The Institute for Integrating Statistics in Decision Sciences

Technical Report TR-2017-6

Hypothesis Testing in Presence of Adversaries

Jorge Gonzalez-Ortega
ICMAT, Spain

David Ríos Insua
ICMAT, Spain

Fabrizio Ruggeri
CNR-IMATI, Italy

Refik Soyer
Department of Decision Sciences
The George Washington University, USA

Hypothesis Testing in Presence of Adversaries

Jorge González-Ortega[1], David Ríos Insua[1],
Fabrizio Ruggeri[2], Refik Soyer[3]

[1] Instituto de Ciencias Matemáticas, CSIC-UAM-UC3M-UCM, Madrid, Spain
[2] Istituto di Matematica Applicata e Tecnologie Informatiche, CNR, Milano, Italy
[3] Department of Decision Sciences, GWU, Washington, DC

Abstract

We consider the fundamental problem of hypothesis testing extended by including the decisions of an adversary which aims at distorting the relevant data process observed so as to confound the decision maker, thus attaining a certain benefit. We provide an adversarial risk analysis approach to this problem and illustrate its usage in a batch acceptance context.

Keywords: Statistical decision theory, Bayesian analysis, Game theory, Adversarial risk analysis, Security, Batch acceptance.

1 Introduction

Hypothesis testing is one of the fundamental problems in statistical inference, see French and Ríos Insua (2000). Though subject to debate, Berger and Sellke (1987) or Berger (2003), it has been thoroughly studied from a decision theoretical perspective, both from the frequentist and Bayesian points of view, following the seminal work of Wald (1950).

In recent years, there has been an increasing interest in issues related with hypothesis testing problems in which hostile adversaries perturb the data observed by a decision maker as a way to confound her about the relevant hypothesis so as to attain some objectives. Examples come from the fields of adversarial signal processing, see Barni and Pérez-González (2013) for an introduction; adversarial classification, see the pioneer work in Dalvi et al. (2004); and adversarial machine learning, see e.g. Tygar (2011). These cover applications like online fraud detection, watermarking or spam detection, among many others.

Most attempts in this area have focused on game theoretic approaches to hypothesis testing, with the entailed common knowledge assumptions. These normally involve assuming that adversaries not only know their own payoffs, preferences, beliefs and possible actions, but also those of their opponents. For example, Barni and Tondi (2014) provide a framework focusing on zero-sum game theoretic minimax approaches to hypothesis testing. This is not satisfactory since losses for various participants will

be typically asymmetric, and, moreover, the beliefs and preferences of the adversary will not be readily available, frequently violating the above mentioned common knowledge assumptions, see Hargreaves-Heap and Varoufakis (1995). Thus, key assumptions of the customarily proposed solution approaches would not hold.

In this paper, using concepts from Adversarial Risk Analysis (ARA), see Ríos Insua et al. (2009), we provide an alternative novel approach to the Adversarial Hypothesis Testing (AHT) problem. We consider an agent, called the defender (D, she), who needs to assess which of several hypotheses holds, based on observations from a source that might have been perturbed by another agent, which we designate attacker (A, he). We study the AHT problem from the defender's perspective. In doing this, the defender formulates a Bayesian decision making problem but requires to forecast the attacker's decision. We make such forecast by simulating from the attacker's problem, taking into account our uncertainty over the attacker's beliefs and preferences.

We begin by introducing what we term the Adversarial Statistical Decision Theory (ASDT) problem in Section 2, extending the standard Statistical Decision Theory (SDT) formulation to consider an adversarial variation in which the attacker tries to modify the dataflow observed by the defender to confound her and, consequently, attain a profit. In Section 3, we pose the AHT problem formally and provide a conceptual solution focusing on binary point hypothesis testing, as well as illustrating it with a simple numerical example and presenting a game theoretic perspective for comparison purposes. Section 4 describes in depth an application in relation with batch acceptance. We conclude with a discussion of several potential applications and other open issues.

2 Adversarial Statistical Decision Theory

As a motivation, we include first a brief discussion of the standard Bayesian SDT framework. As illustrated in the Influence Diagram (ID) in Figure 1, we consider a decision maker D who needs to make a decision d based on an observation x which depends on a state θ taking values in a set Θ. She obtains a loss $l_D(d, \theta)$ which depends on the decision she makes and the state actually occurring.

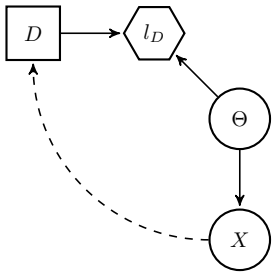


Figure 1: Sketch of the general SDT problem.

To solve her decision making problem, she could describe her prior beliefs over state θ through the prior $p_D(\theta)$ and the dependence of data x on the state θ through the likelihood $p_D(x \,|\, \theta)$. Given such elements, she would seek the decision $d^*(x)$ that minimizes her expected loss, given x, which is

$$d^*(x) = \arg\min_d \int l_D(d, \theta)\, p_D(\theta \,|\, x)\, \mathrm{d}\theta.$$

Note that, for optimization purposes, we may ignore the denominator in Bayes formula and solve the equivalent problem

$$d^*(x) = \arg\min_d \int l_D(d, \theta)\, p_D(x \mid \theta)\, p_D(\theta)\, \mathrm{d}\theta.$$

This general framework covers most standard statistical problems including point estimation, set estimation, hypothesis testing, forecasting and decision analysis. All of the above is reviewed in detail in e.g. French and Ríos Insua (2000).

