When AI helps Wildlife Conservation: Learning Adversary Behaviors in Green Security Games

by

Debarun Kar

A Dissertation Presented to the
FACULTY OF THE GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(Computer Science)

June 2017

Acknowledgements

First, I would like to take this opportunity to thank my advisor, Professor Milind Tambe. I want to express my heartfelt gratitude to him for patiently guiding me during the course of my doctoral pursuit in the past four years. This dissertation would not have been possible without his invaluable guidance, prolific encouragement, constructive criticism and persistent help. I realized very early on how lucky I was to have chosen Milind as my advisor, and my respect for him has constantly grown over the years and still continues to do so. Having attended so many conferences, I have seen first hand the respect and admiration he commands and the resultant respect we as his students get wherever we go. Milind, your hard work and dedication never ceases to amaze me and I hope to emulate such an attitude and aptitude in both my future professional career and personal life. Furthermore, it gives me immense satisfaction and a sense of relief to know that I have a mentor for life. I am deeply humbled to be your 25th graduating PhD student.

Next, I would like to thank Nicole Sintov for being a wonderful mentor during the early years of my PhD, guiding me along with Milind through my first set of research papers. Your invaluable insights on human behavior, empirical research methods and experimental design have immensely benefited my research over the years. Without our frequent interactions, my research would have suffered dearly and so I'll always be grateful for your guidance. I would also like to thank all of my dissertation committee members for their time towards my research and development. My sincere gratitude to all of you: Jelena Mirkovic, Morteza Dehghani, Phebe Vayanos, Richard John and Yiling Chen. Having such an interdisciplinary ensemble of committee members helped shape my thesis in ways that I could not have imagined.

All of my projects at USC were possible due to collaborations with many amazing researchers and rangers at the Uganda Wildlife Authority, Wildlife Conservation Society, and World Wildlife Fund. Their relentless efforts to make the world a better and safer place enabled me to work with them in not only developing novel artificial intelligence models but more importantly testing them out in the field with exciting results. Specifically, I want to thank Andrew Plumptre from WCS and Arnaud Lyet from WWF for working with us and providing us valuable data and insights throughout the course of this work. This made my PhD all the more fun and valuable and so I thank them for giving me this opportunity.

During my time at USC I have also had the honor to work with many great researchers, either while collaborating on papers or while organizing workshops in conferences. In particular, I would like to thank: Fei Fang and Francesco Delle Fave for being amazing collaborators, without your help many of my papers would never have happened; Ben Ford and Shahrzad Gholami for collaborating with me on an exciting wildlife conservation paper; Subhasree Sengupta for helping me complete the final paper in my PhD; Arunesh Sinha for being an amazing friend, your tremendous mathematical expertise and for allowing me to work with you on the only theory paper that I will probably ever write; Jun-young Kwak for allowing me to work with you on the very first paper that I was a part of during my PhD; William Haskell for being the coolest mathematician I have ever seen, thank you for our joint work on fisheries protection; Ece Kamar and Eric Horvitz for being interested in my work and for all the valuable suggestions for our paper on belief modeling; Liz Bondi with whom I hope to significantly impact the field of wildlife conservation in the coming years through the use of drone technology; and, Eugene Vorobeychik, Haifeng Xu and Long Tran-Thanh for your willingness to work with me in conducting two successful adversarial reasoning workshops at AAMAS. I would also like to thank Thanh Nguyen, Matthew Brown, Yasaman Dehghani Abbasi, Pradeep Varakantham, Aram Galstyan and Bo An for working with me on some exciting problems. A big thank you to all the B.S. and M.S. students who have worked with me over the years and for allowing me to guide them: Brian Schwedock, Nikhil Cherukuri, Amit Plaha, Vivek Tiwari, Jie Zheng, Chiao-meng Huang, Soudhamini Radhakrishnan and Donnabell Dmello.

This brings me to the rest of the TEAMCORE family, which has been built through years of hard work, dedication and mutual respect by Milind and his students, many of whom I have had the privilege of spending my PhD career with, while others whom I have met elsewhere. All of you have been a constant source of friendship and support and so I thank you all: Chao Zhang, Yundi Qian, Leandro Marcolino, Aaron Schlenker, Sara Mc Carthy, Bryan Wilder, Elizabeth Bondi, Aida Rahmattalabi, Eric Shieh, Rong Yang, Albert Jiang, Manish Jain, James Pita, Chris Kiekintveld, Matthew Taylor, Janusz Marecki, Praveen Paruchuri, Paul Scerri, and Milind's very first PhD student Gal Kaminka. There is one member of the TEAMCORE family who I want to thank immensely. Amulya Yadav, I am very grateful to have found a friend like you 9000 miles away from home and to have shared my PhD life with you. You are no less than a brother to me. Thank you for the heartfelt discussions during all the good and bad times in my life, and thank you for the constant laughs, your infectious free-spirited attitude and for being a constant source of encouragement. Win more best paper awards buddy!

I would also like to thank a few other people who played an important role during my stay at USC. Thank you Pradipta Ghosh and Nasir Mohammed for rushing to the hospital so late in the night of my accident and for making sure that I reached home safely after my treatments. Thank you Sahil Garg for being a wonderful roommate and for all your help throughout my good and bad times, especially the tough one month when I was bed-ridden after my accident. Thanks also to all my friends at the USC Bengali organization for making me feel at home.

Since the time I came to the United States, I have shared a major part of my life with one person in particular. Thank you Poushali Banerjee for all your love and support and for lighting up my life everyday for these past four years. I want to express my sincerest appreciation to you for being there with me in this long and rocky journey, even during the lengthy periods of time when I retreated to working on my papers. Sharing my life with you has been a constant source of happiness and a learning experience at the same time, and for that I thank you.

Finally, I want to thank my parents, the most important people in my life, for their constant motivation and support in all of my endeavors. Baba and Ma, without your dedication towards my education and well-being throughout the years this research would not have been possible. Thank you for letting me be away from home for so many years now so that I can fulfill my dreams, and thank you for always being there in my good and bad times, no matter how infrequent my phone calls can get from time to time. Thank you immensely for your patience, kindness and support.

Contents

List Of Figures

List Of Tables

Abstract

Whereas previous real-world game-theoretic applications in security focused on protection of critical infrastructure in the absence of past attack data, more recent work has focused on data-driven security and sustainability applications for protecting the environment, including forests, fish and wildlife. One key challenge in such “Green Security Game” (GSG) domains is to model the adversary’s decision making process based on available attack data. This thesis, for the first time, explores the suitability of different adversary behavior modeling approaches in such domains that differ in the type and amount of historical data available.

The first contribution is to provide a detailed comparative study, based on actual human subject experiments, of competing adversary behavior models in domains where attack data is available in plenty (e.g., via a large number of sensors). This thesis demonstrates a new human behavior model, SHARP, which mitigates the limitations of previous models in three key ways. First, SHARP reasons based on successes or failures of the adversary’s past actions to model adversary adaptivity. Second, SHARP reasons about similarity between exposed and unexposed areas of the attack surface to handle the adversary’s lack of exposure to enough of the attack surface. Finally, SHARP integrates a non-linear probability weighting function to capture the adversary’s true weighting of probabilities.

The second contribution relates to domains requiring predictions over a large set of targets by learning from limited (and in some cases, noisy) data. One example dataset on which we demonstrate our approaches to handle such challenges is a real-world poaching dataset collected over a large geographical area at the Queen Elizabeth National Park in Uganda. This data is too sparse to construct a detailed model. The second contribution of this thesis delivers a surprising result by presenting an adversary behavior modeling system, INTERCEPT, which is based on an ensemble of decision trees (i) that effectively learns and predicts poacher attacks based on limited noisy attack data over a large set of targets, and (ii) has fast execution speed. This has led to a successful month-long test of INTERCEPT in the field, a first for adversary behavior modeling applications in the wildlife conservation domain.

Finally, for the my third contribution, we examine one common assumption in adversary behavior modeling that the adversary perfectly observes the defender's randomized protection strategy. However, in domains such as wildlife conservation, the adversary only observes a limited sequence of defender patrols and forms beliefs about the defender's strategy. In the absence of a comparative analysis and a principled study of the strengths and weaknesses of belief models, no informed decision could be made to incorporate belief models in adversary behavior models such as SHARP and INTERCEPT. This thesis provides the first-of-its-kind systematic comparison of existing and new proposed belief models and demonstrates based on human subjects experiments data that identifying heterogeneous belief update behavior is essential in making effective predictions. We also propose and evaluate customized models for settings that differ in the type of belief data available and quantify the value of having such historical data on the accuracy of belief prediction.

Chapter 1

Introduction

Security and sustainability remains a global concern over the years. Challenges in security and sustainability include protecting critical infrastructures such as airports, ports, and transportation networks, preventing smuggling of drugs, urban crimes, and intrusions in cyber systems, as well as protecting our environmental assets such as endangered wildlife from poaching and fisheries from illegal fishing. A unifying theme in all of these challenges is the strategic reasoning between the law enforcement agencies such as police officers and wildlife park rangers (interchangeably referred to as defenders), and the adversaries such as terrorists and poachers. The defenders only have limited resources, and therefore it is not possible to protect everything at all times; while at the same time the adversaries can conduct surveillance to observe the defenders' actions. Therefore, any deterministic allocation of resources can be potentially exploited by the adversaries, and so it is important for the defenders to randomize the allocation of security resources for more effective protection.

In the last decade, game theory has become a well-established paradigm for modeling complex resource allocation and scheduling problems in security and sustainability domains (Tambe, 2011; Gatti, 2008; Agmon, Kraus, & Kaminka, 2008; Basilico, Gatti, & Amigoni, 2009). More specifically, Stackelberg Security Games (denoted as SSGs) which model the interaction between two players, the defender and the adversary, has received significant attention. In an SSG, (Kiekintveld, Jain, Tsai, Pita, Ordonez, & Tambe, 2009) the defender plays the role of a leader by first allocating and scheduling her limited security resources to protect a set of targets from the adversary. The defender commits to a *mixed strategy*, which is a randomized allocation of her resources, specified by a probability distribution over all possible deterministic allocations; the attacker then acts as the follower by taking an action after observing the defender's mixed strategy. Models and algorithms have been proposed to efficiently compute the optimal strategy for the defender to address real-world challenges (Jain, 2013; Yin, 2013; Pita, 2012; Yang, 2014; Shieh, 2015; Brown, 2015). Decision-support systems based on SSG and the proposed algorithms have also been successfully deployed in several domains to assist security agencies (Pita, Jain, Marecki, Ordonez, Portway, Tambe, Western, Paruchuri, & Kraus, 2008; Tsai, Rathi, Kiekintveld, Ordonez, & Tambe, 2009; Shieh, An, Yang, Tambe, Baldwin, DiRenzo, Maule, & Meyer, 2012; Yin, Jiang, Johnson, Kiekintveld, Leyton-Brown, Sandholm, Tambe, & Sullivan, 2012).

A key challenge in such security and sustainability domains is the prediction of adversary behavior so as to optimize the allocation of defender resources against future behavior of the adversary. The first generation of security games research focused on developing optimal defense strategies in the absence of any data to learn the adversary's behavior (Tambe, 2011). However, more recent work has focused on domains involving repeated interactions between the defenders and the adversaries, thus generating significant amount of data. These domains include "Green Security Game" domains (GSG) (see Figure 1.1) (Fang, Stone, & Tambe, 2015), such as the security of wildlife (repeated interactions between rangers and poachers) (Yang, Ford, Tambe, & Lemieux, 2014), fisheries (repeated interactions between coast guard and illegal fishermen)

(a) Wildlife protection (b) Fishery protection (c) Forest protection

Figure 1.1: Different Green Security Game (GSG) domains.

(Haskell, Kar, Fang, Tambe, Cheung, & Denicola, 2014) and forests. These green security domains are fundamentally very different compared to the infrastructure security domain for several reasons. First, as noted above, frequent and repeated attacks are involved. For example, poachers walk in to a protected park or forest area at regular intervals and place snares on the ground to trap animals. Therefore, the frequent attacks generate valuable data about the adversary's behavior which is not possible in infrastructure security domains. The second difference is in the adversary's decision making process. In domains such as wildlife protection, since the adversaries attack at frequent intervals, it is impossible for them to conduct long-term surveillance and careful planning before each attack. Therefore, the adversaries fail to make perfectly rational choices and are boundedly rational in their decision making process.

While prior work has primarily focused on optimization of security resources, this thesis, for the first time, explores the suitability of different adversary behavior modeling approaches in GSG domains that not only differ in the type and amount of historical data available (e.g., plentiful vs sparse data) but also in the type of prediction challenge (e.g., fine-grained prediction on a small set of targets vs coarse prediction on a larger set of targets). The three major contributions towards effective learning and prediction of adversary behavior are presented below.

1.1 Fine-grained Adversary Modeling with Plentiful Attack Data

Our first major contribution is to model adversary behavior in domains which require detailed attack prediction based on plentiful attack data (Kar, Fang, Fave, Sintov, & Tambe, 2015b, 2015c; Kar, Fang, Fave, Sintov, Sinha, Galstyan, An, & Tambe, 2015a; Kar, Fang, Fave, Sintov, Tambe, & Lyet, 2016). Given domains where significant amount of attack data is collected at regular intervals, the problem of learning the adversary's bounded rationality (and computing an optimal defender strategy against the learned adversary behavior) can be modeled via the Green Security Games framework. GSGs correspond to a very general game setting (Fang et al., 2015), where the defender periodically deploys new patrol strategies (in "rounds" of the game) and the adversary responds to a convex combination of current and previous rounds' defender strategies. However, in this thesis, for all such settings where we have enough data to build a detailed adversary behavior model, we will focus on a particular version of the general GSG framework, where we assume that the adversary is responding to the defender strategy deployed in the current round only.

Unfortunately, despite the initial promise of bounded rationality models in such domains, existing models (Haskell et al., 2014; Yang et al., 2014) suffer from three key limitations which are extremely detrimental to defender performance. First, existing models reason about the adversary's future actions based on past actions taken but *not* the associated successes and failures. Our analysis reveals that the adversaries adapt based on past successes and failures. Hence, failing to consider an adaptive adversary leads to erroneous predictions about his[1] future behavior, and thus significantly flawed defender strategies.

[1] By convention in security games literature, the defender is referred to as "she" and the adversary as "he".

Second, existing approaches for learning bounded rationality models perform poorly in the initial rounds. Our analysis reveals that the issue is not just the lack of data in the initial rounds, but insufficient exposure of *attack surface* (Jajodia, Ghosh, Swarup, Wang, & Wang, 2011; Manadhata & Wing, 2011) in the initial rounds which prevents the defender from collecting sufficient information about adversary responses to various strategies and learn a reliable model. This issue of limited attack surface exposure leads to erroneous learned results as the learning is biased towards the limited available information and hence significant losses are incurred by the defender until enough of the *right kind of data* becomes available.

Finally, existing adversary behavior models in the literature have failed to include probability weighting functions (how humans "perceive" probabilities), even though it is well known that probability weighting curves for humans – e.g., in prospect theory (Tversky & Kahneman, 1992) – are typically nonlinear. In light of this, we show that the direct application of existing models in the literature, such as the Quantal Response (QR) model (Yang, Ordonez, & Tambe, 2012; McKelvey & Palfrey, 1995) and the Subjective Utility Quantal Response (SUQR) model (Nguyen, Yang, Azaria, Kraus, & Tambe, 2013), which assume a linear probability function, provide results that would be extremely detrimental to defender performance.

The first main contribution is a new model called SHARP (**S**tochastic **H**uman behavior model with **A**tt**R**activeness and **P**robability weighting), that mitigates these three limitations: (i) Modeling the adversary's adaptive decision making process, SHARP reasons based on success or failure of the adversary's past actions on exposed portions of the attack surface; (ii) Addressing limited exposure to significant portions of the attack surface in initial rounds, SHARP reasons about similarity between exposed and unexposed areas of the attack surface, and also incorporates a discounting parameter to mitigate adversary's lack of exposure to enough of the attack surface; (iii) Addressing shortcomings of existing models in learning the adversaries' weighting of probabilities, we incorporate a two parameter probability weighting function in existing human behavior models. .

The second main contribution is to provide evidence from the first "repeated-measures study" of competing adversary behavior models. In our study, a suite of well-established models and SHARP are compared in human subjects experiments on Amazon Mechanical Turk (AMT). We show that: (i) SHARP outperforms existing approaches consistently over all rounds, most notably in initial rounds. (ii) As discussed earlier, existing approaches perform poorly in initial rounds with some performing poorly throughout all rounds. (iii) Surprisingly, simpler models which were originally proposed for infrastructure security domains performed better than recent advances which are geared specifically towards addressing green security domains. Furthermore, we demonstrate the effectiveness of SHARP's modeling considerations and the robustness of our experimental results through comprehensive analysis on the collected human subjects data.

Since the data was collected through human subjects experiments on AMT, it was essential to validate the findings from the AMT experiments by conducting further experiments with domain experts. Therefore, we conducted one repeated-measures study for SHARP in the real world: with wildlife security experts from the provinces of Lampung and Riau, Sumatra, Indonesia. Participants were from the local government and from the following NGOs: Yayasan Badak Indonesia (YABI), World Wildlife Fund (WWF) and Wildlife Conservation Society (WCS). The results are consistent with the findings from our experiments on AMT.

As mentioned earlier, GSG domains which could benefit from the detailed predictions of SHARP include domains where plentiful attack data is collected at periodic intervals. This includes parks where, in addition to recording data collected through foot patrols, there has been significant amount of recent focus on using advanced technology (e.g., different types of sensors) to collect attack data. For example, wildlife protection agencies are flying Unmanned Aerial Vehicles (UAVs) at various wildlife conservation sites in Africa and Asia to collect significant amount of poaching data (Fieldstadt, 2015; Desikan, Karunakaran, & Gokulnath, 2013). Furthermore, deployment of technologies such as Synthetic Aperture Radars (SAR) and near infrared cameras which are predominantly used for wildlife habitat analysis (Cushman & Huettmann, 2010; Collen, Pettorelli, Baillie, & Durant, 2013) are becoming increasingly popular for wildlife, forests and fisheries conservation (Kim, 2013; Desikan et al., 2013; Meyer & Hinzb, 2009; Casbeer, Kingston, Beard, & McLain, 2006). Detailed models such as SHARP could be used effectively in such data-rich settings. In addition to GSGs, another domain where models such as SHARP could have significant impact is the urban crimes domain. In fact, a model similar to SHARP has been successfully applied in recent work on this domain (Zhang, Sinha, & Tambe, 2015; Zhang, Bucarey, Mukhopadhyay, Sinha, Qian, Vorobeychik, & Tambe, 2016; Zhang, Jiang, Short, Brantingham, & Tambe, 2014; Abbasi, Short, Sinha, Sintov, Zhang, & Tambe, 2015).

1.2 Coarse-grained Adversary Modeling with Sparse Attack Data

Unlike domains discussed in the previous section where the goal is to make fine-grained predictions with plentiful attack data, some GSG problem domains may require coarse-grained predictions based on datasets that pose a different set of challenges (e.g., limited and noisy attack data collected over a large number of targets). One example dataset on which we demonstrate our approaches to handle such challenges is a real-world poaching dataset collected over a large geographical area (approx. 2500 square kilometers) at the Queen Elizabeth National Park in Uganda. The attack data recorded in this dataset is too sparse and noisy to construct a detailed model.

To resolve this, we developed INTERCEPT (INTERpretable Classification Ensemble to Protect Threatened species) (Kar, Ford, Gholami, Fang, Plumptre, Tambe, Driciru, Wanyama, & Rwetsiba, 2017a), a new adversary behavior modeling application, which unlike SHARP, does *not* capture the temporal adaptiveness of adversaries; instead it learns and effectively predicts adversary behavior based on aggregate data from the past. The contributions here are as follows. First, given the limitations of traditional approaches in adversary behavior modeling for the QENP dataset, INTERCEPT takes a fundamentally different modeling approach, decision trees, and delivers a surprising result: although decision trees are simpler and do not take temporal correlations into account, they perform significantly better than more recent sophisticated adversary models similar to SHARP that consider temporal relationships, and other popular machine learning models (e.g., Logistic Regression, SVMs, and AdaBoost). Furthermore, decision trees satisfy the fundamental requirement of speedy execution; without which, relevant authorities would not test INTERCEPT in the field. However, decision trees do not take into account the spatial correlations present in this dataset, and so we introduce a spatially aware decision tree algorithm, BoostIT, that significantly improves prediction performance. To further augment INTERCEPT's performance, we developed an ensemble of the best classifiers which boosts predictive performance to a factor of 3.5 over existing sophisticated models.

Second, as a first for adversary behavior modeling applications applied to the wildlife crime domain, we present in this thesis the results of a *month long* real-world deployment of INTERCEPT: compared to historical observation rates of illegal activity, rangers that used INTERCEPT observed 10 times the number of findings than the average. In addition to many signs of trespassing, rangers found a poached elephant, a roll of elephant snares, and a cache of 10 antelope snares before they were deployed. While the rangers' finding of a poached elephant carcass is a grim reminder that poachers are active, each confiscated snare represents an animal's life saved–this demonstrates the effectiveness of INTERCEPT in problem domains requiring coarse-grained predictions on a large number of targets with sparse attack data.

1.3 Adversary Modeling with Belief Data

One common assumption while developing adversary behavior models in security games is that the adversaries have access to the actual mixed strategy of the defender while optimizing their own attack strategies (Tambe, 2011; Fang et al., 2015; Nguyen, Delle Fave, Kar, Lakshminarayanan, Yadav, Tambe, Agmon, Plumptre, Driciru, Wanyama, et al., 2015; Kar et al., 2015b; Yang et al., 2014; Fang, Nguyen, Pickles, Lam, Clements, An, Singh, Tambe, & Lemieux, 2016; Nguyen, Sinha, Gholami, Plumptre, Joppa, Tambe, Driciru, Wanyama, Rwetsiba, Critchlow, et al., 2016; Haskell et al., 2014; Yang, Kiekintveld, Ordonez, Tambe, & John, 2011; Nguyen et al., 2013). This assumption holds to some extent in domains such as counter-terrorism where the adversary conducts careful surveillance of the defender's deployed pure strategies over a long period of time (Southers, 2011). However, the above assumption of careful surveillance does not always hold in domains such as wildlife poaching. In such domains, the adversary attacks frequently based on limited observations of the instantiations of the defender's randomized strategy. Therefore, a key challenge in these settings is the modeling of adversary's belief formation about the defender's mixed strategy based on limited observations.

Several models have been proposed, both in the SSG literature (An, Kempe, Kiekintveld, Shieh, Singh, Tambe, & Vorobeychik, 2012; Pita, Jain, Ordonez, Tambe, Kraus, & Magori-Cohen, 2009; Yin, Jain, Tambe, & Ordonez, 2011; Nguyen, Yadav, An, Tambe, & Boutilier, 2014) as well as in psychology (See, Fox, & Rottenstreich, 2006) that address this problem in different ways. While (An et al., 2012) proposed a Bayesian belief update model assuming perfectly rational adversaries, (Pita et al., 2009) proposed a linear mixture model of belief formation assuming boundedly rational adversaries. (Yin et al., 2011) and (Nguyen et al., 2014) model the observational uncertainty of the adversary in terms of an interval uncertainty around the actual mixed strategy. However, there are certain issues with the existing literature on belief modeling which we address in this thesis (Kar, Sengupta, Kamar, Horvitz, & Tambe, 2017b).

First, the literature lacks empirical evaluation or a head-to-head comparison of existing belief formation models. Indeed, in the absence of a comprehensive analysis and a principled study of the strengths and weaknesses of belief models, it is unclear as to which model(s) are better suited for estimating adversary beliefs and should be included in the adversary behavior prediction models such as SHARP and INTERCEPT. To address this shortcoming, we conduct the first-of-its-kind systematic comparison of existing and our proposed models of adversary belief formation and update. We developed a game to simulate a GSG scenario where each participant (acting as the adversary) observes the defender's pure strategies sampled from a chosen mixed strategy for multiple days and is required to enter their beliefs about the defender's actual mixed strategy after each observation. Extensive analysis with 24 different models on data collected through this game deployment on Amazon Mechanical Turk (AMT) highlights key insights about the human belief update process and demonstrates the strengths and weaknesses of these models.

Second, existing belief update models assume the presence of a homogeneous population of adversaries with the same belief update mechanism (An et al., 2012; Pita et al., 2009). However, our analysis shows the presence of four heterogeneous groups of adversaries with distinct belief update processes. In this thesis, we present a new model called $B\text{-}REACT$ (Belief model for heteRogenEous Adversaries using ClusTering) that addresses this shortcoming by learning about the adversary based on historical belief update data, combined with a clustering based approach. We demonstrate that this new model completely outperforms existing and other proposed models, thus emphasizing the importance of modeling heterogeneity in human belief formation.

Third, existing work in the literature simply assume that no historical data about adversary beliefs will be available. Therefore, the literature lacks models that can take advantage of varying amounts of historical data (when available) so that the data can be used to learn about the adversary's belief update process and for making more accurate belief predictions in the future. we address this shortcoming by customizing our models for settings that differ in the type of data available for belief modeling and quantify the value of having population-wide or historical data on the accuracy of belief prediction.

The goal of this contribution in the thesis, i.e., presenting a comprehensive study of belief formation models applicable to the GSG domains, highlighting their strengths and shortcomings, and introducing new computational belief models that address the shortcomings, is to identify the belief formation and update model(s) that are best suited for estimating adversary beliefs and could therefore be included in the adversary behavior prediction models for improved prediction.

1.4 Thesis Overview

The structure of the thesis is organized as follows: Chapter 2 discusses background material for Stackelberg security games and adversary behavior models. Chapter 3 reviews related work to provide the context for the contributions of the thesis. Chapter 4 discusses the wildlife poaching game used to collect data for the first contribution of adaptive adversary modeling in data-rich scenarios. Chapter 5 and 6 presents our contributions for SHARP, the adaptive adversary model with probability weighting. Chapter 7 presents detailed experimental results with human subjects data collected using the game in Chapter 4. Chapter 8 and 9 analyzes the real-world poaching dataset from QENP, discusses the models proposed to handle such datasets and demonstrates the superior performance from deployments of our proposed model INTERCEPT at QENP. Chapter 10 and 11 explores the problem of modeling the adversary's belief formation and updating procedure, and discusses the effectiveness of the proposed approaches. Finally, chapter 12 summarizes the contributions of this thesis.

Chapter 2

Background

In this section, we introduce Stackelberg Security Games (SSG), key solution concepts related to SSGs, and existing behavioral models used to model boundedly rational adversaries in SSGs.

2.1 Stackelberg Security Games

Stackelberg games were first introduced to model leadership and commitment (von Stackelberg, 1934). A Stackelberg game is a game played sequentially between two players: the first player is the leader who commits to a strategy first, and then the second player, called the follower, observes the strategy of the leader and then commits to his own strategy. The term Stackelberg Security Games (SSG) was first introduced by (Kiekintveld et al., 2009) to describe specializations of a particular type of Stackelberg game for security as discussed below.

In an SSG, the defender plays the role of a leader who protects a set of targets from the adversary who acts as the follower (Kiekintveld et al., 2009). The defender's *pure strategy* is an assignment of a limited number of security resources M to the set of targets T. An assignment of a resource to a target is also referred to as covering a target. A defender's mixed-strategy $\hat{x}$ $(0 \leq \hat{x}_j \leq 1; \forall \hat{x}_j, j \in P; \sum_{j=1}^{P} \hat{x}_j = 1)$ is then defined as a probability distribution over the set of all possible pure strategies P. An equivalent description (Korzhyk, Conitzer, & Parr, 2010; Yang et al., 2011) of these mixed strategies is a probability distribution over the set of targets: x $(0 \leq x_i \leq 1; \forall x_i, i \in T; \sum_{i=1}^{T} x_i = M)$. In the rest of this thesis, we will refer to this latter description as the mixed strategy of the defender.

A pure strategy of an adversary is defined as attacking a single target. The adversary receives a reward R_i^a for selecting i if it is not covered and a penalty P_i^a for selecting i if it is covered. Similarly, the defender receives a reward R_i^d for covering i if it is selected by the adversary and penalty P_i^d for not covering i if it is selected. Then, the expected utility for the defender (while playing mixed strategy x) when target i is selected by the adversary to attack is:

$$U_i^d(x) = x_i R_i^d + (1 - x_i) P_i^d \tag{2.1}$$

Similarly, the expected utility for the adversary for attacking target i is:

$$U_i^a(x) = (1 - x_i) R_i^a + x_i P_i^a \tag{2.2}$$

Although a perfectly rational adversary would choose to attack the target with the highest expected utility, more recent work has focused on modeling boundedly rational adversaries in SSGs (Nguyen et al., 2013; Yang et al., 2014; Haskell et al., 2014; Yang et al., 2011; Ford, Nguyen, Tambe, Sintov, & Fave, 2015; Cui & John, 2014), some of which are discussed in Section 2.2.

Repeated Stackelberg Security Games: As mentioned earlier in Section 1.1, given non-noisy and plentiful attack data, and the repeated nature of some domains (e.g., wildlife conservation), the problem of learning the adversary's bounded rationality and computing an optimal defender strategy against the learned adversary behavior in such domains can be modeled via the Green Security Games framework. GSGs correspond to a very general game setting (Fang et al., 2015), where the defender periodically deploys new patrol strategies (in "rounds" of the game) and the adversary responds to a convex combination of current and previous rounds' defender strategies. In this thesis, for settings where we have enough data to model our setting as a GSG, we will focus on a particular version of the general GSG model where we assume that the adversary is responding to the defender strategy deployed in the current round only. We will call this a repeated Stackelberg Security Game setting for simplicity.

Note that this repeated SSG setting is different from the traditional repeated game setting (Osborne & Rubinstein, 1994) in the following ways. First, in a repeated SSG, in one round, one player acts first by deploying a mixed strategy and then the other player responds. Intuitively, one round in a repeated SSG corresponds to several ($>> 1$) consecutive rounds in a repeated game. Second, in a repeated SSG, the mixed strategy of the defender may change at the end of one round leading to a new mixed strategy, while such a concept of a change of mixed strategy is not part of a traditional repeated game (Osborne & Rubinstein, 1994). In other words, just as a Stackelberg Security Game focuses on commitment to a mixed strategy in a round, rather than commitment to a pure strategy as done in earlier literature on Stackelberg games (Bagwell, 1992), repeated SSG focuses on mixed strategies in each round.

2.2 Human Behavior Models

This section discusses popular human behavior models proposed in the SSG literature.

2.2.1 Subjective Utility Quantal Response (SUQR)

SUQR (Nguyen et al., 2013) builds upon prior work on quantal response (McFadden, 1976) according to which rather than strictly maximizing utility, an adversary stochastically chooses to attack targets, i.e., the adversary attacks a target with higher expected utility with a higher probability. SUQR proposes a new utility function called Subjective Utility, which is a linear combination of key features that are considered to be the most important in each adversary decision-making step. This is based on the Lens model in psychology which is a framework for modeling prediction based on observable cues (Brunswik, 1952; Hammond, 1955). Usually these observable cues are combined in a weighted fashion to get the utility of the decision maker. Nguyen et al. (Nguyen et al., 2013) experimented with three features: defender's coverage probability, adversary's reward and penalty at each target. Thus, according to this model, the probability that the adversary will attack target i is given by:

$$q_i(\omega|x) = \frac{e^{SU_i^a(x)}}{\sum\limits_{j \in T} e^{SU_j^a(x)}} \tag{2.3}$$

where $SU_i^a(x)$ is the Subjective Utility of an adversary for attacking target i when defender employs strategy x and is given by:

$$SU_i^a(x) = \omega_1 x_i + \omega_2 R_i^a + \omega_3 P_i^a \tag{2.4}$$

The vector $\omega = (\omega_1, \omega_2, \omega_3)$ encodes information about the adversary's behavior and each component of ω indicates the relative importance the adversary gives to each attribute in the decision making process. The weights are computed by performing Maximum Likelihood Estimation (MLE) on available attack data.

2.2.2 Bayesian SUQR

SUQR assumes that there is a homogeneous population of adversaries, i.e., a single ω is used to represent an adversary in (Nguyen et al., 2013). However, in the real-world we face an entire population of heterogeneous adversaries. So, (Yang et al., 2014) introduces a set $\Omega \subset \mathbb{R}^3$ to represent the range of all possible ω, i.e. the entire set of adversaries. Therefore Bayesian SUQR is proposed to learn a particular value of ω for each attack. It assumes that there is a prior distribution F over Ω. Bayesian updates are performed on F as more data becomes available. Then the following stochastic optimization problem is solved to obtain the optimal strategy x:

$$\max_{x \in \mathbb{X}} \int_{\Omega} \left[\sum_{t \in T} U_t^d \left(x\right) q_t \left(\omega \,|\, x\right) \right] F \left(d\omega\right) \tag{2.5}$$

Protection Assistant for Wildlife Security (PAWS) is an application which was originally created using Bayesian SUQR. Recent work by (Fang et al., 2015) has also used this notion of a heterogeneous population of boundedly rational adversaries and applied Bayesian updating based algorithms to learn models of these adversaries.

2.2.3 Robust SUQR

Robust SUQR (Haskell et al., 2014) combines data-driven learning and robust optimization to address settings where not enough data is available to provide a reasonable hypothesis about the distribution of ω. It does not require a specific distribution F over the adversary population parameters. Given an uncertainty set $\widehat{\Omega}$, Robust SUQR solves the following robust optimization problem:

$$\max_{x \in \mathbb{X}} \min_{\omega \in \widehat{\Omega}} \sum_{t \in \mathbb{T}} U_t \left(x \right) q_t \left(\omega \,|\, x \right), \tag{2.6}$$

We now explain Eqn. 2.6 starting with the uncertainty set $\widehat{\Omega}$. There are various ways to construct the uncertainty set $\widehat{\Omega}$. Haskell et al (Haskell et al., 2014) suggests combining the robust optimization in Eqn. 2.6 with a data-driven approach by using the set of all ω learned from each attack by the adversary as the uncertainty set $\widehat{\Omega}$. Therefore, Robust SUQR computes the worst-case expected utility over all previously seen SUQR models of the adversary and deploys the optimal strategy against the adversary type that reduces the defender's utility the most. Robust SUQR has been applied to fisheries protection domain (Haskell et al., 2014).

Chapter 3

Related Work

We have already discussed related work in SSGs and GSGs in the previous section, including key behavioral models. Here we discuss additional areas of related work for each of our contributions.

3.1 Research in repeated games

In this section we discuss past work on repeated games which are relevant to our setting.

3.1.1 Learning in repeated Stackelberg games

The problem of learning the adversary's payoffs to alleviate uncertainty in an SSG by launching a minimum number of games against a perfectly rational adversary is studied in (Letchford, Conitzer, & Munagala, 2009; Blum, Haghtalab, & Procaccia, 2014). (Letchford et al., 2009) propose an approach to learn a single attacker's payoffs by making a number of best-response queries which is polynomial in the number of pure strategies. A query here refers to the defender's execution of a mixed strategy, and letting an adversary respond, thereby providing information about adversary's payoffs. This work was the first in the security game context for learning adversary payoffs. They extend their results to Bayesian Stackelberg games with a known distribution over attacker types by running the single-attacker learning algorithm, where they repeat each best response query until the response of the desired attacker type is observed.

Noticing that (Letchford et al., 2009) may still lead to a large number of queries, particularly given that number of pure strategies may grow exponentially, (Blum et al., 2014) design an algorithm that learns an ϵ-optimal strategy for the defender with a certain probability by asking a significantly lower number of queries. However, (Blum et al., 2014) only study the interaction between the defender and a single attacker.

Building upon previous work (Blum et al., 2014; Letchford et al., 2009) as described above, (Balcan, Blum, Haghtalab, & Procaccia, 2015) provides two contributions in terms of learning the randomized defender strategy to commit to in each round against perfectly rational adversaries: (i) an online learning algorithm where the defender observes the adversary type that is attacking a particular target (full-information); and (ii) an online learning algorithm where the defender only observes a particular target being attacked in each round (partial information). In each interaction, the attacker is assumed to be adversarially chosen from a set of known attacker types.

Additionally, (Marecki, Tesauro, & Segal, 2012) focused on optimizing the defender's overall utility during the learning process when faced with a perfectly rational adversary with unknown payoffs. Their analysis is focused on the repeated interaction between the defender and a single attacker type drawn initially from a distribution. Although their algorithm is shown to converge in the long-term, they do not provide any guarantees for the convergence of their algorithm.

3.1.2 Robust strategies in repeated games

In cases where the opponent cannot be successfully modeled, (McCracken & Bowling, 2004) proposed techniques to generate ϵ-safe strategies which bound the loss from a safe value by

ϵ. (Johanson, Zinkevich, & Bowling, 2007; Johanson & Bowling, 2009) studied the problem of generating robust strategies in a repeated zero-sum game while exploiting the tendency in the adversary's decision making and evaluated their technique in a game of two-player, Limit Texas Hold'em. Following up on this work, (Johanson & Bowling, 2009) proposed methods to minimize losses due to limited data while also exploiting an unknown opponent's weaknesses and evaluated their technique in a game of two-player, Limit Texas Hold'em. Recently, (Ponsen, Jong, & Lanctot, 2011) proposed techniques to compute robust best responses in partially observable stochastic games using sampling methods.

All of the above work in learning adversary behavior differs from ours in three ways: (i) They do not model bounded rationality in human behavior; (ii) They do not consider how humans weigh probabilities; and (iii) None of these existing work address the important problem of significant initial round losses when the problem can be modeled as a repeated SSG in plentiful data settings. Initial round losses is a critical problem in domains such as wildlife security as explained earlier; requiring a fundamental shift at least in the learning paradigm for SSGs. Work on learning in SSGs differ because in our game, the payoffs are known but we are faced with boundedly rational adversaries whose parameters in their behavioral model are to be learned.

3.1.3 Learning from reinforcements in repeated games

Skinner (Skinner, 1938, 1948, 1953) first proposed the theory of operant conditioning in which he explained through experiments that behavior which is reinforced tends to be repeated (i.e. strengthened) and behavior which is not reinforced tends to die out-or be extinguished (i.e. weakened). This reinforcement or lack thereof, happens due to actions and its associated consequences.

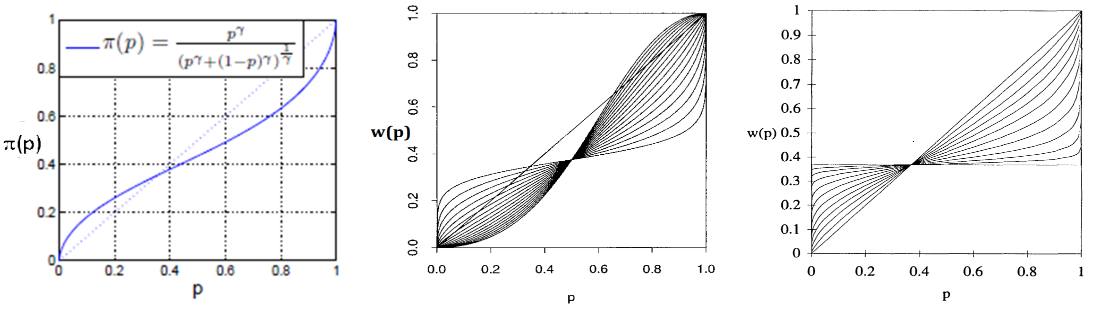
More specifically, Skinner identified three types of responses or operants that can alter behavior, the most relevant among them are: (i) Reinforcers: Responses from the environment that increase the probability of a behavior being repeated; and (ii) Punishers: Responses from the environment that decrease the likelihood of a behavior being repeated. Since this early research, such behavior where the subject learns based on "superstitious" beliefs due to past actions and consequences has come to be known as superstitious learning (Devenport, 1979; Zollo, 2009) in psychology. The influence of superstitious learning on the adaptive behavior of humans has also been studied in the literature (Beck & Forstmeier, 2007). We will show later how superstitious learning, induced by these reinforced/punished responses plays a crucial role in predicting human subject behavior in repeated Stackelberg security games. More specifically, SHARP models this phenomenon and that leads to a significant improvement in SHARP's performance.

"Superstitious Learning", as it is widely known in the psychology literature, is closely related to "Reinforcement Learning" in the computer science literature. Reinforcement learning has been widely studied in the context of game theory (Erev & Roth, 1998; Feltovich, 2000; Hopkins, 2001; Beggs, 2005; Cominetti, Melo, & Sorin, 2010; Ceren, Doshi, Meisel, Goodie, & Hall, 2013). Erev and Roth (Erev & Roth, 1998) describes one such popular Reinforcement Learning (RL) model that predicts people's behaviors while playing repeated, simultaneous move games. However, the paper has major differences as compared to our game setting. First, the models in Erev and Roth's paper were developed for simultaneous move games without any notion of prior commitment to a mixed strategy by any player. This is different from our leader-follower setting where one player moves first by playing a mixed strategy and the other player moves next by playing a pure strategy after having observed the first player's mixed strategy. Therefore, the notion of surveillance and reacting to a mixed strategy by the adversary is missing in (Erev & Roth, 1998). Second, in our games, the defender responds after each round by playing an optimal mixed strategy based on the learned adversary model from past rounds' data. Assuming an adversary who follows an RL model as described in (Erev & Roth, 1998), the optimal defender response would then be to play a pure strategy in each round given the fixed adversary strategy in each round. This will have significantly detrimental effects in terms of the defender utilities obtained in each round, as shown in Section 7.1. Therefore, while not directly applicable, we attempt to translate the main concept of the RL model to our setting of leader-followers with mixed strategies. To that end, in Section 6.5, we have adapted the basic RL model in (Erev & Roth, 1998) to compute optimal mixed strategies for the defender in repeated SSG settings. In Chapter 7, we also show results of conducting human subjects experiments with this model. Specifically, the RL based approach performs poorly as compared to other models in the experiments. Therefore, significant new work would need to be done to understand how the RL models in the literature (Erev & Roth, 1998; Feltovich, 2000; Hopkins, 2001; Beggs, 2005; Cominetti et al., 2010) could be adapted more efficiently for SSGs.

