference on Smart Grid Communications*, Oct 2010, pp. 238–243.

[99] H. Yang, L. Cheng, and M. C. Chuah, "Evaluation of utility-privacy trade-offs of data manipulation techniques for smart metering," in *2016 IEEE Conference on Communications and Network Security (CNS)*, Oct 2016, pp. 396–400.

[100] B. Rassouli and D. Gunduz, "On perfect privacy," *IEEE Intl Symposium on Information Theory (ISIT)*, June 2018.

[101] J. Gomez-Vilardebo and D. Gündüz, "Smart meter privacy for multiple users in the presence of an alternative energy source," in *2013 IEEE Global Conference on Signal and Information Processing*, Dec 2013, pp. 859–862.

[102] H. Kim, M. Marwah, M. F. Arlitt, G. Lyon, and J. Han, "Unsupervised disaggregation of low frequency power measurements," *Proc. SIAM Conf. Data Mining*, vol. 11, pp. 747–758, Apr 2011.

[103] R. R. Avula, J.-X. Chin, T. J. Oechtering, G. Hug, and D. Månsson, "Design framework for privacy-aware demand-side management with realistic energy storage model," *IEEE Transactions on Smart Grid*, vol. 12, no. 4, pp. 3503–3513, 2021.

[104] R. Bellman, "On the theory of dynamic programming," *Proceedings of the National Academy of Sciences*, pp. 716–719, 1952.

[105] A. Mishra, D. Irwin, P. Shenoy, J. Kurose, and T. Zhu, "Smartcharge: Cutting the electricity bill in smart homes with energy storage," in *3rd Int'l Conf. on Future Energy Systems: Where Energy, Computing and Communication Meet*, ser. e-Energy '12. New York, NY, USA: ACM, 2012, pp. 29:1–29:10.

[106] T. Hubert and S. Grijalva, "Modeling for residential electricity optimization in dynamic pricing environments," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 2224–2231, Dec 2012.

[107] O. Tan, J. Gómez-Vilardebó, and D. Gündüz, "Privacy-cost trade-offs in demand-side management with storage," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 6, pp. 1458–1469, Jun 2017.

[108] Z. Li, T. J. Oechtering, and D. Gündüz, "Privacy against a hypothesis testing adversary," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1567–1581, Jun 2019.

[109] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W.-Y. Ma, "Understanding mobility based on gps data," in *Int'l Conf. Ubiquitous Comput.* New York, NY, USA: ACM, 2008, pp. 312–321.

[110] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, "Mining interesting locations and travel sequences from gps trajectories," in *Proc. of the 18th Int'l Conf. World Wide Web.* ACM, 2009, pp. 791–800.

[111] E. Erdemir, P. L. Dragotti, and D. Gündüz, "Privacy-aware location sharing with deep reinforcement learning," in *IEEE Workshop on Information Forensics and Security (WIFS)*, Delft, The Netherlands, Dec 2019.

[112] A. Karatzoglou, D. Koehler, and M. Beigl, "Semantic-enhanced multi-dimensional markov chains on semantic trajectories for predicting future locations †," *Sensors (Basel, Switzerland)*, vol. 18, 2018.

[113] J. Torriti, "A review of time use models of residential electricity demand," *Renew. Sust. Energ. Rev.*, vol. 37, pp. 265 – 272, 2014.

[114] B. Chen and Y. Hong, "Testing for the markov property in time series," *Econometric Theory*, vol. 28, no. 1, p. 130–178, 2012.

[115] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of markov decision processes," *Mathematics of Operations Research*, vol. 12, no. 3, pp. 441–450, 1987.

[116] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

[117] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013.

[118] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *KDD*, 1996.

[119] M. Shoaib, H. Scholten, P. J. M. Havinga, and O. D. Incel, "A hierarchical lazy smoking detection algorithm using smartwatch sensors," in *2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)*, 2016, pp. 1–6.

[120] E. Erdemir, P. L. Dragotti, and D. Gündüz, "Privacy-aware time-series data sharing with deep reinforcement learning," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 389–401, 2021.

[121] A. Makhdoumi, S. Salamatian, N. Fawaz, and M. Médard, "From the information bottleneck to the privacy funnel," in *2014 IEEE Information Theory Workshop (ITW 2014)*, 2014, pp. 501–505.