2.1 ASDT: A data manipulating opponent

There are several possible variants of the SDT framework which take into account the presence of an intelligent adversary. Of them, we shall consider the case in which an opponent A is able to modify the data observed by the decision maker D in an attempt to confound her and, consequently, acquire some advantage.

The problem is depicted in Figure 2 through a Bi-Agent Influence Diagram (BAID), see Koller and Milch (2003), which represents the decisions of both agents, D and A. White nodes correspond to D; grey nodes, to A; and striped nodes are shared by both agents. Square nodes refer to decisions, circle nodes to uncertainties and, finally, hexagonal nodes to losses. Arrows represent conditional relations, except for dashed arrows which depict the available information at decision nodes. Depending on the uncertain true state $\theta \in \Theta$ some original data x is derived, which gets perturbed to y by the attacker's action a. Then, y is observed by the defender, who makes her decision without knowing either x or θ.

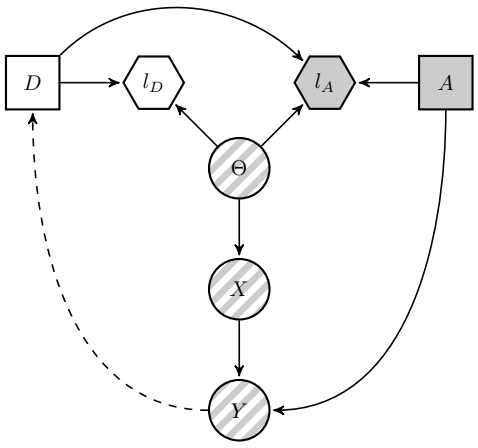


Figure 2: BAID for the data manipulation problem.

As an example, a security agent D may be screening incoming emails. She does not know θ, an indicator of potential security issues associated with the mail. Her observation could be based on the length of the email, the types of attachments, the presence of certain words, the sending address, etc. A data manipulating opponent might perturb that data through subterfuge, e.g. adding or deleting certain words, using an apparently legal sending address and so on.

In this framework, A makes his decision a first, then D makes her decision after observing the manipulated data, and, finally, both agents receive their losses. In general, we assume that the attacker must allow for some loss corresponding to the resources or effort spent in manipulating the data. This is reflected by the dependence of his loss on the action he implements. For example, it costs money to purchase an IP, some time to appropriately craft the email and it is a crime to forge an email, and all constitute a real or potential loss. Opportunity costs may all also be taken into account through this dependence.

The problem the defender needs to solve is described in the ID in Figure 3a. Since she does not know her opponent's decision, his decision node (the circled A) appears as random to her. In a standard decision theoretic approach, D would solve

$$d^*(y) = \arg\min_d \int l_D(d,\theta)\, p_D(\theta \,|\, y)\, \mathrm{d}\theta.$$

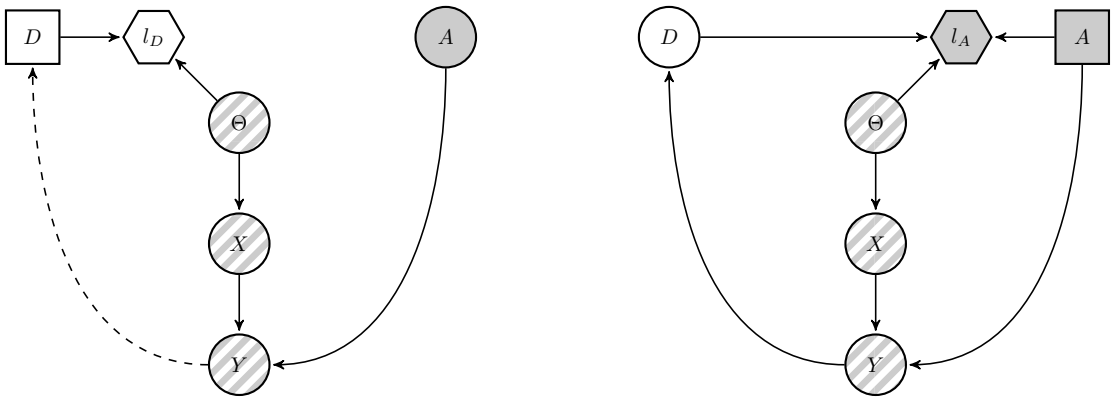
We know that

$$p_D(\theta \,|\, y) = \frac{p_D(\theta, y)}{p_D(y)} = \frac{\iint p_D(y \,|\, x, a)\, p_D(x \,|\, \theta)\, p_D(\theta)\, p_D(a)\, \mathrm{d}x\, \mathrm{d}a}{p_D(y)},$$

so her optimal decision is obtained by solving the equivalent problem

$$d^*(y) = \arg\min_d \iiint l_D(d,\theta)\, p_D(y \,|\, x, a)\, p_D(x \,|\, \theta)\, p_D(\theta)\, p_D(a)\, \mathrm{d}x\, \mathrm{d}\theta\, \mathrm{d}a. \tag{1}$$

Of all the assessments required to evaluate the ID, $l_D(d,\theta)$, $p_D(y \,|\, x, a)$, $p_D(x \,|\, \theta)$ and $p_D(\theta)$ are standard in Bayesian SDT. The only distinctive one is $p_D(a)$ (D's forecast over the action a) as it entails strategic thinking.



(a) D's decision analysis (b) A's decision analysis

Figure 3: Both agents' IDs for the data manipulation model.