3.2 Repeated Measures Studies

Repeated measures studies are conducted to measure a set of variables over a period of time. Repeated measures studies are usually conducted in psychology, political sciences and social sciences and the duration of the experiments can span from a few weeks (Silver, Holman, McIntosh, Poulin, & Gil-Rivas, 2002) to even a few years (Caravolas, Hulme, & Snowling, 2001) [1] . Recently, AMT has become a more favorable choice of conducting these experiments due to the

[1] Whereas "repeated measures study" is often used to describe research that spans years – in which measurement occasions are conducted every X years – we use the term repeated measures study because our study included multiple (5) measurement points with a single population.



(a) Prospect Theory (b) Gonzalez & Wu, 99 (c) Prelec, 98

Figure 3.1: Probability Weighting Functions

ease of collecting data from a huge and diverse subject pool (Parkes, Mao, Chen, Gajos, Procaccia, & Zhang, 2012; Berinsky, Huber, & Lenz, 2012). One of the most important problems in conducting repeated measures studies is handling participant attrition (i.e., people dropping out). Researchers often use imputation and sampling techniques to fill missing data due to participant attrition (Twisk & de Vente, 2002; Goldstein, 2009; Deng, Hillygus, Reiter, Si, & Zheng, 2013). However, for our repeated measures study of comparing human behavior models in data-rich scenarios this may result in extremely biased estimates of the modeling parameters due to the influence of the retained participants' game plays and therefore may generate biased defender strategies.

3.3 Probability Weighting Functions

Probability weighting functions model human perceptions of probability. Perhaps the most notable is the weighting function in Tversky and Kahneman's Nobel Prize-winning work on Prospect Theory (Kahneman & Tversky, 1979; Tversky & Kahneman, 1992), which suggests that people weigh probability non-uniformly, The empirical form of the probability weighting function $\pi(p_i)$, where p_i is the actual probability, from (Kahneman & Tversky, 1979) is shown in Fig. 3.1(a). It indicates that people tend to overweight low probabilities and underweight high probabilities. The diagonal straight line in the figure indicates the linear weighting of probability. However, other works in this domain propose and experiment with parametric models which capture both inverse S-shaped as well as S-shaped probability curves (Abdellaoui, L'Haridon, & Zank, 2010; Gonzalez & Wu, 1999) (Fig. 3.1(b)). We build on this research in the first contribution for data-rich scenarios, incorporating probability weighting functions in SHARP that allow for both S-shaped and inverse S-shaped curves; however, in our work, data supports S-shaped probability curves. Further discussions about this function are in Chapter 5.

There are other popular probability weighting functions in the literature, such as Prelec's one-parameter model (Prelec, 1998), where the weighted probability is

$$w(p) = exp(-(-\ln p)^{\alpha}); 0 < \alpha < 1 \tag{3.1}$$

Although this model has been shown to perform well in the literature, the functional form does not allow for an S-shaped curve to be learned given the allowed range of parameter values– it is only capable of learning an inverse S-shaped curve as shown in Fig. 3.1(c) when $0 < \alpha < 1$. This parameter range of α is due to the necessity that the function satisfies certain properties such as sub-proportionality and compound invariance, which will get violated if $\alpha > 1$. However, it can account for S-shaped curves if we allow $\alpha > 1$. Later in Section 7.2.1, We allow α to be greater than 1 so as to allow learning both an S-shaped as well as an inverse S-shaped curve with this function– our results show that an S-shaped curve is learned on the data. In other words, no matter whether we use Prelec's function or Gonzalez and Wu's function, if we allow for learning both S-shaped as well as inverse S-shaped curves, the data fits an S-shaped probability weighting curve.

We conduct further analysis to show in Section 7.2.1 that, even though both generate S-shaped curves on our data, using the probability weighting function by Gonzalez and Wu (Gonzalez & Wu, 1999) in our model SHARP gives us better prediction accuracy as compared to the case when we use Prelec's function, thus justifying our choice of the probability weighting function in Chapter 5.

3.4 Related research in belief modeling

Earlier work on belief modeling can be broadly classified into two types based on the assumption about the amount of information available. First is the case when no prior data is available to learn about the belief formation and update process of human agents in a given situation. This is what has been used in SSGs. Second is the scenario when historical belief update data for a group of human agents is available (training set), however the training set does not contain data about human agents represented in the test set. This facilitates learning a generalized model of human belief formation and update, and apply the learned model to predict belief updates for a previously unknown set of human agents (testing dataset). This consists of models in the psychology literature and we will discuss one such popular model that could be applied to SSGs.

3.4.1 Setting without training data

In the absence of any training data, previous work on modeling adversary beliefs in SSGs has focused on two aspects depending on the assumptions about the rationality of the adversary: (i) Bayesian belief update models typically associated with a perfectly rational adversary (An et al., 2012) (henceforth referred to as "perfectly rational adversary' models'); and (ii) heuristic belief update associated with boundedly rational adversary (Pita et al., 2009; Pita, Jain, Tambe, Ordonez, & Kraus, 2010). These are described below.

Perfectly Rational Adversary: (An et al., 2012) proposed a Stackelberg Game with Limited Observation (SGLS) model where a perfectly rational adversary updates his beliefs about the defender's actual mixed strategy x given his prior beliefs and τ observations, where each observation is one of the defender's pure strategies j from the set $\mathcal{P}$. They represent the sequence of observations compactly in terms of an observation vector $\mathcal{O}^r =< o_j^r >$ in which o_j^r is the number of times pure strategy j is observed till day r. They assumed that the adversary's belief distribution over the set of all pure strategies can be represented as Dirichlet distributions (Kotz, Balakrishnan, & Johnson, 2000) characterized by a parameter vector $\alpha =< \alpha_1, ..., \alpha_{|\mathcal{P}|} >$. They assumed uniform Dirichlet distribution as prior. Then they use Bayesian updates to compute the posterior belief distribution over pure strategies based on the observation vector $\mathcal{O}^r$. For example, assuming $\alpha_k = 0; \forall k = 1 to |\mathcal{P}|$ before day 1, and then after 5 days (one observation per day) we have observed pure strategy $j \in \mathcal{P}$ three times, then $o_j^5 = 3$ and the posterior $\alpha_j + o_j^5 = 0+3 = 3$ at the end of day 5. The adversary's belief b_i^r about the marginal coverage of target i after the r^{th} observation could then be computed from the posterior belief distribution over pure strategies as in Eqn. 3.2. In our experimental setting, as we will see in Chapter 10, with 9 targets and 3 defenders, $|\mathcal{P}|$ = $\binom{9}{3}$ = 84, and τ=10 (total number of days of observations). j_i is either 1 (or 0) depending on whether target i is protected in pure strategy j (or not).

$$b_i^r = \frac{\sum\limits_{j \in \mathcal{P}} j_i(\alpha_j + o_j^r + 1)}{\sum\limits_{j \in \mathcal{P}} \alpha_j + |\mathcal{P}| + \tau} \tag{3.2}$$

We refer to this model as ${}^{N}B_u$, where u denotes uniform prior and N indicates that this model correspond to the case of an uninformed adversary, i.e., the adversary has *no* information about the number and types of strategies employed by the defender.

Boundedly Rational Adversary: (Pita et al., 2009, 2010) proposed a linear mixture model to account for the belief update of boundedly rational adversaries. They model the adversary's beliefs b based on a weighted linear combination of two components: a prior belief ρ and the actual mixed strategy x. For any target i, this is shown in Eqn. 3.3. They assume the prior (also called the ignorance prior) to be an uniform distribution of the number of defenders over the given set of targets. They further assume a fixed weight ($\mu \in [0, 1]$) on the prior for their experimental setting and do not provide any justification for their choice of the fixed weight. So, given 9 targets and 3 defenders, ρ_i at any target i is $\frac{300}{9} \approx 33$. If x_i at some target i is 50 and μ is 0.60, then the adversary's belief b_i (in percentage) about the defender's coverage at target i is $0.60 * 33 + (1 - 0.60) * 50 \approx 40$. We will refer to this model as ${}_{0.6}M_u^A$, where M denotes mixture models, A represents actual mixed strategy and 0.6 is the fixed weight on the prior.

$$b_i = \mu * \rho_i + (1 - \mu) * x_i \tag{3.3}$$

Interval Uncertainty models: (Yin et al., 2011) and (Nguyen et al., 2014) proposed models that handle observation uncertainty in SSGs by allowing an uncertainty interval around the actual mixed strategy. To capture that insight, we implemented a model as follows: (1) Randomly choose the total number of targets k (out of 9) on which to apply an uncertainty perturbation; (2) Pick k out of the given 9 targets randomly; (3) Pick a very small random number δ independently for each target. (4) Compute beliefs by changing the actual mixed strategy at each of the k

targets by the randomly generated δ for the corresponding target, with type of change (increase or decrease of probability) also chosen randomly; (5) Compute error between model and participant beliefs. This process is repeated 100 times and the results are averaged to remove any selection bias. We experimented with $0 \leq \delta \leq 5$ and $0 \leq \delta \leq 10$ (δ in percentage scale) and only show results for the first case (best of the two) in the paper. We will refer to this model as IU, where IU stands for interval uncertainty. In Chapter 11, we will see experimental results with this model, as well as the other belief models presented above.

3.4.2 Setting with training data

Availability of training data can aid the defender in *learning* a belief model of the adversaries and use that to predict future belief updates for the same or a previously unseen group of adversaries. One such popular belief model in psychology is a non-linear mixture model called the log-odds model (See et al., 2006) shown in Eqn. 3.4. This model assumes that the adversary is boundedly rational and computes the log of odds metric between an event F (in our setting it is the adversary's belief b_i that a target i is covered by the defender) and the alternate event A (the adversary's belief that a target is *not* covered by the defender, i.e., (1-b_i)).

$$ln\frac{b_i}{1-b_i} = a_1 + a_2 * ln\frac{n_F^i}{n_A^i} + a_3 * ln\frac{f^i(F)}{f^i(A)} \tag{3.4}$$

Here, n_F^i (n_A^i) represents the adversary's prior belief about the number of ways target i is protected (or not). Similarly, $f^i(F)$ and $f^i(A)$ represent the influence of the actual observations on the beliefs formed by the adversaries. $ln\frac{n_F^i}{n_A^i}$ and $ln\frac{f^i(F)}{f^i(A)}$ denote the influence of the adversary's prior beliefs and actual observations respectively on his future beliefs. They considered the adversary's prior belief distribution to be uniform. Parameters a_1, a_2 and a_3 are learned by performing linear regression with given training data. We will refer to this model as ${}_{learn}log_u$. We experimented with this model and a variant which we propose. The details along with the results are presented in Chapters 10 and 11 respectively.

Chapter 4

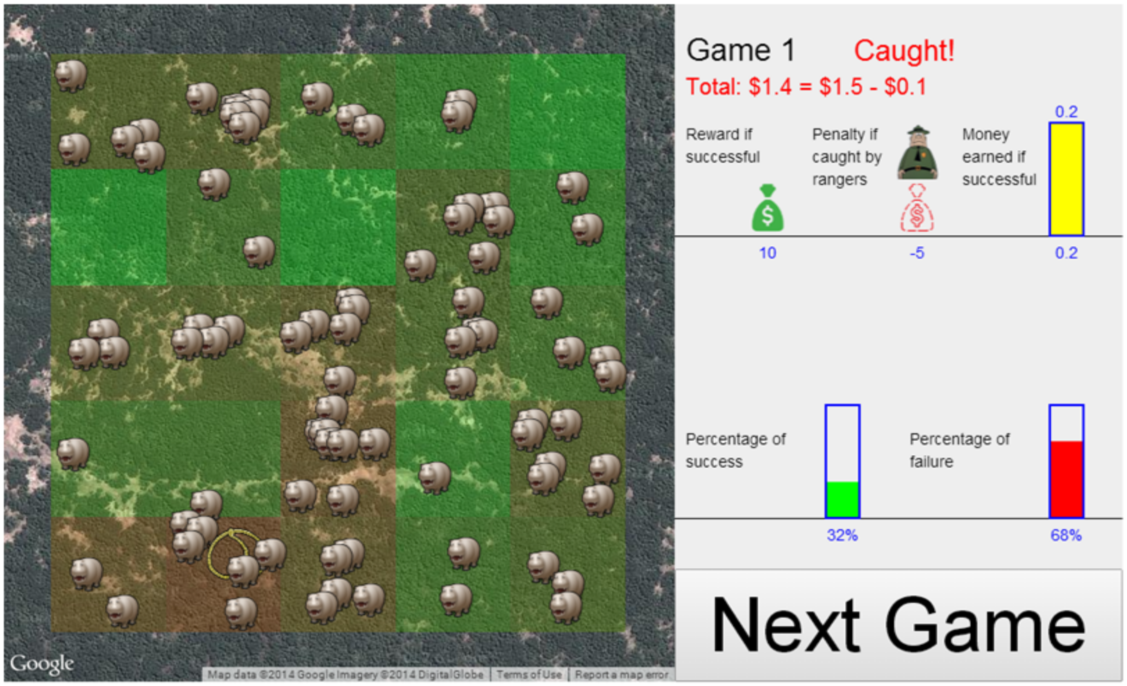
Wildlife Poaching Game

This chapter discusses the process of collecting human subjects data through *repeated measures experiments*, thus providing experimental details about the first contribution of this thesis of providing a benchmark for comparing existing and proposed human behavior models in repeated SSGs. In order to conduct these experiments, we developed an online simulated game. Below is an overview of our experimental game and its properties.

4.1 Game Overview

In our game, human subjects play the role of poachers looking to place a snare to hunt hippopotamus in a protected park[1]. The game interface is shown in Fig. 10.1. In the game, the portion of the park shown in the map is divided into a 5*5 grid, i.e. 25 distinct cells. Overlaid on the Google Maps view of the park is a heat-map, which represents the rangers' mixed strategy x — a cell i with higher coverage probability x_i is shown more in red, while a cell with lower coverage probability is shown more in green. As the subjects play the game, they are given the following detailed

[1] One might argue that since the wildlife poaching game requires participants to place snares with the goal of poaching animals, responses from human subjects and hence the results may be biased due to their moral dilemma. See Section 13.7 for more detailed discussions about the similarity of results between this game and an alternate game scenario designed with the same goal of evaluating human behavior models



Game 1 Caught!

Total: $1.4 = $1.5 - $0.1

Reward if successful
Penalty if caught by rangers
Money earned if successful
0.2
10
-5
0.2
Percentage of success
Percentage of failure
32%
68%
Next Game
Google
Map data ©2014 Google Imagery ©2014 DigitalGlobe | Terms of Use | Report a map error

Figure 4.1: Game Interface for our simulated online repeated SSG (Reward, penalty and coverage probability for a selected cell are shown)

information: R_i^a, P_i^a and x_i for each target i. However, they do not know the exact location of the rangers, i.e., they do not know the pure strategy that will be played by the rangers, which is drawn randomly from the mixed strategy x shown on the game interface. Thus, we model the real-world situation whereby poachers have knowledge of past patterns (mixed strategies) of ranger deployment but not the exact location of ranger patrols when they set out to lay snares.

In our game, there were $M = 9$ rangers protecting this park, with each ranger protecting one grid cell. Therefore, at any point in time, only 9 out of the 25 distinct regions in the park are protected. The players know before they play (place a snare) that only 9 out of the 25 regions will be protected, but as mentioned earlier, they do not know which 9 beforehand. In other words, in a particular round, a player can only know about the presence or absence of a ranger at the location he attacks only after he attacks. A player succeeds if he places a snare in a region which is not protected by a ranger and hence captures a hippo, else he is unsuccessful.

We make two modeling choices in this game. First, we focus on situations where there is no collusion or coordination between poachers in placing of snares as is true in most cases in a large forest area; accordingly, a player in our game can only see his/her snare, but not that of other players and thus cannot coordinate with other players. Second, we assume in our experiments that the defender is able to observe all the attacks conducted by the poachers and hence learn the adversaries' preferences from the complete attack data set. However, later in this article we also provide analysis of defender strategies generated when this assumption is violated, i.e., when the defender can only observe fractions of the entire attack dataset.

4.1.1 Computation of Poacher Reward

A key factor in this game is determination of rewards for poachers and rangers. For poachers animal density is a key factor determining their rewards. In addition to animal density, which is strongly correlated with high-risk areas of poaching (Moreto, 2013; Montesh, 2013; Hamisi, 2008), distance is another important factor in poaching, e.g., recent snare-density studies have found that significant poaching happens within 5 kilometers of South Africa's Kruger National Park border (Lemieux, 2014) and significantly decreases more than 4 kilometers away from the international border in Ksavo West National Park in Kenya (Wato, Wahungu, & Okello, 2006). Therefore, the reward obtained by a poacher in successfully placing a snare at target i is calculated by discounting the animal density by a factor of the distance traveled and is calculated as follows:

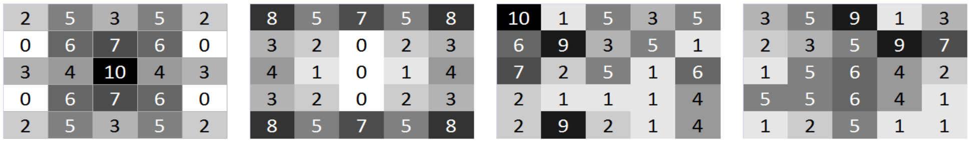
$$R_i^a = int(\phi_i - \zeta * \frac{D_i}{\underset{j}{max}(D_j)}) \tag{4.1}$$

Here, ϕ_i and D_i refer to, respectively, the animal density at target i and the distance to target i from the poacher's starting location. For simplicity, we consider the adversary's reward as an integer. So, $int(y)$ rounds the value y to the closest integer value. The parameter ζ is the importance given to the distance factor in the reward computation and may vary based on the domain. Intuitively, the reward for successfully placing a snare in a region i near the starting location and which has animal density ϕ_i, is higher than the reward obtained by successfully placing a snare in a region with the same animal density but which is at a greater distance from the starting location as compared to i. We used $\zeta = 2$ in our experiments because the use of $\zeta = 1$ did not introduce substantial impact of distance while computing the actual rewards and $\zeta = 3$ was not used to prevent the overwhelming impact distance had on the actual rewards computed.

4.1.2 Non-zero sum game

In our games, the minimum and maximum animal density at each cell were 0 and 10 units respectively. The poacher received a flat penalty of -1 if he was caught at any target. We vary the adversary's actual reward based on the amount of distance traveled because he has to carry the captured animal back to the edge of the forest. However, there is no burden of carrying the animal back when the poacher is caught by the ranger (or equivalently his snare is confiscated), and therefore, in our games we do not vary the penalty based on distance and assume a constant value of -1. Also in our game, when the poacher successfully poaches, he may obtain a reward that is less than the animal density (Eqn. 4.1), but the defender loses a value equal to that of the animal density, i.e., the game is non-zero-sum[2].

[2] Note that in terms of real-world interpretation of the payoff to the adversary in this game, it is to be interpreted as taking into account the probability of catching a hippo. Therefore, higher density leads to a higher payoff. That is, it is the expected reward (in the absence of the defender) in attacking a particular cell — the expected number of hippos captured without the defender.



2	5	3	5	2
0	6	7	6	0
3	4	10	4	3
0	6	7	6	0
2	5	3	5	2

(a) ADS_1

8	5	7	5	8
3	2	0	2	3
4	1	0	1	4
3	2	0	2	3
8	5	7	5	8

(b) ADS_2

10	1	5	3	5
6	9	3	5	1
7	2	5	1	6
2	1	1	1	4
2	9	2	1	4

(c) ADS_3

3	5	9	1	3
2	3	5	9	7
1	5	6	4	2
5	5	6	4	1
1	2	5	1	1

(d) ADS_4

Figure 4.2: Animal density structures (ADS)

4.2 Payoff Structures

The payoff structures used in our human subject experiments vary in terms of the animal densities and hence the adversary rewards. We henceforth refer to payoff structures and animal density structures interchangeably in this article. The total number of animals in all the payoffs we generate is the same (= 96). However, the variation in these payoffs is in the way the animals are spread out in the park. In payoff structure 1 (i.e., Animal Density structure 1 or ADS_1), the animal density is concentrated towards the center of the park, whereas the animal density is higher towards the edges of the park in payoff structure 2. These represent scenarios that might happen in the real world. The animal density for both payoffs is symmetric, thus eliminating any bias due to the participant's starting point in the game.

Contrary to the above, animals in the park may be randomly scattered without any particular orientation. So, we randomly generate two additional animal density structures (payoffs 3 and 4) and test our proposed model on these payoffs. To generate a random structure, one out of 25 cells was chosen uniformly at random and then an animal was allocated to it until the total number of animals in all the cells was 96, keeping in mind that if a cell total reached 10 (maximum animal density in a cell), that cell was not chosen again. Figs. 4.2(a)– 4.2(d) show heatmaps of four animal density structures, denoted as ADS_1, ADS_2, ADS_3 and ADS_4 respectively.

4.3 Online Repeated Measures Experiments

Repeated measures studies are research studies which are typically conducted to observe and understand the changes in and effects of a particular set of variables over a period of time (Menard, 2008; Farrington, Loeber, & Welsh, 2010; Heiman, 2002). In our work the key variable is the adversary's strategy and we show that the adversary's strategy does indeed change over time due to his adaptive nature (as explained later in Section 6) and hence we model such behavior with a novel model called SHARP. Such studies can be conducted with a subject pool at a University lab or by recruiting participants in an online setting like AMT. We conducted our experiments on AMT. We tested a variation (Chapter 5) of the set of behavioral models introduced in Chapter 2 and our new model SHARP by deploying the mixed strategy generated based on each of these models repeatedly over a set of five rounds (Kar et al., 2015b). We observed the strategies employed by the participants in each round, i.e., where they attacked and whether they succeeded or failed, and used that to determine the optimal ranger strategy for the next round. For each model, we recruited a new set of participants to eliminate any learning bias.

We took necessary steps to ensure that participants completely remember the game details and the procedures to play the game during each round of the experiment, as otherwise we may lose significant time and effort in collecting poor quality data, especially because each setting would take more than two weeks to be completed. This was done by setting up proper validation and trial games in each round of the experiment, while not over-burdening the participants with many games and thus keeping their cognitive overload at a minimum. This is discussed next in Sec. 4.3.1, followed by a discussion of participant retention rates in our study in Sec. 4.3.2.

4.3.1 Validation and Trial Games

After viewing the instructions at the beginning, the participants were first asked to play two trial games in round 1, with an option to view the instructions again after each game. After the trial games, they played one validation game, and finally the actual game. The players could choose to stop playing at any point during this process. The validation game consisted of a cell with maximum animal density (=10) and the coverage probability of that cell was zero, while other cells had an animal density of 1 and non-zero but equal coverage probability. The participants were expected to select the target with the maximum animal density and zero coverage. Data from subjects who played the validation game incorrectly were discarded and they were not allowed to participate in future rounds of the experiment.

From second round onwards, participants were only asked to play one trial game and then the actual game. The trial game was kept in order to remind them of the game and its details and the playing procedures. Showing only the actual game without any trial games might have resulted in the participants not playing the game properly due to forgetfulness about the game details.

4.3.2 Participant Retention Rate

For our repeated measures experiments, due to unavailability of data, the strategy shown for each first round of the real game to the participants who passed the validation game was Maximin. We then learned the model parameters based on previous rounds' data, recomputed and redeployed strategies, and asked the $same$ players to play again in the subsequent rounds. For each model, all five rounds were deployed over a span of weeks. When we started conducting the experiments, we observed that there were very high attrition rates (i.e. people dropped out) for the number of participants between rounds of the game. we observed that a delayed compensation scheme

Table 4.1: Experiment Details

Average time taken per model per payoff structure (all 5 rounds)	Average time taken for a set of participants to play each round	Number of participants who played round 1 (minimum, maximum)	Average number of participants who played all 5 rounds	Average retention rate from round 2 to round 5
2.3 weeks	4 days	(42 , 49)	38	83.69%

along with prior participant commitment and repeated reminders throughout the course of the experiment helped in achieving a high average retention rate of 83.69%. This is also shown in Table 4.1 along with other experiment details. For interested readers, a detailed discussion of the set of challenges that faced during our experiments and our methodological contributions towards mitigating those challenges are presented in the appendix.

Chapter 5

SHARP: Probability Weighting

This thesis contributes a novel human behavior model called SHARP for data-rich repeated SSG settings. SHARP has three key novelties: (i) SHARP reasons based on success or failure of the adversary's past actions on exposed portions of the attack surface to model adversary adaptivity; (ii) SHARP reasons about similarity between exposed and unexposed areas of the attack surface, and also incorporates a discounting parameter to mitigate adversary's lack of exposure to enough of the attack surface; and (iii) SHARP integrates a non-linear probability weighting function to capture the adversary's true weighting of probability. In this chapter, we cover the probability weighting aspect of SHARP and also discuss possible causes for the surprising results of incorporating probability weighting in our models. Other aspects are covered in Chapter 6.

5.1 Probability Weighting Mechanism

The need for probability weighting became apparent after our initial experiments. In particular, initially following up on the approach used in previous work on adversary behavior modeling (Nguyen et al., 2013; Yang, Kiekintveld, Ordonez, Tambe, & John, 2013; Yang et al., 2014; Haskell et al., 2014) discussed in Chapter 2, we applied MLE to learn the weights of the SUQR model based on data collected from our human subject experiments using the game discussed in Chapter 4. We found that the weights on coverage probability were positive for all the experiments. That is, counter-intuitively, humans were modeled as being attracted to cells with high coverage probability, even though they were *not* attacking targets with very high coverage but they were going after targets with moderate to very low coverage probability. Examples of the learned weights for SUQR from data collected from the first round deployment of the game for 48 human subjects on ADS_1 and ADS_2 are: (ω_1, ω_2, ω_3)=(2.876, -0.186, 0.3) and (ω_1, ω_2, ω_3)=(1.435, -0.21, 0.3). Here w_1 provides the SUQR code on coverage probability.

We prove a theorem (Theorem 1) to show that, when the weight on the coverage probability in the SUQR model (ω_1) is found to be positive, the optimal defender strategy is a pure strategy. The proof of the theorem can be found in Appendix 2.

Theorem 1. *When $\omega_1 > 0$, the optimal defender strategy is a pure strategy.*

Employing a pure strategy means that there will be no uncertainty about the defender's presence. Several cells will always be left unprotected and in those cells, the attackers will always succeed. In our example domains, even if the top-valued cells are covered by a pure strategy, we can show that such a strategy would lead to significantly worse defender expected utility than what results from the simplest of our defender mixed strategies deployed. For example, if cells of value 4 are left unprotected, the defender expected value will be -4, which is much lower than what we achieve even with a simple strategy like Maximin. In repeated SSG domains like wildlife crime, this would mean that the poachers successfully kill animals in each round without any uncertainty of capture by rangers. In order to show that playing a pure strategy does indeed lead to poor defender utility, we conducted an experiment with human subjects by deploying a SUQR based pure strategy on ADS_1. Results and comparisons with other models that are introduced later in the paper are shown in Section 7.1.

We hypothesize that this counter-intuitive result of a model with $\omega_1 > 0$ may be because the SUQR model may not be considering people's *actual* weighting of probability. SUQR assumes that people weigh probabilities of events in a linear fashion, while existing work on probability weighting (Section 3.3) suggest otherwise. To address this issue, we augment the Subjective Utility function (Eqn. 2.4) with a two-parameter probability weighting function (Eqn. 5.1) proposed by Gonzalez and Wu (Gonzalez & Wu, 1999), that can be either inverse S-shaped (concave near probability zero and convex near probability one) or S-shaped.

$$f(p) = \frac{\delta p^{\gamma}}{\delta p^{\gamma} + (1-p)^{\gamma}} \tag{5.1}$$

The SU of an adversary denoted by 'a' can then be computed as:

$$SU_i^a(x) = \omega_1 f(x_i) + \omega_2 R_i^a + \omega_3 P_i^a \tag{5.2}$$

where $f(x_i)$ for coverage probability x_i is computed as per Eqn. 5.1. The two parameters δ and γ control the elevation and curvature of the function respectively. $\gamma < 1$ results in an inverse S-shaped curve while $\gamma > 1$ results in an S-shaped curve. We will henceforth refer to this as the PSU (Probability weighted Subjective Utility) function and the models (SUQR, Bayesian SUQR and Robust SUQR) augmented with PSU will be referred to as P-SUQR, P-BSUQR and P-RSUQR respectively. *Our SHARP model will also use PSU.* We will use these PSU-based models in our experiments. Although we have already shown in Theorem 1 that models without probability weighting may result in pure defender strategies being generated for subsequent rounds and would thus perform poorly in repeated SSG experiments, for verification pruposes we still deployed and compared models without probability weighting (for example, SUQR) against PSU-based models (for example, P-SUQR). Results are shown in Chapter 7.

One of our key findings, based on experiments with the PSU function is that the curve representing human weights for probability is *S-shaped in nature, and not inverse S-shaped* as prospect theory suggests. The S-shaped curve indicates that people would overweight high probabilities and underweight low to medium probabilities. Some learned curves will be shown in Section 7.2. Recent studies in economics (Alarie & Dionne, 2001; Humphrey & Verschoor, 2004; Etchart-Vincent, 2009) have also found S-shaped probability curves which contradict the inverse S-shaped observation of prospect theory. In addition, other recent work on security games, specifically Opportunistic Crime Security Games, has also found the existence of S-shaped probability weighting curves (Abbasi et al., 2015)[1]. Furthermore, in previous literature (Kahneman & Tversky, 1979; Tversky & Kahneman, 1992) where they experimented with insurance and lotteries, they dealt with smaller number of alternatives (2 or 3 alternatives). In addition to the domain, one possible reason for observing S-shaped curves in our games could be that the participants are shown larger number of alternatives, i.e. they have to choose one from a set of 25 targets. To the best of our knowledge, participants' weighting of probabilities in such games with larger number of alternatives has not been studied before.

[1] Note that although we did not consider the value function from Prospect Theory in our experiments, (Abbasi et al., 2015) showed in her experiments that considering both the value function and probability weighting function still results in the same S-shaped probability weighting curve.

Given S-shaped probability weighting functions, the learned ω_1 was negative as it accurately captured the trend that a significantly higher number of people were attacking targets with low to medium coverage probabilities and *not* attacking high coverage targets.

Feature Selection and Weight Learning: In Section 13.7, we introduced a new feature – distance – that affected the reward and hence the obvious question for us was to investigate the effect of this new feature in predicting adversary behavior. We considered several variations of PSU with different combinations of features. Notice that each combination of the features could be used in each of our models, like P-SUQR, P-BSUQR, etc. In addition to Eqn. 5.2, three more are discussed below (Eqns. 5.3,5.4,5.5). Recall that ϕ_i denotes the animal density at target i. Now, although it is true that the subjects were explicitly told the values of rewards and penalties at each cell and not the animal densities, the animal densities were still visually observable. ϕ has been used in Eqn. 5.3 to check if the participants may have been considering animal densities only and ignoring the effect of distance while playing the game, thus not paying attention to the effective reward. ϕ was used in Eqn. 5.5 to check if participants may have been considering animal density and distance as two separate features and weighting them in a linear fashion instead of the way we provided them the reward values. These models are designed and compared against each other to verify possible ways in which participants may actually have considered the features of the game while making decisions.

$$SU_i^a(x) = \omega_1 f(x_i) + \omega_2 \phi_i + \omega_3 P_i^a \tag{5.3}$$

$$SU_i^a(x) = \omega_1 f(x_i) + \omega_2 R_i^a + \omega_3 P_i^a + \omega_4 D_i \tag{5.4}$$

$$SU_i^a(x) = \omega_1 f(x_i) + \omega_2 \phi_i + \omega_3 P_i^a + \omega_4 D_i \tag{5.5}$$

To compare these variations, we need to learn the behavioral parameters for each of the variations (e.g, for Eqn. 5.5, a 6-tuple $b =< \delta, \gamma, \omega_1, \omega_2, \omega_3, \omega_4 >$ is to be learned; δ and γ due to inclusion of Eqn. 5.1) from available data and evaluate their effectiveness in predicting the attack probability. To learn the behavioral parameters b from available data, we propose an algorithm based on Repeated Random Sub-sampling Validation (Algorithm 2 – see Appendix 1). For P-SUQR , we learn a single b, while for P-BSUQR and P-RSUQR we learn a set of $b \in \mathbb{B}$ for each attack. Note that, for our probability weighting function, we use all possible combinations of δ and γ, with values of each ranging from 0 to 4, at an interval of 0.1. Therefore, our analysis also contains $\delta = 1$ and $\gamma = 1$, which correspond to linear weighting of probabilities — the probability weights used in SUQR.

To test the performance of Algorithm 2 against a non-linear solver (Microsoft Excel's Generalized Reduced Gradient (GRG) nonlinear solver function) and also to compare between models with various feature sets, we learned the weights of the four behavioral models (Eqn. 5.2 to 5.5) using both Algorithm 2 and our non-linear solver. In order to do this, we deployed P-SUQR on ADS_1 and then collected participants' responses to the deployed strategy. Then, we performed the following steps, which conform to standard practices in machine learning for splitting data into training-validation-test sets (Hastie, Tibshirani, & Friedman, 2009; Bishop, 2007):

1. We divided the first round data for the experiment with P-SUQR on ADS_1 into 10 random train-test splits.

2. For each of the 10 training sets, we performed 10-fold cross-validation to obtain the best model weights that give the lowest validation error on the corresponding validation sets.

That is, each of the 10 training sets was randomly partitioned into 10 equal sized subsamples. Of the 10 subsamples for each training set, a single subsample was retained as the validation data for validating the model, and the remaining 9 subsamples were used as training data. The cross-validation process was then repeated 10 times (the 10 folds), with each of the 10 subsamples used exactly once as the validation data, and the model weights that gave the lowest validation error (out of the 10 validation errors) was chosen after the cross validation process. Since we did this for each of the 10 training splits, we obtained 10 best learned model weights after applying 10-fold cross validation on each of the training splits.

3. Each of these 10 best learned model weights was then tested on the corresponding hold-out test data set by computing the sum of squared errors (SE) of predicting the attack probability over all the targets.

4. Finally, we computed the average of these SE values over the 10 test data sets. we computed this average SE for each of the four behavioral models using the best model weights learned by Algorithm 2 and the non-linear solver.

We report these average SE values for both the weight learning approaches on all the four behavioral models in Table 5.1. Results in **boldface** indicate significant differences (with two-tailed t-tests at confidence=0.05) in the performance of Eqn. 5.5 as compared to all other feature combinations. We can see that Eqn. 5.5 achieves the lowest average SE as compared to all other feature combinations.

Our approach helps to significantly improve the robustness of our results. Note that, we used 10-fold cross-validation in our approach *not* to compute the average error over all the validation sets and using it to compare against other models (other feature combinations in our case), but instead to select the best parameters for a particular model, and using those model parameters to test on independent test data sets. We do this multiple times and report the average test set error, thus making the process of comparison between different behavioral models more robust. That is, instead of applying 10-fold cross validation once on one random train split of the original dataset, we performed 10-fold cross-validation on 10 separate training data sets randomly constructed from the original dataset. Cross-validation is in itself a well established model validation technique in machine learning and statistics to assess the generalizability of learned models on independent test data sets. The effectiveness of this approach to derive an accurate estimation of model prediction performance is well established in the machine learning literature (Kohavi, 1995; Seni & Elder, 2010; Hastie et al., 2009; Bishop, 2007). Our approach of not just performing 10-fold cross validation once to select the best model weights, but multiple (10) times and then taking an average of the test set errors of the best learned model weights is also similar to what is traditionally adopted in machine learning literature (Dietterich, 1998) to improve robustness.

Our results show that we can achieve higher accuracy in modeling by generalizing the subjective utility form used in (Nguyen et al., 2013) that relied on just three parameters, by adding more features as shown in Eqn. 5.5. This opens the door to novel subjective utility functions for different domains that exploit different domain features. Results accompanied by * imply significant differences in performance of Algorithm 2 as compared to the non-linear solver. Thus, Algorithm 2 is more efficient in learning model weights as compared to the GRG non-linear solver.

We also present in Tables 5.2 and 5.3 the mean and standard deviations of the weights learned on the 10 training datasets for the best model (Eqn. 5.5) on both the payoff structures ADS_1 and ADS_2 and for both the learning algorithms (Algorithm 2 and Non-linear Solver). Based on our

Table 5.1: Performance (Squared Errors) of various feature sets. Results accompanied by * imply significant differences (with two-tailed t-tests at confidence=0.05) in performance of Algorithm 2 as compared to the non-linear solver. Results in **boldface** indicate significant differences in the performance of a particular feature combination with respect to other feature combinations.

	Eqn. 10	Eqn. 11	Eqn. 12	Eqn. 13
P-SUQR ADS_1 Algorithm 2	0.1965*	0.2031*	0.1985	**0.1025***
P-SUQR ADS_1 Non-linear Solver	0.2545	0.2589	0.2362	**0.1865**
P-SUQR ADS_2 Algorithm 2	0.2065*	0.2156*	0.2625	**0.1136***
P-SUQR ADS_2 Non-linear Solver	0.2546	0.2935	0.3062	**0.1945**

Table 5.2: Mean of the weights learned for the 10 training sets for the model in Eqn. 5.5 and all algorithm and payoff combinations in Table 5.1

	Eqn. 13
P-SUQR ADS_1 Algorithm 2	$< 2.36, 2.78, -2.3, 0.688, -0.3, -0.286 >$
P-SUQR ADS_1 Non-linear Solver	$< 3.88, 2.86, -4.3, 0.38, -0.3, -0.4 >$
P-SUQR ADS_2 Algorithm 2	$< 2.62, 2.92, -1.57, 0.38, -0.3, -0.34 >$
P-SUQR ADS_2 Non-linear Solver	$< 3.92, 3.02, -4.3, 0.34, -0.3, -0.28 >$

detailed experiments, in addition to $\omega_1 < 0$, we found that $\omega_2 > 0$, $\omega_3 < 0$ and $\omega_4 < 0$. The rest of the formulations in this article will be based on these observations about the feature weights.

5.2 Discussions:

Section 5.1 provides an S-shaped probability weighting curve (learned curves are shown in Figs. 7.6(a) and 7.6(b) Section 7.2) as one explanation of the human players' behavior data. Given the surprising nature of this result, it is important to discuss other possible hypotheses that may explain why those human behaviors may have been observed. This section shows however that evidence does not support these alternatives to S-shaped probability weighting curve discussed earlier.

One potential hypothesis is that the participants may have misinterpreted aspects of the game interface design shown in Figure 10.1. However, we took several steps to guard against such

Table 5.3: Standard Deviation of the weights learned for the 10 training sets for the model in Eqn. 5.5 and all algorithm and payoff combinations in Table 5.1

	Eqn. 13
P-SUQR ADS_1 Algorithm 2	$< 0.279, 0.4, 0.9, 0.41, 0, 0.18 >$
P-SUQR ADS_1 Non-linear Solver	$< 0.139, 0.32, 0.88, 0.09, 0, 0.07 >$
P-SUQR ADS_2 Algorithm 2	$< 0.19, 0.39, 0.69, 0.37, 0, 0.12 >$
P-SUQR ADS_2 Non-linear Solver	$< 0.1, 0.34, 0.88, 0.02, 0, 0.08 >$

misinterpretations: (i) We asked the participants to play two trial games and one validation game in the first round and one trial game in each subsequent round; and (ii) We explained key facets of the game in the instructions and the participants could switch to the instructions after playing each of the trial and validation games to verify their understanding before they played the actual game. In addition to ensuring that the participants were given clear instructions and provided enough practice through trial games, we also checked the results of the validation game and it showed that 860 out of 1000 participants passed the validation game — indicating an understanding of the game. Note that we then discarded data from 140 out of 1000 participants (an average of 7 participants per group) who played the validation game incorrectly.

Another hypothesis could be that the validation game had introduced some misinterpretations. Specifically, in our validation game the participants had to choose between an option which is good on two scales (highest animal density of 10 and zero coverage) and other options which are bad on both scales (lowest animal density of 1 and non-zero but equal coverage of 0.375). Therefore, this could potentially have caused the participants to incorrectly interpret the scales in the actual games they played and hence they may have misinterpreted the coverage probabilities in the actual games. However, there is little support for this hypothesis as well. Note that the validation game is one of three games being played by each participant before the actual game in the first round. Also, the validation game is only played once in the first round and never played again in future rounds. However, the participants played two trial games in the first round and one trial game in the future rounds before playing the actual game in each round, and these trial games do not have the same "two scales" property as the validation game as discussed earlier.