[122] M. Sun and W. P. Tay, "Inference and data privacy in IoT networks," in *2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2017, pp. 1–5.

[123] A. Tripathy, Y. Wang, and P. Ishwar, "Privacy-preserving adversarial networks," in *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2019, pp. 495–505.

[124] E. Erdemir, P. L. Dragotti, and D. Gündüz, "Active privacy-utility trade-off against a hypothesis testing adversary," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 2660–2664.

[125] N. Tishby, F. Pereira, and W. Bialek, "The information bottleneck method," *Proceedings of the 37th Allerton Conference on Communication, Control and Computation*, vol. 49, Jul 2001.

[126] J. Liao, L. Sankar, V. Y. F. Tan, and F. du Pin Calmon, "Hypothesis testing under mutual information privacy constraints in the high privacy regime," *IEEE Trans. on Information Forensics and Security*, vol. 13, no. 4, pp. 1058–1071, 2018.

[127] D. Barber and F. Agakov, "The im algorithm: a variational approach to information maximization," in *NIPS 2003*, 2003.

[128] E. Erdemir, D. Gündüz, and P. L. Dragotti, "Smart meter privacy," in *Privacy in Dynamical Systems*, 1st ed., F. Farokhi, Ed. Springer Singapore, 2020.

[129] D. P. Bertsekas and S. E. Shreve, *Stochastic Optimal Control: The Discrete-Time Case.* Belmont, CA: Athena Scientific, 2007.

[130] M. Bloch and J. Barros, *Physical-Layer Security: From Information Theory to Security Engineering*, Jan 2011.

[131] M. Z. Hameed, A. György, and D. Gündüz, "The best defense is a good offense: Adversarial attacks to avoid modulation detection," *IEEE Trans. Info. Forensics and Security*, vol. 16, 2021.

[132] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, "Machine learning in the air," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2184–2199, 2019.

[133] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Tran. on Cognitive Comms. and Networking*, vol. 3, no. 4, pp. 563–575, 2017.

[134] E. Bourtsoulatze, D. B. Kurka, and D. Gündüz, "Deep joint source-channel coding for wireless image transmission," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 567–579, 2019.

[135] K.-L. Besser, P.-H. Lin, C. R. Janda, and E. A. Jorswieck, "Wiretap code design by neural network autoencoders," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3374–3386, 2020.

[136] T. Marchioro, N. Laurenti, and D. Gündüz, "Adversarial networks for secure wireless communications," in *IEEE Int'l Conf. on Acoustics, Speech and Signal Proc. (ICASSP)*, 2020, pp. 8748–8752.

[137] R. Fritschek, R. F. Schaefer, and G. Wunder, "Deep learning for the gaussian wiretap channel," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.

[138] ——, "Deep learning based wiretap coding via mutual information estimation," ser. WiseML '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 74–79.

[139] N. Merhav, "Shannon's secrecy system with informed receivers and its application to systematic coding for wiretapped channels," *IEEE Transactions on Information Theory*, vol. 54, no. 6, pp. 2723–2734, 2008.

[140] M. Bloch, O. Günlü, A. Yener, F. Oggier, H. V. Poor, L. Sankar, and R. F. Schaefer, "An overview of information-theoretic security and privacy: Metrics, limits and applications," *IEEE Journal on Selected Areas in Information Theory*, vol. 2, no. 1, pp. 5–22, 2021.

[141] B. Poole, S. Ozair, A. Van Den Oord, A. Alemi, and G. Tucker, "On variational bounds of mutual information," in *International Conference on Machine Learning.* PMLR, 2019, pp. 5171–5180.

[142] K. Choi, K. Tatwawadi, A. Grover, T. Weissman, and S. Ermon, "Neural joint source-channel coding," in *International Conference on Machine Learning.* PMLR, 2019, pp. 1182–1192.

[143] M. F. Zeager, A. Sridhar, N. Fogal, S. Adams, D. E. Brown, and P. A. Beling, "Adversarial learning in credit card fraud detection," in *2017 Systems and Information Engineering Design Symposium (SIEDS).* IEEE, 2017, pp. 112–116.

[144] N. Liu, H. Yang, and X. Hu, "Adversarial detection with model interpretation," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 1803–1811.