The ARA approach to ASDT determines $p_D(a)$ by focusing on the problem that the attacker solves, represented in Figure 3b. This analysis assumes that he wants to minimize his expected loss. Also note that, unlike the defender, the attacker's actions are specified before observing the data. For his decision theoretic solution, A solves

$$a^* = \arg\min_a \iiint l_A(d,a,\theta)\, p_A(d \,|\, y)\, p_A(y \,|\, x, a)\, p_A(x \,|\, \theta)\, p_A(\theta)\, \mathrm{d}y\, \mathrm{d}x\, \mathrm{d}\theta.$$

However, D lacks knowledge about the probabilities and loss function used by A. Suppose she models her uncertainty about them through random probabilities and losses $F \sim \big(L_A(d,a,\theta), P_A(d\,|\,y), P_A(y\,|\,x,a), P_A(x\,|\,\theta), P_A(\theta)\big)$. Then, she would solve

$$A^* = \arg\min_a \iiint L_A(d,a,\theta)\, P_A(d\,|\,y)\, P_A(y\,|\,x,a)\, P_A(x\,|\,\theta)\, P_A(\theta)\,\mathrm{d}y\,\mathrm{d}x\,\mathrm{d}\theta,$$

to find the optimal random decision A^*, whose distribution is induced by the above random probabilities and loss function in F. Thus, the defender has found the distribution $p_D(a) = P(A^* = a)$ that she needs to calculate her best decision $d^*(y)$. That distribution properly incorporates all of her uncertainty about the attacker's situation.

In general, to approximate $p_D(a)$, one will typically use simulation, by drawing K samples $\big(L_A^k(d,a,\theta), P_A^k(d\,|\,y), P_A^k(y\,|\,x,a), P_A^k(x\,|\,\theta), P_A^k(\theta)\big)$, $k = 1,\ldots,K$ from F, finding

$$A_k^* = \arg\min_a \iiint L_A^k(d,a,\theta)\, P_A^k(d\,|\,y)\, P_A^k(y\,|\,x,a)\, P_A^k(x\,|\,\theta)\, P_A^k(\theta)\,\mathrm{d}y\,\mathrm{d}x\,\mathrm{d}\theta,$$

and approximating

$$\hat{p}_D(A \le a) \approx \#\{A_k^* \le a\}/K.$$

Within F, four of the elements are relatively easy to model, see Banks et al. (2015):

- $P_A(\theta)$ could be based on $p_D(\theta)$, with some uncertainty about it. For example, should $p_D(\theta)$ be a discrete distribution, $P_A(\theta)$ could be modeled as a Dirichlet distribution with mean $p_D(\theta)$. Similarly, should $p_D(\theta)$ be a continuous distribution, $P_A(\theta)$ could be modeled as a Dirichlet process with base measure $p_D(\theta)$.

- This would also be the case for $P_A(y\,|\,x,a)$ which could be based on $p_D(y\,|\,x,a)$, with some uncertainty around it.

- Analogously, $P_A(x\,|\,\theta)$ could be based on $p_D(x\,|\,\theta)$, with additional uncertainty about it (although in many cases it will be reasonable to assume that they actually coincide).

- For $L_A(d,a,\theta)$, one could typically reflect upon the adversary's interests, formulate a parametric form for the loss function, and assess a subjective distribution over its parameters.

On the other hand, $P_A(d\,|\,y)$ is not easy to assess. It entails strategic thinking since the defender needs to understand her opponent's beliefs about what decision she will make given that she observes y. This could be the beginning of a hierarchy of decision making problems; see Ríos and Ríos Insua (2012) for a description of the potentially infinite regress in a simpler class of problems. We illustrate here just the next stage of the hierarchy for our case. Note that in expression (1) to be solved by D, the adversary A does not know the terms in the integral. By assuming uncertainty over them through random distributions $P_D^A(y\,|\,x,a)$, $P_D^A(x\,|\,\theta)$, $P_D^A(\theta)$ and $P_D^A(a)$ and a random loss $L_D^A(d,\theta)$, he would get the corresponding random optimal decision by replacing the corresponding elements. Again, this requires assessment of $P_D^A(a)$ (what the defender believes that the attacker thinks about her beliefs concerning his action to be implemented) for which

there is a strategic component, leading to the next stage in the hierarchy. Within the pertinent iteration in the loop, one could stop at a level in which no more information is reasonably available. At that stage, one could use a non-informative prior over the involved probabilities and losses.

We adapt now this ASDT framework to the hypothesis testing context.

3 Adversarial Hypothesis Testing

We shall focus on the problem of testing two simple hypotheses described by $\Theta = \{\theta_0, \theta_1\}$. As an example, suppose the defender needs to decide whether a batch of e-mails includes spam or not. She has beliefs about the standard flow of legit and spam messages. The attacker perturbs such flow by adding, deleting or modifying some of the messages, in an attempt to confound the defender and obtain some benefit. Both agents get different rewards depending on whether the batch is accepted or not by the defender and the batch includes just legit messages or not.

The backbone structure of the AHT problem coincides with that in Figure 2. Depending on the uncertain hypothesis $\theta \in \Theta$, there will be an observation data flow x which gets perturbed to y by the attacker's action a. The perturbed data flow y is observed by the defender, who needs to decide which is the relevant hypothesis. She makes such decision d without observing neither x nor θ. Depending on d, and the actual hypothesis θ, both agents receive the corresponding losses. Besides, we assume that the attacker spends some effort in performing the attack, as reflected by the dependence of his loss on the attack he implements. Our aim is to support the defender in deciding which is the appropriate hypothesis.