Another possible hypothesis for such an S-shaped curve for the probability weighting function could potentially be that we use the weighted probabilities as a separate additive feature in our model — P-SUQR implies that we take a weighted sum of the different model features. This is contrary to how the probability weighting function is used in the prospect theory literature (Kahneman & Tversky, 1979; Tversky & Kahneman, 1992). In that literature, the weighted probabilities are used to weight the values of outcomes; could that perhaps explain the S-shaped curve in our results? Unfortunately, evidence does not support this hypothesis as well. First, note that there have been existing works in the literature that show learning of S-shaped probability weighting curves even when conforming to the traditional prospect theoretic model, i.e., when the prospect theoretic values of outcomes are weighted by transformed probabilities (Abbasi et al., 2015; Leclerc, 2014). Thus, there already exists evidence of S-shaped probability curves in other domains even for the traditional prospect theoretic function. Furthermore, to verify the shape of the probability weighting curve in our game setting when we consider values of outcomes to be weighted by the transformed probabilities, we explored an alternate form of our P-SUQR model, called PWV-SUQR (Probability Weighted Values SUQR). In PWV-SUQR, the rewards and penalties are weighted by the transformed coverage probabilities, as shown in Eqn. 5.6. In Section 7.2.2, we show that even while learning adversary behavior using Eqn. 5.6, we get S-shaped probability curves. This result indicates that the learned S-shape of the probability curves is not merely the outcome of the additive nature of our P-SUQR model.

$$SU_i^a(x) = \omega_1(1 - f(x_i))R_i^a + \omega_2 f(x_i) P_i^a \tag{5.6}$$

Chapter 6

SHARP: Adaptive Utility Model

A second major innovation in SHARP is the adaptive nature of the adversary and addressing the issue of attack surface exposure. First, we define key concepts, present evidence from our experiments, and then present SHARP's innovations.

Definition 1. *The* **attack surface** α *is defined as the n-dimensional space of the features used to model adversary behavior. Formally,* $\alpha =< F^1, F^2, ..., F^n >$ *for features* $F^j (\forall j; 1 \leq j \leq n)$.

For example, as per the PSU model in Eqn. 5.5, this would mean the space represented by the following four features: coverage probability, animal density, adversary penalty and distance from the starting location.

Definition 2. *A* **target profile** $\beta_k \in \alpha$ *is defined as a point on the attack surface* α *and can be associated with a target. Formally,* $\beta_k =< F_k^1, F_k^2, ..., F_k^n >$ *denotes the* k*th target profile on the attack surface.*

In our example domain, the kth target profile can be represented as $\beta_k =< x_{\beta_k}, \phi_{\beta_k}, P^a_{\beta_k}, D_{\beta_k} >$, where x_{β_k}, ϕ_{β_k}, $P^a_{\beta_k}$ and D_{β_k} denote values for coverage probability, animal density, attacker penalty and distance from starting location respectively[1]. For

[1] In our experiments, $\phi_{\beta_i} > 0$, $P^a_{\beta_i} < 0$ and $D_{\beta_i} > 0$

example, $< 0.25, 2, -1, 4 >$ is the target profile associated with the top-leftmost cell of ADS_1 in round 1. We can add more features like terrain information, vegetation, etc. if available. Exposing the adversary to a lot of different target profiles would therefore mean exposing the adversary to more of the attack surface and gathering valuable information about their behavior. While a particular target location, defined as a distinct cell in the 2-d space, can only be associated with one target profile in a particular round, more than one target may be associated with the same target profile in the same round. β_k^i denotes that target profile β_k is associated with target i in a particular round.

6.1 Observations and Evidence

Below is an observation from our human subjects data, based on the above concepts, that reveal interesting trends in attacker behavior in repeated SSGs.

Observation 1. *Consider two sets of adversaries: (i) those who have succeeded in attacking a target associated with a particular target profile in one round; and (ii) those who have failed in attacking a target associated with a particular target profile in the same round. In the subsequent round, the first set of adversaries are significantly more likely than the second set of adversaries to attack a target with a target profile which is 'similar' to the one they attacked in the earlier round.*

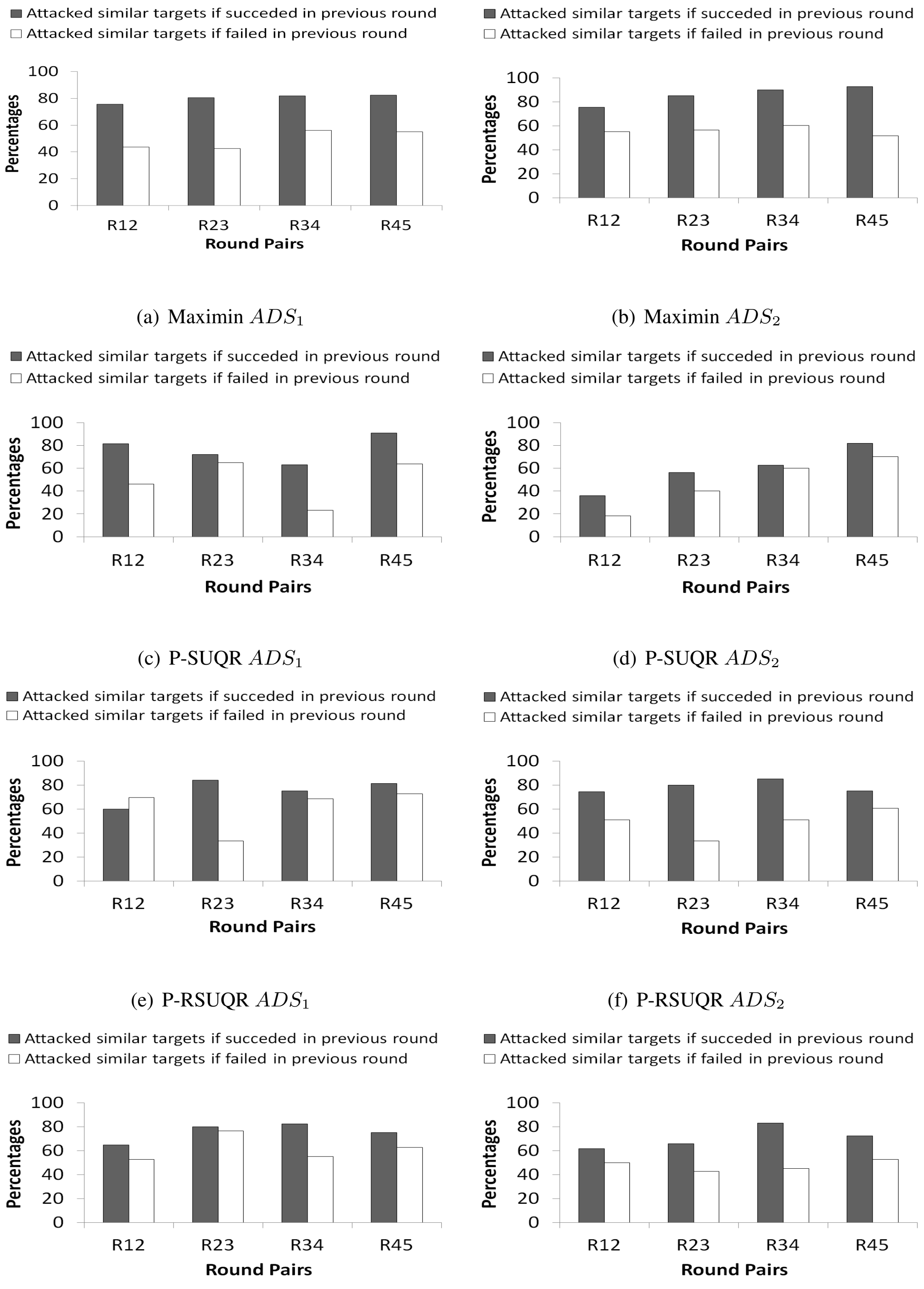
In order to provide evidence in support of Observation 2, we show results from our data highlighting these trends on ADS_1 and ADS_2 in Figs. 6.1(a) - 6.1(h). In each plot, the y-axis denotes the percentage of (i) attacks on similar targets out of the total successful attacks in the previous round (ζ_{ss}) and (ii) attacks on similar targets out of the total failed attacks in the previous round (ζ_{fs}). Here, by *similar*, we mean the k nearest neighbors to the target profile under consideration and these are determined by computing the Euclidean distances between the target profiles on the attack surface. In this case, we set k=5, i.e., 5 nearest neighbors. The x-axis denotes pairs of rounds for which we are computing the percentages, for example, in R23, 2 corresponds to round $(r-1)$ and 3 means round r in our claim. Thus, ζ_{ss} corresponding to R23 in ADS_2 is 80%, meaning that out of all the people who succeeded in round 2, 80% attacked similar target profiles in round 3. Similarly, ζ_{fs} corresponding to R23 in ADS_2 is 33.43%, meaning that out of all people who failed in round 2, 33.43% attacked similar target profiles in round 3.

From Figs. 6.1(a)-6.1(h), we can observe that, as opposed to the failed attackers, a statistically significant number of successful attackers returned to attack the same or similar targets in the subsequent round. The average (over all four models on two payoffs and for all round pairs) of ζ_{ss} is 75.2% and the average of ζ_{fs} which is 52.45%. This difference is statistically significant (two-tailed t-tests at confidence=0.05), thus supporting Observation 2.

One might however argue that successful poachers return to attack the same or similar targets in future rounds due to some inherent bias towards specific targets and not because they succeeded on such targets in the previous rounds. Therefore, we conducted additional human subjects experiments to test the extent to which successes and failures alone affect their decision making process.

We recruited two groups of human subjects and conducted two rounds of repeated experiments with each group. We showed the Maximin strategy to both groups in both rounds of the experiment. We ensured that all the participants of Group 1 succeeded in round 1, i.e., even though there were coverage probabilities shown, no rangers were actually “deployed”. In round 2, Maximin strategy was again deployed and the same set of players were asked to play. We observed that 96% of the human subjects attacked the same or similar (k=5) target profiles. We observed that out of the 96%, 70.83% attacked the exact same target profile as they had attacked in round 1. Group 2 was shown Maximin strategy in round 1 and all the participants were made to fail in round 1, i.e., despite the coverage probabilities, there was a “ranger” deployed in every cell. In round 2, Maximin strategy was again deployed and the same set of players were asked to play. We observed that only 36% of the participants attacked the same or similar (k=5) targets in round 2. This shows that successes and failures are important factors that players take into account while deciding on their strategy in subsequent rounds. Similarly, when k=6, we observe that 38% of the participants from Group 2 who failed in round 1, had actually attacked the same or similar target profiles. In Fig. 6.2, we show for various values of k, the percentage of successful participants in round 1 who returned to attack the same or similar targets in round 2 and the percentage of failed participants in round 1 who returned to attack same or similar targets in round 2.

Notice that failure does not lead all attackers to abandon their target profile (and vice versa with successful attacker). This shows that attackers have some inherent weights for defender coverage, animal density, penalty and distance, as is captured by the PSUweight vectors, but they do adapt their strategies based on their past successes and failures. Therefore, we will observe later in Chapter 7 that even though P-SUQR is outperformed by our model SHARP in the initial rounds, P-SUQR is still a valuable model.



(a) Maximin ADS_1

(b) Maximin ADS_2

(c) P-SUQR ADS_1

(d) P-SUQR ADS_2

(e) P-RSUQR ADS_1

(f) P-RSUQR ADS_2

(g) P-BSUQR ADS_1

(h) P-BSUQR ADS_2

Figure 6.1: Evidence for adaptivity of attackers

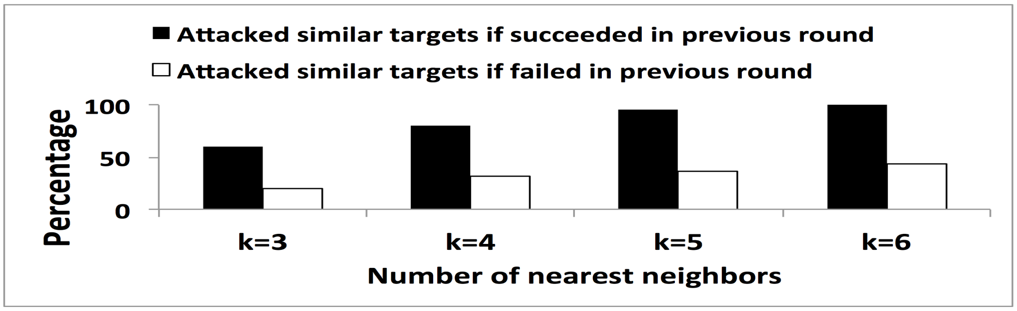


Figure 6.2: For various values of k (the number of nearest neighbors), percentage of people who attacked similar targets in round 2 after succeeding or failing in the previous round

These observations about successes and failures on the adversary's future behavior are also consistent with the "spillover effect" in psychology (Elster, 2005), which in our case suggests that an adversary will tend to associate properties of unexposed target profiles with knowledge about similar target profiles to which he has been exposed, where similarity is expressed in terms of the Euclidean distance between two points on the attack surface. Smaller distance indicates higher similarity. The above aspects of adversary behavior currently remain unaccounted for, in BR-RSG models. Based on observations above, we define two key properties below to capture the consequences of past successes and failures on the adversary's behavior and reason based on them.

Definition 3. *The* **vulnerability** *associated with a target profile β_i which was shown to the adversary in round r, denoted $V^r_{\beta_i}$, is defined as a function of the total number of successes and failures on the concerned target profile in that round (denoted by $success^r_{\beta_i}$ and $failure^r_{\beta_i}$ respectively). This is shown in Eqn. 6.1:*

$$V^r_{\beta_i} = \frac{success^r_{\beta_i} - failure^r_{\beta_i}}{success^r_{\beta_i} + failure^r_{\beta_i}} \tag{6.1}$$

Therefore, more successful attacks and few failures on a target profile indicate that it was highly vulnerable in that round. Because multiple targets can be associated with the same target profile and the pure strategy generated based on the mixed strategy x in a particular round may result in a defender being present at some of these targets while not at others, there may be both successes and failures associated with the same target profile in that round.

Definition 4. *The* **attractiveness** *of a target profile β_i at the end of round R, denoted $A^R_{\beta_i}$, is defined as a function of the vulnerabilities for β_i from round 1 to round R. It is computed using Eq. 6.2.*

$$A^R_{\beta_i} = \frac{\sum_{r=1}^{R} V^r_{\beta_i}}{R} \tag{6.2}$$

Therefore, we model the attractiveness of a target profile as the average of the Vulnerabilities for that target profile over all the rounds till round R. This is consistent with the notion that a target profile which has led to more successful attacks over several rounds will be perceived as more attractive by the adversary[2].

6.2 SHARP's Utility Computation

Existing models (such as SUQR, which is based on the subjective utility function (Eqn. 2.4)) only consider the adversary's actions from round $(r-1)$ to predict their actions in round r. However, based on our observation (Obs. 2) it is clear that the adversary's actions in a particular round are dependent on his past successes and failures. The *adaptive* probability weighted subjective

[2] Although here we give equal weight to the vulnerability values in each round, we can modify this easily to consider the recency effect in human decision making by discounting vulnerability values of earlier rounds and giving more importance to recent rounds. Such models of human discounting of past actions, such as hyperbolic discounting and exponential discounting, have been explored in (Azaria, Gal, Kraus, & Goldman, 2015; Chabris, Laibson, & Schuldt, 2006; Gans, Knox, & Croson, 2007). Exploring such models in our formulation would be an interesting area for future work.

utility function proposed in Eq. 6.3 captures this adaptive nature of the adversary's behavior by capturing the shifting trends in attractiveness of various target profiles over rounds.

$$\begin{aligned} ASU^R_{\beta_i} &= (1 - d * A^R_{\beta_i})\omega_1 f(x_{\beta_i}) + (1 + d * A^R_{\beta_i})\omega_2 \phi_{\beta_i} \\ &\quad + (1 + d * A^R_{\beta_i})\omega_3 P^a_{\beta_i} + (1 - d * A^R_{\beta_i})\omega_4 D_{\beta_i} \end{aligned} \tag{6.3}$$

There are three main parts to SHARP's computation: (i) Adapting the subjective utility based on past successes and failures on exposed parts of the attack surface; (ii) Discounting to handle situations where not enough attack surface has been exposed; and (iii) Reasoning about similarity of unexposed portions of the attack surface based on other exposed parts of the attack surface.

The intuition behind the adaptive portion of this model is that, the subjective utility of target profiles which are highly attractive to the adversary should be increased, and that of less attractive target profiles should be decreased, to model the adversary's future decision making. Hence, for a highly attractive target profile β_i, the attacker would view the coverage x_{β_i} and distance from starting location D_{β_i} to be of much lower value, but the animal density ϕ_{β_i} to be of higher value, as compared to the actual values. The contribution of the penalty term would also increase the utility (recall that $P^a_{\beta_i} < 0$ and $\omega_3 < 0$).

Let us take an example from our game. Suppose we take the target profile $\beta_i =< 0.25, 2, -1, 4 >$. This profile had $A^1_{\beta_i} = 1$ after round 1, because $success^r_{\beta_i} = 9$ and $failure^r_{\beta_i} = 0$, i.e., the target was highly attractive to the attacker. The weights learned were $b =< \delta, \gamma, \omega_1, \omega_2, \omega_3, \omega_4 > =< 2.2, 2.4, -3, 0.9, -0.3, -0.5 >$, P-SUQR would compute the subjective utility as -0.29, while (assuming d (explained later) $= 0.25$, for example) SHARP's adaptive utility function would compute the subjective utility as 0.855. In comparison to the original subjective utility function, this function is adaptive due to the positive or negative boosting of model weights based on the defender's knowledge about the adversary's past experiences. Here, learning the model parameters b has been decoupled from boosting the model parameters for future prediction to ensure simplicity in terms of both the model formulation as well the weight learning process. This also ensures that the linearity in terms of the features of the model (as in the original SUQR model) remains intact. Through an example in Section 6.4, we show the effect of this design decision on the defender mixed strategy generated.

Now we turn to the next aspect of SHARP's utility computation. Recall the problem that the defender does not necessarily have information about the attacker's preferences for enough of the attack surface in the initial rounds. This is because, the attacker is exposed to only a limited set of target profiles in each round and the defender progressively gathers knowledge about the attacker's preferences for only those target profiles. We provide evidence in support of this observation in Section 7.3.

The parameter d ($0 \leq d \leq 1$) in Eqn. 6.3 mitigates this attack surface exposure problem. It is a discounting parameter which is based on a measure of the amount of attack surface exposed. d is low in the initial rounds when the defender does not have enough of the right kind of data, but would gradually increase as more information about the attacker's preferences about various regions of the attack surface become available. For our experiments, we varied d based on Eqn. 6.4:

$$d = \frac{1}{N_r - r} \tag{6.4}$$

where N_r is the total number of rounds and r is the round whose data is under consideration. For example, $N_r = 5$ and $r = 1$ for data collected in round 1 of an experiment conducted over five rounds. The intuition behind this formulation is that, as more rounds are played, more cumulative information about the adversary' preferences for a lot of the attack surface will be available and hence d will increase from a very small value gradually as rounds progress.

Finally, we look at how we reason about unexposed portions of the attack surface based on the exposed areas. If a target profile β_u was not exposed to attacker response in round r, the defender will not be able to compute the vulnerability $V^r_{\beta_u}$. Therefore, we will also not be able to estimate the attractiveness for β_u and hence the optimal defender strategy. So, in keeping with our analysis on available data and based on the spillover effect introduced earlier, we use the distance-weighted k-nearest neighbors algorithm (Dudani, 1976) to obtain the Vulnerability $V^r_{\beta_u}$ of an unexposed target profile β_u in round r, based on the k most similar target profiles which were exposed to the attacker in round r (Eqns. 6.5 and 6.6).

$$V^r_{\beta_u} = \frac{\sum_{i=1}^{k} \theta_i * V^r_{\beta_i}}{\sum_{i=1}^{k} \theta_i} \tag{6.5}$$

$$\theta_i \equiv \frac{1}{d(\beta_u, \beta_i)^2} \tag{6.6}$$

where, $d(\beta_u, \beta_i)$ denotes the Euclidean distance between β_u and β_i in the feature space. We use $k = 5$ for our experiments.

6.3 Generating Defender Strategies Against SHARP

While SHARP provides an adversary model, we must now generate defender strategies against such a model. To that end, we first learn the parameters of SHARP from available data (See Section 5). We then generate future round strategies against the boundedly rational adversary characterized by the learned model parameters by solving the following optimization problem:

$$\max_{x \in \mathbb{X}} \left[\sum_{i \in \mathbb{T}} U_i^d\left(x\right) q_i^R\left(x \middle| \, \omega\right) \right] \tag{6.7}$$

$q_i^R(\omega|x)$ is the probability that the adversary will attack target i in round R and is calculated based on the following equation:

$$q_i^R(\omega|x) = \frac{e^{ASU^R_{\beta^i_k}(x)}}{\sum\limits_{i \in \mathbb{T}} e^{ASU^R_{\beta^i_k}(x)}} \tag{6.8}$$

β_k^i denotes that target profile β_k is associated with target i. $ASU^R_{\beta^i_k}$ and $U_i^d(x)$ are calculated according to Eqns. 6.3 and 2.1 respectively.

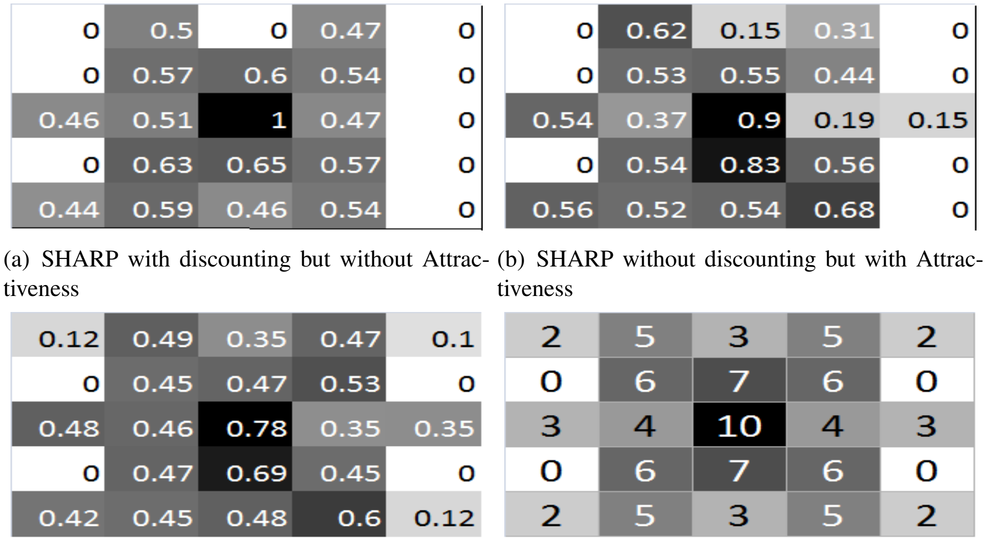
To solve the non-linear and non-convex optimization problem in Eqn. 6.7 and generate the optimal defender strategy, we use PASAQ (Yang et al., 2012) as it provides an efficient approximated computation of the defender strategy with near-optimal solution quality.

6.4 SHARP in action: An example

In this section, we give an example to show the effectiveness of SHARP in terms of the design of each component: (i) adaptive utility, (ii) similarity learning, and (iii) confidence based discounting. Figs. 6.3(a), 6.3(b) and 6.3(c) show second round strategies generated by SHARP with discounting of Eqn. 6.4 but without Attractiveness, SHARP without discounting, i.e., $d = 1$ but with Attractiveness, and SHARP, based on parameters learned ($b =< \delta, \gamma, \omega_1, \omega_2, \omega_3, \omega_4 >=< 1.2, 1.6, -3.2791, 0.1952, -0.3, -0.8388 >$) from first round data collected for the experiment with SHARP on ADS_1 (shown in Fig. 6.3(d)). The strategy generated by SHARP with discounting but without Attractiveness (see Fig. 6.3(a)) can be easily exploited due to several unprotected cells with positive animal density. SHARP without discounting but with Attractiveness (see Fig. 6.3(b)) generates a comparatively more robust strategy than SHARP with discounting but without Attractiveness (Fig. 6.3(a)) due to its adaptive utility function and similarity learning mechanism. SHARP generates the best strategy (see Fig. 6.3(c)) due to its capability to model all the design parameters together into one single framework.

6.5 RL-SSG: A Descriptive Reinforcement Learning Algorithm for SSGs

In this section, we translate the basic Reinforcement Learning (RL) model proposed by Erev and Roth (Erev & Roth, 1998) to our setting; we use their RL approach to compute the optimal mixed strategy for the defender in repeated SSGs. The primary reason for adapting an existing RL based approach for our problem is to compare our models against another popular learning framework which has been used earlier in the context of two-player games. However, as explained



0	0.5	0	0.47	0
0	0.57	0.6	0.54	0
0.46	0.51	1	0.47	0
0	0.63	0.65	0.57	0
0.44	0.59	0.46	0.54	0

(a) SHARP with discounting but without Attractiveness

0	0.62	0.15	0.31	0
0	0.53	0.55	0.44	0
0.54	0.37	0.9	0.19	0.15
0	0.54	0.83	0.56	0
0.56	0.52	0.54	0.68	0

(b) SHARP without discounting but with Attractiveness

0.12	0.49	0.35	0.47	0.1
0	0.45	0.47	0.53	0
0.48	0.46	0.78	0.35	0.35
0	0.47	0.69	0.45	0
0.42	0.45	0.48	0.6	0.12

(c) SHARP

2	5	3	5	2
0	6	7	6	0
3	4	10	4	3
0	6	7	6	0
2	5	3	5	2

(d) ADS_1

Figure 6.3: (a,b,c): Round 2 strategies generated by SHARP (with discounting without Attractiveness), SHARP (no discounting but with Attractiveness) and SHARP respectively; (d) ADS_1

in Section 3.1.3, the main challenge of using the same framework as in (Erev & Roth, 1998) is that the models in Erev and Roth's paper were developed for simultaneous move games without any notion of prior commitment to a mixed strategy by any player. Therefore, we developed a new RL model for our leader-follower setting where the defender moves first by playing a *mixed* strategy and the other player moves next by playing a pure strategy after having observed the first player's mixed strategy. Since we deploy and compute optimal mixed strategy responses for the defender per round based on all the attacks observed in the earlier rounds, we do not explicitly require a lot of defender and attacker pure strategy combinations to be deployed. We describe below the RL based algorithm (Algorithm 1) to compute the optimal mixed strategy.

The algorithm first starts with an initial propensity for the defender to play any pure strategy k (Line 1). The probability distribution over all possible pure strategies is then computed by normalizing the propensities (Line 2). The defender then computes the mixed strategy and deploys this strategy (which results in a coverage probability for each target as discussed in Section 2.1), and collects all pure strategy responses of the attacker to the defender's mixed strategy in the corresponding round (Line 2). Using this data collected in a particular round, the defender computes her utility of playing each pure strategy k (Line 3). More specifically, this utility is computed for each pure strategy as the reward that would result to the defender given the observed adversary response if the defender were playing only this pure strategy. The reinforcement of playing each pure strategy is then computed as the difference between the corresponding utility and the minimum possible utility over all the pure strategies (Line 4). The defender then updates her propensities of playing each pure strategy by adding the reinforcements to the propensities computed in the earlier round (Line 5). The mixed strategy to be deployed in the next round is then computed in the same way as Line 2, and the game proceeds to future rounds in this fashion.

In our game, starting from an equal initial propensity for each pure strategy (as proposed in (Erev & Roth, 1998)) would result in a mixed strategy with equal coverage probability at each target. Due to the differences in animal densities, if we start from a uniform mixed strategy, it would leave many of the targets of high animal density to be attacked in round 1 (as evidenced from other human subject experiments conducted in the past (Pita et al., 2010)) and that would result in a defender utility which is much lower than the cumulative utility for any of our models over five rounds (see Chapter 7 for details). Therefore, for our experiment with the RL approach, we assumed that the defender starts with the robust Maximin strategy in round 1. The use of Maximin as the initial strategy also ensures that we allow the RL model to start from the same starting point as all other models, and that it does not have an initial advantage or disadvantage compared to other models. Thus, we used Maximin to compute the initial propensities for pure strategies in this setup; we used Comb Sampling on the Maximin mixed strategy (Tsai, Yin, young Kwak, Kempe, Kiekintveld, & Tambe, 2010) to compute the probability of playing each pure strategy and considered those to be the initial propensities for each pure strategy. We then updated the propensities based on round 1 attack data for Maximin on ADS_1 and computed the corresponding mixed strategy and deployed that as the round 2 strategy. Based on this experiment, results and comparisons of the RL based approach against other models are shown in Chapter 7.

Algorithm 1 Algorithm to learn RL based defender strategy for repeated Stackelberg games

INPUT: Set of Targets T; Number of security resources M.

OUTPUT: Optimal defender strategy for any round r.

1: In round $r = 1$, the defender has an initial propensity to play pure strategy k, denoted by $q_k(1); \forall k \in P$, where P is the set of all possible pure strategies as described earlier in Chapter 2. (1) denotes round 1.

2: The defender computes the probability of playing a particular pure strategy k in round r based on the propensities as follows:

$$p_k(r) = \frac{q_k(r)}{\sum\limits_{j \in P} q_j(r)} \tag{6.9}$$

The defender can then directly compute the mixed strategy $x(r)$ for round r from the $p_k(r)$'s, deploy $x(r)$ and collect attack data for round r.

3: The defender uses data collected in round r, i.e. D^r, to compute the expected utility of playing pure strategy k in round $r+1$ as follows:

$$U_k(r+1) = \frac{\sum\limits_{i \in T} D_i(r) * B_{k,i}}{\sum\limits_{i \in T} D_i(r)} \tag{6.10}$$

Here, $D_i(r)$ denotes the number of attacks on target i in round r, $B_{k,i}$ denotes the payoff to the defender if she plays pure strategy k and the adversary plays pure strategy i (i.e. attacks target i). It is calculated as follows:

$$\mathrm{B_{k,i}} = \begin{cases} \mathrm{R_i^d} & \text{if 'i' is protected in pure strategy 'k'} \\ \mathrm{P_i^d} & \text{if 'i' is not protected in pure strategy 'k'} \end{cases}$$

Here, R_i^d is the defender's reward for covering i if it is selected by the adversary and P_i^d is the penalty for not covering i.

4: The reinforcement of playing pure strategy k is then computed as:

$$I_k = U_k(r+1) - \underset{k}{min}(U_k(r+1)) \tag{6.11}$$

Here, $\underset{k}{min}(U_k(r+1))$ denotes the minimum utility over all the computed utilities.

5: The propensities of the defender for pure strategy $k (\forall k \in P)$ in round $(r+1)$ is then updated as follows:

$$q_k(r+1) = q_k(r) + I_k \tag{6.12}$$

6: Repeat Step 2 to Step 5.

Chapter 7

Results with Human Subjects on AMT

This chapter first shows the results of our human subjects experiments on AMT, and then discusses results against security experts, based on the experimental setting discussed in Chapter 4. In Section 7.1, we show average defender expected utilities for five models (P-SUQR, P-BSUQR, P-RSUQR, SHARP and Maximin) against actual human subjects, for various rounds of our experiment on two payoff structures (ADS_1 and ADS_2). In Section 7.2, we show results of learning the shape of the probability weighting function for the adversaries on ADS_1 and ADS_2 for three different scenarios: (i) when Gonzalez and Wu's probability weighting function Eqn. 5.1 is used in Eqn. 5.5; (ii) when Prelec's probability weighting (Eqn. 3.1) is used in Eqn. 5.5; and, (iii) when transformed probabilities with Gonzalez and Wu's function is used to weight the values of outcomes (Eqn. 5.6). In the same section, we also show prediction performances of models for cases (ii) and (iii) above. Section 7.3 shows the effect of attack surface exposure on the performance of P-SUQR. In Section 7.4, we show the adaptiveness of SHARP based strategies over rounds while P-SUQR based strategies converge to a particular strategy at the end of a few rounds. Plots for all the analysis on payoff structures ADS_3 and ADS_4 are reported in Appendix 6.

7.1 Defender Utilities

In Figs. 7.1(a)-7.1(b), we show actual defender utilities obtained over 5 rounds for P-SUQR, P-BSUQR, P-RSUQR, SHARP and Maximin on ADS_1 and ADS_2 respectively, with an average of 38 human subjects playing per round. Similar to human subjects experiment results in previous work (Pita, John, Maheswaran, Tambe, & Kraus, 2012; Yang et al., 2011; Nguyen et al., 2013), in the plots, y-axis corresponds to defender expected utility. However, what is different here is that, we are now reporting results on repeated rounds and hence the round numbers are shown on the x-axis. For example, in Fig. 7.1(b), P-SUQR performs worst in round 2 with a utility of -5.26. In Fig. 7.1(b), we also show (inset) zoomed in results of the second round to highlight the difference in performance between Maximin (= -0.18) and SHARP (= -0.1). Figs. 7.2(a)-7.2(b) show cumulative defender utility over five rounds on ADS_1 and ADS_2 respectively. Note that the first round utilities for all models are same as Maximin strategy was played due to absence of data. All significance results reported below are computed with bootstrap t-test. Following are key observations from our experiments.

- **Heavy initial round losses**: For all models except SHARP, there is statistically significant (p=0.05) loss in defender utility as compared to Maximin in second round on all the payoffs. P-SUQR recovers from initial round losses and outperforms Maximin in rounds 3, 4 and 5 for ADS_1 (statistically significant at p=0.05), and in round 4 (statistically significant at p=0.15) and round 5 for ADS_2. P-SUQR also outperforms Maximin in rounds 3, 4 and 5 on ADS_3 and ADS_4 (see Appendix 6). P-RSUQR, which is a robust model, also outperforms Maximin in rounds 4 and 5 (statistically significant at p=0.05) for ADS_1, ADS_3 and ADS_4 after initial round losses. Surprisingly, P-BSUQR, which is the basis for wildlife security application PAWS (Yang et al., 2014), performs worst on all payoffs over all rounds.

From Fig. 7.2(a), we can observe that it takes five rounds for P-SUQR to recover from initial round losses and outperform Maximin in terms of cumulative defender utility for ADS_1. P-SUQR does not recover from initial round losses and outperform Maximin on any other payoffs. None of the other models recover from the initial round losses on any of the payoffs in five rounds, thus highlighting the impact initial round losses have on model performance for a long period of time.

- **Performance of SHARP against other models**: SHARP consistently outperforms (statistically significant at p=0.05) all the models over all rounds (Figs. 7.1(a)-7.1(b), and 13.3(a)-13.3(b)), most notably in initial rounds (round 2) and ends up with significantly high cumulative utility at the end of all rounds (Figs. 7.2(a)-7.2(b) and 13.4(a)-13.4(b)).

  Therefore, our results on extensive human subjects experiments on repeated SSGs show SHARP's ability to perform well throughout, including the important initial rounds.

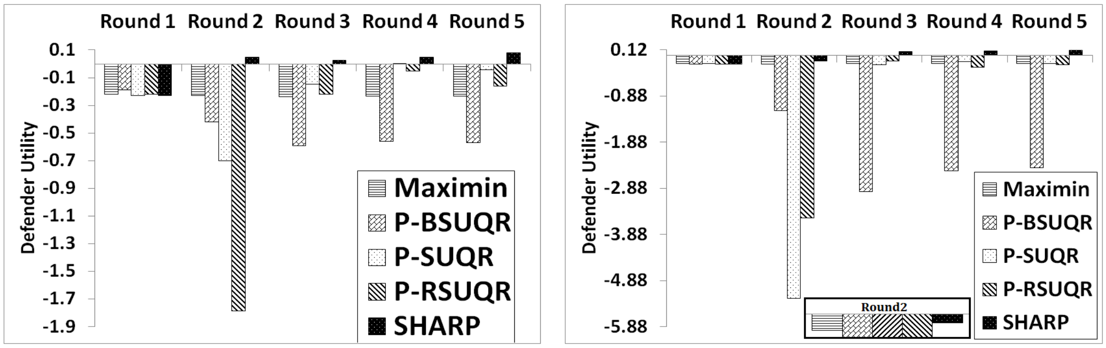
- **Performance of SHARP (with and without discounting)**: To test the effectiveness of the design decisions in SHARP, we considered SHARP both with and without discounting. SHARP with $d = 1$ is compared against SHARP and P-SUQR on ADS_1 and ADS_2 in Figs. 7.3(a) and 7.3(b). SHARP($d = 1$) outperforms P-SUQR (statistically significant at p=0.05) because it captures the adaptive nature of the adversary. However, it performs worse than SHARP (statistically significant at p=0.01) as SHARP also trusts the data less when we don't have enough information about the adversary's responses to most of the attack surface; in this case the initial rounds.

- **Comparison with SUQR** ($w_1 > 0$): As mentioned earlier in Section 7.1, we conduct additional human subjects experiments on ADS_1 to show that the performance of SUQR without probability weighting is worse than any of the other models. We deployed an experiment on AMT with the defender strategy computed based on the SUQR model learned from round 1 data of ADS_1. The resulting SUQR weight vector had a positive weight on coverage probability and thus resulted in a defender pure strategy. The game was deployed with this strategy on AMT. 60 people played the game, and out of them 48 participants passed the validation test. For the experimental results, we considered the data from only the participants who passed the validation test. The average expected defender utility obtained was -4.75. This average expected defender utility obtained by deploying a pure strategy based on a learned SUQR model is significantly less than that of all the other models on ADS_1 in Round 2 (Figure 7.4(a)). Furthermore, the SUQR (Pure Strategy) model's average expected defender utility in this one round is significantly less than the cumulative average expected defender utility of all the other models after five rounds (Figure 7.4(c)). Given that this strategy performs worse in one round than the cumulative average expected defender utility of all the other models, it demonstrates the point that the performance of SUQR without probability weighting is worse than any of the other models that include probability weighting.

  Notice that the reason this SUQR pure strategy performs so poorly is that it leaves 16 out of 25 targets completely exposed, and among these targets are ones with animal densities 4 and 5. Pure strategies for other reward structures similarly leave other targets of high value completely exposed. Also, as mentioned in Chapter 1, degraded performance in initial rounds may have severe consequences for the reasons outlined there. Thus, the poor performance in this initial round of the pure strategy on ADS_1 and its leaving targets of high value completely exposed illustrates that pure strategy SUQR is completely useless as a strategy for deployment. Therefore, we did not conduct any further experiments for future rounds with this model.

- **Comparison with RL based approach**: We conducted human subjects experiments on ADS_1 with the RL based approach (Algorithm 1) to compare its performance against our behavioral models. We deployed an experiment on AMT with the defender strategy computed based on the RL model learned from round 1 data of Maximin on ADS_1 (as explained earlier in Section 6.5). 63 participants played the game, out of which 49 participants passed the validation test. For the experimental results, we considered the data from only the participants who passed the validation test. The average expected defender utility obtained was -4.139. This average expected defender utility obtained by deploying the defender strategy based on a learned RL model is significantly less than that of all the other models on ADS_1 in Round 2 (Figure 7.4(b)). Furthermore, the RL model's average expected defender utility in this one round is significantly less than the cumulative average expected defender utility of all the other models after five rounds (Figure 7.4(d)). Given that this strategy performs worse in one round than the cumulative average expected defender utility of all the other models, it has very little chance of ever recovering and outperforming the other models we have discussed earlier after more rounds.

  In addition, we show the deployed defender strategy for round 2 on ADS_1 in Figure 7.5. Based on Figure 7.5, notice that the reason the RL based approach performs so poorly is that after learning from attacks in the previous round, it places a significant amount of coverage on target cells with high number of attacks (the cell with a resultant coverage probability of 0.492 in round 2 had an animal density of 6 and was attacked 7 times in round 1), while it reduces the coverage on cells with very few attacks (the middlemost cell with a resultant round 2 coverage probability of 0.371 had an animal density of 10 and only 1 attack in round 1). This is because, for cells with zero or very few attacks, the propensities for playing strategies that correspond to protecting those targets are not updated as much as cells which have been attacked more frequently and have simultaneously resulted in higher gains for the defender. This leaves cells with high rewards but very few attacks in the past rounds less protected in the subsequent round and therefore completely exposed to a lot of attacks. RL based strategies for other reward structures similarly leave other targets of high value (but very attacks) with little protection and therefore open to exploitation by the attacker in the subsequent round. Thus, the poor performance in this initial round of the RL model on ADS_1 and its leaving targets of high value exposed to exploitation in the subsequent round illustrates that significant new work would need to be done to adapt the proposed RL framework (based on the model by Erev and Roth (Erev & Roth, 1998)) for our Stackelberg Security Game setting. As mentioned in Chapter 1, degraded performance in initial rounds may have severe consequences for the reasons outlined there. Therefore, given the poor initial round performance of the RL model on ADS_1, we did not conduct any further experiments for future rounds.



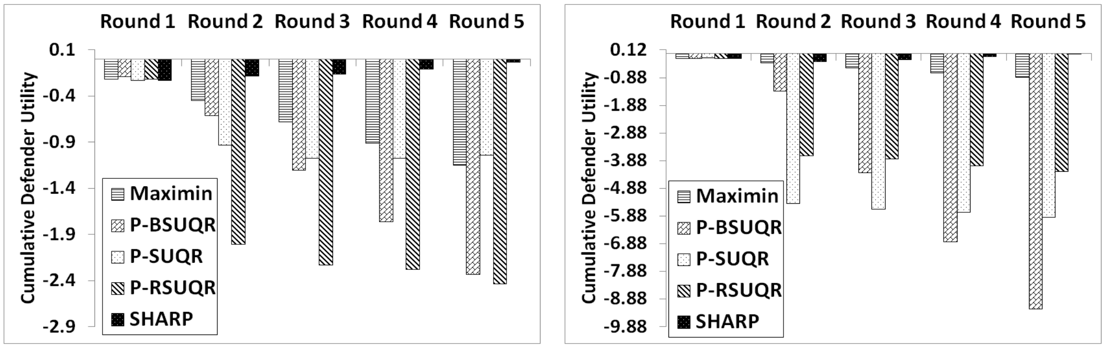
(a) Results on ADS_1

(b) Results on ADS_2

Figure 7.1: Defender utilities for various models on ADS_1 and ADS_2 respectively.