[145] V. Ballet, X. Renard, J. Aigrain, T. Laugel, P. Frossard, and M. Detyniecki, "Imperceptible adversarial attacks on tabular data," *arXiv preprint arXiv:1911.03274*, 2019.

[146] E. Levy, Y. Mathov, Z. Katzir, A. Shabtai, and Y. Elovici, "Not all datasets are born equal: On heterogeneous data and adversarial examples," *arXiv preprint arXiv:2010.03180*, 2020.

[147] D. Li and Q. Li, "Adversarial deep ensemble: Evasion attacks and defenses for malware detection," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3886–3900, 2020.

[148] F. Pierazzi, F. Pendlebury, J. Cortellazzi, and L. Cavallaro, "Intriguing properties of adversarial ml attacks in the problem space," in *2020 IEEE Symposium on Security and Privacy (SP).* IEEE, 2020, pp. 1332–1349.

[149] I. Rosenberg, S. Meir, J. Berrebi, I. Gordon, G. Sicard, and E. O. David, "Generating end-to-end adversarial examples for malware classifiers using explainability," in *2020 Int'l Joint Conf. on Neural Networks (IJCNN).* IEEE, 2020, pp. 1–10.

[150] W. Guo, D. Mu, J. Xu, P. Su, G. Wang, and X. Xing, "Lemna: Explaining deep learning based security applications," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 364–379.

[151] P. C. Mahalanobis, "On the generalized distance in statistics," *Proceedings of the National Institute of Sciences*, vol. 2, pp. 49–55, 1936.

[152] L. S. Shapley, *Notes on the n-Person Game -; II: The Value of an n-Person Game.* Santa Monica, CA": RAND Corporation, 1951.

[153] J. Han, M. Kamber, and J. Pei, “Data transformation and data discretization,” *Data Mining: Concepts and Techniques. Elsevier*, pp. 111–118, 2011.

[154] H. Wang, T. Chen, S. Gui, T.-K. Hu, J. Liu, and Z. Wang, “Once-for-all adversarial training: In-situ tradeoff between robustness and accuracy for free,” *arXiv preprint arXiv:2010.11828*, 2020.

[155] S. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Thirty-First Conference on Neural Information Processing Systems (NeurIPS)*, Dec 2017.

[156] H. S. Anderson and P. Roth, “EMBER: an open dataset for training static PE malware machine learning models,” *arXiv preprint arXiv: 1804.04637*, 2018.

[157] Pdfrate. [Online]. Available: https://github.com/csmutz/pdfrate

[158] C. Smutz and A. Stavrou, “Malicious pdf detection using metadata and structural features,” in *28th Annual Computer Security Applications Conference*, New York, NY, USA, 2012, p. 239–248.

[159] W. Fleshman, *Evading Machine Learning Malware Classifiers*, 2019. [Online]. Available: https://towardsdatascience.com/evading-machine-learning-malware-classifiers-ce52dabdb713

[160] DEFCON, *Machine learning static evasion competition*, 2019. [Online]. Available: https://www.elastic.co/blog/machine-learning-static-evasion-competition

[161] VirusTotal. [Online]. Available: http://www.virustotal.com/

[162] L. Tong, B. Li, C. Hajaj, C. Xiao, N. Zhang, and Y. Vorobeychik, “Improving robustness of ML classifiers against realizable evasion attacks using conserved features,” in *28th USENIX Security Symposium*, Santa Clara, CA, Aug 2019, pp. 285–302.

[163] W. Xu, Y. Qi, and D. Evans, “Automatically evading classifiers: A case study on pdf malware classifiers,” in *NDSS*, 2016.

[164] C. J. Merz and P. Murphy, *UCI repository of machine learning databases*, 1996. [Online]. Available: http://www.cs.uci.edu/~mlearn/MLRepository.html

[165] K. Lee, B. D. Eoff, and J. Caverlee, “Seven months with the devils: A long-term study of content polluters on twitter.” in *5th International AAAI Conference on Weblogs and Social Media (ICWSM)*, Barcelona, 2011.