3.1 Solving the defender's problem

The problem the defender needs to solve was described in Figure 3a. Now, her decision space is $\mathcal{D} = \{d_0, d_1\}$, with d_j representing her support for θ_j, $j = 0, 1$. Following a standard Bayesian decision theoretic approach, assume that we may elicit from the defender the following judgements:

$\mathcal{D}1$. At node Θ, $p_D(\theta)$ models her beliefs about the various hypotheses. We designate such beliefs

$$p_D(\theta = \theta_i) = \pi_D^i, \quad i = 0, 1,$$

with $\pi_D^i \geq 0$ and $\pi_D^0 + \pi_D^1 = 1$.

$\mathcal{D}2$. At node X, $p_D(x \,|\, \theta)$ represents her beliefs about how data would depend on the hypothesis, described by

$$X \,|\, \theta_i \sim p_D(x \,|\, \theta_i), \quad i = 0, 1.$$

$\mathcal{D}3$. At node Y, $p_D(y \,|\, x, a)$ models her beliefs about how data will be perturbed: it reflects her notion about what would the observed y be, if x is the actual data and a is the attacker's selected action.

$\mathcal{D}4$. At node A, $p_D(a)$ represents her beliefs about which attack a would be undertaken by the attacker.

$\mathcal{D}5$. At node l_D, $l_D(d, \theta)$ models the defender's loss function. We use a standard 0-1-c_D loss as in Table 1, where 0 is the best loss (associated with a system functioning as expected), and 1 is the worst loss (associated with a non-functioning system). We assume that $c_D \leq 1$.

		Actual Hypothesis	
		θ_0	θ_1
D's Decision	d_0	0	1
	d_1	c_D	0

Table 1: Defender's loss function.

The defender would then solve

$$\arg\min_{d \in \mathcal{D}} \sum_{i=0}^{1} l_D(d, \theta_i)\, p_D(\theta_i \,|\, y).$$

After simple computations, it follows that the optimal decision for the defender would be to support θ_0 if

$$p_D(\theta_1 \,|\, y) \leq c_D\, p_D(\theta_0 \,|\, y).$$

We have that

$$p_D(\theta_i \,|\, y) = \frac{p_D(\theta_i, y)}{p_D(y)} = \frac{\pi_D^i \iint p_D(y \,|\, x, a)\, p_D(x \,|\, \theta_i)\, p_D(a)\, \mathrm{d}x\, \mathrm{d}a}{p_D(y)}, \quad i = 0, 1.$$

Therefore, the optimal decision for the defender is to support θ_0 if

$$\begin{gathered} \pi_D^1 \iint p_D(y \,|\, x, a)\, p_D(x \,|\, \theta_1)\, p_D(a)\, \mathrm{d}x\, \mathrm{d}a \\ \leq \\ c_D\, \pi_D^0 \iint p_D(y \,|\, x, a)\, p_D(x \,|\, \theta_0)\, p_D(a)\, \mathrm{d}x\, \mathrm{d}a. \end{gathered} \tag{2}$$

Among the required assessments $\mathcal{D}1$–$\mathcal{D}5$, as in Section 2.1, the only non-standard one is $\mathcal{D}4$ referring to $p_D(a)$ – the defender's forecast over the attacks a – as it entails strategic thinking. We facilitate its estimation by considering the problem that the attacker should solve.

3.2 Modeling the attacker's problem

Figure 3b provided the influence diagram of the attacker's decision making problem, assuming that he aims at minimising expected loss. For its decision theoretic solution, being the attacker's decision space $\mathcal{A}$, the attacker would need:

$\mathcal{A}1$. At node Θ, $p_A(\theta)$ models his beliefs about the likelihood of the hypotheses, which we designate

$$p_A(\theta = \theta_i) = \pi_A^i, \quad i = 0, 1,$$

with $\pi_A^i \geq 0$ and $\pi_A^0 + \pi_A^1 = 1$.

$\mathcal{A}2$. At node X, $p_A(x \,|\, \theta_i)$ reflects his beliefs about the dataflow, for each hypothesis θ_i, $i = 0, 1$.

$\mathcal{A}3$. At node Y, $p_A(y \,|\, x, a)$ represents his beliefs about what would the effect of his actions be in transforming the data.

$\mathcal{A}4$. At node D, $p_A(d \,|\, y)$ reflects his beliefs about the defender's decision d provided that she observes y.

$\mathcal{A}5$. At node l_A, $l_A(d, a, \theta)$ models the attacker's loss function, with form as in Table 2. Typically, it will be $l_{00}(a) \geq l_{01}(a)$ and $l_{10}(a) \leq l_{11}(a)$, since it is better for the attacker when the defender makes mistakes.

		Actual Hypothesis	
		θ_0	θ_1
D's Decision	d_0	$l_{00}(a)$	$l_{01}(a)$
	d_1	$l_{10}(a)$	$l_{11}(a)$

Table 2: Attacker's loss function, given attack a.

An important case is described in Table 3, where $0 \leq c_A^0 \leq c_A^1 \leq 1$ to reflect that the best loss for the attacker (0) is attained when the defender supports θ_0, and she should not, while the worst (1) holds when the defender supports θ_0, and she should. The intermediate cases reflect that it is worse for the attacker that the defender supports θ_1 when the actual hypothesis is θ_1 (taking into account the attacker's costs and the induction of a feeling of insecurity) than when it is θ_0 (no costs for the attacker and sense of security).

		Actual Hypothesis	
		θ_0	θ_1
D's Decision	d_0	1	0
	d_1	c_A^0	c_A^1

Table 3: Attacker's loss function.