7.2 Learned Probability Curves

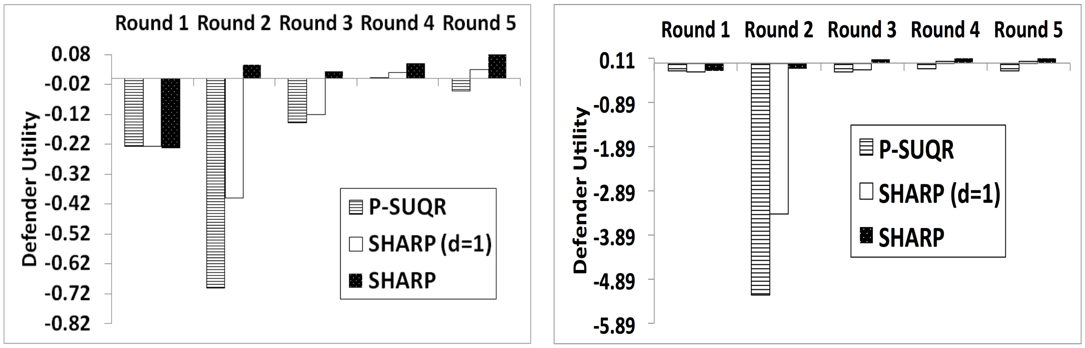
Figs. 7.6(a)-7.6(b) and 13.5(a)-13.5(b) show human perceptions of probability in rounds 1 to 4 when the participants were exposed to P-SUQR based strategies on ADS_1, ADS_2, ADS_3 and ADS_4 respectively. Learned curves from P-SUQR on all payoffs have this S-shaped nature, showing that even though there is a little change in the curvature between rounds, it retains the same S-shape throughout all rounds. The curves indicate that people weigh high probabilities to be higher and low to medium probabilities to be lower than the actual values. Even though this is contrary to what Prospect theory (Tversky & Kahneman, 1992) suggests, this is an intuitive result for our Stackelberg Security Games domain because we would expect the adversary to be deterred from targets with very high coverage probabilities and that they would prefer to attack targets with low to medium coverage probabilities.



(a) Results on ADS_1

(b) Results on ADS_2

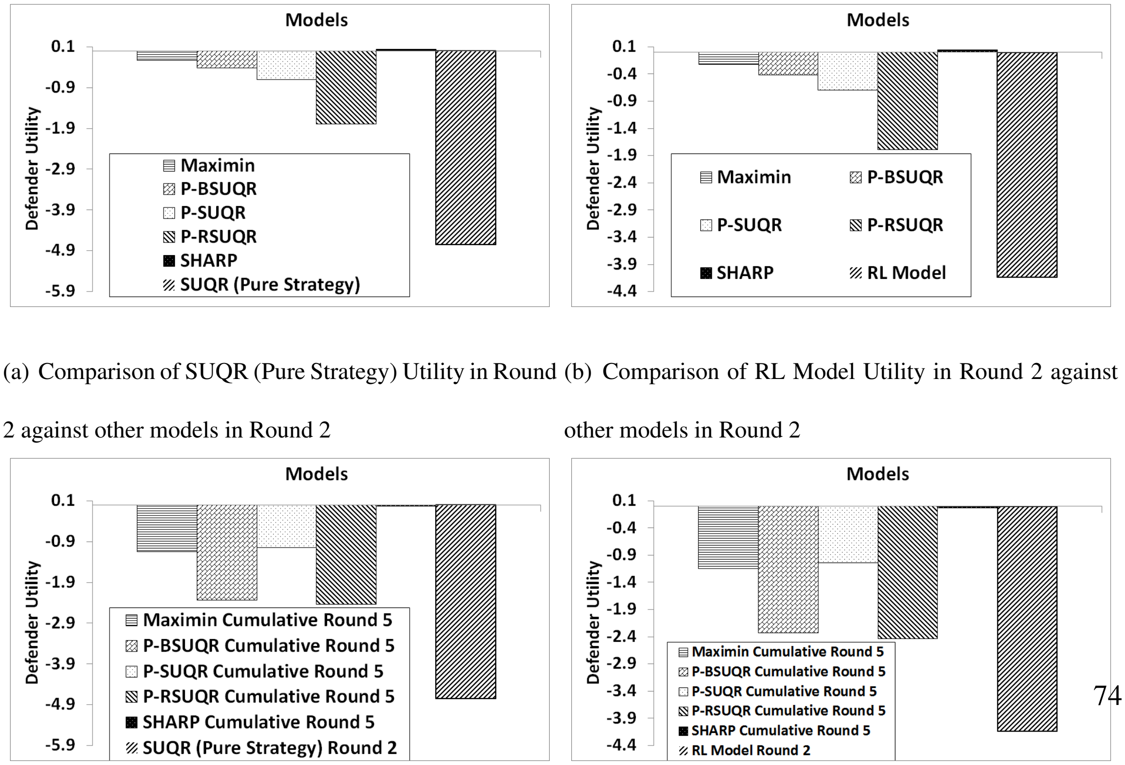
Figure 7.2: Cumulative defender utilities for various models on ADS_1 and ADS_2 respectively.



(a) Results on ADS_1

(b) Results on ADS_2

Figure 7.3: (a) Comparison of defender utilities between P-SUQR, SHARP and SHARP(d=1) on ADS_1 and ADS_2 respectively

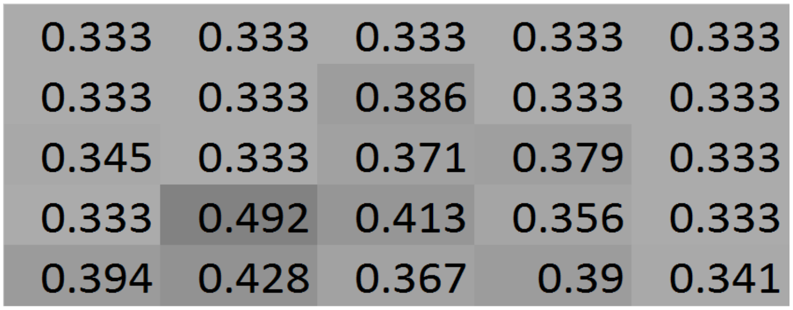


(a) Comparison of SUQR (Pure Strategy) Utility in Round 2 against other models in Round 2

(b) Comparison of RL Model Utility in Round 2 against other models in Round 2

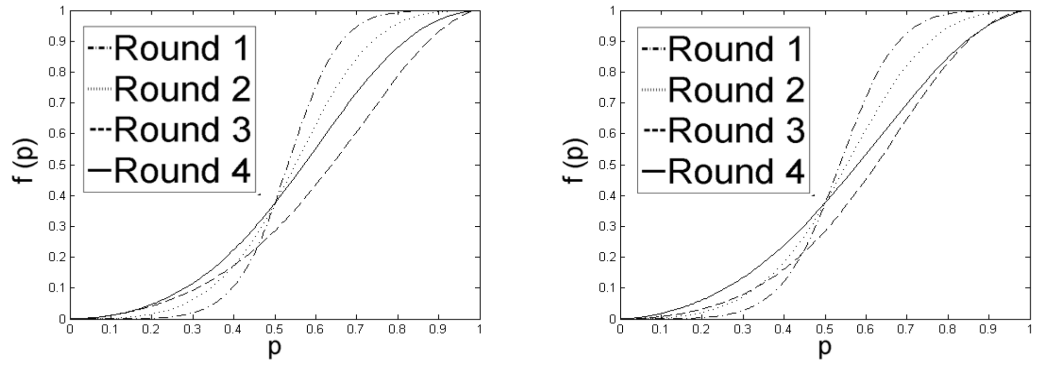
(c) Comparison of SUQR (Pure Strategy) Utility in Round 2 against Cumulative Utility of other models after Round 5

(d) Comparison of RL Model Utility in Round 2 against Cumulative Utility of other models after Round 5



0.333	0.333	0.333	0.333	0.333
0.333	0.333	0.386	0.333	0.333
0.345	0.333	0.371	0.379	0.333
0.333	0.492	0.413	0.356	0.333
0.394	0.428	0.367	0.39	0.341

Figure 7.5: RL model based defender strategy for round 2 on ADS_1.



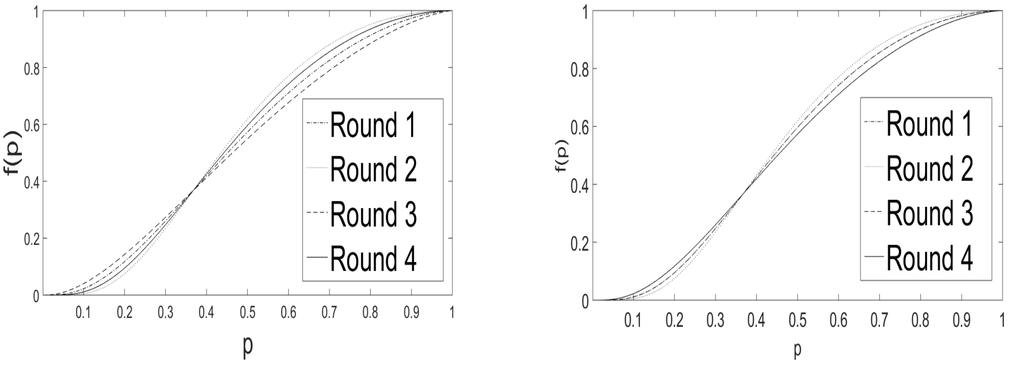
(a) ADS_1

(b) ADS_2

Figure 7.6: Learned probability curves for P-SUQR on ADS_1 and ADS_2 respectively.

7.2.1 Comparison with Prelec's probability weighting function

As mentioned earlier in Chapter 3.3, we also experiment with Prelec's one-parameter model while allowing α to be any value greater than zero. In this case too, we learn S-shaped curves on all of our payoff structures as shown in Figs. 7.7(a)-7.7(b) and 13.6(a)-13.6(b). This indicates that the shape of the learned curve is not dependent on the probability weighting function used, as long as the function allows for learning both an S-shaped and an inverse S-shaped curve. In addition, the prediction performance (average of the sum of squared errors for all rounds and animal density structures) of P-SUQR with Gonzalez and Wu's probability weighting function (Eqn. 5.1 and



(a) ADS_1

(b) ADS_2

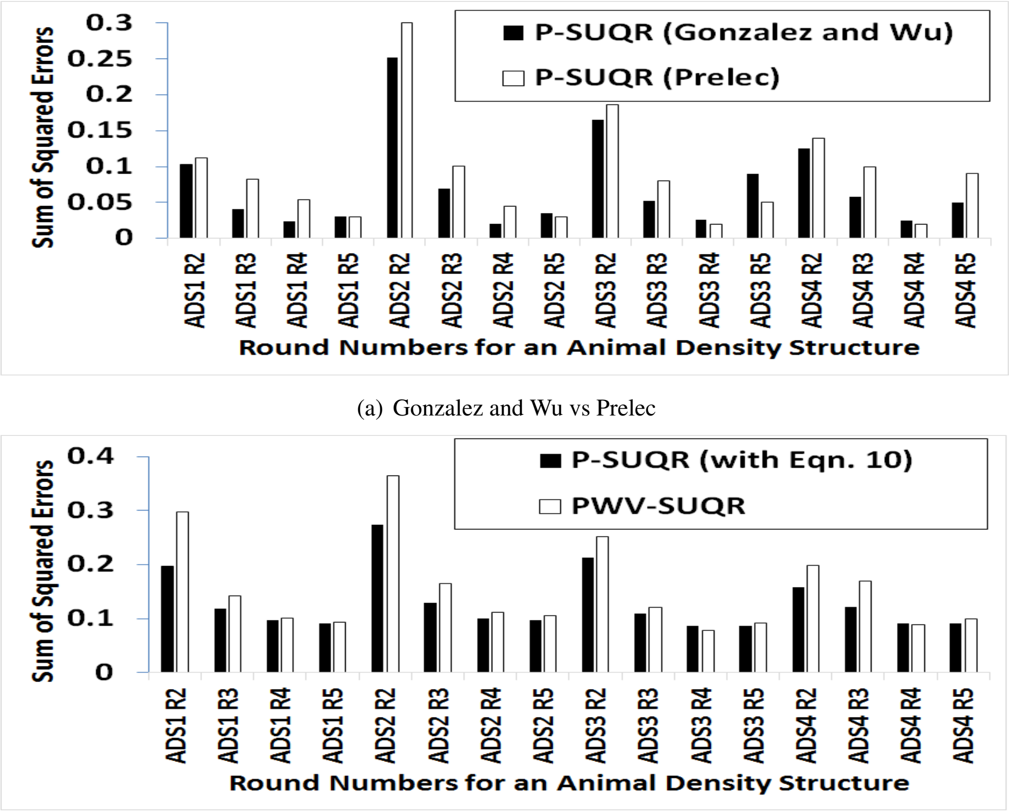
Figure 7.7: Learned probability curves with Prelec's probability weighting function for P-SUQR on ADS_1 and ADS_2 respectively.

Eqn. 5.5) and P-SUQR with Prelec's probability weighting function (Eqn. 3.1 and Eqn. 5.5) are 0.072 and 0.09 respectively and this is statistically significant at p=0.02. The sum of squared errors in prediction for each of the four rounds (round 2 to 5) and each animal density structure are shown in Figure 7.8(a), where the x-axis shows each possible combination of animal density structures and rounds, and the y-axis shows the sum of squared errors.

7.2.2 Comparison with PWV-SUQR

As mentioned earlier in Chapter 5.2, the adversary behavior model PWV-SUQR is one plausible alternative that could be considered for comparison with our models. Therefore, in this section, we first show the probability weighting curves learned (Figs. 7.9(a)-7.9(b) and 13.7(a)-13.7(b)) when we consider Eqn. 5.6 (see Chapter 5.2) as the subjective utility function in our adversary model. We observe that the curves are S-shaped in nature and this indicates that the shape of the probability weighting curves in our domain is not dependent on the use of the P-SUQR model [1].

[1] Note that, instead of Eqn. 5.6, even if we use prospects where the transformed probabilities weight the transformed values (Kahneman & Tversky, 1979; Tversky & Kahneman, 1992), we still get S-shaped curves in our game setting.

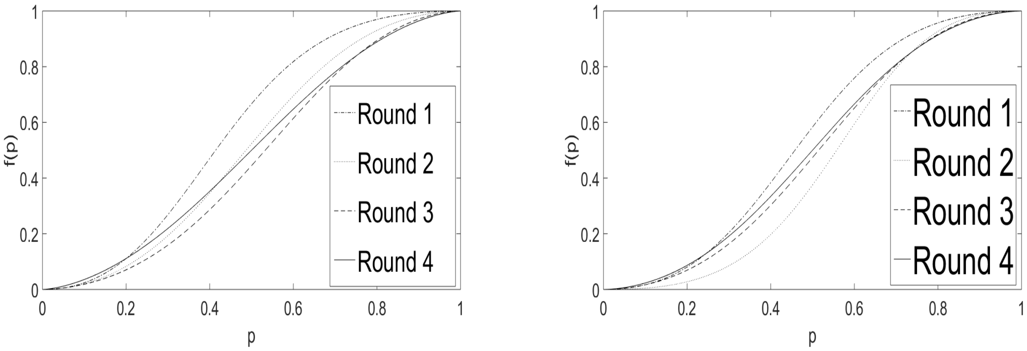


(a) Gonzalez and Wu vs Prelec

(b) P-SUQR vs PWV-SUQR

Figure 7.8: (a) Comparison of the sum of squared errors for P-SUQR with Gonzalez and Wu, and P-SUQR with Prelec's probability weighting function respectively; (b) Comparison of the sum of squared errors for P-SUQR and PWV-SUQR respectively

Nonetheless, PWV-SUQR does raise an intriguing possibility as a plausible alternative for P-SUQR and thus the performance of PWV-SUQR should be compared with P-SUQR. Therefore, we compare the performance of P-SUQR (with the PSU function in Eqn. 5.2) and PWV-SUQR in terms of predicting future round attacks. We show that P-SUQR (with the PSU function in Eqn. 5.2) performs better (with statistical significance) as compared to PWV-SUQR. The sum of squared errors in prediction for each of the four rounds (round 2 to 5) and each animal density structure are shown in Figure 7.8(b), where the x-axis shows each possible combination of animal density structures and rounds, and the y-axis shows the sum of squared errors. The prediction



(a) ADS_1

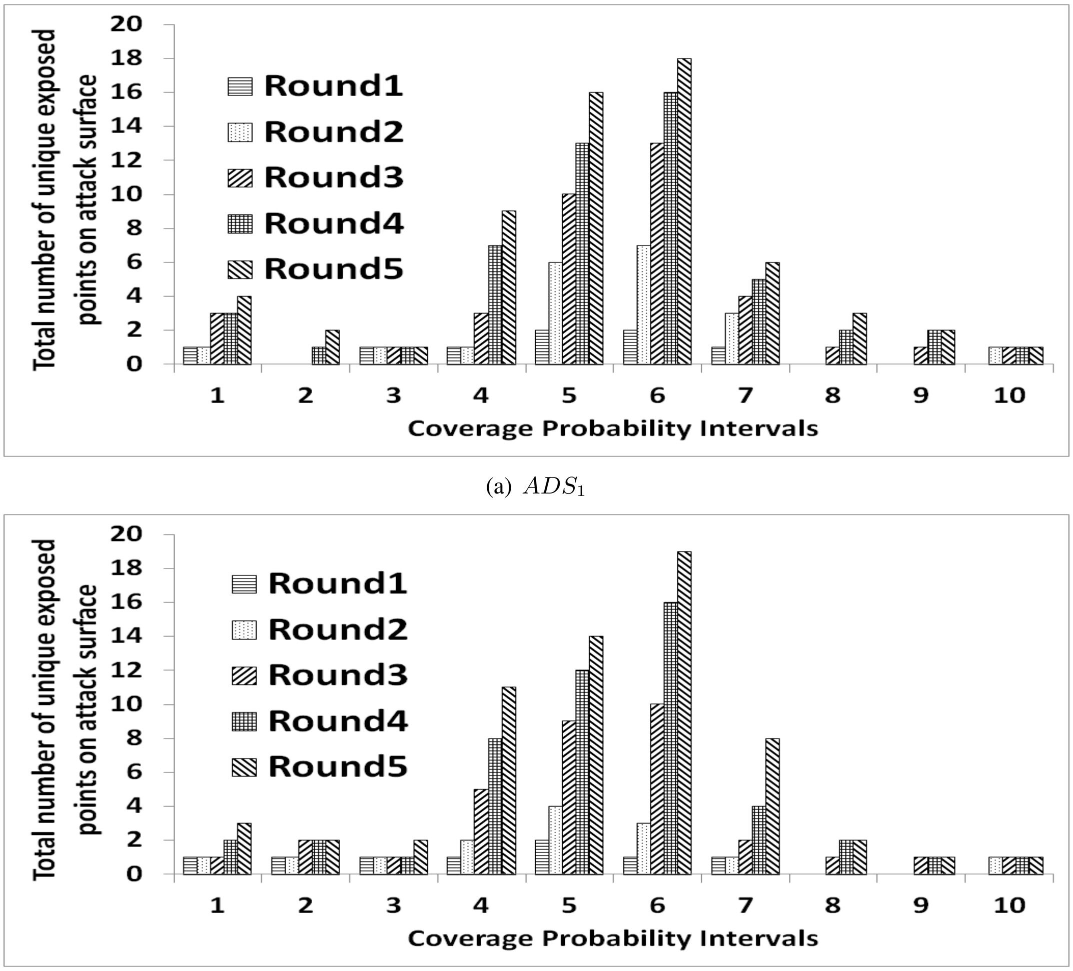
(b) ADS_2

Figure 7.9: (a) - (d) Learned probability curves for PWV-SUQR on ADS_1 and ADS_2 respectively.

performance (average of the sum of squared errors for all rounds and animal density structures) of P-SUQR (with the PSU function in Eqn. 5.2) and PWV-SUQR are 0.128 and 0.155 respectively and this is statistically significant at p=0.01. This justifies the use of P-SUQR and its variants while modeling the adversary.

7.3 Attack surface exposure

In our repeated SSG, the only variation in terms of feature values for our model (Eqn. 6.3) from round to round is the mixed strategy x and hence the coverage x_i at each target. Hence, exposure to various regions of the attack surface means exposure to various values of x_i for fixed values of the other model parameters. Fig. 7.10(a)-7.10(b) and Fig. 13.8(a)-13.8(b) show how the adversary was exposed to more unique values of the coverage probability, and hence attack surface, when conducting experiments with P-SUQR over the five rounds for ADS_1, ADS_2, ADS_3 and ADS_4 respectively. We discretize the range of x_i, i.e. [0,1] into 10 intervals (x-axis) and show the total number of unique coverages exposed till a particular round (y-axis) for each interval. Observe that more interval ranges and more unique coverage probabilities get exposed



(a) ADS_1

(b) ADS_2

Figure 7.10: Total number of unique exposed target profiles till the end of each round for each coverage probability interval for ADS_1 and ADS_2.

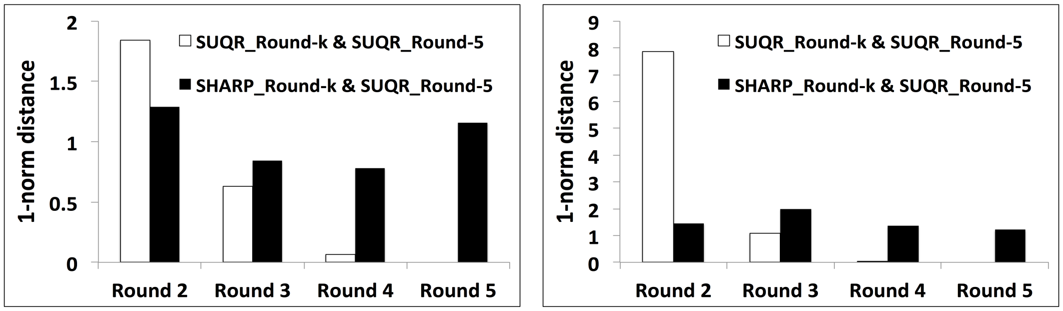
in rounds 3 to 5, thus exposing more of the attack surface. Based on our earlier discussion in Section 6, this phenomenon of revealing more of the attack surface would lead to improved gain in information about the adversary and would thus help us to perform better in the future rounds. As we showed in Figs. 7.1(a)-7.1(b) and 13.3(a)-13.3(b), the defender performance for P-SUQR improves significantly in rounds 4 and 5.

7.4 Adaptiveness of SHARP

Recall that P-SUQR assumes the presence of a homogeneous adversary type and attempts to learn that adversary type from past attack data. So we should expect that as we learn the model parameters over various rounds, the learned parameters and hence the generated defender strategy should converge. On the contrary, SHARP models the adaptive nature of a homogeneous adversary type based on his past successes and failures. Hence the convergence of the defender strategy generated based on SHARP in each round is not guaranteed. Figs. 7.11(a)-7.11(b) and 13.9(a)-13.9(b) show the 1-norm distance between defender strategies generated by SHARP (and P-SUQR) over rounds with respect to the strategy generated by P-SUQR in round 5. While P-SUQR converges to a particular strategy in round 5 for all four animal density structures, SHARP does not converge to any strategy. To further illustrate that the SHARP based strategy does indeed change over rounds, we show SHARP based strategies on ADS_2 from rounds 2 to 5 in Figs. 7.12(a) - 7.12(d). For ADS_2, the 1-norm distances between the defender strategies in rounds 2 and 3, rounds 3 and 4, and rounds 4 and 5 are 2.324, 2.19 and 1.432 respectively, showing that the strategies are changing from round to round. All these results demonstrate the adaptivity of SHARP over rounds based on the successes and failures of the adversaries in the past.

7.5 Validation and Testing Robustness of AMT findings

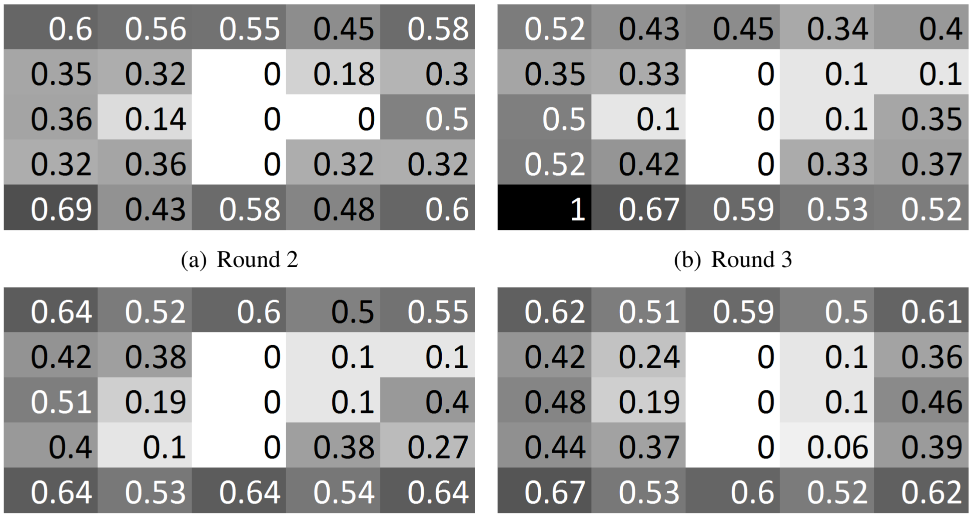
While in general findings from AMT have been validated with human subject experiments in the lab, the first question we ask is whether domain experts would perform similarly to what was observed of human subjects in AMT studies, i.e., we wish to further validate the findings from AMT. To that end, we deploy SHARP-based strategies against security experts at a national park



(a) ADS_1

(b) ADS_2

Figure 7.11: Adaptivity of SHARP and Convergence of P-SUQR on payoff structures ADS_1 and ADS_2 respectively.



0.6	0.56	0.55	0.45	0.58
0.35	0.32	0	0.18	0.3
0.36	0.14	0	0	0.5
0.32	0.36	0	0.32	0.32
0.69	0.43	0.58	0.48	0.6

(a) Round 2

0.52	0.43	0.45	0.34	0.4
0.35	0.33	0	0.1	0.1
0.5	0.1	0	0.1	0.35
0.52	0.42	0	0.33	0.37
1	0.67	0.59	0.53	0.52

(b) Round 3

0.64	0.52	0.6	0.5	0.55
0.42	0.38	0	0.1	0.1
0.51	0.19	0	0.1	0.4
0.4	0.1	0	0.38	0.27
0.64	0.53	0.64	0.54	0.64

(c) Round 4

0.62	0.51	0.59	0.5	0.61
0.42	0.24	0	0.1	0.36
0.48	0.19	0	0.1	0.46
0.44	0.37	0	0.06	0.39
0.67	0.53	0.6	0.52	0.62

(d) Round 5

Figure 7.12: SHARP based strategy for the defender on payoff structure ADS_2.

in Indonesia and analyze the results (Chapter 7.5.1) by comparing them with our observations on human subjects data from AMT. A second question that may be raised is with regard to our assumption that all attack data is perfectly observed in AMT studies. Therefore, we analyze SHARP-based strategies with only a fraction of the entire data (Chapter 7.5.2).

7.5.1 Results with Security Experts in Indonesia

In order to validate our AMT findings, we also conducted human subjects experiments for SHARP in the real world: with wildlife security experts from the provinces of Lampung and Riau, Sumatra, Indonesia. The 33 participants were from the local government, and from the following NGOs YABI, WWF and WCS. Each of the 33 participants played SHARP based strategies over 4 rounds. As in our AMT experiments, the first round strategy was Maximin.

In Fig. 7.13, we show actual defender utilities obtained over 4 rounds for SHARP on ADS_3. Interestingly, the defender utility obtained in round 2 was not only significantly higher than other rounds, but is also significantly higher than the utility obtained in round 2 for the same animal density structure for AMT participants (see Fig. 13.3(a)). This is because 96% of the experts who were successful in round 1 had attacked the same or similar targets in round 2. This is comparatively higher than the number of successful participants on AMT on ADS_3 in round 1 who returned to attack the same or similar targets in round 2: it was 78%. Hence, our model SHARP which captures the adversary's adaptiveness based on their past successes and failures, completely outperforms the experts. The defender's utility drops in round 3 as compared to that in round 2, because the experts, now aware of SHARP's adaptiveness, adjust their strategy. However, SHARP is robust enough to still generate high utility for the defender.

Similarity between AMT and Indonesia experts data: We earlier conducted a set of analyses and made certain observations based on our human subjects experiments data from AMT. We conducted the same analysis on the attack data obtained from real-world experts to validate our AMT results.

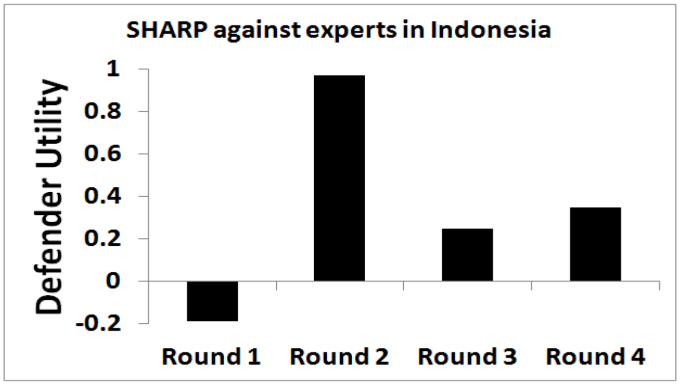


Figure 7.13: Defender utility for SHARP against security experts in Indonesia

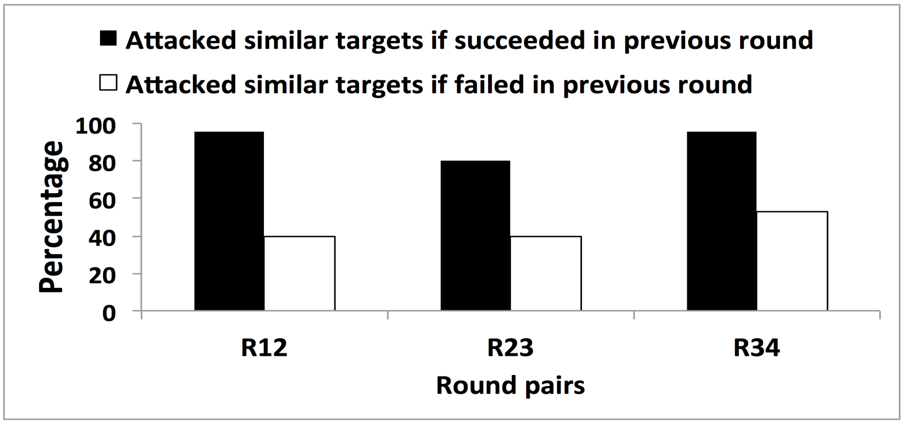


Figure 7.14: Evidemce for adaptivity of attackers (security experts in Indonesia)

First, in our human subjects experiments on AMT, we made Observation 2. We conducted analysis on the security experts data to see if we observe the same phenomenon in this data. Fig. 7.14 shows how the adversaries (security experts in this case) adapted to SHARP based strategy depending on past successes and failures. The x-axis denotes pairs of rounds for which we are computing the percentages; for example, in R23, 2 corresponds to round $(r-1)$ and 3 means round r in our claim. The results obtained are consistent with the ones obtained from our AMT data (see Figs. 6.1(a) - 6.1(h)), i.e., successful adversaries tend to return to attack the same or similar targets in the subsequent round while failed adversaries will not tend to return to attack the same or similar targets in the subsequent round.

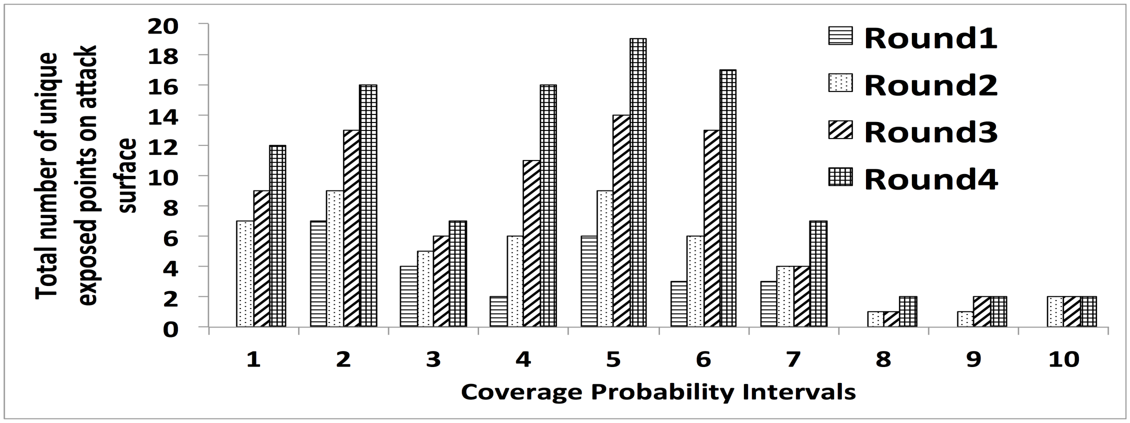


Figure 7.15: Total number of unique exposed target profiles till the end of each round for each coverage probability interval for Indonesia experts data.

Second, we conducted analysis to see how the attack surface is exposed to the adversary over various rounds. The amount of attack surface exposed to the adversary over the four rounds in the wildlife experts data is shown in Fig. 7.15. This is consistent with similar plots obtained from our AMT data (see Fig. 7.10(a)-7.10(b) and Fig. 13.8(a)-13.8(b)) which show that as rounds progress, more number of coverage probability values from various intervals are exposed to the adversary.

Third, we show in Fig. 7.5.1, the human perceptions of probability in rounds 1 to 4 when the security experts were exposed to SHARP based strategies on ADS_3. The learned curves have an S-shaped nature for each of the rounds, which is consistent with our AMT findings (Chapter 7.2).

7.5.2 Results with fraction of complete attack data

In our human subjects experiments, we assume that the defender can observe all the attacks that occurred in each target region of the park at the end of every round. However, this may not be true in reality, i.e., defenders may miss some large fraction of the attacks. Therefore, we conduct

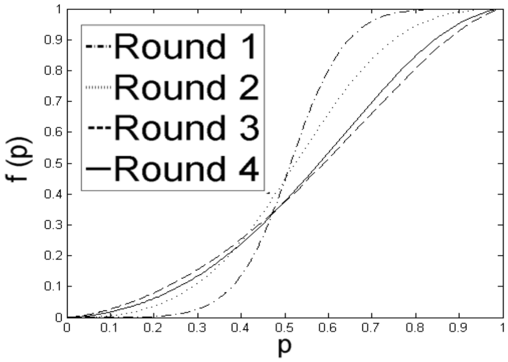
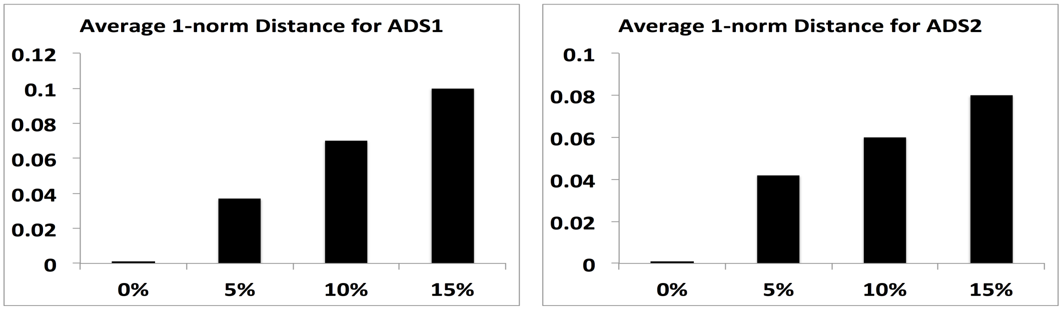


Figure 7.16: Learned probability curves for SHARP on ADS_3 on the security experts dataset.

additional analysis to understand the effects of considering a fraction of the original dataset on our defender strategy.

We generated round 2 defender strategies for all four payoffs with 50% of the data sampled randomly to test the robustness of our model. Here, by *robustness* we mean that the deviation of the strategy generated will be very similar to the original one, i.e., the 1-norm distance of the strategy generated with a fraction of the data will be very small when compared with the strategy generated with the full dataset. We randomly sampled several such fractional datasets but show results for four different sampled datasets (0%, 5%, 10% and 15% deviations from original attack data) for each payoff for the fraction size of 50%. By random sampling, we mean that, if there were $|\chi|$ attacks in the original dataset, we randomly picked a target cell and removed one attack data point and repeated this until 50% of the attack data (i.e. $round(|\chi|/2)$ attack data points) remained. Therefore, by 0% deviation, we mean that we removed 50% attacks from each target cell to make the new dataset. Similarly, by 5% deviation, we mean that the 1-norm distance between the new dataset obtained by removing 50% of the attack data and the original dataset is 0.05, and so on.

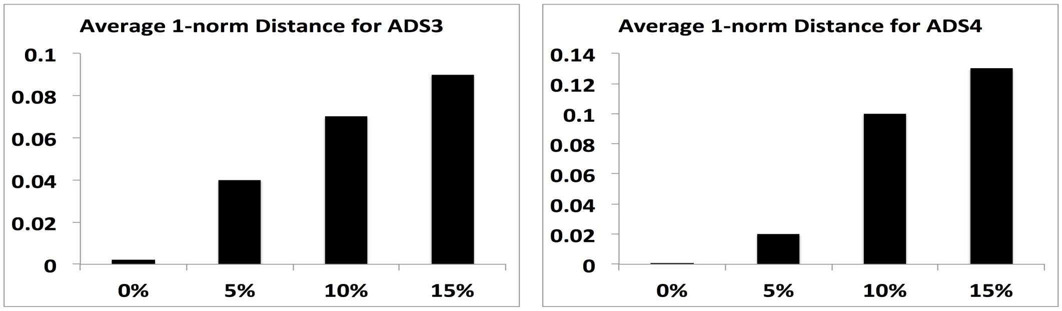
For each payoff structure we show (Figs. 7.17(a) and 7.18(b)) the average 1-norm distances between the round 2 defender strategies generated when datasets with various deviations (0%, 5%, 10% and 15%) from the original dataset were used to learn the model parameters, as opposed to learning from the complete dataset. We can observe from Figs. 7.17(a)-7.18(b) that the average 1-norm distance between the coverage probability $x_i; 0 \leq x_i \leq 1$ for any target i between the original and 5% deviation datasets is no more than 0.044 for any of the payoffs. However, when the deviation from the original dataset increases to 15%, the average 1-norm distance also increases. Note that if the proportion of attacks over the targets were same as that of the original dataset, then the defender strategy generated would also be exactly the same modulo rounding errors.



(a) Average 1-norm distance for ADS1

(b) Average 1-norm distance for ADS2

Figure 7.17: (a) and (b): Average 1-norm distances between defender strategies generated by SHARP when the model is learned based on randomly sampled 50% data (0%, 5%, 10% and 15% deviation from actual data) and when the model is learned from the complete data set. Results are shown for ADS_1 and ADS_2 respectively.



(a) Average 1-norm distance for ADS3

(b) Average 1-norm distance for ADS4

Figure 7.18: (a) and (b): Average 1-norm distances between defender strategies generated by SHARP when the model is learned based on randomly sampled 50% data (0%, 5%, 10% and 15% deviation from actual data) and when the model is learned from the complete data set. Results are shown for ADS_3 and ADS_4 respectively.

Chapter 8

INTERCEPT[1]

While the previous chapters discussed the fine-grained prediction problem with plentiful attack data, this chapter, as well as Chapter 9, focuses on the coarse-grained prediction problem with sparse attack data collected at the Queen Elizabeth National Park (QENP) in Uganda. We first discuss the dataset in Section 8.1, followed by a description of existing models (and its variants) that were proposed for the QENP dataset. Section 8.3 discusses in detail my modeling system called INTERCEPT.

8.1 Wildlife Crime Dataset

The following discussion is on wildlife crime data collected over 13 years at the Queen Elizabeth National Park (QENP) in Uganda. QENP (Figure 8.1) is a wildlife conservation area covering 1,978 square kilometers. Among their many duties, wildlife park rangers there conduct foot patrols to monitor wildlife habitat, apprehend any poachers sighted inside the park, and collect data on animal signs and signs of illegal human activity.

[1] This chapter is based on the paper (Kar et al., 2017a) where Benjamin Ford is a joint first author.

8.1.1 Dataset Challenges

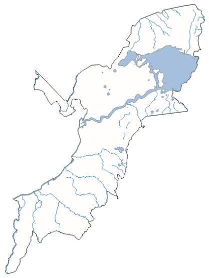


Figure 8.1: QENP

Because this is a real-world geospatial crime dataset, it is important to understand the inherent challenges in analyzing its contents, such as nonlinear relationships between features (Kanevski, Pozdnoukhov, & Timonin, 2008). Additionally, data can only be collected in areas that are patrolled, and even in the areas that are patrolled, poaching signs may remain undetected. This occurs because poaching signs (such as snares) are often well-hidden, and rangers may need to conduct a thorough patrol in order to detect any attack – an infeasible task to undertake for all targets all the time due to limited patrolling resources. This real-world constraint not only leads to uncertainty in the negative class labels (i.e., when poaching signs are not observed we are uncertain whether an attack actually happened at the corresponding target or not) but also results in a small number of positive samples being recorded in the dataset thus creating a huge class imbalance. As such, it is necessary to evaluate the attack prediction model's performance with metrics that account for this uncertainty, such as those for Positive and Unlabeled Learning (PU Learning) (Lee & Liu, 2003), and are discussed in more detail in the following sections.