[166] X. Huang, *Twitter bot detection*, 2017. [Online]. Available: https://github.com/tapilab/is-xhuang1994

[167] N. Liu, H. Yang, and X. Hu, "Interpretation to adversary," 2018. [Online]. Available: https://github.com/ninghaohello/Interpretation2Adversary

[168] M.-I. Nicolae, M. Sinn, M. N. Tran, B. Buesser, A. Rawat, M. Wistuba, V. Zantedeschi, N. Baracaldo, B. Chen, H. Ludwig, I. M. Molloy, and B. Edwards, "Adversarial robustness toolbox v1.0.0," 2019.

[169] L. McInnes, J. Healy, N. Saul, and L. Grossberger, "Umap: Uniform manifold approximation and projection," *The Journal of Open Source Software*, vol. 3, no. 29, p. 861, 2018.

[170] J. M. Cohen, E. Rosenfeld, and J. Z. Kolter, "Certified adversarial robustness via randomized smoothing," *arXiv preprint arXiv:1902.02918*, 2019.

[171] H. Salman, G. Yang, H. Zhang, C.-J. Hsieh, and P. Zhang, "Benchmark for lp-relaxed robustness verification of relu-networks," 2019. [Online]. Available: https://github.com/Hadisalman/robust-verify-benchmark

[172] J. Cohen, E. Rosenfeld, and Z. Kolter, "Certified adversarial robustness via randomized smoothing," 2019. [Online]. Available: https://github.com/locuslab/smoothing

[173] U. G. Assembly *et al.*, "Universal declaration of human rights," *UN General Assembly*, vol. 302, no. 2, pp. 14–25, 1948.

[174] Y. You, Z. Li, and T. J. Oechtering, "Energy management strategy for smart meter privacy and cost saving," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1522–1537, 2021.

[175] M. Shateri, F. Messina, P. Piantanida, and F. Labeau, "Privacy-cost management in smart meters with mutual information-based reinforcement learning," *IEEE Internet of Things Journal*, pp. 1–1, 2021.

Appendix A

Proofs for Chapter 7

A.1 Proof of Theorem 7.2

For an AE x that is generated under the Mahalanobis distance constraint, i.e., $x \in \tilde{\Delta}_{\epsilon,2}$, we can write the following bound:

$$\min_{x \in \tilde{\Delta}_{\epsilon,2}} \log f(x \mid y) \;=\; C - \frac{1}{2}\delta^T \Sigma_y^{-1} \delta \;=\; \log \gamma \tag{A.1}$$

where the second equality is a result of γ-consistency assumption. Then, by using the upper limit of ℓ_2 Mahalanobis distance of δ for $M = \Sigma_y^{-1}$, we get

$$\sqrt{\delta^T \Sigma_y^{-1} \delta} \;=\; \sqrt{2C - 2\log\gamma} \;\leq\; \epsilon. \tag{A.2}$$

A.2 Proof of Theorem 7.3

The linear program with non-uniform input perturbation and relaxed ReLU constraints can be written as

$$\begin{aligned}
\underset{\hat{z}_k}{\text{minimize}} \quad & c^T \hat{z}_k \\
\text{s.t.} \quad & \hat{z}_{i+1} = W_i z_i + b_i, i = 1, \ldots, k-1 \\
& ||\Omega(z_1 - x)||_p \leq \epsilon \\
& z_{i,j} = 0, i = 2, \ldots, k-1, j \in \mathcal{I}_i^- \\
& z_{i,j} = \hat{z}_{i,j}, i = 2, \ldots, k-1, j \in \mathcal{I}_i^+ \\
& \left.\begin{aligned} & z_{i,j} \geq 0, \quad z_{i,j} \geq \hat{z}_{i,j}, \\ & ((u_{i,j} - l_{i,j}) z_{i,j} - u_{i,j}\hat{z}_{i,j}) \leq -u_{i,j} l_{i,j} \end{aligned}\right\}_{\substack{i=2,\ldots,k-1 \\ j \in \mathcal{I}_i}}.
\end{aligned} \tag{A.3}$$

We associate the following Lagrangian variables with each of the constraints except the ℓ_p norm constraint in Problem A.3,

$$\begin{aligned}
\hat{z}_{i+1} &= W_i z_i + b_i \Rightarrow \nu_{i+1} \\
\delta &= z_1 - x \Rightarrow \psi \\
-z_{i,j} &\leq 0 \Rightarrow \mu_{i,j} \\
\hat{z}_{i,j} - z_{i,j} &\leq 0 \Rightarrow \tau_{i,j} \\
((u_{i,j} - l_{i,j})z_{i,j} - u_{i,j}\hat{z}_{i,j}) &\leq -u_{i,j}l_{i,j} \Rightarrow \lambda_{i,j}.
\end{aligned} \tag{A.4}$$