Should the above assessments be available, the optimal decision a^* for him would be

$$a^* = \arg\min_{a\in\mathcal{A}} \sum_{j=0}^{1}\sum_{i=0}^{1} \iint l_A(d_j, a, \theta_i)\, p_A(d_j \,|\, y)\, p_A(\theta_i)\, p_A(y \,|\, x, a)\, p_A(x \,|\, \theta_i)\, \mathrm{d}y\, \mathrm{d}x.$$

However, the defender lacks knowledge about assessments $\mathcal{A}1$–$\mathcal{A}5$ for the attacker. As in Section 2.1, suppose we are capable of modeling her uncertainty through random probabilities P_A and losses L_A and finding the optimal random attack

$$A^* = \arg\min_{a\in\mathcal{A}} \sum_{j=0}^{1}\sum_{i=0}^{1} \iint L_A(d_j, a, \theta_i)\, P_A(d_j \,|\, y)\, P_A(\theta_i)\, P_A(y \,|\, x, a)\, P_A(x \,|\, \theta_i)\, \mathrm{d}y\, \mathrm{d}x. \quad (3)$$

Then, we have the required distribution through

$$p_D(a) = P(A^* = a),$$

assuming that $\mathcal{A}$ is discrete, and, similarly, if it is continuous.

3.3 AHT: A numerical example

We illustrate the previous ideas with a numerical example in which the defender monitors continuous positive observations perturbed by an attacker. The two entertained hypotheses are $\theta_0 = 2$ and $\theta_1 = 1$. We display the elements introduced in Section 3 for the defender's problem:

$\mathcal{D}1$. As for the priors over the hypotheses, we assume that both are equally likely a priori, so that $\pi_D^0 = \pi_D^1 = 1/2$.

$\mathcal{D}2$. The defender receives data $X \,|\, \theta_i$ exponentially distributed $\mathcal{E}(\theta_i)$, with uncertainty about the parameter θ_i.

$\mathcal{D}3$. The attacker can modify the data according to a strategy which allows for keeping, doubling or halving x. We call such actions a_0, a_1 and $a_{\text{-}1}$, respectively. Thus, if x is the actual value, the defender will observe $y = x$ if the attacker chooses a_0, whereas $y = 2x$ and $y = x/2$ will be the observed values if the attacker chooses a_1 and $a_{\text{-}1}$, respectively. Then, the distributions $p_D(y \,|\, x, a)$ are Dirac measures correspondingly assigning probability 1 to $(y = x,\, a_0)$, $(y = 2x,\, a_1)$ and $(y = x/2,\, a_{\text{-}1})$.

$\mathcal{D}4$. As an illustration, we start considering the case in which the defender knows the probabilities $p_D(a)$ with which the attacker chooses among his actions. Suppose, for the moment, that $p_D(a_0) = 1/2$, $p_D(a_1) = 1/6$ and $p_D(a_{\text{-}1}) = 1/3$.

$\mathcal{D}5$. We consider the loss function in Table 1, with $c_D = 3/4$.

Recall condition (2), leading the defender to adopt decision d_0 (accept θ_0). In this case, using $\mathcal{D}2$, $\mathcal{D}3$ and $\mathcal{D}5$, such condition becomes

$$\pi_D^1 \left[\theta_1\, e^{-\theta_1 y}\, p_D(a_0) + \theta_1\, e^{-\theta_1 \frac{y}{2}}\, p_D(a_1) + \theta_1\, e^{-\theta_1 2y}\, p_D(a_{\text{-}1})\right]$$
$$\leq$$
$$\tfrac{3}{4}\, \pi_D^0 \left[\theta_0\, e^{-\theta_0 y}\, p_D(a_0) + \theta_0\, e^{-\theta_0 \frac{y}{2}}\, p_D(a_1) + \theta_0\, e^{-\theta_0 2y}\, p_D(a_{\text{-}1})\right].$$

Plugging in the values of θ_0 and θ_1 and the probabilities in $\mathcal{D}1$, we get

$$\begin{gathered}\tfrac{1}{2}\left[p_D(a_0)\,e^{-y}+p_D(a_1)\,e^{-\frac{y}{2}}+p_D(a_{\text{-}1})\,e^{-2y}\right]\\ \leq\\ \tfrac{3}{8}\left[2p_D(a_0)\,e^{-2y}+2p_D(a_1)\,e^{-y}+2p_D(a_{\text{-}1})\,e^{-4y}\right].\end{gathered}\tag{4}$$

Finally, we incorporate the values in $\mathcal{D}4$, so that

$$\frac{1}{2}\left[\frac{1}{2}e^{-y}+\frac{1}{6}e^{-\frac{y}{2}}+\frac{1}{3}e^{-2y}\right]\leq\frac{3}{8}\left[e^{-2y}+\frac{1}{3}e^{-y}+\frac{2}{3}e^{-4y}\right],$$

which gets simplified to checking the inequality

$$2e^{-\frac{y}{2}}+3e^{-y}-5e^{-2y}-6e^{-4y}\leq 0.$$

We can show that decision d_0 should be made when a value $y\lesssim 0.37$ is observed. However, note that a slight change of the parameters might produce a completely different result. For example, with $\pi_D^0=1/3$ (and $\pi_D^1=2/3$), and all other probabilities and costs as above, d_1 is optimal regardless of the observed y.

Consider now the case in which the defender does not accurately know $p_D(a)$ ($\mathcal{D}4$). We resort to an ARA. Suppose the following assessments are made:

$\mathcal{A}1$. The defender assumes $P_A(\theta_1)$ is drawn uniformly over the interval $[1/4,3/4]$ (and $P_A(\theta_0)=1-P_A(\theta_1)$).