8.1.2 Dataset Composition

The entire QENP area was discretized into 1 square kilometer grid cells (total 2,522 cells), each as a potential target of poaching. For each target, the ranger patrol effort level (i.e., coverage) and observed illegal human activity signs (e.g., poached animal carcasses, snares) were recorded. In addition, each target is associated with a non-static average ranger patrol effort value and a set of static features (that are constant throughout the entire time period): terrain features such as habitat (the terrain type and relative ease of travel) and terrain slope; distances to nearby roads, water bodies, patrol posts, and villages; and animal density.

For the following analysis, we examine poaching data from 2003-2015. We aim to find the targets that are liable to be attacked since predicting the attackability of targets can guide future patrols. We assume a target is attackable if an attack is ever observed at that target at any point in time. Therefore, when creating training sets, we combine observations from the entire training period for each target and label it as attackable if any observations were made.

Given the uncertainty in negative labels, there are bound to be training and testing samples that contradict one another. We consider a sample in the training set and a sample in the testing set to be contradictory when they have the same combination of static domain features values (e.g., terrain, distances, animal density) and non-static patrol coverage amount (i.e., low or high coverage) but different class labels (attacked or not attacked). These contradictions introduce additional noise in evaluating the performance of learned models and would thus cause any model to perform poorly on said contradictory data. As such, we remove these contradictions, about 10% of the data, from testing sets.

8.2 CAPTURE and Proposed Variants

The natural first step towards predicting future poaching attacks based on the real-world wildlife crime dataset was to use the best previous model proposed for this dataset, CAPTURE (Nguyen et al., 2016). CAPTURE was shown to have superior predictive performance to a number of other standard models in the behavioral game theory literature (e.g., Quantal Response (QR) (Yang et al., 2011), Subjective Utility Quantal Response (SUQR) (Nguyen et al., 2013)). Unfortunately, as we will discuss later, the number of attacks (and hence, the total number of successes and failures) recorded in the dataset was very low, even over 13 years, and SHARP could not be applied effectively on this data for comparison. Therefore, to make attackability predictions, we discretized the protected area into a set of targets I. Each target $i \in I$ has a set of domain-specific features $x_i \in x$ such as animal density d_i and distance to water. In a given time period t, a target i will be patrolled/covered by rangers with probability $c_{t,i}$.

CAPTURE consists of a two-layered behavior model. CAPTURE's first layer, the attackability layer, computes the probability that a poacher will attack a given target i at time step t. Similar to SUQR, which has been used to describe human players' stochastic choice of actions in security games, CAPTURE predicts attacks based on a linear combination of domain features $x_{t,i}$, ranger coverage probability $c_{t,i}$ at the current time step t, and whether the target was attacked in the previous time step $a_{t-1,i}$. With this last feature, $a_{t-1,i}$, CAPTURE models attacker behavior as being temporally dependent on past attacks.

$$p(a_{t,i} = 1 | a_{t-1,i}, c_{t,i}, x_{t,i}) = \frac{e^{\lambda^\intercal [a_{t-1,i}, c_{t,i}, x_{t,i}, 1]}}{1 + e^{\lambda^\intercal [a_{t-1,i}, c_{t,i}, x_{t,i}, 1]}} \tag{8.1}$$

λ is a parameter vector representing the importance of the features.

CAPTURE's second layer, the observation layer, computes the probability that rangers will observe an attack if poachers did attack that patrolled area based on a subset of domain features (e.g., habitat and slope) $\hat{x}_{t,i}$ and ranger coverage probability $c_{t,i}$.

$$p(o_{t,i} = 1 | a_{t,i} = 1, c_{t,i}, \hat{x}_{t,i}) = c_{t,i} \times \frac{e^{\omega^\intercal [\hat{x}_{t,i}, 1]}}{1 + e^{\omega^\intercal [\hat{x}_{t,i}, 1]}} \tag{8.2}$$

ω is a parameter vector that measures how domain features impact observation probability. The model parameters (λ, ω) that can maximize the likelihood of observations are estimated via the Expectation Maximization (EM) algorithm.

However, CAPTURE has a few limitations that lead to poor predictive performance in its *attackability layer*. First, CAPTURE's attackability predictions would sometimes predict too many targets to be attacked with a high probability (e.g., 80% of the targets will be attacked with almost 100% probability), leading to poor performance (see Section 9.2). One explanation is CAPTURE's parameter learning algorithm focuses on maximizing the performance of the observation layer rather than on the attackability layer. As the observation layer acts as a filter for the attackability layer, CAPTURE's learning process will converge to solutions that obtain decent performance for the observation layer even if the attackability layer's performance is poor.

Therefore, we propose several novel variants of CAPTURE as attempts to improve its predictions. In an attempt to restrict the degrees of freedom in the observation layer, and thus restrict the values the attackability layer can take in the learning process, we propose **CAPTURE-LB** which replaces the observation layer with a simpler observation layer, adapted from (Critchlow, Plumptre, Driciru, Rwetsiba, Stokes, Tumwesigye, Wanyama, & Beale, 2015), described as follows:

$$p(o_{t,i} = 1 | a_{t,i} = 1, c_{t,i}) = 1 - e^{-\beta \times c_{t,i}} \tag{8.3}$$

where $\beta \in [0, 1]$ is the parameter that estimates the detection efficiency. This not only provides a straightforward way of assessing the performance of patrol effort to observations but also has a smaller chance of overfitting, due to fewer parameters. For a given attack probability $p(a_{t,i} = 1)$, the unconditional probability of observing an attack at target i at time step t is given by:

$$p(o_{t,i}) = p(a_{t,i} = 1) \times p(o_{t,i} = 1 | a_{t,i} = 1, c_{t,i}) \tag{8.4}$$

Second, CAPTURE's attackability layer assumes that poachers plan attacks based on the patrol coverage in the current time step, which may not be realistic in the real world as the poachers may not get up-to-date information about the current patrol strategy and thus would rely on historical patrol coverage instead (Fang et al., 2015). Therefore, we propose another variant of CAPTURE, **CAPTURE-PCov**, that learns based on the previous time step's patrol coverage instead of the current time step's patrol coverage (Equation 8.5). Similarly, we propose **CAPTURE-PCov-LB**, a model that uses the attackability layer of CAPTURE with previous coverage as a feature but instead uses the LB observation layer defined in Equation 8.3.

$$p(a_{t,i} = 1 | a_{t-1,i}, c_{t-1,i}, x_{t,i}) = \frac{e^{\lambda^\intercal [a_{t-1,i}, c_{t-1,i}, x_{t,i}, 1]}}{1 + e^{\lambda^\intercal [a_{t-1,i}, c_{t-1,i}, x_{t,i}, 1]}} \tag{8.5}$$

Finally, CAPTURE's attackability predictions fail to take into account the domain knowledge that inaccessible and unattractive areas of the park will not be attacked with high probability, and I thus propose another variant **CAPTURE-DKHO**, which is the same as CAPTURE-PCov-LB except that it exponentially penalizes the attractiveness of inaccessible areas (Equation 8.6).

$$p(a_{t,i} = 1 | a_{t-1,i}, c_{t-1,i}, x_{t,i}) = \frac{e^{\lambda^\intercal [a_{t-1,i}, c_{t-1,i}, x'_{t,i}, 1]}}{1 + e^{\lambda^\intercal [a_{t-1,i}, c_{t-1,i}, x'_{t,i}, 1]}} \tag{8.6}$$

x' corresponds to the linear combination of features x but with the modified habitat feature $\sigma_i' = -\sigma_i e^{\sigma_i}$ which heavily penalizes high habitat values (i.e., hard to access areas).

8.3 INTERCEPT

The attempts of using the best previous model CAPTURE and the more complex variants of CAPTURE, proposed to address the limitations of CAPTURE, all suffered from poor attackability prediction performance as shown in Section 9.2. The natural progression then would have been to pursue more complex models in this behavioral game theory family of models with the expectation that they would improve performance on my real-world data. However, as reported in (Nguyen et al., 2016), complex models such as CAPTURE and its variants incur heavy computational costs; it takes approximately 6 hours for these models to complete execution. In addition, these models become more difficult to interpret when the dimensionality of the feature space increases (e.g., more numerical values to simultaneously account for in a single interpretation). We wanted to use models that would address all of these shortcomings by, not only significantly reducing computational costs so as to be usable by rangers with limited computing power in Uganda, but also remain interpretable to domain experts as the feature space dimensionality increases. All of these factors pointed against using more complex behavioral models. Therefore, we break from the current trend in behavior modeling in security games and model adversary behavior in terms of decision tree-based behavior models, even thoughwe were initially skeptical about its predictive capabilities. Surprisingly, this simpler approach led to significant improvements in performance over the prior state-of-the-art (i.e., CAPTURE).

Algorithm 2 BoostIT

```
$D^0 \leftarrow learnDecisionTree(\Theta^0)$
while Stopping Criteria Not Reached do
    $h^\Theta \leftarrow calcHotspotProximity(P_{\Theta^{m-1}}(D^{m-1}), \alpha)$
    $h^\Psi \leftarrow calcHotspotProximity(P_{\Psi^{m-1}}(D^{m-1}), \alpha)$
    $\Theta^m \leftarrow addFeature(\Theta^0, h_\Theta)$
    $\Psi^m \leftarrow addFeature(\Psi^0, h_\Psi)$
    $D^m \leftarrow learnDecisionTree(\Theta^m)$
    $m = m + 1$
end while
return $P$
```

8.3.1 BoostIT

A binary decision tree D is trained on a set Θ of independent variables x (the domain features), a dependent variable o (attack observations), and outputs a binary classification D_i for each target i: $\{$not attacked ($D_i = 0$), attacked ($D_i = 1$)$\}$. A decision tree's negative predictions for a test set Ψ are denoted by $P^-_\Psi(D)$ and positive predictions by $P^+_\Psi(D)$ (i.e., vectors of binary predictions).

Crime hot spots are part of a well-known theory in Criminology (Eck, Chainey, Cameron, & Wilson, 2005) that views crime as an uneven distribution; crime is likely to be concentrated in particular areas called hot spots. If a particular geographic area has a high concentration of predicted attacks, it is reasonable to interpret these predictions as a hot spot prediction (i.e., predicting a high concentration of crime). While CAPTURE explicitly models attacks as a probability distribution decided by a linear combination of feature values and thus can implicitly represent the hot spots with soft boundaries in the geographic space, decision trees' rules with hard boundaries in the feature space would lead to fine-grained segmentations in the geographic space and is thus less capable of representing hot spots. As such, we designed the **Boost**ed decision tree with an **It**erative learning algorithm (henceforth referred to as BoostIT) (Algorithm 2), where proximity to a predicted hot spot is encoded as an additional input feature.

D^0 is the initial decision tree learned without the hot spot proximity feature h, and Θ^0 and Ψ^0 correspond to the initial training and test sets, respectively. For each level of iteration m, a feature h^Θ (and h^Ψ) is computed for each target $i \in I$ that corresponds to whether that target is close to a predicted hot spot in the training (and test sets); for example, if a target $i \in P_{\Theta^{m-1}}(D^{m-1})$ is adjacent to α or more targets in $P^+_{\Theta^{m-1}}(D^{m-1})$ (i.e., targets that are predicted to be positive), then $h_i^\Theta = 1$. We then re-learn the decision tree at each iteration m with a feature augmented dataset Θ^m. As an example, BoostIT may add a feature to a target i that i is near a hot spot if there are two adjacent targets that are predicted to be attackable. In the next iteration, this new feature ("near a hot spot") will get used in learning about predicting attacks on i. This continues until an iteration criterion is reached. Note that the test set Ψ is not used while learning new decision trees (only training data Θ is used) and is only used to update the test set prediction P_Ψ. In the rest of the paper, we will refer to BoostIT with an α as BoostIT-αNearestNeighbors (or BoostIT-αNN). With this algorithm, the final decision tree D^m would generally predict more positive predictions with concentrated areas (i.e., hot spots) compared to D^0, but the set of predictions of D^m is not necessarily a superset of the set of predictions of D^0.

Although we are primarily interested in predicting attackability, we can also predict where patrollers would observe attacks by cascading attackability predictions with the LB observation layer (Equation 8.3). We convert the unconditional observation probability, derived from the cascaded model (Equation 8.4), to binary predictions by classifying samples as observed/not observed based on whether they are above or below the mean respectively.

8.3.2 INTERCEPT: Ensemble of Experts

We investigated the predictions of the traditional decision tree and BoostIT and observed that they are diverse in terms of their predictions. Here, by diversity, we mean that they predict attacks at a variety of targets. Therefore, while one model may fail to correctly classify a particular target as attacked, another model may succeed. This indicates the ability of different models to correctly learn and predict on different regions of the feature space. For example, let us consider the following three models: (i) DecisionTree, (ii) BoostIT-3NN and (iii) BoostIT-2NN. While computing pairwise disagreement between the models' attackability predictions, I observed that: (i) DecisionTree and BoostIT-3NN disagree on 105 out of 2211 target samples; (ii) DecisionTree and BoostIT-2NN disagree on 97 out of 2211 samples; and (iii) BoostIT-3NN and BoostIT-2NN disagree on 118 out of 2211 samples. This observation led us to consider combining the best decision tree and BoostIT based models, thus forming INTERCEPT–an ensemble of experts.

Because of uncertainty in negative labels, INTERCEPT considers not only decision tree models with the standard false positive (FP) cost of one, but also decision trees with various FP costs. For a decision tree with FP cost of 0.6, during the learning process, the decision tree will not receive the full penalty of 1 but will instead receive a penalty of 0.6 for each false positive prediction it makes.

In INTERCEPT, each expert model voted for the final attack prediction on a particular target. We considered three types of voting rules to determine whether a target should be predicted to be attacked by the ensemble: (a) majority of the experts predict an attack; (b) all experts predict an attack; and (c) any one expert predicts an attack. INTERCEPT uses the best voting rule: majority.

We considered ensembles with three and five experts. Having at most 5 experts makes the ensemble easily interpretable. In other words, the final prediction at a target is due to only 5 decision rules at a maximum, and it is easy to walk the human domain experts through the 5 rules in a way that the logic is easily verified.

Chapter 9

INTERCEPT Results

In this chapter, we will present results from our experiments with different models on the QENP wildlife poaching dataset and also results from ranger deployments at QENP based on our best performing model.

9.1 Evaluation Metrics

To evaluate INTERCEPT and other models, we first prepared two separate train/test splits on the dataset. For one dataset, we trained on data from 2003 to 2013 and evaluated our models on data in 2014, and for the other dataset, we trained on data from 2003 to 2014 and evaluated on data from 2015. Prior to discussing the evaluation results, we briefly discuss the metrics we use for computing our performance on predicting attackability and observed attacks.

Any metric to evaluate targets' *attackability* in domains such as wildlife poaching must account for the uncertainty in negative class labels. Therefore, in addition to standard metrics (Precision, Recall, and F1-score) that are used to evaluate models on datasets where there is no uncertainty in the underlying ground truth, we also evaluate our models with a metric that accounts for the uncertainty present in our dataset. The metric introduced in (Lee & Liu, 2003), henceforth referred to as L&L, is an appropriate metric since it is specifically designed for models learned on Positive and Unlabeled (PU) datasets (i.e., datasets with uncertain negative labels). L&L is defined in equation 9.1, where r denotes the recall and $Pr[f(Te) = 1]$ denotes the probability of a classifier f making a positive class label prediction. I compute $Pr[f(Te) = 1]$ as the percentage of positive predictions made by our model on a given test set.

$$L\&L(D, Te) = \frac{r^2}{Pr[f(Te) = 1]} \tag{9.1}$$

As we are certain about the positive samples in our dataset, L&L rewards a classifier more for correctly predicting where attacks have occurred (i.e., positive labels). However, it also prevents models from predicting attacks everywhere, via its denominator, and ensures that the model is selective in its positive predictions.

We also evaluate the models in terms of *observation* predictions. Here, we report standard metrics (Precision, Recall, and F1-score). We also compute the area under the Precision-Recall curve (PR-AUC). PR-AUC is a more appropriate metric for evaluating models on datasets with severe class imbalance (Davis & Goadrich, 2006) compared to area under the ROC curve. When there are many more negative points than positive points, the model can make many false positive predictions and the false positive rate would still be low, and thus, the ROC curve becomes less informative. In contrast, precision better captures how well the model is making correct positive predictions given a small number of positive examples. L&L is no longer used to evaluate the observation probability model as there is no uncertainty in terms of the observations, i.e., we either observed or did not observe an attack, and we are measuring the model's ability to predict whether we will observe attacks at already attacked targets.

9.2 Evaluation on Historical Real-world Patrol Data

To compare INTERCEPT with its competitors, we conducted a thorough investigation of the performance of 41 different models and 193 variants. Only a subset of the best performing ones are presented in this thesis. This is one of the largest evaluation efforts on a real-world dataset in the wildlife crime domain, and we compared INTERCEPT against the previous best model CAPTURE, its variants, and other machine learning approaches such as Support Vector Machines (SVM), AdaBoosted Decision Trees, and Logistic Regression[1]. All the numbers highlighted in **bold** in the tables indicate the results of the best performing models in that table. The best performing INTERCEPT system is an ensemble of five decision trees with majority voting. The five decision trees are: a standard decision tree, two BoostIT decision trees ($m = 1$) with $\alpha = 2$ and $\alpha = 3$ respectively, and two decision trees with modified false positive costs 0.6 and 0.9 respectively. Note that, due to data collection methodology changes in 2015, the distribution of attack data in 2015 is significantly different than all other previous years; 2015 is a difficult dataset to test on when the training dataset of 2003-2014 represents a different distribution of attack data, and we will demonstrate this impact in the following evaluation.

9.2.1 Attackability Prediction Results

In Tables 9.1 and 9.2, we show a comparison of the performance between our best INTERCEPT system (the five decision tree ensemble with majority voting), the current state-of-the-art CAPTURE, its variants, and other baseline models towards accurately predicting the attackability of targets in QENP for years 2014 and 2015, respectively. The PositiveBaseline corresponds to a

[1] Note that due to data confidentiality agreements, we are unable to show an example decision tree in this thesis.

Classifier Type	F1	L&L	Precision	Recall
PositiveBaseline	0.06	1	0.03	1
UniformRandom	0.05	0.51	0.03	0.50
CAPTURE	0.31	3.52	0.25	0.39
CAPTURE-PCov	0.13	1.29	0.08	0.48
CAPTURE-PCov-LB	0.08	0.87	0.04	0.58
CAPTURE-DKHO	0.10	1.05	0.06	0.67
INTERCEPT	**0.41**	**5.83**	0.37	0.45

Table 9.1: Attackability Prediction Results on 2014 Test Data

Classifier Type	F1	L&L	Precision	Recall
PositiveBaseline	0.14	1	0.07	1
UniformRandom	0.19	0.50	0.11	0.50
CAPTURE	0.21	1.08	0.13	0.63
CAPTURE-PCov	0.19	0.87	0.11	0.57
CAPTURE-PCov-LB	0.18	0.69	0.11	0.46
CAPTURE-DKHO	0.20	0.71	0.12	0.5
INTERCEPT	**0.49**	**3.46**	0.63	0.41

Table 9.2: Attackability Prediction Results on 2015 Test Data

model that predicts every target to be attacked ($p(a_{t,i}) = 1; \forall i, t$), and the UniformRandom corresponds to the baseline where each target is predicted to be attacked or not attacked with equal probability. Note that, in this subsection, when evaluating two-layered models such as CAPTURE and its variants, we are examining the performance of just the attackability layer output, and we defer the evaluation of the observation predictions to Section 9.2.2. Since we evaluate the attackability predictions of our models on metrics for binary classification, the real-valued output of the attackability layer of CAPTURE and its variants were converted to a binary classification where probabilities greater than or equal to the mean attack probability were classified as positive.

We make the following observations from these tables: First, INTERCEPT completely outperforms the previous best model CAPTURE and its variants, as well as other baseline models in terms of L&L and F1 scores. For 2014, INTERCEPT outperforms CAPTURE in terms of precision, recall, F1, and L&L score. For 2015 test data, INTERCEPT represents an even larger performance increase by approximately 3.50 times (L&L score of 3.46 vs 1.08) over CAPTURE and even more so for CAPTURE-PCov (L&L score of 3.46 vs 0.87). CAPTURE-PCov doesn't even outperform the positive baseline. Second, CAPTURE performs better on the 2014 dataset (when the training and testing data were similarly distributed) than on the 2015 dataset. In contrast, INTERCEPT remained flexible enough to perform well on the difficult 2015 testing set. However, CAPTURE-PCov, the more realistic variant of CAPTURE that can actually be used for forecasting, fails to make meaningful predictions about the attackability of targets. Its similar performance to PositiveBaseline demonstrates the need for models to learn the attackability of targets independently of observation probability to avoid learning models that make incorrect inferences about the attackability of the park (e.g., the entire park can be attacked). This is particularly important in the wildlife poaching domain because, due to the limited number of security resources, rangers cannot patrol every target all the time. Therefore, the attack probability model's predictions need to be extremely precise (high precision) while also being useful indicators of poaching activities throughout the park (high recall). Third, CAPTURE-PCov-LB performs even worse than CAPTURE-PCov in terms of L&L score for these attackability predictions, although the only difference between the two models is the observation layer. This occurs because the attackability prediction layer and the observation layer are not independent of one another; with the EM algorithm, the parameters are being learned for both layers simultaneously. In addition, by incorporating domain knowledge and penalizing the unattractive areas, CAPTURE-DKHO unfortunately does not lead to a significant improvement in performance. Fourth, INTERCEPT's precision values are significantly better compared to CAPTURE-PCov in 2014 and both CAPTURE and CAPTURE-PCov in 2015 with only modest losses of recall, indicating a significant reduction in the number of false positive predictions made throughout the park.

Classifier Type	F1	L&L	Precision	Recall
Weighted DecisionTree	0.11	1.01	0.06	0.48
SVM-BestFPCost-0.3	0.13	1.18	0.46	0.45
Logistic Regression	-	-	-	0
AdaBoostDecisionTree-BestFPCost-0.2	0.13	1.22	0.07	0.48
INTERCEPT	**0.41**	**5.83**	0.37	0.45

Table 9.3: Additional Attackability Prediction Results on 2014 Test Data

Classifier Type	F1	L&L	Precision	Recall
Weighted DecisionTree	0.25	1.42	0.15	0.69
SVM-BestFPCost-0.25	0.19	0.72	0.12	0.43
Logistic Regression	-	-	-	0
AdaBoost-DT-BestFPCost-0.15	0.21	0.86	0.13	0.49
INTERCEPT	**0.49**	**3.46**	0.63	0.41

Table 9.4: Additional Attackability Prediction Results on 2015 Test Data

In Tables 9.3 and 9.4, we also compare INTERCEPT with other models including: (i) a decision tree where each sample was weighted based on the patrol intensity for the corresponding target (Weighted Decision Tree); (ii) the best performing SVM; (iii) Logistic Regression (which predicted no attacks and thus metrics could not be computed); and (iv) the best performing AdaBoosted Decision Tree. INTERCEPT provides significantly better performance than these other models as well.

9.2.2 Observation Prediction Results

Tables 9.5 and 9.6 correspond to how accurately each model predicted the observations in our test datasets. For a fair comparison, we also cascade the attackability predictions of the PositiveBaseline and UniformRandom baselines with an LB observation layer, and convert those unconditional observation probabilities to binary predictions with a mean threshold, as was done for CAPTURE's attackability predictions. We observe the following. First, incorporating the observation model in Equation 8.4 improved the PR-AUC score of CAPTURE in both test datasets (for

Classifier Type	F1	Precision	Recall	PR-AUC
PositiveBaseline	0.13	0.07	0.79	0.12
UniformRandom	0.09	0.05	0.46	0.07
CAPTURE	0.14	0.08	0.73	0.33
CAPTURE-PCov	0.12	0.07	0.61	0.31
CAPTURE-PCov-LB	0.13	0.08	0.48	0.36
CAPTURE-DKHO	0.16	0.09	0.72	0.33
INTERCEPT	**0.36**	0.32	0.89	**0.45**

Table 9.5: Observation Prediction Results on 2014 Test Data

Classifier Type	F1	Precision	Recall	PR-AUC
PositiveBaseline	0.26	0.16	0.66	0.20
UniformRandom	0.19	0.12	0.45	0.14
CAPTURE	0.29	0.18	0.70	0.29
CAPTURE-PCov	0.29	0.18	0.70	0.29
CAPTURE-PCov-LB	0.34	0.21	0.85	0.32
CAPTURE-DKHO	0.36	0.24	0.79	0.32
INTERCEPT	**0.50**	0.65	0.41	**0.49**

Table 9.6: Observation Prediction Results on 2015 Test Data

2014, 0.36 vs 0.33; for 2015, 0.32 vs 0.29). Second, INTERCEPT outperforms the other models by a large margin, both in terms of F1 and PR-AUC, for both test datasets. Combined with the attackability results, these results demonstrate the benefit of learning more precise attackability models in order to better predict observation probability.

9.2.3 Impact of Ensemble and Voting Rules

INTERCEPT consists of five experts with a majority voting rule. We now investigate the impact of combining different decision trees into an ensemble, and the impact of different voting rules. Tables 9.7 and 9.8 show that constructing an ensemble, INTERCEPT, significantly improves the performance of the system as a whole, compared to the performance of its individual decision tree and BoostIT members. The standard decision tree is more conservative as it predicts less false positives, leading to higher precision, but suffers from low recall.

Table 9.9 shows the impact that a voting rule has on performance on 2015 test data (2014 test data results are omitted as it exhibits the same trends). We evaluate the performances of the best ensemble compositions, with three and five experts for each voting rule. We observe that: (i) Ensembles which predict an attack if any one expert predicts an attack (*Any*) are significantly better in terms of recall (0.68), but do poorly in terms of precision (0.23). This is because such ensembles are more generous in terms of predicting an attack, and this leads to a significantly higher number of false positives; (ii) Ensembles with a voting rule where all experts have to agree (*All*) perform worse in terms of recall (0.16), but do best in terms of precision (0.89) as it makes less positive predictions (both true positives as well as false positives). This would mean that it would miss a lot of attacks in our domain, however; (iii) The majority voting based ensembles (*Maj*), used by INTERCEPT, provide an important balance between precision (0.63) and recall (0.41) as they are neither extremely conservative nor generous in terms of their predictions and therefore outperform other voting rules significantly (L&L of 3.46).

This analysis provides important guidance for selecting ensembles depending on the requirements of the domain. For example, if it is extremely crucial to predict as many true positives as possible and a high number of false positives is acceptable, then using an *Any* voting method would be beneficial. However, in our wildlife poaching prediction problem, we have limited security resources and therefore cannot send patrols to every target all the time. Therefore, we not only wish to limit the number of false positives but also increase the number of correct poaching predictions. The majority voting rule provides this important balance in our domain.

Classifier Type	F1	L&L	Precision	Recall
PositiveBaseline	0.06	1	0.03	1
DecisionTree	0.2	1.8	0.14	0.36
BoostIT-1NN	0.19	2.23	0.12	0.55
BoostIT-2NN	0.21	2.13	0.13	0.45
BoostIT-3NN	0.2	2.01	0.13	0.45
INTERCEPT	**0.41**	**5.83**	0.37	0.45

Table 9.7: Attackability Prediction Results For Decision Tree Models on 2014 Test Data

Classifier Type	F1	L&L	Precision	Recall
PositiveBaseline	0.14	1	0.07	1
DecisionTree	0.39	2.01	0.39	0.38
BoostIT-1NN	0.39	2.16	0.32	0.50
BoostIT-2NN	0.37	2.00	0.30	0.50
BoostIT-3NN	0.42	2.45	0.35	0.52
INTERCEPT	**0.49**	**3.46**	0.63	0.41

Table 9.8: Attackability Prediction Results For Decision Tree Models on 2015 Test Data

9.3 Evaluation on Real-World Deployment

INTERCEPT represents a paradigm shift from complex logit-based models such as CAPTURE (Nguyen et al., 2016), and many others, to decision tree-based models. During development, we worked with a domain expert from the Wildlife Conservation Society to improve and validate our decision tree models and their corresponding predictions. Indeed, one advantage of shifting to a decision tree-based approach (as opposed to methods like CAPTURE) is that the underlying rules can be easily expressed to experts in non-AI fields.

After this development and evaluation on historical data was completed, we deployed INTERCEPT to the field. Based on INTERCEPT's predictions, we chose two patrol areas for QENP rangers to patrol for one month. We selected these areas (approximately 9 square km each) such that they (1) were predicted to have multiple attacks but did not have any attack observations in the past and (2) were previously infrequently patrolled as rangers did not previously consider these as important as other areas (and thus are good areas to test our predictions). Selecting areas that our model deemed attackable but where no past observations exist was important for our field

Classifier Type	F1	L&L	Precision	Recall
BoostIT-3Experts-Any	0.36	2.11	0.26	0.59
BoostIT-5Experts-Any	0.34	2.13	0.23	0.68
BoostIT-3Experts-All	0.36	2.68	0.88	0.22
BoostIT-5Experts-All	0.28	1.97	0.89	0.16
BoostIT-3Experts-Maj	**0.49**	3.34	0.58	0.43
INTERCEPT	**0.49**	**3.46**	0.63	0.41

Table 9.9: Attackability Prediction Results For Different Ensembles on 2015 Test Data

test to demonstrate that a naive approach which only focuses on patrolling areas where at least one observation has been made in the past will not be able to perform as well as our learning model which has the capability to generalize across the entire park. After providing the rangers with GPS coordinates of particular points in these areas, they patrolled these areas on foot and utilized their expert knowledge to determine where exactly in these areas they were most likely to find snares and other signs of illegal human activity (e.g., salt licks, watering holes). On each patrol, in addition to their other duties, rangers recorded their observations of animal sightings (i.e., 21 animals were sighted in one month) and illegal human activity.

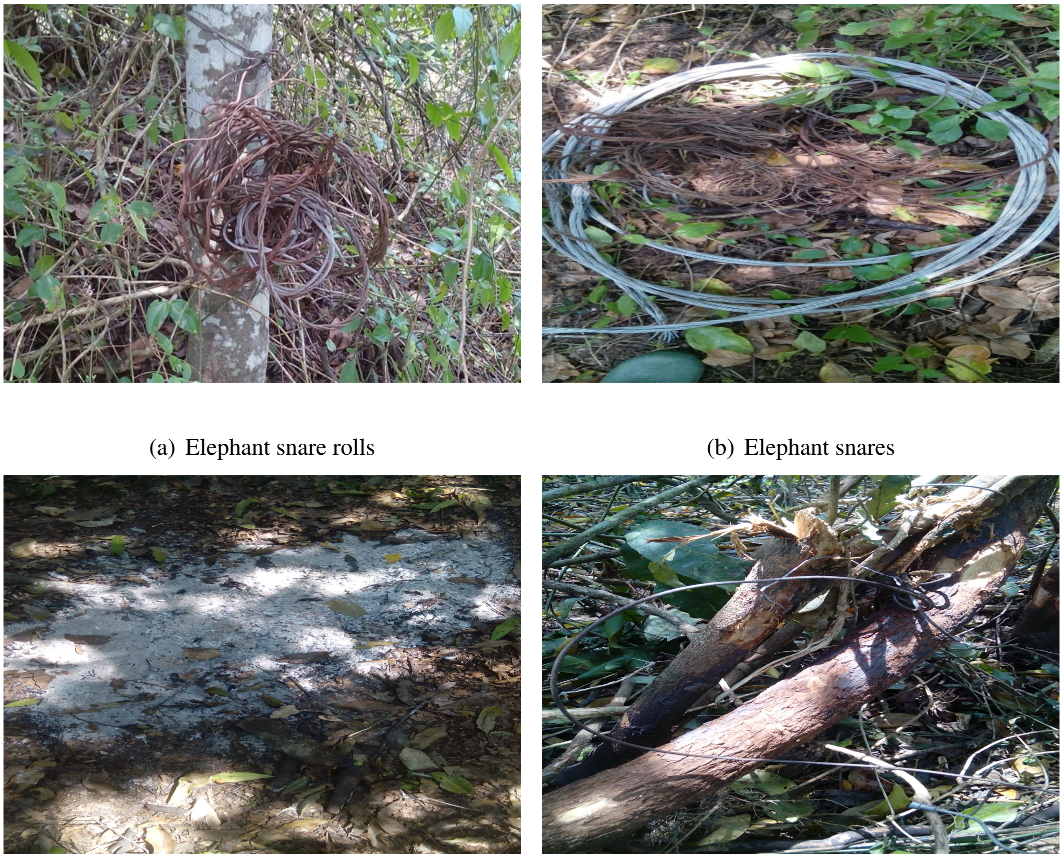
We now present our key findings in Tables 9.10 and 9.11 and provide a selection of photos in Figures 9.1(a)–9.1(d). The most noteworthy findings of these patrols are those related to elephant poaching; rangers, unfortunately, found one poached elephant with its tusks removed. However, this result demonstrates that poachers find this area, predicted by our model, attractive for poaching. On a more positive note, our model's predictions led rangers to find many snares before they caught any animals: one large roll of elephant snares, one active wire snare, and one cache of ten antelope snares. INTERCEPT's predictions assisted rangers' efforts in potentially saving the lives of *multiple animals including elephants*.

In addition to wildlife signs, which represent areas of interest to poachers, the findings of trespassing (e.g., litter, ashes) are significant as these represent areas of the park where humans

Week#	Illegal Activity	Count
2	Trespassing	19
3	Active Snares	1
	Plant Harvesting	1
4	Poached Elephants	1
	Elephant Snare Roll	1
	Antelope Snares	10
	Fish Roasting Racks	2

Table 9.10: Real World Patrol Results: Illegal Activity

were able to enter illegally and leave without being detected; if we can continue to patrol areas where poachers are visiting, rangers will eventually encounter the poachers themselves.



(a) Elephant snare rolls

(b) Elephant snares

(c) Illegal campfire ashes

(d) Antelope snare rolls

Figure 9.1: Illegal activities detected by rangers directed by INTERCEPT. Photo credit: Uganda Wildlife Authority ranger

Crime Type	**INTERCEPT**	Average	Percentile
AnimalCom	**1**	0.16	89%
AnimalNoncom	**3**	0.73	91%
Fishing	**1**	0.73	79%
PlantNoncom	**1**	0.46	76%
Trespassing	**19**	0.20	100%
Total	**25**	2.28	

Table 9.11: Base Rate Comparison: Hits per Month

So as to provide additional context for these results, we present a set of base rates in Table 9.11. These base rates, computed in and around our proposed patrol areas, correspond to the average number of observed crimes per month from 2003-2015. Animal commercial (AnimalCom) crimes correspond to elephant, buffalo, and hippopotamus poaching; animal noncommercial (AnimalNoncom) corresponds to all other poaching and poaching via snares; and plant noncommercial (PlantNoncom) corresponds to illegal harvesting of non-timber forest products (e.g., honey). The percentile rank corresponds to the number of months where our deployed patrols recorded more observations than in the historical data. For animal noncommercial crime, there was an average of 0.73 attacks observed monthly; for our deployed patrols, there were 3 separate observations (such as a roll of elephant snares), and in 91% of the months from 2003-2015, 2 or fewer observations were recorded.

9.4 Lessons Learned

After our extensive modifications to the CAPTURE model and our subsequent evaluation, it is important to identify the reasons why we obtained such a surprising result: decision trees outperformed a complex, domain-specific temporal model. (1) The amount of data and its quality need to be taken into consideration when developing a model. The QENP dataset had significant noise (e.g., imperfect observations) and extreme class imbalance. As such, attempting to develop a complex model for such a dataset can backfire when there does not exist sufficient data to support it. Our decision tree approach, generally regarded as simpler, benefits from being able to express complex relationships with limited noisy data. SVMs, also able to express non-linear relationships, appear to fail due to their complexity and attempt to define very fine-grained divisions of the dataset. (2) Model interpretability and speed are a necessity when working in the real-world. Our decision tree model was deployed because, not only did it have superior performance to CAPTURE, but it was also easy to directly look at the rules the decision tree had learned and evaluate whether or not those rules were reasonable (according to a domain expert). (3) The tradeoff between interpretability and performance, studied in domains where interpretability is key (e.g., biopharmaceutical classification) (Johansson, Sönströd, Norinder, & Boström, 2011), may not always exist. Indeed, the most interpretable and the fastest executable model, out of all that we evaluated, was also the best performing (by a large margin!); future research should (i) not always forego interpretability and speed in favor of performance under the assumption that there is always a tradeoff but (ii) instead be sure to investigate simpler models in case there isn't a tradeoff.

Chapter 10

Belief Modeling

As highlighted in the introduction (Chapter 1), a major drawback of existing adversary behavior models is the assumption that adversaries perfectly observe the defender's mixed strategy and acts based on that. In domains such as GSGs, the adversary only observes few pure strategies sampled from the defender's mixed strategy and reasons based on that. This chapter first presents a belief modeling game that we developed to collect belief data in our game settings. The subsequent sections constitute of three different settings with respect to the amount and type of data available. Each of these settings lead to the development of different models that exploit the situations in different ways to make more effective predictions based on the amount and type of data available.

10.1 Belief Modeling Game

We conducted human subjects experiments on AMT to collect data about how humans update their beliefs about the defender's mixed strategy while acting as adversaries based on their observations about the defender. Each observation is a pure strategy sampled from one of four different defense strategies implemented by the defender (discussed later). Below is an overview of our

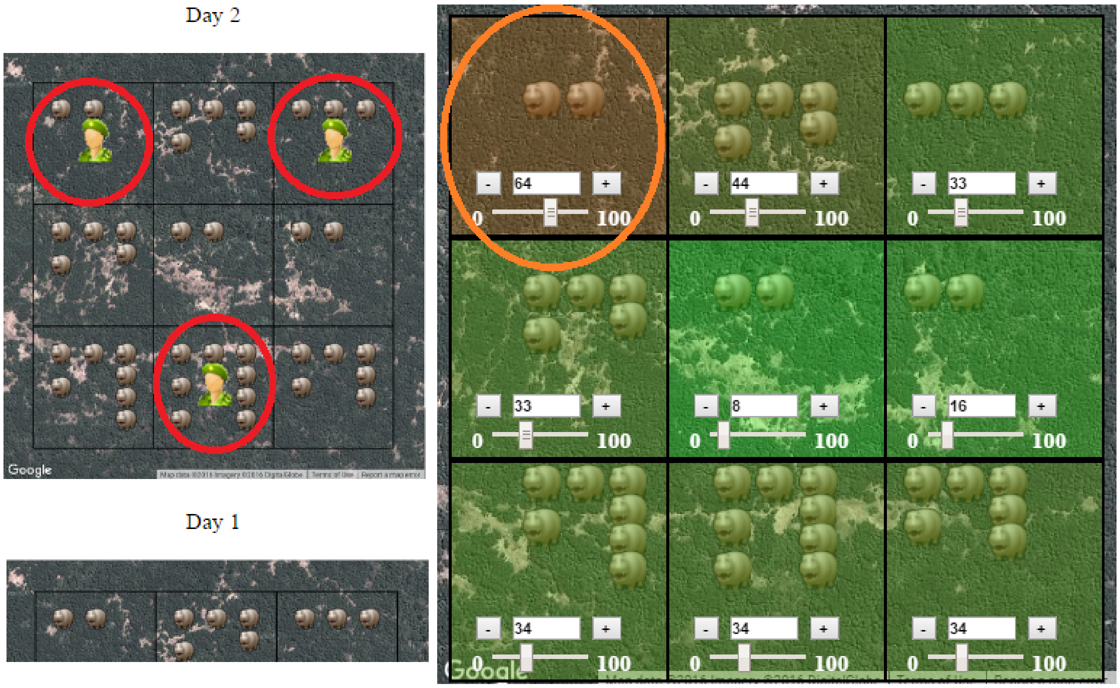


Figure 10.1: Game Interface for simulated online belief modeling game

experimental game, the payoff structures and defender strategies used and the model categories tested.

10.1.1 Game Overview

In our game, human subjects play the role of poachers (a type of adversary) who are trying to estimate the defender's mixed strategy by observing 10 consecutive pure strategies sampled independently from the corresponding defender mixed strategy. Each pure strategy corresponds to the strategy used by the defender on one particular day for patrolling the protected park area. At the end of each day, the participants were required to enter their beliefs about the defender's mixed strategy based on their pure strategy observations till the current day. The game interface is shown in Fig. 10.1.

In our game, the Google maps view of the portion of the park shown in the interface is divided into a 3*3 grid, i.e. 9 distinct target cells. Overlaid on this map to the right of the interface is a heat-map which represents the participants' current belief about the rangers' mixed strategy x — a cell i where the participant believes that a defender has higher coverage probability x_i is shown more in red, while a cell with lower coverage probability is shown more in green. The participants can use the sliders, text boxes and +/- buttons to enter their beliefs about the percentage likelihood of a ranger being present in each cell and this change will be reflected by the color of that cell. As the subjects play the game, they are given information about the presence/absence of a ranger for each target i for each day as shown by the map in the left of the game interface. In Fig. 10.1, you can observe the defender's pure strategy for Day 2 in the map on the left (three rangers are circled) and in the right map the participant is currently entering his/her beliefs (64% coverage on top leftmost target) about the defender's strategy after having observed two days of defender patrols. The participant can check all the previous days' patrols (pure strategies) by scrolling down in the left side of the interface before entering their beliefs. In our game, $M = 3$ rangers were protecting 3 out of 9 grid cells in the park. So, for any day, only 3 out of the 9 targets are shown to be protected in the per day maps shown in the left of the interface.