We do not define explicit dual variables for $z_{i,j} = 0$ and $z_{i,j} = \hat{z}_{i,j}$ since they will be zero in the optimization. Then, we create the following Lagrangian by grouping up the terms with z_i, $\hat{z}_i$:

$$\begin{aligned}
L(\mathbf{z}, \hat{\mathbf{z}}, \nu, \delta, \lambda, \tau, \mu, \psi) = &-\sum_{\substack{i=2 \\ j\in\mathcal{I}_i}}^{k-1} (\mu_{i,j} + \tau_{i,j} - \lambda_{i,j}(u_{i,j} - l_{i,j}) + (W_i^T \nu_{i+1})_j) z_{i,j} \\
&+ \sum_{\substack{i=2 \\ j\in\mathcal{I}_i}}^{k-1} (\tau_{i,j} - \lambda_{i,j}u_{i,j} + \nu_{i,j})\hat{z}_{i,j} + (c + \nu_k)^T \hat{z}_k - \sum_{i=1}^{k-1} \nu_{i+1}^T b_i \\
&+ \sum_{\substack{i=2 \\ j\in\mathcal{I}_i}}^{k-1} \lambda_{i,j}u_{i,j}l_{i,j} + \psi^T x + \psi^T \delta - (W_1^T \nu_2 + \psi)^T z_1 \\
\text{subject to} \quad & ||\Omega\delta||_p \leq \epsilon
\end{aligned} \tag{A.5}$$

Now, we take the minimum of $L(.)$ w.r.t $\mathbf{z}$, $\hat{\mathbf{z}}$ and δ:

$$\begin{aligned}
\inf_{\mathbf{z},\hat{\mathbf{z}},\delta} L(\mathbf{z}, \hat{\mathbf{z}}, \nu, \delta, \lambda, \tau, \mu, \psi) = &-\inf_{z_{i,j}} \sum_{\substack{i=2 \\ j\in\mathcal{I}_i}}^{k-1} \big(\mu_{i,j} + \tau_{i,j} - \lambda_{i,j}(u_{i,j} - l_{i,j}) + (W_i^T \nu_{i+1})_j\big) z_{i,j} \\
&+ \inf_{\hat{\mathbf{z}}} \big(\sum_{\substack{i=2 \\ j\in\mathcal{I}_i}}^{k-1} (\tau_{i,j} - \lambda_{i,j}u_{i,j} + \nu_{i,j})\hat{z}_{i,j} + (c + \nu_k)^T \hat{z}_k \big) - \sum_{i=1}^{k-1} \nu_{i+1}^T b_i \\
&+ \sum_{\substack{i=2 \\ j\in\mathcal{I}_i}}^{k-1} \lambda_{i,j}u_{i,j}l_{i,j} + \psi^T x + \inf_{||\Omega\delta||_p \leq \epsilon} \psi^T \delta - \inf_{z_1} (W_1^T \nu_2 + \psi)^T z_1.
\end{aligned} \tag{A.6}$$

We can represent the term $\inf_{||\Omega\delta||_p \leq \epsilon} \psi^T \delta$ independent of δ using the following dual norm definition.

Cauchy-Schwarz inequality for dual norm:

We can write the Cauchy-Schwarz inequality as $\alpha^T\beta \leq ||\alpha||_p||\beta||_q$, where $\frac{1}{p}+\frac{1}{q}=1$ and q norm represents the dual of p norm. Let $\hat{u}=\frac{\alpha}{||\alpha||_p}$, the definition of dual norm is

$$||\beta||_q = \sup_{||\hat{u}||_p\leq 1} \hat{u}^T\beta. \tag{A.7}$$

We can write $\inf\limits_{||\Omega\delta||_p\leq\epsilon} \psi^T\delta = -\sup\limits_{||\Omega\delta||_p\leq\epsilon}(-\psi^T\delta) = -\sup\limits_{||\Omega\delta||_p\leq\epsilon}\psi^T\delta$. For $\alpha=\frac{\Omega\delta}{\epsilon}$ and $\beta=\epsilon\Omega^{-1}\psi$, we get $\delta^T\psi \leq ||\frac{\Omega\delta}{\epsilon}||_p||\epsilon\Omega^{-1}\psi||_q$ which implies $-\sup\limits_{||\Omega\delta||_p\leq\epsilon}\psi^T\delta = -\epsilon||\Omega^{-1}\psi||_q$.