$\mathcal{A}2$. We model the defender's knowledge of $P_A(x\,|\,\theta)$, where $\theta\in\{\theta_0,\theta_1\}$, as a Gamma distribution $\mathcal{G}a(\alpha,\beta)$ with mean $\theta=\alpha/\beta$ and variance $\sigma^2=\alpha/\beta^2$ uniformly chosen over the interval $[1/2,2]$. This variance randomness induces that of $P_A(x\,|\,\theta)$.

$\mathcal{A}3$. $P_A(y\,|\,x,a)$ will be Dirac distributions, coinciding with $p_D(y\,|\,x,a)$.

$\mathcal{A}4$. We build $P_A(d\,|\,y)$ based on the likelihood $h(y\,|\,d,a)$ of y under different choices of d and a, mixing them through a random allocation of probabilities to each action. Suppose the attacker assumes the defender is modeling the data with an exponential distribution, with the defender assessing the probabilities $(\epsilon_0,\epsilon_1,\epsilon_{\text{-}1})$ assigned by the attacker to each strategy through a Dirichlet distribution $\mathcal{D}ir(1,1,1)$. Then, $P_A(d=d_1\,|\,y)$ has the form

$$\begin{aligned}g(\epsilon_0,\epsilon_1,\epsilon_{\text{-}1},y) &= \frac{\sum_{j=\text{-}1}^{1}\epsilon_j\,h(y\,|\,d_1,a_j)}{\sum_{j=\text{-}1}^{1}\epsilon_j\,h(y\,|\,d_0,a_j)+\sum_{j=\text{-}1}^{1}\epsilon_j\,h(y\,|\,d_1,a_j)}\\ &= \frac{\epsilon_0\,e^{-y}+\epsilon_1\,e^{-\frac{y}{2}}+\epsilon_{\text{-}1}\,e^{-2y}}{2\left(\epsilon_0\,e^{-2y}+\epsilon_1\,e^{-y}+\epsilon_{\text{-}1}\,e^{-4y}\right)+\epsilon_0\,e^{-y}+\epsilon_1\,e^{-\frac{y}{2}}+\epsilon_{\text{-}1}\,e^{-2y}}.\end{aligned}$$

The distribution of $(\epsilon_0,\epsilon_1,\epsilon_{\text{-}1})$ induces the randomness of $P_A(d=d_1\,|\,y)$. Finally, $P_A(d=d_0\,|\,y)=1-P_A(d=d_1\,|\,y)$.

$\mathcal{A}5$. The random loss function $L_A(d,a,\theta)$ is based on Table 3, where C_A^0 is degenerated at 0 and C_A^1 is uniformly distributed over $[1/2,1]$.

Taking into account assessments $\mathcal{A}3$ and $\mathcal{A}5$ and expression (3), the attacker's random expected losses for the three actions will be:

$$\Psi_A(a_0) = \int \left[P_A(d_0 \,|\, y = x)\, P_A(\theta_0)\, P_A(x \,|\, \theta_0) + C_A^1\, P_A(d_1 \,|\, y = x)\, P_A(\theta_1)\, P_A(x \,|\, \theta_1)\right] \mathrm{d}x$$

$$\Psi_A(a_1) = \int \left[P_A(d_0 \,|\, y = 2x)\, P_A(\theta_0)\, P_A(x \,|\, \theta_0) + C_A^1\, P_A(d_1 \,|\, y = 2x)\, P_A(\theta_1)\, P_A(x \,|\, \theta_1)\right] \mathrm{d}x$$

$$\Psi_A(a_{\text{-}1}) = \int \left[P_A(d_0 \,|\, y = \tfrac{x}{2})\, P_A(\theta_0)\, P_A(x \,|\, \theta_0) + C_A^1\, P_A(d_1 \,|\, y = \tfrac{x}{2})\, P_A(\theta_1)\, P_A(x \,|\, \theta_1)\right] \mathrm{d}x$$

The random models in $\mathcal{A}1$, $\mathcal{A}2$, $\mathcal{A}4$ and $\mathcal{A}5$ induce the randomness in these expected losses. We estimate the attack probabilities as follows:

Algorithm 1 AHT: Numerical example - Simulating the attacker's problem.

Data: Considered hypotheses θ_0 and θ_1; number of iterations K.

Set $p_j = 0, \quad j = \text{-}1, 0, 1.$

For $k = 1$ **to** K

  Generate $\pi_A^{1,k} \sim \mathcal{U}(1/4, 3/4)$. Compute $\pi_A^{0,k} = 1 - \pi_A^{1,k}$.