As mentioned earlier, the pure strategies shown to the left were drawn independently from a particular mixed strategy x used by the defender. This is the mixed strategy that the participants were asked to estimate based on the pure strategy observations. This setting simulates a real-world situation where poachers have knowledge of previous ranger deployments in terms of their exact locations per day and they are tasked to form beliefs about the actual mixed strategy based on these observations. In this paper, we are only interested in modeling the belief formation and update procedures in such scenarios and hence only collect data about their beliefs and do *not* ask them to choose a target to attack after any day of play in the game. This would be an interesting direction for future work.

10.1.2 Experimental Procedure

After an introduction to the game setting, the participants had to answer two validation questions which tested their understanding of the game, and were allowed to proceed to a trial and then the actual game if they answered them correctly. In the actual game, one of four mixed strategies was randomly selected for each participant to eliminate any bias and he/she was shown the 10 pure strategies sampled from the chosen mixed strategy.

Payment Scheme: We set up the payment scheme to not only reward participation but also to incentivize truthful reporting of the participants' beliefs. Specifically, each participant was paid a 'base compensation' of $0.50\$$ for participation. To motivate the participants to enter their beliefs accurately after each day, we gave them an incentive called 'performance bonus', based on the difference between the entered beliefs after each day and the actual mixed strategy from which the pure strategies were sampled. For a particular day, the maximum amount (M) a participant can earn is $0.30\$$, i.e., when their belief estimation is identical to the actual mixed strategy. Let the deviation between the entered strategy for any day i and the actual mixed strategy be d_i. Also, D represents the maximum possible deviation from the actual mixed strategy. Then the performance bonus for the belief entered for day$i$ is $M - (d_i/D) * M$. The total reward was the sum of their performance bonuses and base compensation.

Payoff Structures: We randomly generated two game boards showing how animals are spread out across the 9 targets, which determines the payoff structure for the game. We henceforth refer to payoff structures and animal density structures interchangeably in this paper. The

2	5	3
5	2	2
7	8	6

(a) ADS_1

8	3	5
10	4	2
4	2	2

(b) ADS_2

0	49	15
49	0	0
64	68	58

(c) Maximin

0	41	0
41	0	0
66	100	52

(d) SUQR

Figure 10.2: (a,b): Animal Densities; (c) Maximin; (d) SUQR

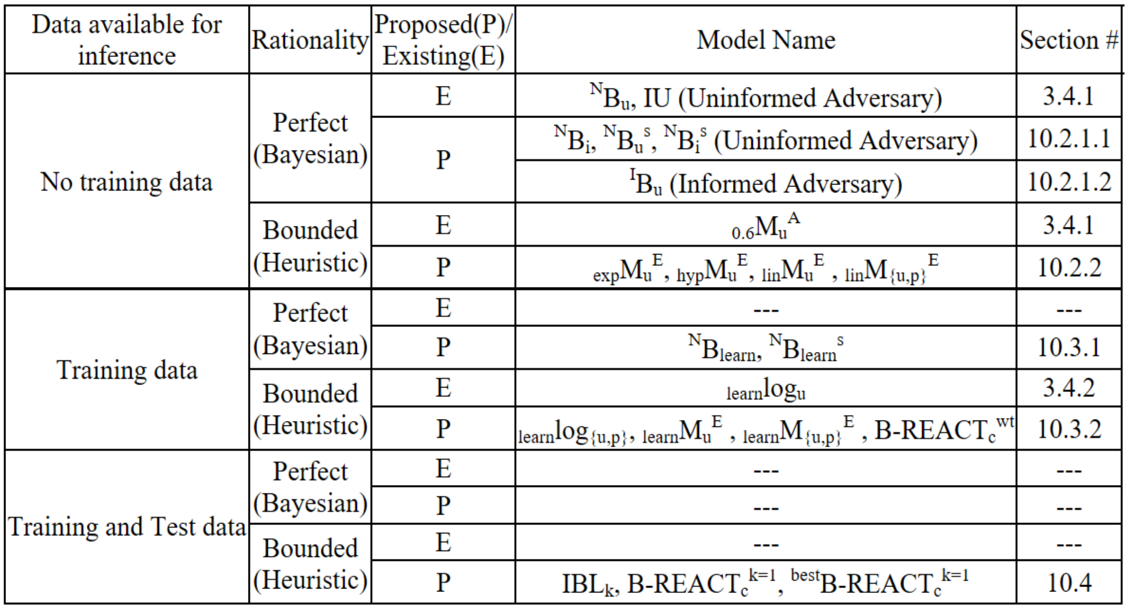
total number of animals on the board is constant across games (= 40). Figs. 10.2(a)–10.2(b) show animal densities used; they are referred to as ADS_1 and ADS_2 respectively in the paper.

Defender Strategies: We experimented with four different defender strategies to test how humans form and update their beliefs when faced with different strategies. These are: (i) Maximin, (ii) Proportional, (iii) SUQR, and (iv) Uniform. Maximin and SUQR strategies for ADS_1 are shown in Figs. 10.2(c) – 10.2(d). As explained in Chapter 2, while Maximin is a robust game-theoretic strategy, an SUQR based strategy is generated by assuming a model of the human adversary learned from prior data. SUQR strategy was computed based on learned weights in the SUQR model as reported in (Nguyen et al., 2013) from a previous human subjects experiment in security games. Proportional strategy puts coverage probabilities on targets in proportion to the number of animals in that target. In a Uniform strategy, each target is covered with equal probability by the defender. Since three defender resources were protecting 9 targets, sum of the coverages (in terms of percentages) is $\leq$ 300. Coverages and adversary's beliefs about the coverages could be computed in terms of either probabilities or percentages.

We deployed our game on AMT and collected data for 191 and 160 participants for ADS_1 and ADS_2 respectively. Since each participant was randomly allocated to a condition corresponding to one of the four mixed strategies, the number of participants for each condition in the resulting data set varies. In the experiments with ADS_1, Maximin, Proportional, SUQR and Uniform strategies were played by 35, 55, 44 and 57 participants respectively. We divided each of these four groups of participants randomly into 10 train-test splits with 70% of the participants in the training data and remaining 30% from the same split in the test data. Training data (whenever used) is for learning our models. We will make belief predictions for participants in the test sets. Non-learning models were evaluated on the same test sets as the learning models to enable fair comparison.

Models Tested: The literature on belief modeling can be broadly categorized as: (a) Bayesian updating models; (b) Heuristic belief updating; (c) Bayesian Theory of Mind (BTOM); and (d) Level-k models. Here we will provide a description of models that fall in categories (a) and (b) only, as these were earlier shown to be the best performing models in the SSG literature and other related fields (e.g., psychology). We have extensively experimented with such models and we present those results in Chapter 11. Models that belong to categories (c) and (d) will not be presented in this thesis because: BTOM models ((Baker, Saxe, & Tenenbaum, 2011)) use POMDPs to model beliefs, and are therefore not easily applicable in our setting due to infinite state space (all possible mixed strategies); and Level-k models ((Wright & Leyton-Brown, 2014)) have only been used to predict actions in simultaneous-move games and it is non-trivial to adapt to our belief updating setting in repeated SSGs.

Earlier work on belief modeling in categories (a) and (b) can be broadly classified into two types based on the assumption about the amount of information available. First is the case when no prior data is available to learn about the belief formation and update process of human agents in a given situation. This is what has been used in SSGs. Second is the scenario when historical belief update data for a group of human agents is available (training set). This facilitates learning a generalized model of human belief formation and update, and apply the learned model to predict



Data available for inference	Rationality	Proposed(P)/ Existing(E)	Model Name	Section #
No training data	Perfect (Bayesian)	E	${}^{N}B_u$, IU (Uninformed Adversary)	3.4.1
		P	${}^{N}B_i$, ${}^{N}B_u^{s}$, ${}^{N}B_i^{s}$ (Uninformed Adversary)	10.2.1.1
			${}^{I}B_u$ (Informed Adversary)	10.2.1.2
	Bounded (Heuristic)	E	${}_{0.6}M_u^{A}$	3.4.1
		P	${}_{exp}M_u^{E}$, ${}_{hyp}M_u^{E}$, ${}_{lin}M_u^{E}$, ${}_{lin}M_{\{u,p\}}^{E}$	10.2.2
Training data	Perfect (Bayesian)	E	---	---
		P	${}^{N}B_{learn}$, ${}^{N}B_{learn}^{s}$	10.3.1
	Bounded (Heuristic)	E	${}_{learn}\log_u$	3.4.2
		P	${}_{learn}\log_{\{u,p\}}$, ${}_{learn}M_u^{E}$, ${}_{learn}M_{\{u,p\}}^{E}$, B-REACT$_c^{wt}$	10.3.2
Training and Test data	Perfect (Bayesian)	E	---	---
		P	---	---
	Bounded (Heuristic)	E	---	---
		P	IBL$_k$, B-REACT$_c^{k=1}$, ^{best}B-REACT$_c^{k=1}$	10.4

Figure 10.3: Model names and assumptions

belief updates for a previously unknown set of human agents (testing set). The assumption about having access to training data is common in the psychology literature and we adapt one popular model from that literature to SSGs. In this thesis, we will also discuss another setting where, in addition to the training data about a group of participants, we will use information about the previously unseen (test set) participants' past beliefs (when available) to predict their future beliefs. Fig. 10.3 provides a summary of the models presented in this paper along with the corresponding assumptions.

10.2 Proposed Models: Setting without training data

For this setting where we have no training data, we first applied previous existing models (Section 3.4.1) and observed that these models perform poorly in terms of predicting beliefs of the test set adversaries. Therefore, we developed new models assuming both perfectly rational and boundedly rational adversaries that improve the state-of-the-art by providing new methods for (a) prior initialization and (b) the updating scheme. Performance results for all models proposed in this section are reported in Section 11.1.

10.2.1 Perfectly Rational Adversary

In this section, we consider two scenarios for modeling perfectly rational adversaries that make different assumptions about the amount of information the adversary may have about the strategies the defender is employing. In the first setting, we assume the adversary knows nothing about the possible set of defender strategies. In the second setting, the adversary knows a set of candidate strategies of size $|\Theta|$ (=4 in our case) the defender may employ but does not know which strategy among this set the defender chooses to implement. In my experiments, this candidate set is composed of Maximin, SUQR, Uniform and Proportional strategies. The motivation for the second scenario is that there may be an inside informant on the defender side who has secretly revealed this information to the adversary, and therefore we were interested in investigating the performance of a belief prediction model that accounts for this.

10.2.1.1 Uninformed Adversary

In the existing belief update model for a perfectly rational adversary in an SSG setting (Section 3.4.1), the adversary has no information about the types of strategies the defender may deploy. In their model, (An et al., 2012) further assume that the adversary (a) starts from a uniform Dirichlet prior and (b) only updates the prior corresponding to the observed pure strategies. We relax these assumptions and improve the existing approach by proposing an informative Dirichlet prior based on domain features and a similarity based updating mechanism.

Informative Dirichlet Prior: We hypothesize that instead of starting from a uniform Dirichlet prior (see Sec. 3.4.1) the adversary may start with an informative Dirichlet prior based on the features of the domain. Intuitively, in our game, since animal density is the most important factor in determining defender allocations in the wildlife crime domain, we hypothesize that this would be an important feature that would influence the adversary's prior beliefs about the defender's mixed strategy. Therefore, we compute an informative Dirichlet prior which puts prior values on each pure strategy in proportion to the sum of the animal densities at the targets protected by that pure strategy. We refer to this model as ${}^{N}B_i$, where B will henceforth represent Bayesian models, i denotes informative prior, and N indicates that these models correspond to the case of an non-informed adversary. The existing model (Section 3.4.1) will be referred to as ${}^{N}B_u$, where u stands for uniform prior.

Updating Method: The intuition behind our novel updating method is generalizing the observations about pure strategies employed by the defender to other, similar pure strategies so that a more informed updated belief can be generated even after making limited pure strategy observations. In this work, we assume that two pure strategies are similar if they differ in terms of defender allocation in only one of the three protected targets. For example, in Eqn. 3.2, if a pure strategy $j \in \mathcal{P}$ is observed three times in 5 days, then not only is $\alpha_j + o_j^5 = 3$, but also $\alpha_k + o_k^5 = 3$ for pure strategy k which was never observed during the 5-day timeframe but is similar to pure strategy j. The uniform and informative Dirichlet prior models with similar pure strategy updating will henceforth be referred to as ${}^{N}B_u^s$ and ${}^{N}B_i^s$ respectively, where s denotes the similarity based updating procedure.

10.2.1.2 Informed Adversary

Let us denote the set of mixed strategies that the defender chooses from as $\Theta =< \theta_1, \theta_2, ..., \theta_{|\Theta|} >$. For the case where the adversary has complete knowledge that the defender is deploying one of these $|\Theta|$ different mixed strategies, a perfectly rational adversary will perform Bayesian updates on their belief distribution over these strategies (represented as $\xi =< \xi_1, \xi_2, ..., \xi_{|\Theta|} >$) based on the sequence of pure strategy observations. The updated probability for the k^{th} mixed strategy θ_k (a vector denoting the coverage probabilities over all the targets) after observing the pure strategy on day r, denoted as ξ_k^r is computed using Eqn. 10.1 , where S_r denotes the set of all targets protected in pure strategy observation on day r, and x_i^k denotes the coverage probability at target i for the kth mixed strategy. His belief of the defender's mixed strategy after observing pure strategy on day r (denoted as b^r, which is a vector denoting the beliefs over all the targets) can then be computed as a weighted average of all the mixed strategies, where the weights are the updated probabilities (Eqn. 10.2). We denote this model as ${}^{I}B_u$, where I denotes informed adversary and u indicates that we start with a uniform prior over the set of mixed strategies.

$$\xi_k^r = \frac{\xi_k^{r-1} * \prod_{i \in S_r} x_i^k}{\sum_k (\xi_k^{r-1} * \prod_{i \in S_r} x_i^k)} \tag{10.1}$$

$$b^r = \frac{\sum_k (\xi_k^r * \theta_k)}{\sum_k (\xi_k^r)} \tag{10.2}$$

10.2.2 Boundedly Rational Adversary

The previously proposed belief update model for boundedly rational adversaries (Pita et al., 2009) in an SSG setting with no training data (Sec. 3.4.1) assumed that the adversary forms beliefs based on: (a) the *actual* mixed strategy of the defender, (b) uniform prior belief about the coverage at each target, and (c) fixed weight on the prior for all days. They did not propose any generic method to weight the prior beliefs over days of the game. However, the above assumptions may not hold in reality and that could be the reason for this model's poor performance (see results in Sec. 11.1).

First, the adversary would only observe the defender's pure strategies and *not* know the exact mixed strategy. Therefore, he can only reason based on the empirical probability distribution of protection at each target. Second, he can have non-uniform prior beliefs about the coverage probabilities. Finally, an exploration of different weighting methods is necessary as the adversary can have any arbitrary weighting function for the prior weights over days of the game. None of these has ever been taken into consideration in any of the existing work in SSGs. Our contributions here are to address these shortcomings and improve the state-of-the-art belief model for boundedly rational adversaries.

First, we incorporate in the existing model (Eqn. 3.3) the empirical mixed strategy (instead of actual mixed strategy) of the defender computed using all the pure strategy observations till the current day under consideration. So, when reasoning about the adversary's beliefs for day i, the model would compute the empirical strategy based on all pure strategy observations till day i. Empirical mixed strategy is denoted as x^E. Eqn. 10.3 shows this new model. Second, since we assume that the defender has no prior training data about belief updates, it is not possible to learn about the belief update patterns of humans in this scenario. Therefore, instead of learning a function of how the adversary's reliance on his prior beliefs changes over days, we experiment with three different types of discounting functions and compare their performances: (a) linear, (b) hyperbolic, and (c) exponential. We chose hyperbolic and exponential since these are the most popular discounting methods in the literature (Frederick, Loewenstein, & O'Donoghue, 2002;

Samuelson, 1937; Farmer & Geanakoplos, 2009). These models are denoted as ${}_{lin}M_u^E$, ${}_{hyp}M_u^E$ and ${}_{exp}M_u^E$. Linear discounting based mixture model is shown in Eqn. 10.3. Similarly for hyperbolic (Eqn. 10.4) and exponential discounting (Eqn. 10.5).

$$b = \mu^{lin} * \rho_u + (1 - \mu^{lin}) * x^E \tag{10.3}$$

$$b = \mu^{hyp} * \rho_u + (1 - \mu^{hyp}) * x^E \tag{10.4}$$

$$b = \mu^{exp} * \rho_u + (1 - \mu^{exp}) * x^E \tag{10.5}$$

We will henceforth use M to denote mixture models. u denotes uniform prior, E denotes that empirical strategy is used and lin, hyp and exp denote linear, hyperbolic and exponentially decreasing weighting functions respectively. $\mu =< \mu_1, \mu_2, ..., \mu_\tau >$ denotes the weight on the prior for each of the τ days of observations. In terms of the ith day of the game, μ^{hyp} and μ^{exp} are computed as in Eqns. 10.6 and 10.7 respectively.

$$\mu_i^{hyp} = \frac{1}{i} \tag{10.6}$$

$$\mu_i^{exp} = \frac{1}{\exp(i-1)} \tag{10.7}$$

We will refer to the original model by Pita et al. as discussed in Section 3.4.1 as ${}_{0.6}M_u^A$, where A represents actual mixed strategy and 0.6 is the fixed weight on the prior.

Although the above models assume a uniform prior, we observed during my analysis that not all participants start with a uniform prior; in fact while some participants start with a uniform prior and update their beliefs, others start with a proportional prior. We observed that 46$\%$ of participants start with a uniform prior and the rest 54$\%$ start with a proportional prior in the Maximin dataset for ADS_1 and similar trends are observed in the other datasets as well. Whether a participant starts from a uniform or proportional prior is determined in two ways: (i) by computing the difference between the uniform mixed strategy and day 1 beliefs of participants, and (ii) comparing the average (over all days) errors between a model that starts with a uniform mixed strategy as the prior vs a model that starts with a proportional prior (denoted simply as p). Therefore, since it is unknown which category of prior belief a previously unseen adversary would belong to, we apply a model ${}_{lin}M^{E}_{\{u,p\}}$ shown in Eqn. 10.8 that uses a weighted (weight=β) combination of uniform and proportional strategies as the prior (Eqn. 10.9). Due to absence of data to learn from, we assume β=0.5 in our experiments. We show interesting observations with these models in Section 11.1.

$$b = \mu^{lin} * \rho_{comb} + (1 - \mu^{lin}) * x^{E} \tag{10.8}$$

$$\rho_{comb} = \beta * \rho_u + (1 - \beta) * \rho_p \tag{10.9}$$

10.3 Proposed Models: Setting with training data

In order to explore the benefits of having prior training data on model performances, we present models that learn adversary's belief update process using training data and then use those learned models to predict belief updates for new participants in the test dataset and evaluate their performances. In this scenario, the set of participants in the train and test sets are disjoint. We first applied the popular non-linear log-odds model from the psychology literature (Section 3.4.2) for the first time in an SSG setting. However, contrary to our expectations the performance of the more complex log-odds model was similar or poor as compared to the existing models that do not assume data availability(Section 11.2). Here, we propose an improved version of the log-odds model and customize existing and our proposed models from previous section to take advantage of training data when available.

10.3.1 Perfectly Rational Adversary

The first model assuming perfectly rational adversaries improves upon the previously existing model (Eqn. 3.2) by learning the Dirichlet prior that best fits the training set participants' beliefs, uses the learned prior as the starting prior for any test set participant and then updates the prior based on the observation sequence in the same way as Eqn. 3.2. We refer to this model as ${}^{N}B_{learn}$ where $learn$ denotes learned prior, and N indicates that this model corresponds to the case of an uninformed adversary, i.e., the adversary has *no* information about the number and types of strategies employed by the defender. In addition to learning the prior from data, we also use my proposed updating scheme from Section 10.2.1.1 in a new model to update the prior corresponding to unobserved pure strategies that are similar to the observed pure strategies. Hereafter, the learned Dirichlet prior model with similar (denoted by s) pure strategy updating will be referred to as ${}^{N}B^{s}_{learn}$.

10.3.2 Boundedly Rational Adversary

In this section, we propose three types of learning models assuming boundedly rational adversaries: (i) linear mixture models; (ii) non-linear mixture models; and (iii) clustering based models that exploit the heterogeneity in adversary behavior.

Linear Mixture Models: Given training data about belief formation and update from a set of participants, the defender can learn the weighting function for the prior that best fits the training data. Here, by best fit we mean that we compute the weight $\mu =< \mu_1, \mu_2, ..., \mu_\tau >$ that minimizes the average root mean squared error (rmse) between the model's predicted beliefs and those of the training set participants. Therefore, instead of using a fixed weighting function as in our proposed models in Section 10.2.2, we use model $_{learn}M_u^E$ shown in Eqn. 10.10.

$$b = \mu^{learn} * \rho_u + (1 - \mu^{learn}) * x^E \tag{10.10}$$

Consistent with our non-learning model with combined prior in Sec. 10.2.2, we propose a learning variant $_{learn}M_{\{u,p\}}^E$ (Eqn. 10.11) where we learn β in Eqn. 10.9 along with μ.

$$b = \mu^{learn} * \rho_{comb}^{learn} + (1 - \mu^{learn}) * x^E \tag{10.11}$$

Non-linear Mixture Models: As discussed earlier in Section 3.4.2, Fox et al. (See et al., 2006) considered one ignorance prior term representing the uniform belief distribution. We relax this assumption and consider another informative prior term. We re-write Eqn. 3.4 as:

$$ln\frac{b_i}{1 - b_i} = a_1 + a_2 * ln\frac{^u n_F^i}{^u n_A^i} + a_3 * ln\frac{f^i(F)}{f^i(A)} + a_4 * ln\frac{^p n_F^i}{^p n_A^i} \tag{10.12}$$

Here, $^p n_F^i$ corresponds to the new prior term (represents proportional distribution) for the event that target i is covered. Similarly for the alternate event A that the target is not covered. We then perform regression analysis(See et al., 2006) with training data to learn the model parameters. We refer to this model as $_{learn}log_{\{u,p\}}$.

Clustering based Models: During my comprehensive analysis of the performances of different models on the belief data collected on AMT, we observed a heterogeneous behavior among adversaries in terms of their belief formation process. We primarily made the following observation in terms of adversary belief updates.

Observation 2. *Adversaries can be clustered into four distinct groups based on their belief updates: (a) participants who start from a uniform prior and then update their beliefs by taking into account the empirical distribution, (b) participants who start from a proportional prior and then update their beliefs by taking into account the empirical distribution, (c) participants who only update based on the empirical distribution and starts with no prior, and (d) participants whose updates have no clear pattern and could be termed as random players.*

This observation inspired us to apply clustering techniques on the belief data of the training set participants and learn a separate model for each cluster and use those learned models to predict the beliefs of the test set participants. With respect to our proposed clustering based models, we look at two scenarios based on the amount of information available to the defender about belief updates of test set participants. As mentioned earlier, in this section, we assume that the defender only has training data from a set of participants and *no* data for any of the days for the test set participants. We will look at other models that also have additional information about test set participants in the next section.

For the case where the defender can only learn from given training data, we propose a weighted clustering based approach to model and predict beliefs of a heterogeneous population of adversaries. First, we perform c-means clustering on the 10 day belief data of the training set participants to determine the clusters of training set participants. Since the c-means algorithm depends on the initial cluster centroids provided, each run of c-means can lead to different clusters and hence result in different cluster centroids being generated. Since we did not want to influence the clustering by providing any initial starting point, in order to prevent the algorithm from generating clusters with huge variations in the cluster centroids for every run, we ran the algorithm for 1000 iterations each time for a total of 1000 initial seeds chosen using the k-means++ heuristic initialization algorithm. We then selected the clustering (out of the 1000) with the lowest within-cluster sums of point-to-centroid distances and used it for the next steps. We used the c-means++ heuristic initialization (Arthur & Vassilvitskii, 2007) algorithm with 1000 random initial seeds and chose the clustering with the lowest within-cluster sums of point-to-centroid distances. Once the clusters are generated, we learn the model ${}_{learn}M^{E}_{\{u,p\}}$ (Eqn. 10.11) for each of the c clusters. ${}_{learn}M^{E}_{\{u,p\}}$ was chosen as it performed best (see Section 11.2) among all previously discussed models, and it is also the most generalized mixture model presented. Next, we compute the model's predicted beliefs for any participant after observing pure strategy for day r as a weighted average of the predictions of each of the models (Eqn. 10.13):

$$b^{r} = \frac{\sum_{i=1}^{c} \gamma_i * {}_{i}b^{r}}{\sum_{i=1}^{c} \gamma_i} \tag{10.13}$$

Here, $\gamma_i = N_c$ is the weight given to cluster i and is the number of training set participants that belong to that cluster. ${}_{i}b^{r}$ denotes the belief predicted for day r by the learned model ${}_{learn}M^{E}_{\{u,p\}}$ for cluster i. The intuition behind weighting each cluster's model with the number of participants in that cluster is that we assume that the test set participant distribution will be similar to the training set participant distribution and hence we give higher importance to clusters containing higher number of participants, and vice versa. We will henceforth refer to this model as B-$REACT_c^{wt}$.

10.4 Proposed Models: Setting with training and testing data

In the setting studied in the previous section, training data collected from a group of participants are used to predict belief updates of a completely new set of participants in test set. Here, I assume that in addition to the training data I also have some data collected from the participants in the test set (earlier days of belief updates), which are used to predict belief updates in following days.

10.4.1 Instance based Learning Models

Instance-Based Learning Theory (IBLT) (Gonzalez, Lerch, & Lebiere, 2003) is a popular model used in Cognitive Science that attempts to explain human decision making in dynamic tasks. Based on past data about various situations and actions of different agents in such situations, IBLT attempts to predict the behavior of an agent in some situation by reasoning about known actions of other agents in similar situations. We propose an IBL model for belief prediction of an unknown adversary T_m after observing pure strategy j^r on day r.

We assume that in addition to knowing beliefs over all days of a set of adversaries (training set), we also gain information about the beliefs of the previously unseen test set adversaries at the end of each day. This could be achieved by placing an informant or spy among the poachers who would provide the defender information about the poacher's day-to-day beliefs. This allows the defender to make future belief predictions about the test set adversaries using their leaked beliefs till the current day by reasoning about beliefs of similar adversaries that are in the training data. In order to achieve this task, my model first computes similarity between beliefs (till day $r-1$) of a test set adversary and beliefs (till day $r-1$) of all training set adversaries. I then choose the k most similar training set adversaries and compute the belief of test set adversary T_m upon observing the r^{th} pure strategy based on day r beliefs of the k most similar training set participants based on Eqn. 10.14, where $\theta_i \equiv \frac{1}{d(i,T_m)^2}$ and $d(i,T_m)$ denotes the dissimilarity between the test set adversary T_m and its ith most similar training set adversary.

$$^{T_m}b^r = \frac{\sum_{i=1}^{k} \theta_i * {}^{k}b^r}{\sum_{i=1}^{k} \theta_i} \tag{10.14}$$

We will refer to this as the IBL_k model, e.g., a model based on four nearest neighbors will be referred to as $IBL_{k=4}$. Comparison results for various values of k are shown in Section 11.3.

10.4.2 Clustering based Models

We customize my previously proposed clustering based model to take advantage of additional information about the test set participants (when available). We consider two cases of information availability: (a) before each day the defender has complete information about a test set adversary's beleifs till the previous day– this is same as the assumption for IBL models; and (b) the defender knows the exact cluster a test set adversary belongs to.

For case (a), we compute for each test set adversary, the nearest (k=1) cluster he belongs to based on the known beliefs of that participant till day $i-1$ and apply the model for that cluster to predict his/her day i beliefs. This model is represented as $B\text{-}REACT_c^{k=1}$.

For case (b), since we assume that the exact cluster for each test set adversary is known to the defender, we apply the corresponding cluster's learned model to predict their beliefs for any day

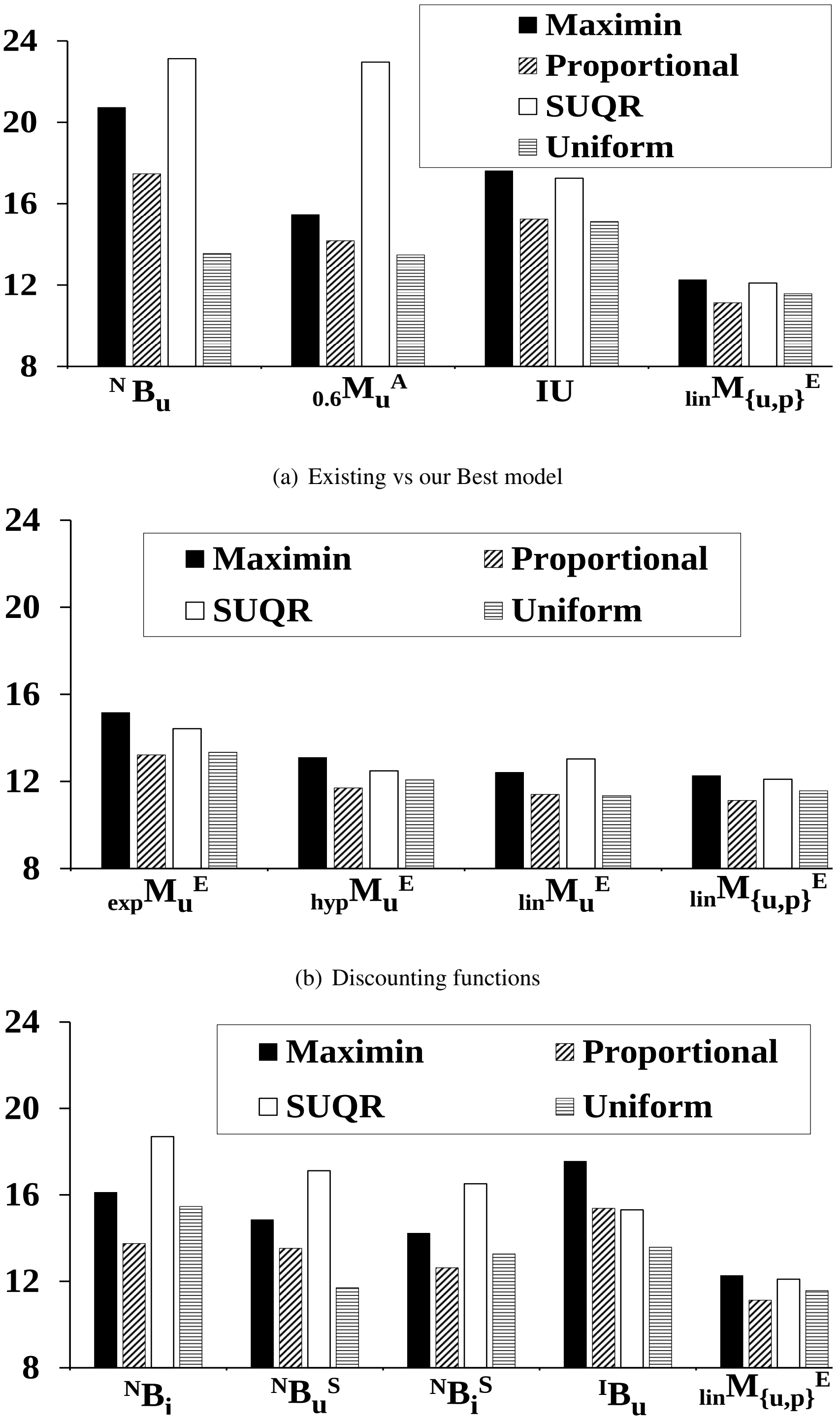
i. This is a somewhat unrealistic best-case scenario which gives us an important lower bound and therefore forms a baseline for comparing other models. In order to implement this, an important question is: how do we determine the exact cluster for a test set participant? In our game, since we have each participant's belief information for each of the 10 days, we assume the ideal scenario where we know the beliefs of all the 10 days for any test set participant ahead of time. This allows us to perform an exact nearest neighbor computation w.r.t. the c cluster centroids and determine the cluster for any test set adversary. The model is henceforth referred to as ${}^{best}B-REACT_c^{k=1}$ and its performance is shown in Sec. 11.3.

Chapter 11

Belief Modeling Experiment Results

In this chapter, we present results for existing and our proposed models (see Figure 10.3 for all model names and their assumptions). We report the performance of all the models in estimating the beliefs of the test data set participants in terms of the average root mean squared errors (rmse) between the human entered beliefs and the models' predicted beliefs. The averaging is done over all targets for all days over the total number of participants in the respective test sets and over the total number of train-test splits. We only show results on ADS_1 data in this thesis. Results on ADS_2 have the same trends for all the models that we tested, thus confirming the value of our modeling and analysis. So, those results are omitted from this thesis. *In the figures, model names are on the x-axis and average rmse (lower is better) is on the y-axis. We start y-axis from 8 instead of 0 to show differences between the model performances more prominently.* The four defender strategies for which we conducted our experiments (Maximin, Proportional, SUQR and Uniform) are shown by the colored/patterned bars for each model in each of the figures. Any mention of statistical significance indicates that the discussed model performances are statistically significant based on two-tailed t-tests at confidence=0.05.

11.1 Setting without training data



(a) Existing vs our Best model

(b) Discounting functions

(c) Bayesian models

Figure 11.1: Belief Estimation Errors (average RMSE) for models with NO training data

In Figs. 11.1(a), 11.1(b) 11.1(c), we first show performances for previously existing and our proposed models that do *not* learn on training data. We discuss important observations about these models below:

Comparison w.r.t. our best model: In Fig. 11.1(a), we demonstrate that our best performing non-learning mixture model ${}_{lin}M^{E}_{u,p}$ completely outperforms (statistically significant) the three existing non-learning models in SSGs (${}^{N}B_u$, ${}_{0.6}M^{A}_{u}$ and IU) in terms of predicting beliefs for any defender strategy. We also observe that although for existing models, Maximin and SUQR are hardest to estimate due to their non-intuitiveness (as is evident by comparing their performances on Maximin and SUQR data against their performances on Proportional and Uniform data), our best model's performance on these strategies is similar to intuitive strategies such as Uniform and Proportional. Our model's performance further highlights the impact of using the mixed empirical strategy instead of actual mixed strategy, a linear discounting function to capture the adversary's decreasing reliance on their prior beliefs, and a weighted combination of uniform and proportional prior so as to perform well against an unknown adversary who can belong to either one of these two groups.

Linear discounting performs best: We show in Fig. 11.1(b) that a simple linearly decreasing weighting function on the prior belief in the mixture models assuming boundedly rational adversaries (${}_{lin}M^{E}_{u}$) surprisingly performs similarly or better when compared to models that consider more complex discounting functions such as hyperbolic (${}_{hyp}M^{E}_{u}$) and exponential (${}_{exp}M^{E}_{u}$). Results for ${}_{lin}M^{E}_{u}$ are statistically significant w.r.t. ${}_{exp}M^{E}_{u}$ for all strategies except only on Maximin and Uniform datasets w.r.t. ${}_{hyp}M^{E}_{u}$. This demonstrates that human adversaries have extremely strong initial biases towards a prior strategy in these game settings and they only linearly decrease their reliance on that bias over days of the game. Furthermore, we observe that applying a combined prior (the best non-learning model ${}_{lin}M^E_{u,p}$ shown at the end in Fig. 11.1(b)) improves performance on SUQR data with statistical significance but only slightly on Maximin and Proportional datasets. Although intuitively one would expect ${}_{lin}M^E_{u,p}$ to perform better, this model: (a) applies a fixed weight 0.50 on each of proportional and uniform priors; and (b) fails to personalize the weight for each adversary. We'll show later that a learning model with clustering significantly improves the performance.

Informed prior and similarity based updating performs best: We show in Fig. 11.1(c) that our best non-learning model ${}_{lin}M^E_{u,p}$ significantly outperforms all proposed non-learning Bayesian update models. Furthermore, we make the following observations: (i) the performance of the previously existing model (${}^NB_u$ shown in Fig. 11.1(a)) assuming perfectly rational adversaries significantly improves (20.73 to 16.11 for Maximin data) due to our proposed informative prior model (${}^NB_i$). ${}^NB_i$ performs comparatively worse when compared to ${}^NB_u$ on only the Uniform dataset because it performs poorly for participants who start with a uniform prior and stay there due to observations from a Uniform mixed strategy. When deploying any of the other three strategies we show that it is beneficial to start with a more informative prior. The improvement is more pronounced if we only apply our proposed updating scheme to the original model (20.73 for ${}^NB_u$ to 14.85 for ${}^NB^s_u$ for Maximin data). The best performance is by our model ${}^NB^s_i$, which combines the above ideas. This emphasizes the benefit of starting with an informative prior and updating similar pure strategies when faced with limited observations. The informed rational adversary model ${}^IB_u$ which assumes that adversaries have prior knowledge about the set of defender mixed strategies, doesn't perform as well as the uninformed adversary models which make no such assumption.

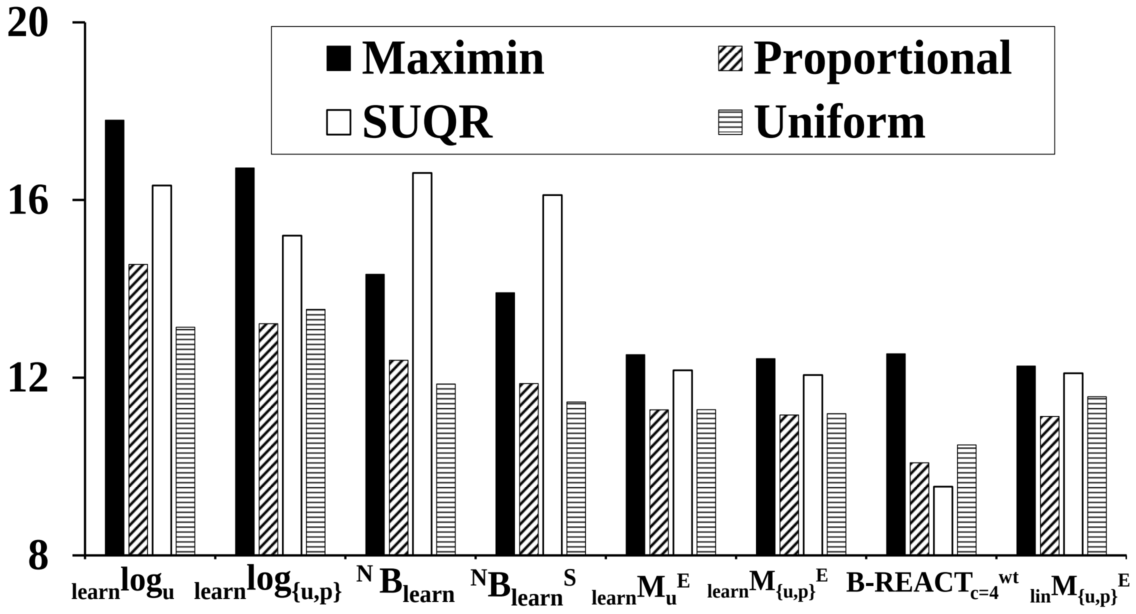


Figure 11.2: Belief Estimation Errors (average RMSE) for models with train data only

11.2 Setting with training data

In Fig. 11.2, we show performances for previously existing and our proposed models that learns on training data. Important observations are highlighted below:

Existing learning models perform poorly: We observe that the performance of ${}_{learn}log_u$ is the worst among all the learning models although its performance improves when an additional ignorance prior term is incorporated (${}_{learn}log_{u,p}$) in the existing model.

Clustering significantly improves performance: First, we show that our proposed clustering based model $B\text{-}REACT_c^{wt}$ with $c = 4$ clusters outperforms (statistically significant) the previous best non-learning model ${}_{lin}M_{u,p}^E$ (shown as the last model) on 3 out of 4 datasets. More importantly, it also outperforms all other learning models. The significant difference between the non-clustering model ${}_{learn}M_{u,p}^E$ and $B\text{-}REACT_{c=4}^{wt}$ which learns the same model but on different clusters, can be attributed to our earlier Observation 2 in Sec. 10.3.2 about the four distinct groups of adversary belief updates. This is consistent with our observation about the weights learned for each model for each of the four clusters: (a) Cluster 1: The learned model's weight μ^{learn} on the prior decreases almost linearly from approx. 0.95 to 0.05 over 10 days, and the fixed weight on proportional prior ($1 - \beta$ in Eqn. 10.9) is high (approx. 0.95 for most datasets), representing a group of adversaries who start with proportional prior and then linearly updates their reliance on the empirical strategy as they observe more pure strategies; (b) Cluster 2: Both μ^{learn} and β are 0, representing adversaries who only update based on the empirical strategy and do not start from any prior; (c) Cluster 3: It represents a group of adversaries who start with a uniform prior and then update their beliefs with more importance on their observations as days progress– the learned model has a high β value (approx. 0.97 for most datasets) and a μ that is high initially but gradually decreases (approx. 0.98 to 0.23); and, (d) Cluster 4: The model learns a high weight on β (approx. 0.97 for most datasets) and a μ that decreases from 0.90 to 0.50 (approx.) over 10 days, thus representing a unique group of adversaries who start with a uniform prior and update at random on most days, and are not influenced by the pure strategy observations.

Learning Dirichlet prior improves performance: Learning a Dirichlet prior significantly improves the predictions of the resulting models (${}^{N}B_{learn}$ and ${}^{N}B^{s}_{learn}$ respectively). For Proportional data, the average rmse for ${}^{N}B^{s}_{learn}$ is 11.86 as opposed to 17.47 for the original model with no training data (${}^{N}B_{u}$ in Fig. 11.1(a)).