Hence, the minimization of $L(.)$ becomes,

$$\inf_{\mathbf{z},\hat{\mathbf{z}},\delta} L(.) = \begin{cases} -\sum\limits_{i=1}^{k-1}\nu_{i+1}^T b_i + \sum\limits_{\substack{i=2\\ j\in\mathcal{I}_i}}^{k-1}\lambda_{i,j}u_{i,j}l_{i,j} + \psi^T x - \epsilon||\Omega^{-1}\psi||_q & \text{if } \textit{cond.}\\ \\ -\infty & \text{o.w.,}\end{cases} \tag{A.8}$$

where the conditions are

$$\begin{aligned} &\nu_k = -c\\ &W_1^T\nu_2 = -\psi\\ &\nu_{i,j} = 0,\ j\in\mathcal{I}_i^-\\ &\nu_{i,j} = (W_i^T\nu_{i+1})_j,\ j\in\mathcal{I}_i^+\\ &\left.\begin{aligned}((u_{i,j}-l_{i,j})\lambda_{i,j}-\mu_{i,j}-\tau_{i,j}) &= (W_i^T\nu_{i+1})_j\\ \nu_{i,j} &= u_{i,j}\lambda_{i,j}-\tau_{i,j}\end{aligned}\right\}_{\substack{i=2,\ldots,k-1\\ j\in\mathcal{I}_i}}. \end{aligned} \tag{A.9}$$

The dual problem can be rearranged and reduced to the standard form

$$\underset{\nu,\psi,\lambda,\tau,\mu}{\text{maximize}} \quad -\sum_{i=1}^{k-1}\nu_{i+1}^T b_i + \psi^T x - \epsilon||\Omega^{-1}\psi||_q + \sum_{i=2}^{k-1}\lambda_i^T(u_i l_i) \tag{A.10}$$

$$\text{s.t.} \quad \nu_k = c \tag{A.11}$$

$$W_1^T\nu_2 = -\psi \tag{A.12}$$

$$\nu_{i,j} = 0,\ j\in\mathcal{I}_i^- \tag{A.13}$$

$$\nu_{i,j} = (W_i^T\nu_{i+1})_j,\ j\in\mathcal{I}_i^+ \tag{A.14}$$

$$\left.\begin{aligned}&((u_{i,j}-l_{i,j})\lambda_{i,j}-\mu_{i,j}-\tau_{i,j}) = (W_i^T\nu_{i+1})_j\\ &\nu_{i,j} = u_{i,j}\lambda_{i,j}-\tau_{i,j}\end{aligned}\right\}_{\substack{i=2,\ldots,k-1\\ j\in\mathcal{I}_i}} \tag{A.15}$$

$$\lambda,\tau,\mu\geq 0. \tag{A.16}$$

The insight of the dual problem is that it can also be written in the form of a deep network. Consider the equality constraint (A.15), the dual variable λ corresponds to the upper bounds in the convex ReLU relaxation, while μ and τ correspond to the lower bounds $z \geq 0$ and $z \geq \hat{z}$, respectively. By the complementary property, these variables will be zero of ReLU constraint is non-tight, and non-zero if the ReLU constraint is tight. since the upper and lower bounds cannot be tight simultaneously, either λ or $\mu + \tau$ must be zero. Hence, at the optimal solution to the dual problem,

$$\begin{aligned} (u_{i,j} - l_{i,j})\lambda_{i,j} &= [(W_i^T \nu i + 1)_j]_+ \\ \tau_{i,j} + \mu_{i,j} &= [(W_i^T \nu i + 1)_j]_-. \end{aligned} \tag{A.17}$$