  Generate $\sigma_{i,k}^2 \sim \mathcal{U}(1/2, 2)$. Compute $\alpha_i^k = \theta_i^2/\sigma_{i,k}^2$; $\beta_i^k = \theta_i/\sigma_{i,k}^2, \quad i = 0, 1.$

  Generate $(\epsilon_0^k, \epsilon_1^k, \epsilon_{\text{-}1}^k) \sim \mathcal{D}ir(1, 1, 1)$ and $C_A^{1,k} \sim \mathcal{U}(1/2, 1)$.

  $$\psi_A^k(a_0) = \pi_A^{0,k} \int (1 - g(\epsilon_0^k, \epsilon_1^k, \epsilon_{\text{-}1}^k, x))\, f(x \,|\, \alpha_0^k, \beta_0^k)\, \mathrm{d}x + C_A^{1,k}\, \pi_A^{1,k} \int g(\epsilon_0^k, \epsilon_1^k, \epsilon_{\text{-}1}^k, x)\, f(x \,|\, \alpha_1^k, \beta_1^k)\, \mathrm{d}x.$$

  $$\psi_A^k(a_1) = \pi_A^{0,k} \int (1 - g(\epsilon_0^k, \epsilon_1^k, \epsilon_{\text{-}1}^k, 2x))\, f(x \,|\, \alpha_0^k, \beta_0^k)\, \mathrm{d}x + C_A^{1,k}\, \pi_A^{1,k} \int g(\epsilon_0^k, \epsilon_1^k, \epsilon_{\text{-}1}^k, 2x)\, f(x \,|\, \alpha_1^k, \beta_1^k)\, \mathrm{d}x.$$

  $$\psi_A^k(a_{\text{-}1}) = \pi_A^{0,k} \int (1 - g(\epsilon_0^k, \epsilon_1^k, \epsilon_{\text{-}1}^k, x/2))\, f(x \,|\, \alpha_0^k, \beta_0^k)\, \mathrm{d}x + C_A^{1,k}\, \pi_A^{1,k} \int g(\epsilon_0^k, \epsilon_1^k, \epsilon_{\text{-}1}^k, x/2)\, f(x \,|\, \alpha_1^k, \beta_1^k)\, \mathrm{d}x.$$

  Find $j^* = \underset{j \in \{\text{-}1,0,1\}}{\arg\min}\, \psi_A^k(a_j)$.

  Set $p_{j^*} = p_{j^*} + 1$.

End For

Set $\hat{p}_D(a_j) = p_j/K, \quad j = \text{-}1, 0, 1.$

An application of the previous scheme with $K = 10^5$ leads to estimates $\hat{p}_D(a_0) \approx 0.04$, $\hat{p}_D(a_1) \approx 0.85$ and $\hat{p}_D(a_{\text{-}1}) \approx 0.11$. Plugging such values in (4), the optimal decision is d_0 when a value $y \lesssim 0.74$ is observed, which differs from the Bayesian solution obtained earlier.

3.4 A game theoretic perspective

In order to provide a broader understanding of the benefits of the ARA approach to the AHT problem, we present here an alternative game theoretic perspective to it.

First, recall the standard SDT framework illustrated in Figure 1. The ARA approach required the decision maker to model her prior beliefs $p_D(\theta)$ and $p_D(x \mid \theta)$. However, in the absence of such priors, she could apply a game theoretic approach by means of the minimax model

$$d^* = \arg\min_d \max_\theta l_D(d, \theta).$$

Unfortunately, this worst case scenario approach would neglect all information that could be derived from observing data x.

Now, consider the AHT problem posed in Sections 3.1 and 3.2. To avoid ignoring data y, we assume that priors for $p_D(\theta)$, $p_D(x \mid \theta)$, $p_D(y \mid x, a)$ and $p_A(\theta)$ are available. Using a game theoretic approach, we redefine both agent's loss functions in terms of their combined decisions. That is, if the defender's decision is d and the attacker's is a, the defender's (equivalent) loss function is defined as

$$\Psi_D(d, a, y) = \sum_{i=0}^{1} \pi_D^i \int l_D(d, \theta_i)\, p_D(y \mid x, a)\, p_D(x \mid \theta_i)\, \mathrm{d}x;$$

and the attacker's as

$$\Psi_A(d, a) = \sum_{i=0}^{1} \pi_A^i\, l_A(d, a, \theta_i).$$

Under common knowledge assumptions, if a Nash equilibrium $(d^*(y), a^*)$ exists, then it must satisfy

$$\Psi_D(d^*(y), a^*, y) \leq \Psi_D(d, a^*, y),\ \forall d \in \mathcal{D}; \qquad \Psi_A(d^*(y), a^*) \leq \Psi_D(d^*(y), a),\ \forall a \in \mathcal{A}.$$

With an illustrative purpose, we can replicate the numerical example in Section 3.3 making use of this game theoretic approach. The elements involved in the defender's problem will actually coincide, and those required for the attacker's problem will be based on the assessments made by the defender employing the means of the specified probability distributions. Thus, $\pi_A^1 = E\left[\mathcal{U}(1/4, 3/4)\right] = 1/2 = \pi_A^0$, $C_A^0 = 0$ and $C_A^1 = E\left[\mathcal{U}(1/2, 1)\right] = 3/4$. The defender's loss function is specified then as

$$\Psi_D(d_0, a_0, y) = \frac{e^{-y}}{2}, \qquad \Psi_D(d_0, a_1, y) = \frac{e^{-\frac{y}{2}}}{2}, \qquad \Psi_D(d_0, a_{\text{-}1}, y) = \frac{e^{-2y}}{2},$$
$$\Psi_D(d_1, a_0, y) = \frac{3e^{-2y}}{4}, \qquad \Psi_D(d_1, a_1, y) = \frac{3e^{-y}}{4}, \qquad \Psi_D(d_1, a_{\text{-}1}, y) = \frac{3e^{-4y}}{4};$$

and the attacker's as

$$\Psi_A(d_0, a) = \frac{1}{2}, \qquad \Psi_D(d_1, a) = \frac{3}{8}, \qquad \forall a \in \{0, 1, \text{-}1\}.$$

We can only find a mixed strategies Nash equilibrium in which the agents choose each of three actions with probability 1/3. The attacker relies on any of his three possible

attacks (a_0, a_1 and $a_{\text{-}1}$) and the defender follows one of the three respectively matching decision rules: choose d_0 if $y \lesssim 0.41$ (s_0), choose d_0 if $y \lesssim 0.81$ (s_1)), and choose d_0 if $y \lesssim 0.20$ ($s_{\text{-}1}$). The defender could also deviate from the mixed strategy and just adopt one of the decision rules in terms of her risk attitude, where $s_{\text{-}1}$ is the most conservative, s_1 the least and s_0 in between.