Learning weights in mixture models do *not* help: Mixture models that learn the weighting function on the prior from training data (${}_{learn}M^{E}_{u}$ and ${}_{learn}M^{E}_{u,p}$) are similar in performance to the best mixture model that does *not* learn on any training data (shown as the last graph in Fig. 11.2 for comparison). This is a surprising observation, especially because we observed significant improvement in performances due to learning for perfectly rational adversary models

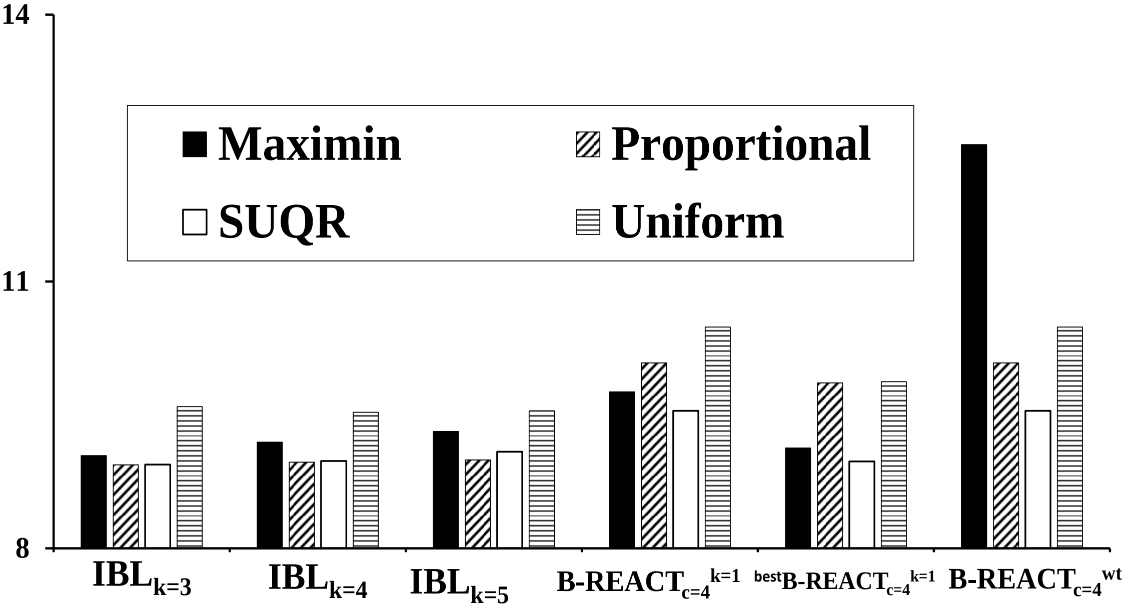


Figure 11.3: Belief Estimation Errors (average RMSE) for models with train and test data

(discussed above). Further investigation reveals that the shape of the learned weighting function is approximately linear for majority of the datasets, and hence the similar performance. Although surprising, this is a significant observation because it demonstrates that in the absence of data we could simply apply a linear decreasing weighting function irrespective of the deployed mixed strategy and expect to perform as well as if we had prior data to learn from. Furthermore, this demonstrates that human adversaries have extremely strong initial biases towards a prior strategy in our game settings and they only linearly decrease their reliance on that bias over days of the game.

11.3 Setting with training and testing data

This section discussed results for models that use both training as well as additional information of test set adversaries to predict future beliefs. In Fig. 11.3, we compare the performance of such models against the best performing model discussed in the previous section ($B\text{-}REACT^{wt}_{c=4}$ shown as the last model for comparison).

Testing set information does *not* help clustering models: $B\text{-}REACT^{k=1}_{c=4}$, a model that uses past beliefs of test set participants to infer their clusters, has similar performance to $B\text{-}REACT^{wt}_{c=4}$ which does not have this information. An ideal model that assumes complete knowledge about each test participant's exact cluster (${}^{best}B\text{-}REACT^{k=1}_{c=4}$ has rmse of 9.8 for Proportional data) shows improved performance over $B\text{-}REACT^{wt}_{c=4}$ which has an rmse of 10.08. However, the near similar performance of $B\text{-}REACT^{wt}_{c=4}$ to models that assume additional information about test set adversaries demonstrates the validity of our weighted clustering based technique towards making accurate predictions about adversary beliefs even in the absence of additional information.

IBL models perform best: IBL models (Section 10.4.1) outperform (rmse for $IBL_{k=3}$ is 8.93 for Proportional dataset) $B\text{-}REACT^{wt}_{c=4}$ (rmse of 10.08), $B\text{-}REACT^{k=1}_{c=4}$ and ${}^{best}B\text{-}REACT^{k=1}_{c=4}$ with statistical significance. This shows that, while clustering based models suffer from abstraction due to clustering, IBL models are able to make more personalized predictions **when information about past beliefs of test set participants is available**.

We demonstrate in Fig. 11.3 that the performance gain of our proposed model that learns only on the training data ($B\text{-}REACT^{wt}_{c=4}$) when compared to previously existing models is due to our improved performance through all days of observations, with a decreasing trend of the rmse's over days. This is also true for the best performing non-learning model (${}_{lin}M^{E}_{u,p}$). However, there is no clear trend in the performance of the previously existing models. In fact, it is surprising that they perform worse with more observations. We also observe that the model that takes advantage of

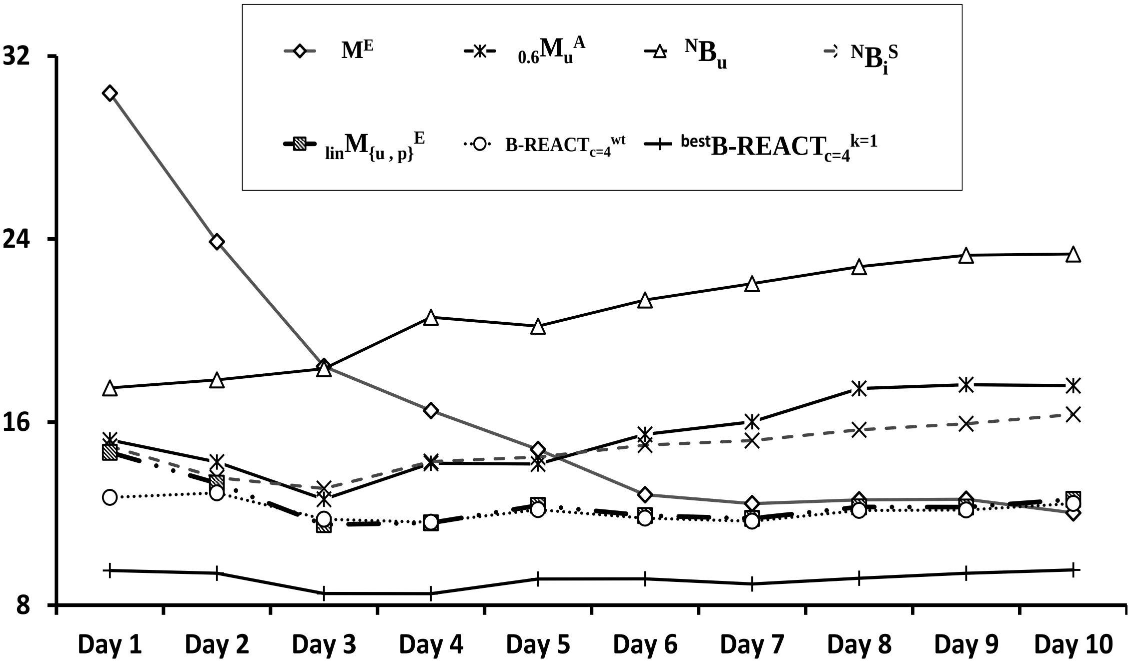


Figure 11.4: Per Day Belief Estimation Errors (average RMSE) for some of the models

both existing training data as well as available data from test set participants ${}^{best}B\text{-}REACT_{c=4}^{k=1}$ performs significantly better over all days as compared to any of the other models.

In conclusion, we highlight three key observations from our extensive analysis with existing and our proposed belief models. First, contrary to most research in psychology, we observed surprisingly that a linear discounting function best fits adversary behavior in our setting as opposed to more complex hyperbolic and exponential discounting. Second, we demonstrated the benefit of modeling heterogeneous groups of adversaries for improved belief prediction. Third, we show that our models significantly outperform existing models; the difference in performance further increases when using learning models and becomes even more pronounced when personalized prediction models (e.g., IBL) are employed.

Chapter 12

Conclusions and Future Directions

Although several competing human behavior models have been proposed to model and protect against boundedly rational adversaries in various security and sustainability domains, no study has yet been conducted either against actual human subjects or based on real-world data to show which is the best model in different settings. This article provides three major contributions towards answering that question and therefore, provides an advancement to the field of adversary behavior modeling. Given the important applications in security and sustainability, such as protecting wildlife and fisheries, my contributions in this thesis are critical for such domains.

My first major contribution is to provide a a novel human behavior model called SHARP for domains where the challenge is to make fine-grained predictions on a small set of targets based on plentiful attack data. SHARP has three major novelties: (i) It models the adversary's adaptive decision making process by reasoning based on success or failure of the adversary's past actions on exposed portions of the attack surface. (ii) It accounts for lack of information about the adversary's preferences due to insufficient exposure to attack surface by reasoning about similarity between exposed and unexposed areas of the attack surface, and also incorporating a confidence based discounting parameter to model the learner's trust in the available data. (iii) It integrates a non-linear probability weighting function to model the adversary's perception of probabilities. Based on a repeated measures study of competing models we provide results analyzing the performance of SHARP along with other existing approaches. Besides data collected from AMT, we also demonstrate SHARP's superiority by conducting experiments with security experts from the government of Indonesia as well as various NGOs (WWF, WCS, YABI, etc.) who are in charge of protecting wildlife in the national park. Results from my experiments show that: (i) Human perceptions of probability are S-shaped, contradicting the inverse S-shaped observation from prospect theory. (ii) Existing human behavior models and algorithms perform poorly in initial rounds; and (iii) SHARP consistently performs significantly better than existing approaches, most notably in the initial rounds.

Our second major contribution is to present INTERCEPT, a paradigm shift from complex temporal models for data rich scenarios to simpler decision tree-based models for domains requiring coarse-grained predictions over large geographical areas based on sparse and noisy historical data. Previous state-of-the-art models developed for such settings suffer from poor performance and other critical limitations that preclude its actual deployment in the field. Indeed, in the process of conducting the most extensive empirical evaluation in the AI literature of one of the largest real-world poaching datasets from QENP, we show a surprising result: INTERCEPT, based on a simpler model, significantly outperformed the more complex models in the presence of sparse (and noisy) attack data. Furthermore, INTERCEPT satisfied other requirements such as fast execution speed, which are beneficial when deploying such models in the field. Therefore, as a first for behavior modeling applications for the wildlife conservation domain, we presented results from a month-long test of INTERCEPT by rangers in QENP where rangers found and confiscated an active snare and almost a dozen additional snares, including multiple elephant snares, before they were deployed. Given that the rangers also found a poached elephant, their finding and confiscating of new elephant snares before they were deployed is significant; this research has potentially saved the lives of elephants and other animals in QENP.

As the third major contribution, we address the lack of empirical evaluation of belief formation models by conducting the first-of-its-kind systematic comparison of existing and new proposed belief models on belief data collected through human subjects experiments on AMT. This is important because it relaxes a common assumption in adversary behavior modeling – the adversaries have access to the actual mixed strategy of the defender while optimizing their own attack strategies, which is not true for domains such as wildlife protection. Modeling the adversary's belief formation and update procedure is therefore crucial in developing more effective attack prediction models by incorporating the best belief model for a particular scenario in the best behavioral model for that scenario. We highlight three key observations based on my experiments in belief modeling. First, we show that surprisingly a linear discounting function on the adversary's prior beliefs best fits adversary belief update process, as opposed to more complex weighting functions such as hyperbolic and exponential discounting. Second, we demonstrate the presence of four different types of belief formation and update procedure among human subjects and show the benefit of modeling such heterogeneous groups of adversaries for improved belief prediction. Third, we show that my proposed models significantly outperform existing models for various settings with and without past data; the difference in performance significantly improves when using learning models for heterogeneous adversaries in data-driven settings. This research provides a clear guidance towards incorporating such belief models in the relevant attack prediction models for improved predictions in different scenarios in the future.

Future work could potentially explore the following important research areas: (a) Effects of opponent player on probability weighting: explore the hypothesis that different shapes of the probability weighting function could be obtained depending on the opponent in the game, i.e., people possibly think about probability differently when playing against nature as opposed to when playing against an adversary; (b) Information leakage for strategic gain: how the defender (rangers) can use various signaling schemes to strategically reveal information (Xu, Freeman, Conitzer, Dughmi, & Tambe, 2016) to catch the adversaries (poachers); (c) Modeling deception: how the defender (rangers) can strategically deceive the adversaries (poachers) with misinformation and get higher utility; (d) Cooperative adversary behavior modeling: address the problem of modeling the attackers' bounded rationality in a more complicated, cooperation-enabled (Gholami, Wilder, Brown, Sinha, Sintov, & Tambe, 2016) and repeated SSG setting for wildlife protection; (e) Modeling more intelligent adversaries such as the ones who can influence (not just learn) the distribution of animals through changes to the environment (food sources, water, natural shelter, etc); and (f) Application of adaptive adversary models in other domains: explore the suitability of adaptive adversary behavior models proposed in this thesis, for different domains such as cybersecurity and urban crimes.

Chapter 13

Appendix

13.1 Algorithm to learn PSU model parameters

The algorithm first randomly divides all data from past rounds into training and testing data (Line 2), and then splits the training data into K training and validation data sets (Line 3). Considering a set of $\{\delta\ ,\ \gamma\}$ combinations, we learn the other parameters of the model and compute the average validation error over all the validation datasets for each $\{\delta\ ,\ \gamma\}$ combination (Line 14). We then choose the $\{\delta\ ,\ \gamma\}$ combination with the minimum average validation error (Line 16) and using that combination we re-learn the other model parameters and that forms our final weight vector (Line 17).

Algorithm 3 Algorithm to learn the weights of P-SUQR and its variations in repeated SSGs

```
INPUT: Data from $R$ rounds: $D^1$, $D^2$, ... , $D^R$.

OUTPUT: Learned weights ($\delta_p$, $\gamma_p$, $\omega_1$, $\omega_2$, $\omega_3$, $\omega_4$).

for $r$=1 to $R$ do
    Randomly divide the collected data $D^r$ into one training ($Tr^r$) and test ($Te^r$) set.
    Take the training samples ($Tr^r$) and randomly divide it into $K$ training (${}^kTrv^r$) and validation (${}^kVal^r$) splits ($1 \leq k \leq K$).
end for
Consider a range of values for both $\delta$ and $\gamma$ (Eqn. 5.1 in the article).
Discretize each range and consider all possible $M$ pairs for $\{\delta , \gamma\}$ in that range.
for $i$=1 to $M$ do
    for $k$=1 to $K$ do
        Given training splits ${}^kTrv^1$, ${}^kTrv^2$, ... , ${}^kTrv^R$, learn the weights ${}^k\omega$=(${}^k\omega_1$, ${}^k\omega_2$, ${}^k\omega_3$, ${}^k\omega_4$) of Eqn. 10 using MLE to maximize the sum of log-likelihoods.
        Predict using learned weights ${}^k\omega$ on the validation splits ${}^kVal^1$, ${}^kVal^2$, ... , ${}^kVal^R$.
        Calculate the prediction errors ${}^kErr^1$, ${}^kErr^2$, ... , ${}^kErr^R$ on the validation sets ${}^kVal^1$, ${}^kVal^2$, ... , ${}^kVal^R$ respectively.
        Calculate the sum of all prediction errors ${}^kErr^1$, ${}^kErr^2$, ... , ${}^kErr^R$ and let it be ${}^kErr$.
    end for
    Calculate the average of all $K$ prediction errors ${}^kErr$ ($1 \leq k \leq K$) and let that be $AvgErr_i$.
end for
Let $p$ be the index of the $\{\delta , \gamma\}$ pair with the minimum $AvgErr_i$ (i=1 to $M$). Choose $\{\delta_p , \gamma_p\}$ as the best parameter values for the probability weighting function.
Given training sets $Tr^1$, $Tr^2$, ... , $Tr^R$ and $\{\delta_p , \gamma_p\}$, learn the weights $\omega$=($\omega_1$, $\omega_2$, $\omega_3$, $\omega_4$) of Eqn. 10 using MLE . The final learned weight set is then ($\delta_p$, $\gamma_p$, $\omega_1$, $\omega_2$, $\omega_3$, $\omega_4$).
```

13.2 Proof of Theorem 1

Proof. Assume $R_i^d(>0)$, $P_i^d(<0)$, $R_i^a(>0)$ and $P_i^a(<0)$ Also assume that the defender has $M \in \mathbb{N}^+$ defender resources. Let q_i be the attacking probability for target i. According to SUQR model,

$$q_i = \frac{e^{\omega_1 x_i + \omega_2 R_i^a + \omega_3 P_i^a}}{\sum_j e^{\omega_1 x_j + \omega_2 R_j^a + \omega_3 P_j^a}}$$

We rewrite the defender's expected utility $U_i^d(x)$ as: $U_i^d(x) = (R_i^d - P_i^d)x_i + P_i^d$. Then defender's overall expected utility can be represented as

$$f(x) = \sum_i q_i U_i^d(x)$$

and the optimization problem is to maximize $f(x)$ under the constraints $\sum_i x_i \leq K$ and $0 \leq x_i \leq 1$.

Assume $\bar{x}$ is the optimal defender strategy and *opt* is the optimal value of defender's overall expected utility. Let $\bar{S}$ be the set of targets with positive coverage probability, i.e., $\bar{S} = \{i|\bar{x_i} > 0\}$. Then $\forall i \in \bar{S}$, $\frac{\partial f}{\partial x_i}|_{\bar{x}} \geq 0$. Otherwise, a defender strategy with a lower coverage probability on target i will achieve a higher defender expected utility than $\bar{x}$, contradict with the optimality. Formally, let $\Delta_i = (0, 0, ..., \delta, 0, 0)$ be a vector with an infinitesimal positive value in i^{th} row. If $\frac{\partial f}{\partial x_i} < 0$, then $f(\bar{x} - \Delta_i) = f(\bar{x}) - \delta \frac{\partial f}{\partial x_i}|_{\bar{x}} > f(\bar{x})$.

Further, the targets in $\bar{S}$ can be devided into two subsets $\bar{S_1}$ and $\bar{S_2}$ where $\bar{S_1} = \{i|\bar{x_i} = 1\}$ and $\bar{S_2} = \{i|0 < \bar{x_i} < 1\}$. Then $\forall i, j \in \bar{S_2}$, $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}} \geq 0$. Otherwise, a defender strategy that moves a little bit coverage probability from a target with higher partial derivative to a target with a lower partial derivative will achieve a higher defender expected utility than $\bar{x}$, contradict with the optimality. Formally, if $\frac{\partial f}{\partial x_i}|_{\bar{x}} > \frac{\partial f}{\partial x_j}|_{\bar{x}}$,

$$\begin{aligned}
f(\bar{x}+\Delta_i-\Delta_j)-f(\bar{x}) &= f(\bar{x}+\Delta_i-\Delta_j)-f(\bar{x}+\Delta_i)+f(\bar{x}+\Delta_i)-f(\bar{x}) && (13.1)\\
&= -\delta\frac{\partial f}{\partial x_j}|_{\bar{x}+\Delta_i}+\delta\frac{\partial f}{\partial x_i}|_{\bar{x}} && (13.2)\\
&= -\delta(\frac{\partial f}{\partial x_j}|_{\bar{x}}+\delta\frac{\partial f^2}{\partial x_i\partial x_j}|_{\bar{x}})+\delta\frac{\partial f}{\partial x_i}|_{\bar{x}} && (13.3)\\
&= \delta(\frac{\partial f}{\partial x_i}|_{\bar{x}}-\frac{\partial f}{\partial x_j}|_{\bar{x}})-\delta^2\frac{\partial f^2}{\partial x_i\partial x_j}|_{\bar{x}} && (13.4)\\
&> 0 && (13.5)
\end{aligned}$$

The last inequality is achieved by neglecting the second order term. So $f(\bar{x}+\Delta_i-\Delta_j) > f(\bar{x})$.

Moreover, when $\omega_1 > 0$, $\forall i,j \in \bar{S}_2$, $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}} = 0$. Select two targets $i,j \in \bar{S}_2$. As $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}}$, $f(\bar{x}+\Delta_i-\Delta_j)-f(\bar{x}) = -\delta^2\frac{\partial f^2}{\partial x_i\partial x_j}|_{\bar{x}}$ according to line (4). The partial derivative of function f is

$$\frac{\partial f}{\partial x_i}|_{\bar{x}} = \omega_1 q_i(U_i^d - f) + q_i(R_i^d - P_i^d)$$

If $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}} > 0$, then we have

$$\omega_1(f - U_i^d) < R_i^d - P_i^d \tag{13.6}$$

and

$$\omega_1(f - U_j^d) < R_j^d - P_j^d \tag{13.7}$$

Thus

$$
\begin{aligned}
\frac{\partial^2 f}{\partial x_i \partial x_j} &= \omega_1^2 q_i q_j (2f - U_i^d - U_j^d) - \omega_1 q_i q_j (R_i^d - P_i^d + R_j^d - P_j^d) && (13.8)\\
&= \omega_1 q_i q_j (\omega_1 (f - U_i^d) + \omega_1 (f - U_j^d) - (R_i^d - P_i^d + R_j^d - P_j^d)) && (13.9)\\
&< \omega_1 q_i q_j (R_i^d - P_i^d + R_j^d - P_j^d - (R_i^d - P_i^d + R_j^d - P_j^d)) && (13.10)\\
&= 0 && (13.11)
\end{aligned}
$$

The inequality in line (10) comes from (6) and (7) and the fact the $\omega_1 > 0$. So we have $f(\bar{x} + \Delta_i - \Delta_j) - f(\bar{x}) > 0$, which means moving some coverage probability from target i to target j in $\bar{S}_2$ leads to a defender strategy with higher expected utility. It contradicts with the optimality.

We now prove that $\|\bar{S_2}\| < 2$. As we know$\forall i, j \in \bar{S}_2$, $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}} = 0$ and $\frac{\partial f^2}{\partial x_i \partial x_j} = 0$, so $f(\bar{x} + \Delta_i - \Delta_j) - f(\bar{x}) = 0$ and we can move some coverage probability from target i to target j to get another optimal strategy $\hat{x}$ with the same expected defender utility *opt* while $i, j \in \hat{S}_2$. As the $\hat{x}$ is also an optimal strategy, it also satisfies $\frac{\partial f}{\partial x_i}|_{\hat{x}} = \frac{\partial f}{\partial x_j}|_{\hat{x}} = 0$. So we have

$$\omega_1(opt - U_i^d(\hat{x_i})) = R_i^d - P_i^d \tag{13.12}$$

$$\omega_1(opt - U_i^d(\bar{x_i})) = R_i^d - P_i^d \tag{13.13}$$

From (12)-(13), we get

$$(R_i^d - P_i^d)(\hat{x_i} - \bar{x_i}) = 0 \tag{13.14}$$

As $\hat{x_i} \neq \bar{x_i}$, we get $R_i^d = P_i^d$, which contradicts with the payoff structure of the game.

Next, we prove $\|\bar{S_2}\| \neq 1$. Assume target i is the only element in $\bar{S_2}$. If $\frac{\partial f}{\partial x_i}|_{\bar{x}} > 0$, we can increase x_i to get a better defender strategy without violating the constraint of total number of resources as $K \in \mathbb{N}$ and all targets other than i are covered with probability 0 or 1. This contradicts with optimality. So $\frac{\partial f}{\partial x_i}|_{\bar{x}} = 0$, i.e., $\omega_1(f - U_i^d(\bar{x_i})) = R_i^d - P_i^d$. Then

$$\begin{aligned}
\frac{\partial^2 f}{\partial x_i^2} &= \omega_1 q_i(\omega_1(1-2q_i)(U_i^d - f) + 2(R_i^d - P_i^d)(1-q_i)) \\
&= 3\omega_1 q_i(R_i^d - P_i^d) \\
&> 0
\end{aligned}$$

So x_i is a minimum point and increasing x_i can get a better defender strategy. Again, this contradicts with optimality.

So $\bar{S_2} = \emptyset$ and the coverage probabilities of the optimal strategy are chosen only from $0, 1$, i.e., the optimal defender strategy is a pure strategy. □ □

13.3 Sample Email

Hi,

Thank you for participating in our experiment. Your base compensation for round 3 has been paid to you via AMT. Thank you also for your valuable comments and suggestions about the game and its strategies. We will definitely take those into account later on. Now, we would want you to participate in the 4th round of our experiment. Please follow the link below to participate: `http://cs-server.usc.edu:16568/gamelink/index.jsp`

In the first page, please read carefully the compensation details. You will be starting with the performance bonus that you earned in the last round. **The last date to participate in this round of our experiment is Wednesday (November 6 2014) 4 pm PST**. Please try to complete the experiment by the deadline because otherwise deployment of the next round gets delayed.

You are very important to the study and your continued participation is critical. Don't be discouraged if you got caught by a ranger in this round. The chance to play again and earn performance and completion bonuses are coming in a few days. We look forward to your continued participation.

Thank you.

13.4 Challenges and Remedies of Online Repeated Measures Experiments

In this section we discuss a set of challenges that we faced during our repeated measures experiments on AMT and our methodological contributions towards mitigating those challenges.

For our repeated measures experiments, due to unavailability of data, the strategy shown for each first round of the real game was Maximin. We then learned the model parameters based on previous rounds' data, recomputed and redeployed strategies, and asked the *same* players to play again in the subsequent rounds. For each model, all five rounds were deployed over a span of weeks. Such repeated measures studies on AMT are rare in game-theoretic studies; and certainly none have been conducted in the context of SSGs. Indeed, while the total time of engagement

Table 13.1: Average time (in seconds) taken to play the actual game per round

Round 1	Round 2	Round 3	Round 4	Round 5
61	52	47	43	39

over our 20 experimental settings was 46 weeks, each setting required on average 2.3 weeks (See Table 4.1). One interesting statistic to note is that the average amount of time taken by the participants to play the actual game based on which the results in our experiments are generated, is 45 seconds, as obtained by computing from the data over all four payoff structures. The average time spent on the actual game per round is shown in Table 13.1. This, in addition to comments and feedback from the participants (Appendix 5), indicates that the participants were spending time considering the trade-offs between the risk of getting captured and obtaining high rewards.

When we started conducting the experiments, we observed that there were very high attrition rates (i.e. people dropped out) for the number of participants between rounds of the game. The varying number of participants from one round to another made it difficult to not only compare between the performance of the model between rounds but also at the end of the five rounds. We hypothesized that the low participant retention rates were due to the following reasons: (i) our initial payment scheme for the participants did not have a large payout at the end of all the rounds of the experiment and therefore participants could potentially leave the experiment at any time depending on how much money they were satisfied with; (ii) initially, each round on average lasted 3.5 weeks as some participants would complete the experiments quickly while others would take a long time to respond: hence several participants may have been dropping out due to such lengthy rounds; and (iii) the lack of commitment to complete a few weeks long repeated measures experiment could also be an issue, as has previously been found in similar repeated measures studies (Estrada, Woodcock, & Schultz, 2014).

To mitigate the above challenges, we took the following steps: (i) We set up the payment scheme to consistently reward participation in each round plus offering a relatively high completion incentive at the end of the experiment; (ii) Although we allowed respondents sufficient time (3-4 days on average) to respond (Menard, 2008), as giving them an immediate deadline to finish a particular task can result in high attrition rates, we also maintained persistent contact by sending repeated reminders (Cotter, Burke, Stouthamer-Loeber, & Loeber, 2005) to the participants, especially to participants who did not respond immediately; (iii) Prior to beginning the first round of our experiment, we asked participants to commit to completing all five rounds, i.e, remain in the study through completion, to be eligible for study enrollment.

The problems addressed by the development of each of our strategy (for example, the choice of concrete metrics, payoffs and incentive mechanisms) does indeed lead to the development of some methodological contributions towards conducting such repeated measures experiments on crowdsourcing platforms like AMT. Since we did not find any related research in this area which specifies a minimum accepted participant retention rate for a repeated measures study, in our work we attempted to achieve a retention rate of at least 80%. this also ensured sufficient number of participants for statistical significance tests. We therefore implemented the steps mentioned above, in order, and measured the effect of the implementation of the corresponding strategy until we achieved the desired participant retention rate. Next, we discuss the implementation details of each of the steps taken (in the order they were implemented) to reduce attrition rates and also provide results showing the improvements due to our approaches.

13.4.1 Step 1: Payment Scheme

In our initial payment scheme, shown in Table 13.2, column 2, participants were paid a fixed 'base compensation' (=\$0.50) for participation in each round of the experiment and a 'performance bonus' based on the points earned (or lost) in each round by attacking a particular target region in the game. The participants started with an initial amount of \$0.50 as the 'performance bonus' in each round. For each reward point earned in a particular round (i.e., if they successfully poached), \$0.10 was added to the initial 'performance bonus'. For each point lost (i.e., if they were captured by the ranger), \$0.10 was deducted from their current 'performance bonus'. The bonus at the end of a particular round was *not* carried forward to the next round and was paid along with the fixed 'base compensation' for that round. For example, for an experiment with two rounds and \$0.50 as the 'base compensation' for each round, if a participant earned a reward point of 9 in the first round and got a penalty of 1 in the second round, (s)he was paid \$(0.50+(0.50+9*0.10)) = \$1.90 at the end of round 1 and \$(0.50+(0.50-0.10*1)) = \$0.90 at the end of round 2. With this payment scheme in place, we observed that there were very high attrition rates, i.e., very few people returned to play in each round, thus making it difficult to compare the performances on various models on a varying number of participants for each model. This is shown in Fig. 13.1(a), where the x-axis shows rounds of the game and the y-axis shows retention rates. Note that we had to abandon the experiments due to high attrition rates (low retention rates) in round 5 for one of the models (PSUQR) in the first trial and rounds 4 and 5 for PSUQR in the second trial. The failure of this method led us to implement a new payment scheme which is discussed below.

We made three changes to our first method of compensation. First, we introduced a 'completion bonus' (=\$2.50) for completing all the rounds of the experiment. Second, like before, to motivate the subjects, the participants were incentivized based on the reward/penalty of the region they chose to attack, i.e., ‘performance bonus’. However now, while the base compensation was paid after each round was completed, the ‘performance bonus’ was carried forward from one round to the next and paid along with the ‘completion bonus’ at the end of all the rounds of the experiment. Third, the players now started with an initial ‘performance bonus’ of \$1.50 in round 1 and they could win up to a maximum and a minimum amount in each round and hence a very high ‘performance bonus’ at the end of all the rounds, based on how successful they were. We still had the same ‘base compensation’ for each round as \$0.50, thus resulting in a total base compensation of \$2.50 over 5 rounds. However, the maximum amount they could potentially earn at the end of all the rounds from only the performance and completion bonus was as high as \$7.60. The performance and completion bonus together at the end of all the rounds was much higher as compared to the total base compensation earned for playing all the 5 rounds. This ensured that majority of the participants remained motivated and returned to play all the rounds. A detailed comparison of the initial and modified payment schemes are shown in Table 13.2.

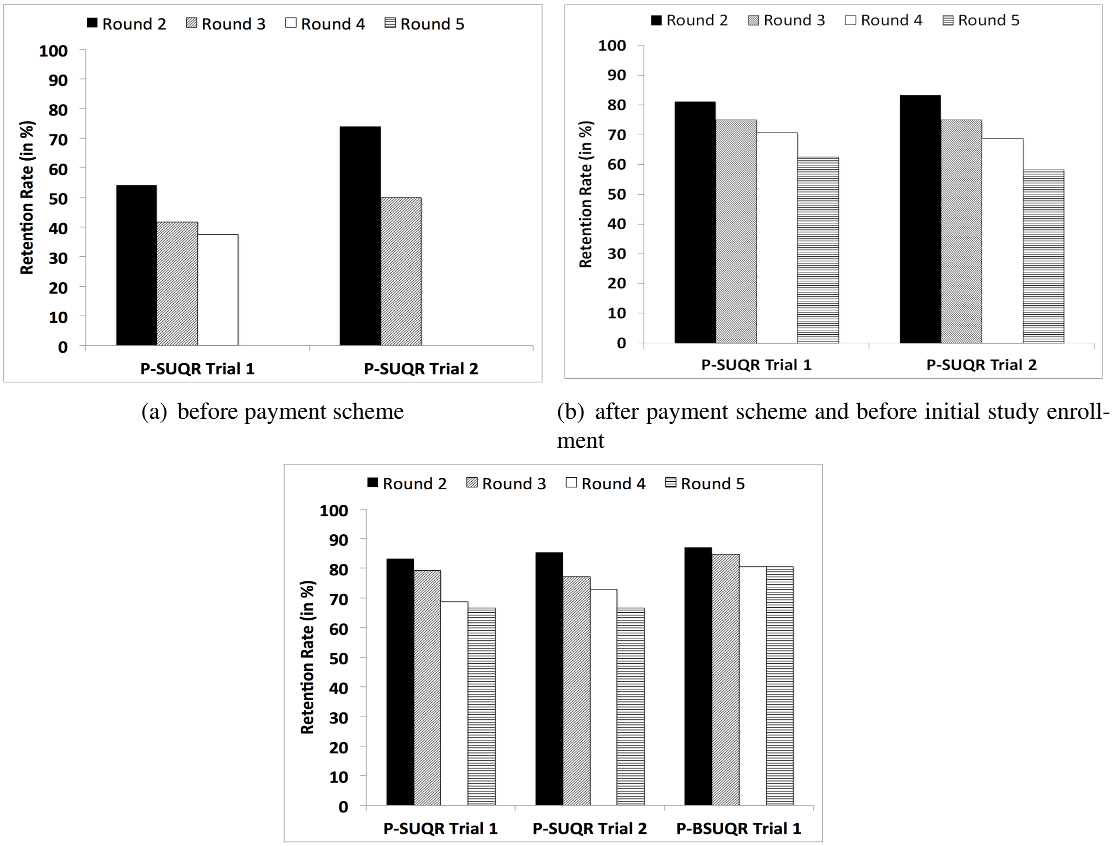
To better understand the impact of our new payment scheme, let us take the previous example of a two-round experiment where a participant earned a reward point of 9 in the first round and a penalty of 1 in the second round. According to our new payment scheme, (s)he was paid \$0.50 at the end of round 1 (the bonus compensation for round 1). (S)he also earned a performance bonus of \$(1.50+9*0.1) = \$2.40 in round 1 which was carried forward to round 2 and *not* paid at the end of round 1. Then at the end of round 2 she was paid \$(0.50+(2.40-1*0.1)+2.50) = \$5.30 (base compensation for round 2 (=\$0.50) + performance bonus at the end of round 2 (=\$2.30) + completion bonus (=\$2.50)). As mentioned before, as compared to our initial payment scheme, this high amount at the end of all the five rounds of our experiments ensured that a relatively high

Table 13.2: Comparison between payment schemes

Types of Compensation	Initial Payment Scheme	Modified Payment Scheme
Base Compensation per round	0.50\$ — paid after each round	0.50\$ — paid after each round
Performance Bonus per round	0.50\$ + 0.10\$ per reward point (or -0.10\$ per penalty point) — paid after each round	start with 1.50\$ in round 1, then 0.10\$ per reward point (or -0.10\$ per penalty point) in every round — added to previous rounds' performance bonus and gets caried forward; total accumulated amount is paid after 5 rounds
Completion Bonus	0\$	2.50\$ —paid after 5 rounds

number of participants were retained till the end of the study. On an average, including all the compensations, each participant was paid \$7.60 upon completion of our five-round experiments. There were also participants who earned as high as \$9 at the end of the five rounds including all the compensations. The effect of this payment scheme on participant retention rate can be seen in Fig.13.1(b).

Although the new payment scheme proved effective in retaining more participants, one possibility to be considered is that the performance bonus should not have caused any bias such that the subjects who performed well are more likely to participate in future rounds but who performed poorly are more likely to drop off. We observe from our data that the average retention rates over all games for people who succeeded in the previous round and those who failed in the previous round are 90% and 92% respectively. Therefore, we conclude based on our data that no bias was introduced due to the design of our payment schemes.



(a) before payment scheme

(b) after payment scheme and before initial study enrollment

(c) after payment scheme and after initial study enrollment

Figure 13.1: Retention Rates for various models (a) before implementation of our payment scheme, and (b) after implementation of our payment scheme and before implementation of initial study enrollment procedure, and (c) after implementation of our payment scheme and initial study enrollment.

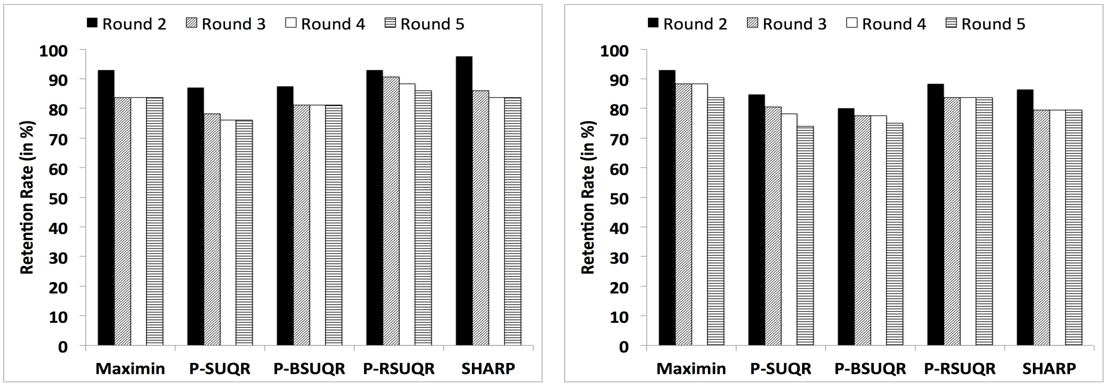
13.4.2 Initial Study Enrollment

Even though the implementation of the new payment scheme saw an increase in retention rate as shown in Fig. 13.1(b), there was still a decrease in retention rates over rounds. Therefore, we implemented an approach where the participants had to commit to completing all five rounds before starting the first round of the game. Commitment has been shown to be effective in the past in various scenarios (Aharonovich, Amrhein, Bisaga, Nunes, & Hasin, 2008; Baca-Motes, Brown, Gneezy, Keenan, & Nelson, 2013). In our game, the participants were asked to either 'agree' or 'disagree' to this commitment. On an average, 96% of the participants who enrolled in AMT for our study agreed to this commitment. These participants were then allowed to proceed towards playing the first round of the game. On the other hand, if they did not agree, they were thanked for their interest in our study, but not allowed to participate any further. The effect of this on the retention rate can be seen in Fig. 13.1(c). This clearly shows that a significant number of participants with prior commitment towards completing all the rounds of the experiment, returned and completed all the rounds.

13.4.3 Reminder Emails

Even though the implementation of the payment scheme and initial study enrollment procedures increased the retention rate as shown in Fig. 13.1(c), the retention rate still decreased over rounds for some of the experiments, even though at a slower rate. Therefore, we sent repeated reminders to the participants with clearly stated deadlines to ensure that they (i) do not forget to participate in the current round, and (ii) also remain motivated throughout the study. The emails were worded carefully and a sample email is shown in Appendix 3. Results are shown in Figs. 13.2(a) and 13.2(b).

In this section, we gave an overview of our trial and validation games, tested a set of hypothesis to improve participant retention rates for our AMT repeated measures experiments and showed the results of the deployment of each of our strategies to mitigate the challenges in retaining participants. We observed that a delayed compensation scheme along with prior participant commitment and repeated reminders throughout the course of the experiment helped in achieving an average retention rate of 83.69%, which is above the 80% retention rate. In Section 7,



(a) ADS_1

(b) ADS_2

Figure 13.2: Retention Rates for various models over 4 rounds, starting from round 2 to round 5, on (a) ADS_1 and (b) ADS_2 respectively.

we will show results from the comparison of our models based on the data obtained from the corresponding number of participants retained per round.

13.5 Participant Feedback

After the actual game was over, we asked the participants for feedback regarding the games they played. We asked them two specific questions regarding: (i) their experiences playing the game; and (ii) any strategy they employed while playing the actual game. For point (i), participants primarily mentioned that they enjoyed playing the game and that the instructions were easy to understand while some even mentioned that it was interesting to play a game that involved taking decisions while balancing risk and reward. For point (ii), most participants mentioned that they tried to balance risk and rewards by looking for target areas close to their starting point that had relatively high animal density but still a reasonable probability of success. This risk-reward balance is consistent with the model we learned which put different weights on defender coverage and adversary reward and penalty. This feedback in essence supports formulations such as

SHARP studied in this article. Few participants mentioned that they took risks by attacking target areas with high animal density even if the coverage probabilities in that target area was relatively high. Below we share some key feedback by the participants regarding their game playing experiences. Note that these are actual comments from the participants and have not been modified in any way.

13.5.1 Feedback for ”Please tell us about your experience playing the game”:

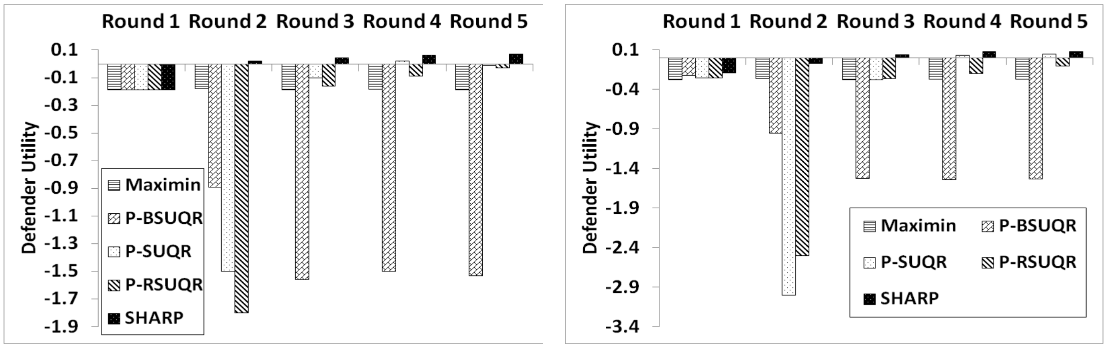
(a) Easy to understand and the visual indications make it even easier, (b) The game was enjoyable and easy to understand, (c) The game was interesting, no bugs encountered, (d) I thought it was fun, very clearly laid out and enjoyable to play, (e) It was fun and kind of exciting. I liked the opportunity and it was interesting to balance risk and reward.