Combining this with the constraint $\nu_{i,j} = u_{i,j}\lambda_{i,j} - \tau_{i,j}$ leads to

$$\nu_{i,j} = \frac{u_{i,j}}{u_{i,j} - l_{i,j}}[(W_i^T \nu i + 1)_j]_+ - \eta[(W_i^T \nu i + 1)_j]_- \tag{A.18}$$

for $j \in \mathcal{I}_i$ and $0 \leq \eta \leq 1$. This is a leaky ReLU operation with a slope of $\frac{u_{i,j}}{u_{i,j} - l_{i,j}}$ in the positive portion and and a negative slope η between 0 and 1. Also note that from (A.12) $-\psi$ denotes the pre-activation variable for the first layer. For the sake of simplicity, we use $\hat{\nu}_i$ to denote the pre-activation variable for layer i, then the objective of the dual problem becomes

$$\begin{aligned} S_{D_\epsilon}(x, \nu) &= -\sum_{i=1}^{k-1} \nu_{i+1}^T b_i + \sum_{i=2}^{k-1} \sum_{j \in \mathcal{I}_i} \frac{u_{i,j} l_{i,j}}{u_{i,j} - l_{i,j}} [\hat{\nu}_{i,j}]_+ - \hat{\nu}_1^T x - \epsilon ||\Omega^{-1} \hat{\nu}_1||_q \\ &= -\sum_{i=1}^{k-1} \nu_{i+1}^T b_i + \sum_{i=2}^{k-1} \sum_{j \in \mathcal{I}_i} l_{i,j} [\hat{\nu}_{i,j}]_+ - \hat{\nu}_1^T x - \epsilon ||\Omega^{-1} \hat{\nu}_1||_q \end{aligned} \tag{A.19}$$

Hence, the final form of the dual problem can be rewritten as a network with objective $S_{D_\epsilon}(x, \nu)$, input $-c$ and activations $\mathcal{I}$ as follows:

$$\begin{aligned} \underset{\hat{\nu}, \nu}{\text{maximize}} \quad & -\sum_{i=1}^{k-1} \nu_{i+1}^T b_i + \sum_{i=2}^{k-1} \sum_{j \in \mathcal{I}_i} l_{i,j} [\hat{\nu}_{i,j}]_+ - \hat{\nu}_1^T x - \epsilon ||\Omega^{-1} \hat{\nu}_1||_q \\ \text{s.t.} \quad & \nu_k = -c \\ & \hat{\nu}_i = (W_i^T \nu_{i+1}), i = k-1, \ldots, 1 \\ & \nu_{i,j} = \begin{cases} 0 & j \in \mathcal{I}_i^- \\ \hat{\nu}_{i,j} & j \in \mathcal{I}_i^+ \\ \frac{u_{i,j}}{u_{i,j} - l_{i,j}} [\hat{\nu}_{i,j}]_+ - \eta [\hat{\nu}_{i,j}]_- & j \in \mathcal{I}_i \end{cases} \qquad i = k-1, \ldots, 2 \end{aligned} \tag{A.20}$$

A.3 Proof of Theorem 7.4

Let X and Y be random variables such that $X \sim \mathcal{N}(x, \Sigma)$ and $Y \sim \mathcal{N}(x + \delta, \Sigma)$. Next, we define the set $\mathcal{A} := \left\{ z \mid \delta^T \Sigma^{-1}(z - x) \leq \sqrt{\delta^T \Sigma^{-1} \delta}\, \Phi^{-1}_{r,d}(\underline{p_a}) \right\}$, where $r := \sqrt{\delta^T \Sigma^{-1} \delta}$ and $\Phi^{-1}_{r,n}(\underline{p_a})$ is the quantile function of the χ distribution of d degree of freedom for the probability p_a, so that $\mathbb{P}(X \in \mathcal{A}) = \underline{p_a}$. Consequently, $\mathbb{P}(Y \in \mathcal{A}) = \Phi_{r,d}\left(\Phi^{-1}_{r,d}(\underline{p_a}) - \sqrt{\delta^T \Sigma^{-1} \delta}\right)$. To ensure that Y is classified as class A, we need

$$\Phi_{r,d}\left(\Phi^{-1}_{r,d}(\underline{p_a}) - \sqrt{\delta^T \Sigma^{-1} \delta}\right) \geq 1/2 \tag{A.21}$$

which can be satisfied if and only if $\sqrt{\delta^T \Sigma^{-1} \delta} \leq \Phi^{-1}_{r,d}(\underline{p_a}) - q_{50}$.