13.5.2 Feedback for ”Did you use a particular strategy in playing the game? If yes, please specify.”:

(a) Find a greenish square with many hippos, as close as possible to the starting location, (b) I would only target areas with greater than 50% success rate, (c) My basic strategy was to find the most populated, greenest and closest square, (d) I stayed away from the darker red areas, (e) I tried to balance the risk and reward factors. That is, what would be acceptable as a loss versus what I could possibly gain, (f) I tried to get the maximum payoff while minimize the risk of getting caught to an acceptable level, (g) I decided to risk it and set traps in areas that payed well, even though there is high chance that I will get caught.

13.6 Additional Experimental Results on ADS_3 and ADS_4

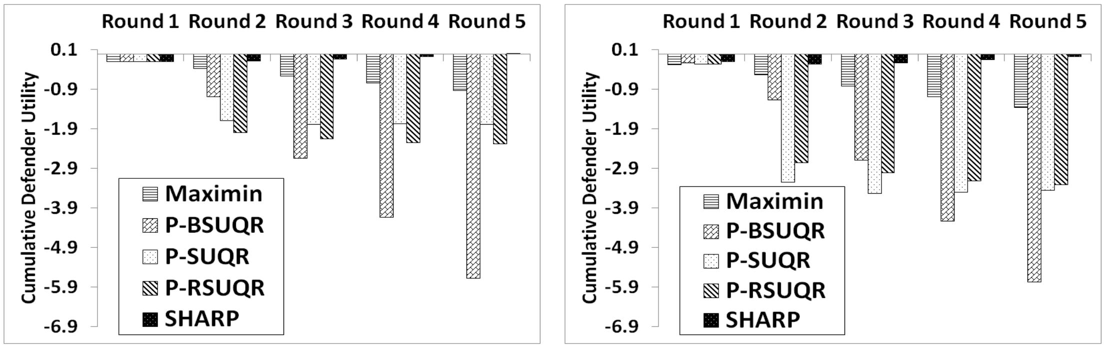
13.6.1 Defender Utilities



(a) Results on ADS_3

(b) Results on ADS_4

Figure 13.3: Defender utilities for various models on ADS_3 and ADS_4 respectively.

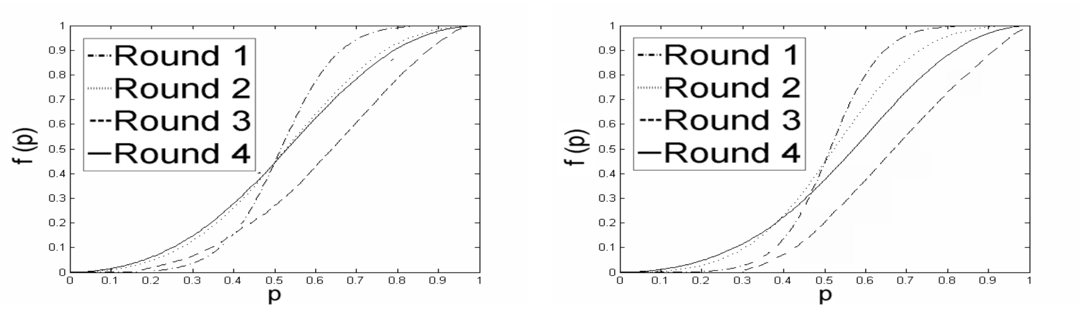


(a) Results on ADS_3

(b) Results on ADS_4

Figure 13.4: Cumulative defender utilities for various models on ADS_3 and ADS_4 respectively.

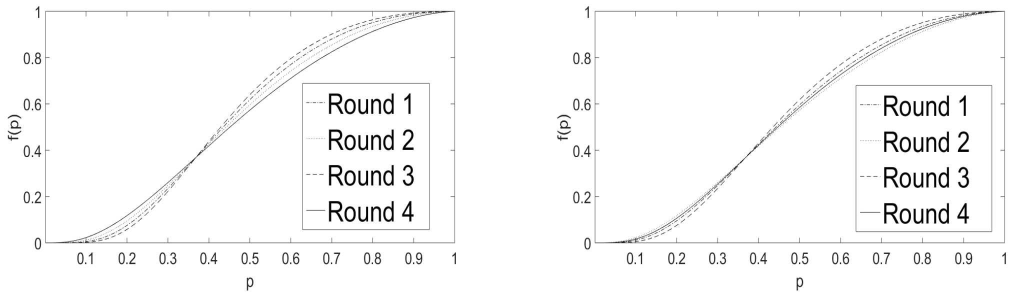
13.6.2 Learned Probability Curves



(a) ADS_3

(b) ADS_4

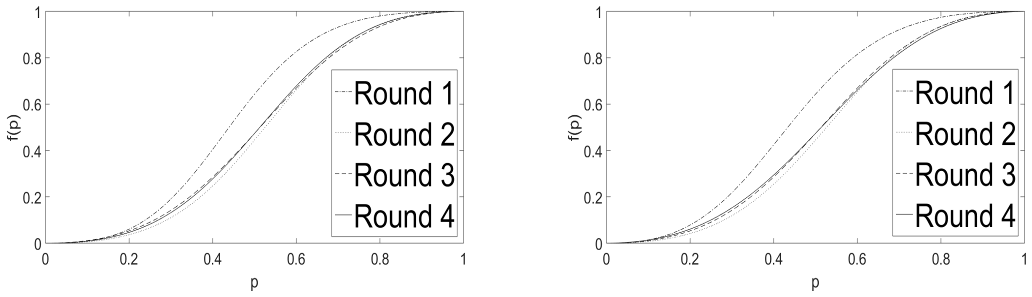
Figure 13.5: Learned probability curves for P-SUQR on ADS_3 and ADS_4 respectively.



(a) ADS_3

(b) ADS_4

Figure 13.6: Learned probability curves with Prelec's probability weighting function for P-SUQR on ADS_3 and ADS_4 respectively.

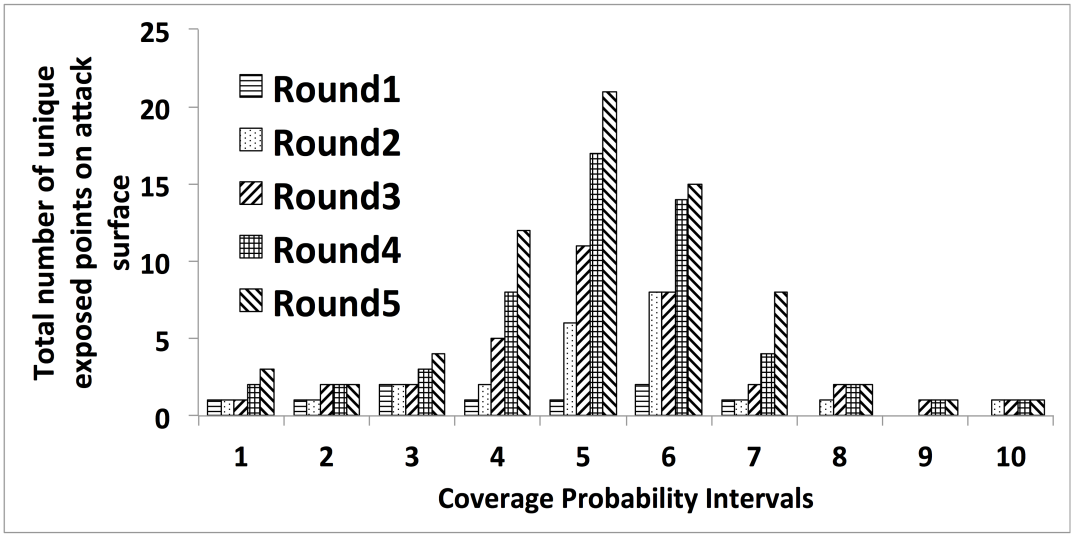


(a) ADS_3

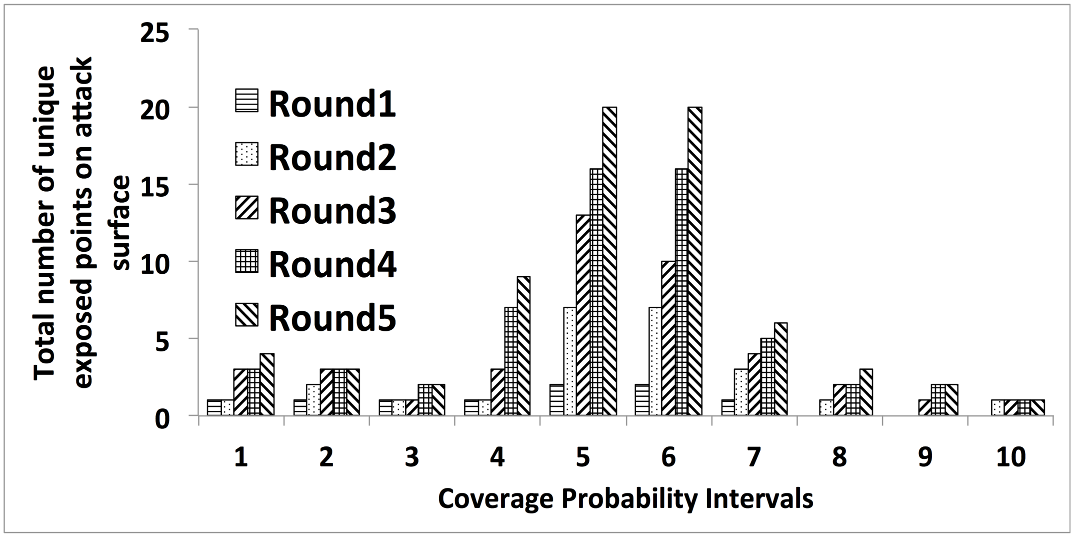
(b) ADS_4

Figure 13.7: Learned probability curves for PWV-SUQR on ADS_3 and ADS_4 respectively.

13.6.3 Evidence of Attack Surface Exposure



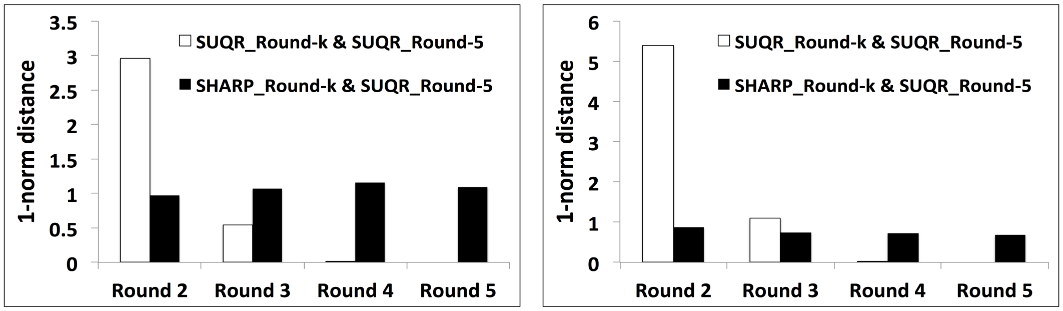
(a) ADS_3



(b) ADS_4

Figure 13.8: Total number of unique exposed target profiles till the end of each round for each coverage probability interval for ADS_3 and ADS_4.

13.6.4 Adaptiveness of SHARP



(a) ADS_3

(b) ADS_4

Figure 13.9: Adaptivity of SHARP and Convergence of P-SUQR on payoff structures ADS_3 and ADS_4 respectively.

13.7 Robustness of SHARP's results across domains

One might argue that since the wildlife poaching game requires participants to place snares with the goal of poaching animals, responses from human subjects and hence the results may be biased due to their moral dilemma. In order to verify this, we conducted a separate set of human subjects experiments on ADS_1 and ADS_3 with a game (see Figure 13.10) where the participants play the role of a soldier who is looking to place a bomb to attack enemy trucks. This is exactly same as the wildlife poaching game and data is also collected in the same way, with the only exception that this game revolves around a different attack scenario. In this game, there is no moral dilemma in terms of conducting the attack because the backstory primes them to acts as soldiers fighting for their as well as the country's honor. The results, in terms of the defender's utilities over rounds of the game, as well as the learned probability weighting functions, are similar to those obtained from the poaching game, as can be seen in Figures 13.11(a) to 13.12(b).

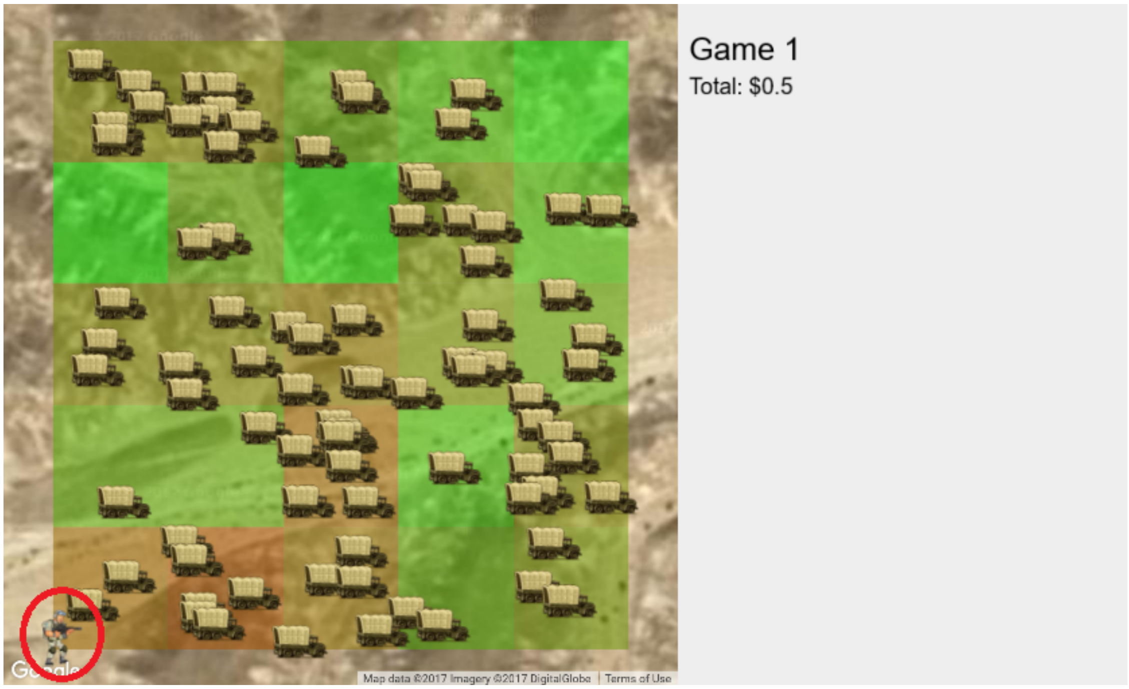
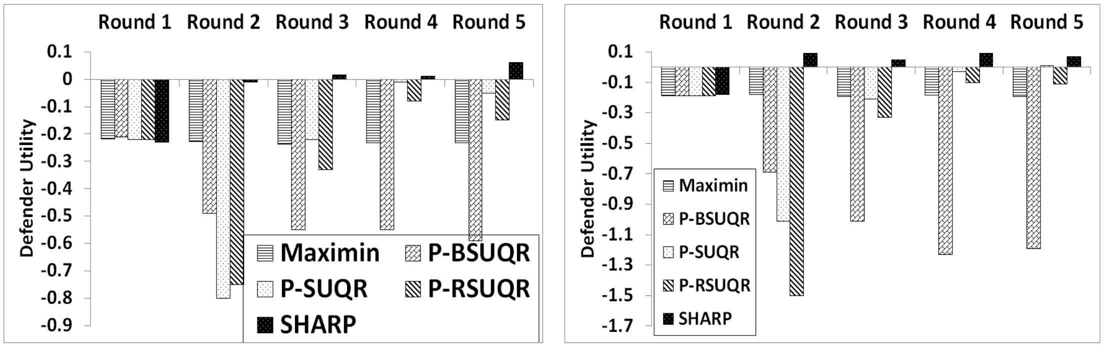


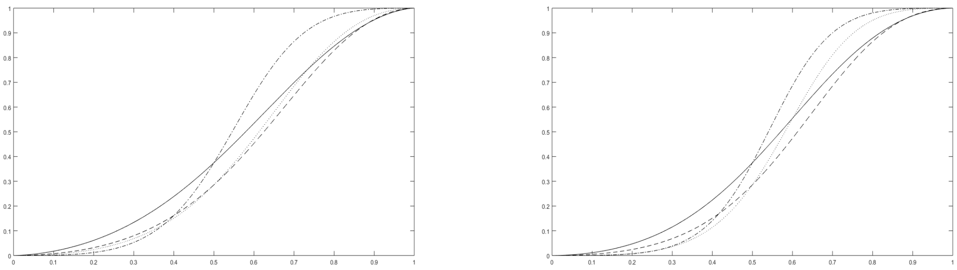
Figure 13.10: Soldier Game Interface for our simulated online repeated SSG



(a) Results on ADS_1

(b) Results on ADS_3

Figure 13.11: Defender utilities for various models on ADS_1 and ADS_3 respectively.



(a) ADS_1

(b) ADS_3

Figure 13.12: Learned probability curves for SHARP on ADS_1 and ADS_3 respectively on the Soldier game data.

Bibliography

Abbasi, Y. D., Short, M., Sinha, A., Sintov, N., Zhang, C., & Tambe, M. (2015). Human adversaries in opportunistic crime security games: Evaluating competing bounded rationality models. In *Advances in Cognitive Systems (ACS)*.

Abdellaoui, M., L'Haridon, O., & Zank, H. (2010). Separating curvature and elevation: A parametric probability weighting function. *Journal of Risk and Uncertainty*, *41*(1), 39–65.

Agmon, N., Kraus, S., & Kaminka, G. A. (2008). Multi-robot perimeter patrol in adversarial settings. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2339–2345.

Aharonovich, E., Amrhein, P. C., Bisaga, A., Nunes, E. V., & Hasin, D. S. (2008). Cognition, commitment language, and behavioral change among cocaine-dependent patients.. *Psychology of addictive behaviors*, *22*(4), 557–567.

Alarie, Y., & Dionne, G. (2001). Lottery decisions and probability weighting function. *Journal of Risk and Uncertainty*, *22*(1), 21–33.

An, B., Kempe, D., Kiekintveld, C., Shieh, E., Singh, S., Tambe, M., & Vorobeychik, Y. (2012). Security games with limited surveillance. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI)*.

Arthur, D., & Vassilvitskii, S. (2007). K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA.

Azaria, A., Gal, Y., Kraus, S., & Goldman, C. (2015). Strategic advice provision in repeated human-agent interactions. *Autonomous Agents and Multi-Agent Systems*, 1–26.

Baca-Motes, K., Brown, A., Gneezy, A., Keenan, E. A., & Nelson, L. D. (2013). Commitment and behavior change: Evidence from the field. *Journal of Consumer Research*, *39*(5), 1070–1084.

Bagwell, K. (1992). Commitment and observability in games. Tech. rep., University, Center for Mathematical Studies in Economics and Management Science.

Baker, C. L., Saxe, R. R., & Tenenbaum, J. B. (2011). Bayesian theory of mind: Modeling joint belief-desire attribution. In *In Proceedings of the Thirtieth Third Annual Conference of the Cognitive Science Society*.

Balcan, M.-F., Blum, A., Haghtalab, N., & Procaccia, A. D. (2015). Commitment without regrets: Online learning in stackelberg security games. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, EC '15.

Basilico, N., Gatti, N., & Amigoni, F. (2009). Leader-follower strategies for robotic patrolling in environments with arbitrary topologies. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '09.

Beck, J., & Forstmeier, W. (2007). Superstition and belief as inevitable by-products of an adaptive learning strategy. *Human Nature*, *18*(1), 35–46.

Beggs, A. W. (2005). On the convergence of reinforcement learning. *Journal of Economic Theory*, *122*(1), 1–36.

Berinsky, A. J., Huber, G. A., & Lenz, G. S. (2012). Evaluating online labor markets for experimental research: Amazon. com's mechanical turk. *Political Analysis*, *20*(3), 351–368.

Bishop, C. (2007). *Pattern Recognition and Machine Learning*. Springer.

Blum, A., Haghtalab, N., & Procaccia, A. (2014). Learning optimal commitment to overcome insecurity. In *In Proceedings of the 28th Annual Conference on Neural Information Processing Systems (NIPS)*.

Brown, M. (2015). *Balancing Tradeoffs in Security Games: Handling Defenders and Adversaries with Multiple Objectives*. Ph.D. thesis, University of Southern California.

Brunswik, E. (1952). The conceptual framework of psychology. In *International Encyclopedia of Unified Science, Volume 1, Number 10*. he University of Chicago Press.

Caravolas, M., Hulme, C., & Snowling, M. J. (2001). The foundations of spelling ability: Evidence from a 3-year longitudinal study. *Journal of memory and language*, *45*(4), 751–774.

Casbeer, D. W., Kingston, D. B., Beard, R. W., & McLain, T. W. (2006). Cooperative forest fire surveillance using a team of small unmanned air vehicles. *International Journal of Systems Science*, *37*(6), 351–360.

Ceren, R., Doshi, P., Meisel, M., Goodie, A., & Hall, D. (2013). On modeling human learning in sequential games with delayed reinforcements. In *Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 3108–3113.

Chabris, C., Laibson, D., & Schuldt, J. (2006). Intertemporal choice. *The new Palgrave dictionary of economics*, *2*.

Collen, B., Pettorelli, N., Baillie, J. E. M., & Durant, S. M. (Eds.). (2013). *Biodiversity Monitoring and Conservation: Bridging the Gap Between Global Commitment and Local Action*. Wiley-Blackwell.

Cominetti, R., Melo, E., & Sorin, S. (2010). A payoff-based learning procedure and its application to traffic games. *Games and Economic Behavior*, *70*(1), 71 – 83.

Cotter, R. B., Burke, J. D., Stouthamer-Loeber, M., & Loeber, R. (2005). Contacting participants for follow-up: how much effort is required to retain participants in longitudinal studies?. *Evaluation and Program Planning*, *28*(1), 15–21.

Critchlow, R., Plumptre, A., Driciru, M., Rwetsiba, A., Stokes, E., Tumwesigye, C., Wanyama, F., & Beale, C. (2015). Spatiotemporal trends of illegal activities from ranger-collected data in a ugandan national park. *Conservation Biology*, *29*(5), 1458–1470.

Cui, J., & John, R. (2014). Empirical comparisons of descriptive multi-objective adversary models in stackelberg security games. In *Conference on Decision and Game Theory for Security (Gamesec)*.

Cushman, S. A., & Huettmann, F. (Eds.). (2010). *Spatial Complexity, Informatics, and Wildlife Conservation*. Springer.

Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML.

Deng, Y., Hillygus, D. S., Reiter, J. P., Si, Y., & Zheng, S. (2013). Handling attrition in longitudinal studies: The case for refreshment samples. *Statistical Science*, *28*(2), 238–256.

Desikan, P., Karunakaran, K., & Gokulnath, G. (2013). Design of an aquatic park and salvation of endangered aquatic species in its natural habitat. *APCBEE Procedia*, *5*, 197 – 202.

Devenport, L. (1979). Superstitious bar pressing in hippocampal and septal rats. *Science*, *18*(1), 35–46.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10*(7), 1895–1923.

Dudani, S. A. (1976). The distance-weighted k-nearest-neighbor rule. *Systems, Man and Cybernetics, IEEE Transactions on*, *SMC-6*(4), 325–327.

Eck, J., Chainey, S., Cameron, J., & Wilson, R. (2005). Mapping crime: Understanding hotspots..

Elster, J. (2005). A plea for mechanisms. *Social mechanisms: an analytical approach to social theory*.

Erev, I., & Roth, A. (1998). Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria. *The American Economic Review*, *88*(4), 848–881.

Estrada, M., Woodcock, A., & Schultz, P. W. (2014). Tailored panel management: A theory-based approach to building and maintaining participant commitment to a longitudinal study.. In *Evaluation Review*.

Etchart-Vincent, N. (2009). Probability weighting and the level and spacing of outcomes: An experimental study over losses. *Journal of Risk and Uncertainty*, *39*(1), 45–63.

Fang, F., Nguyen, T. H., Pickles, R., Lam, W. Y., Clements, G. R., An, B., Singh, A., Tambe, M., & Lemieux, A. (2016). Deploying paws: Field optimization of the protection assistant for wildlife security. In *Innovative Applications of Artificial Intelligence Conference*.

Fang, F., Stone, P., & Tambe, M. (2015). When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *International Joint Conference on Artificial Intelligence (IJCAI)*.

Farmer, J. D., & Geanakoplos, J. (2009). Hyperbolic discounting is rational: Valuing the far future with uncertain discount rates. Tech. rep., Cowles Foundation for Research in Economics, Yale University.

Farrington, D., Loeber, R., & Welsh, B. (2010). Longitudinal-experimental studies. In *Handbook of Quantitative Criminology*. Springer New York.

Feltovich, N. (2000). Reinforcement-based vs. belief-based learning models in experimental asymmetric-information games. *Econometrica*, *68*(3), 605–641.

Fieldstadt, E. (2015). Drones used to stop elephant and rhino poachers in africa..

Ford, B., Nguyen, T., Tambe, M., Sintov, N., & Fave, F. D. (2015). Beware the soothsayer: From attack prediction accuracy to predictive reliability in security games. In *Conference on Decision and Game Theory for Security (Gamesec)*.

Frederick, S., Loewenstein, G., & O'Donoghue, T. (2002). Time Discounting and Time Preference: A Critical Review. *Journal of Economic Literature*, *40*(2), 351–401.

Gans, N., Knox, G., & Croson, R. (2007). Simple models of discrete choice and their performance in bandit experiments. *Manufacturing and Service Operations Management*, *9*(4), 383–408.

Gatti, N. (2008). Game theoretical insights in strategic patrolling: Model and algorithm in normal-form. In *Proceedings of the 18th European Conference on Artificial Intelligence (ECAI)*, pp. 403–407.

Gholami, S., Wilder, B., Brown, M., Sinha, A., Sintov, N., & Tambe, M. (2016). A game theoretic approach on addressing cooperation among human adversaries. In *Workshop on security and multiagent systems, International conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Goldstein, H. (2009). Handling attrition and non-response in longitudinal data. *Longitudinal and Life Course Studies*, *1*(1), 63–72.

Gonzalez, C., Lerch, J. F., & Lebiere, C. (2003). Instance-based learning in dynamic decision making. *Cognitive Science*, *27*(4).

Gonzalez, R., & Wu, G. (1999). On the shape of the probability weighting function. *Cognitive psychology - Vol 38*, 129–166.

Hamisi, M. (2008). *Identification and mapping risk areas for zebra poaching: A case of Tarangire National Park, Tanzania*. Thesis, ITC.

Hammond, G. (1955). A correlation of reaction rates. In *Journal of the American Chemical Society, Volume 77(2)*.

Haskell, W., Kar, D., Fang, F., Tambe, M., Cheung, S., & Denicola, E. (2014). Robust protection of fisheries with compass. In *Innovative Applications of Artificial Intelligence (IAAI)*.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer-Verlag.

Heiman, G. W. (2002). *Research Methods in Psychology, 3rd Edition*. Houghton Mifflin Company, Boston and New York.

Hopkins, E. (2001). Two competing models of how people learn in games. Tech. rep., David K. Levine.

Humphrey, S. J., & Verschoor, A. (2004). The probability weighting function: experimental evidence from Uganda, India and Ethiopia. *Economics Letters*, *84*(3), 419–425.

Jain, M. (2013). *Thwarting Adversaries with Unpredictability: Massive-scale Game-Theoretic Algorithms for Real-world Security Deployments*. Ph.D. thesis, Citeseer.

Jajodia, S., Ghosh, A. K., Swarup, V., Wang, C., & Wang, X. S. (2011). *Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats* (1st edition). Springer Publishing Company, Incorporated.

Johanson, M., & Bowling, M. (2009). Data biased robust counter strategies. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Johanson, M., Zinkevich, M., & Bowling, M. (2007). Computing robust counter-strategies. In *In Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*.

Johansson, U., Sönströd, C., Norinder, U., & Boström, H. (2011). Trade-off between accuracy and interpretability for predictive in silico modeling. *Future medicinal chemistry*, *3*(6), 647–663.

Kahneman, D., & Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, *47*(2), 263–91.

Kanevski, M., Pozdnoukhov, A., & Timonin, V. (2008). Machine learning algorithms for geospatial data. applications and software tools. In *4th Biennial Meeting of the International Environmental Modelling and Software Society*, pp. 7–10.

Kar, D., Fang, F., Fave, F. D., Sintov, N., Sinha, A., Galstyan, A., An, B., & Tambe, M. (2015a). Learning bounded rationality models of the adversary in repeated stackelberg security games. In *Adaptive and Learning Agents Workshop at the International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*.

Kar, D., Fang, F., Fave, F. D., Sintov, N., & Tambe, M. (2015b). "a game of thrones": When human behavior models compete in repeated stackelberg security games. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Kar, D., Fang, F., Fave, F. D., Sintov, N., & Tambe, M. (2015c). Conducting longitudinal experiments with behavioral models in repeated stackelberg security games on amazon mechanical turk. In *Human-Agent Interaction Design and Models (HAIDM) Workshop at the International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*.

Kar, D., Fang, F., Fave, F. M. D., Sintov, N., Tambe, M., & Lyet, A. (2016). Comparing human behavior models in stackelberg security games: An extended study. *Artificial Intelligence Journal (AIJ), Elsevier, DOI: http://dx.doi.org/10.1016/j.artint.2016.08.002*.

Kar, D., Ford, B., Gholami, S., Fang, F., Plumptre, A., Tambe, M., Driciru, M., Wanyama, F., & Rwetsiba, A. (2017a). Cloudy with a chance of poaching: Adversary behavior modeling and forecasting with real-world poaching data. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Kar, D., Sengupta, S., Kamar, E., Horvitz, E., & Tambe, M. (2017b). Believe it or not: Modeling adversary belief formation in stackelberg security games with varying information. In *Advances in Cognitive Systems (ACS)*.

Kiekintveld, C., Jain, M., Tsai, J., Pita, J., Ordonez, F., & Tambe, M. (2009). Computing optimal randomized resource allocations for massive security games.. International Foundation for Autonomous Agents and Multiagent Systems.

Kim, J. W. (2013). Applications of synthetic aperture radar (sar) sar interferometry (insar) for monitoring of wetland water level and land subsidence. Tech. rep., The Ohio State University.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, pp. 1137–1143. Morgan Kaufmann.

Korzhyk, D., Conitzer, V., & Parr, R. (2010). Complexity of computing optimal stackelberg strategies in security resource allocation games. In *In Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 805–810.

Kotz, S., Balakrishnan, N., & Johnson, N. L. (2000). *Continuous Multivariate Distributions. Volume 1: Models and Applications*. Wiley, New York, NY.

Leclerc, P. (2014). *Prospect Theory Preferences in Noncooperative Game Theory*. Ph.D. thesis, Virginia Commonwealth University.

Lee, W. S., & Liu, B. (2003). Learning with positive and unlabeled examples using weighted logistic regression. In *ICML*, Vol. 3.

Lemieux, A. M. (2014). *Situational Crime Prevention of Poaching (Crime Science Series)*. Routledge.

Letchford, J., Conitzer, V., & Munagala, K. (2009). Learning and approximating the optimal strategy to commit to. In *Proceedings of the 2Nd International Symposium on Algorithmic Game Theory*, SAGT '09, pp. 250–262, Berlin, Heidelberg. Springer-Verlag.

Manadhata, P. K., & Wing, J. M. (2011). An attack surface metric. *Software Engineering, IEEE Transactions on*, *37*(3), 371–386.

Marecki, J., Tesauro, G., & Segal, R. (2012). Playing repeated stackelberg games with unknown opponents. In *AAMAS*, pp. 821–828.

McCracken, P., & Bowling, M. (2004). Safe strategies for agent modelling in games. In *In Proceedings of the National Conference on Artificial Intelligence (AAAI)*.

McFadden, D. (1976). Quantal choice analysis: A survey. *Annals of Economic and Social Measurement*, *5*(4), 363–390.

McKelvey, R. D., & Palfrey, T. R. (1995). Quantal response equilibria for normal form games. *Games and Economic Behavior*, *2*, 6–38.

Menard, S. W. (2008). *Handbook of longitudinal research: Design, measurement, and analysis*. Academic Press.

Meyer, F., & Hinzb, S. (2009). Automatic ship detection in space-borne sar imagery..

Montesh, M. (2013). Rhino poaching: A new form of organised crime. Tech. rep., College of Law Research and Innovation Committee of the University of South Africa.

Moreto, W. (2013). *To Conserve and Protect: Examining Law Enforcement Ranger Culture and Operations in Queen Elizabeth National Park, Uganda*. Thesis, Rutgers.

Nguyen, T. H., Delle Fave, F. M., Kar, D., Lakshminarayanan, A. S., Yadav, A., Tambe, M., Agmon, N., Plumptre, A. J., Driciru, M., Wanyama, F., et al. (2015). Making the most of our regrets: Regret-based solutions to handle payoff uncertainty and elicitation in green security games. In *International Conference on Decision and Game Theory for Security*, pp. 170–191. Springer.

Nguyen, T. H., Sinha, A., Gholami, S., Plumptre, A., Joppa, L., Tambe, M., Driciru, M., Wanyama, F., Rwetsiba, A., Critchlow, R., et al. (2016). Capture: A new predictive anti-poaching tool for wildlife protection. In *International Conference on Autonomous Agents & Multiagent Systems*.

Nguyen, T. H., Yadav, A., An, B., Tambe, M., & Boutilier, C. (2014). Regret-based optimization and preference elicitation for Stackelberg security games with uncertainty. In *In Proceedings of the National Conference on Artificial Intelligence (AAAI)*.

Nguyen, T. H., Yang, R., Azaria, A., Kraus, S., & Tambe, M. (2013). Analyzing the effectiveness of adversary modeling in security games.. In *AAAI*.

Osborne, M. J., & Rubinstein, A. (1994). *A Course in Game Theory*. MIT Press, Cambridge.

Parkes, D. C., Mao, A., Chen, Y., Gajos, K. Z., Procaccia, A., & Zhang, H. (2012). Turkserver: Enabling synchronous and longitudinal online experiments. In *Proceedings of the Fourth Workshop on Human Computation (HCOMP'12)*. AAAI Press.

Pita, J. (2012). *The Human Element: Addressing Human Adversaries in Security Domains*. Ph.D. thesis, University of Southern California.

Pita, J., Jain, M., Marecki, J., Ordonez, F., Portway, C., Tambe, M., Western, C., Paruchuri, P., & Kraus, S. (2008). Deployed ARMOR protection: the application of a game theoretic model for security at the Los Angeles International Airport. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems: Industrial Track*, AAMAS '08, pp. 125–132.

Pita, J., Jain, M., Ordonez, F., Tambe, M., Kraus, S., & Magori-Cohen, R. (2009). Effective solutions for real-world stackelberg games: When agents must deal with human uncertainties. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Pita, J., Jain, M., Tambe, M., Ordonez, F., & Kraus, S. (2010). Robust solutions to stackelberg games. *Artificial Intelligence*, *174*(15), 1142–1171.

Pita, J., John, R., Maheswaran, R., Tambe, M., & Kraus, S. (2012). A robust approach to addressing human adversaries in security games. *In ECAI*.

Ponsen, M., Jong, S. D., & Lanctot, M. (2011). Computing approximate nash equilibria and robust best-responses using sampling. *J. Artif. Intell. Res. (JAIR)*.

Prelec, D. (1998). The probability weighting function. *Econometrica*, *66*(3), 497–527.

Samuelson, P. (1937). A note on measurement of utility. *Review of Economic Studies*, *4*(2), 155–161.

See, K., Fox, C., & Rottenstreich, Y. (2006). Between ignorance and truth: Partition dependence and learning in judgment under uncertainty. *Journal of Experimental Psychology: Learning, Memory and Cognition*, *32*(6).

Seni, G., & Elder, J. F. (2010). Ensemble methods in data mining: Improving accuracy through combining predictions. *Synthesis Lectures on Data Mining and Knowledge Discovery*, *2*(1), 1–126.

Shieh, E. (2015). *Not a Lone Ranger: Unleashing Defender Teamwork in Security Games*. Ph.D. thesis, University of Southern California.

Shieh, E., An, B., Yang, R., Tambe, M., Baldwin, C., DiRenzo, J., Maule, B., & Meyer, G. (2012). PROTECT: A deployed game theoretic system to protect the ports of the United States. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '12, pp. 13–20.

Silver, R. C., Holman, E. A., McIntosh, D. N., Poulin, M., & Gil-Rivas, V. (2002). Nationwide longitudinal study of psychological responses to september 11. *Jama*, *288*(10), 1235–1244.

Skinner, B. F. (1938). The behavior of organisms: An experimental analysis. *New York: Appleton-Century*.

Skinner, B. F. (1948). Superstition' in the pigeon. *Journal of Experimental Psychology*, *38*, 168–172.

Skinner, B. F. (1953). Science and human behavior. *Simon and Schuster*.

Southers, E. (2011). Lax - terror target: The history. In Tambe, M. (Ed.), *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*, pp. 27–50. Cambridge University Press.

Tambe, M. (2011). *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. Cambridge University Press, New York, NY.

Tsai, J., Rathi, S., Kiekintveld, C., Ordonez, F., & Tambe, M. (2009). IRIS - a tool for strategic security allocation in transportation networks. In *The Eighth International Conference on Autonomous Agents and Multiagent Systems - Industry Track*, AAMAS '09, pp. 37–44.

Tsai, J., Yin, Z., young Kwak, J., Kempe, D., Kiekintveld, C., & Tambe, M. (2010). Urban security: Game-theoretic resource allocation in networked physical domains. In *National Conference on Artificial Intelligence (AAAI)*.

Tversky, A., & Kahneman, D. (1992). Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty*, *5*(4), 297–323.

Twisk, J., & de Vente, W. (2002). Attrition in longitudinal studies. *Journal of Clinical Epidemiology*, *55*(4), 329–337.

von Stackelberg, H. (1934). *Marktform und Gleichgewicht*. Springer, Vienna.

Wato, Y. A., Wahungu, G. M., & Okello, M. M. (2006). Correlates of wildlife snaring patterns in tsavo west national park, kenya. *Biological Conservation*, *132*(4), 500–509.

Wright, J. R., & Leyton-Brown, K. (2014). Level-0 meta-models for predicting human behavior in games. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*, EC '14.

Xu, H., Freeman, R., Conitzer, V., Dughmi, S., & Tambe, M. (2016). Signaling in bayesian stackelberg games. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Yang, R. (2014). *Human Adversaries in Security Games: Integrating Models of Bounded Rationality and Fast Algorithms*. Ph.D. thesis, University of Southern California.

Yang, R., Ford, B., Tambe, M., & Lemieux, A. (2014). Adaptive resource allocation for wildlife protection against illegal poachers. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Yang, R., Kiekintveld, C., Ordonez, F., Tambe, M., & John, R. (2011). Improving resource allocation strategy against human adversaries in security games. *In IJCAI*.

Yang, R., Kiekintveld, C., Ordonez, F., Tambe, M., & John, R. (2013). Improving resource allocation strategies against human adversaries in security games: An extended study. *Artif. Intell.*, *195*, 440–469.

Yang, R., Ordonez, F., & Tambe, M. (2012). Computing optimal strategy against quantal response in security games. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '12, pp. 847–854.

Yin, Z. (2013). *Addressing Uncertainty in Stackelberg Games for Security: Models and Algorithms*. Ph.D. thesis, University of Southern California.

Yin, Z., Jain, M., Tambe, M., & Ordonez, F. (2011). Risk-averse strategies for security games with execution and observational uncertainty. *In AAAI*.

Yin, Z., Jiang, A. X., Johnson, M. P., Kiekintveld, C., Leyton-Brown, K., Sandholm, T., Tambe, M., & Sullivan, J. P. (2012). TRUSTS: Scheduling randomized patrols for fare inspection in transit systems. In *Proceedings of the Twenty-Fourth Conference on Innovative Applications of Artificial Intelligence (IAAI)*, pp. 2348–2355.

Zhang, C., Bucarey, V., Mukhopadhyay, A., Sinha, A., Qian, Y., Vorobeychik, Y., & Tambe, M. (2016). Using abstractions to solve opportunistic crime security games at scale. In *International Conference on Autonomous Agents and Multiagent Systems*.

Zhang, C., Jiang, A. X., Short, M. B., Brantingham, P. J., & Tambe, M. (2014). Defending against opportunistic criminals: New game-theoretic frameworks and algorithms. In *International Conference on Decision and Game Theory for Security*.

Zhang, C., Sinha, A., & Tambe, M. (2015). Keeping pace with criminals: Designing patrol allocation against adaptive opportunistic criminals. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*.

Zollo, M. (2009). Superstitious learning with rare strategic decisions: Theory and evidence from corporate acquisitions. *Organization Science*, *20*(5), 894–908.