4 A Batch Acceptance Model

As an example of application of the approach in Section 3, we consider now a model for batch acceptance. The problem we deal with is deciding whether to accept a batch of items received over a period of time, some of which could be faulty, thus entailing potential security and/or performance problems. This type of issues arise in areas such as screening containers at international ports, filtering batches of electronic messages or admitting packages of perishable products or electronic components, among others. The main difference with the general AHT problem in Section 3 is that, in this case, the effect of the defender's decision does not depend on the parameters but on the observed data. We first outline a non-adversarial problem which we then modify to include adversaries.

4.1 Problem setting

The problem we initially face is sketched in Figure 4. Its structure is similar to the SDT problem depicted in Figure 1, except for two key differences.

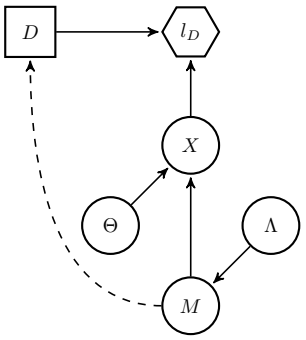


Figure 4: ID for the batch acceptance problem without adversaries.

One is minor, since we consider two influencing parameters, which we call θ and λ, and two pieces of data: the batch size m (observed when making the decision) and the batch composition (unobserved when making the decision) with x acceptable items and $m - x$ unacceptable (faulty) items. The second one is major, since the consequences do not directly depend on the parameters, but will be determined by the data (in particular, by the presence of faulty items).

The problem is specified as follows:

- A decision maker D (the defender) receives a batch with two types of items: 0, which we associate with acceptable items; and 1, corresponding to faulty ones. She needs to decide whether to accept (d_0) or reject (d_1) the batch.

- The defender observes the size m of the batch, which is related with a parameter λ. To fix ideas, assume that, over a period of duration 1, the number of items follow a Poisson distribution $M \,|\, \lambda \sim \mathcal{P}o(\lambda)$. We consider that the prior over λ is a Gamma distribution $\mathcal{G}a(a, b)$. After t periods in which, in total, r items have arrived, the posterior is $\Lambda \,|\, t, r \sim \mathcal{G}a(a + r, b + t)$. Note that λ will have no impact in the non-adversarial problem, as the defender observes the actual value of m. However, it will provide useful information about m in the adversarial version considered in Section 4.2.

- The probability that an item is acceptable is determined by θ. If we use Z to designate this ($z = 0$, an acceptable item; $z = 1$, otherwise), we then have $p_D(z = 0 \,|\, \theta) = \theta$. The number of acceptable items will have a binomial distribution $X \,|\, m, \theta \sim \mathcal{B}in(m, \theta)$. To complete model specification, we assume that we have prior beliefs about θ modeled through a Beta distribution $\mathcal{B}e(\alpha, \beta)$. Suppose that after receiving r items, s have been acceptable (and $r - s$, faulty). Then, we update to the posterior $\Theta \,|\, r, s \sim \mathcal{B}e(\alpha + s, \beta + r - s)$.

As for the loss function l_D, we may consider numerous scenarios. We describe two, although we shall only use the first one in the adversarial problem in Section 4.2.

4.1.1 Scenario A: Winner takes it all

We receive a batch with m items in a given period. In this scenario, just allowing one faulty item is as bad as allowing several of them, because of the entailed security or performance problems. The loss structure is displayed in Table 4, where c describes the (expected) opportunity costs associated with rejecting a batch with all acceptable items.

		Batch of m Items		
		All Acceptable	Some Faulty	
		$p = \theta^m$	$p = 1 - \theta^m$	**Exp. Loss**
D's Decision	Accept, d_0	0	1	$1 - \theta^m$
	Reject, d_1	c	0	$c\,\theta^m$

Table 4: Defender's loss function - Scenario A.

The expected losses of both decisions are:

$$l_D(d_0) = E_\theta\left[1 - \theta^m\right] = 1 - E_\theta\left[\theta^m\right], \qquad l_D(d_1) = E_\theta\left[c\,\theta^m\right] = c\,E_\theta\left[\theta^m\right].$$

Then, the decision is to accept the batch (d_0) if

$$1 - E_\theta\left[\theta^m\right] \leq c\,E_\theta\left[\theta^m\right] \quad \Longleftrightarrow \quad E_\theta\left[\theta^m\right] \geq \frac{1}{1+c}.$$

Since $E_\theta\left[\theta^m\right]$ decreases as m increases, there will be a threshold value m_A such that if $m > m_A$, the decision would be to reject the batch (d_1). In particular, with the posterior

$\mathcal{B}e(\alpha+s, \beta+r-s)$ model for θ, we have

$$E_\theta\left[\theta^m\right] = \prod_{k=0}^{m-1} \frac{\alpha+s+k}{\alpha+\beta+r+k}, \tag{5}$$

and we easily obtain m_A recursively.

4.1.2 Scenario B: Each fault counts

In this second scenario, the loss will depend on the number $m-x$ of faulty items included, because of the increased security or performance issues. The relevant loss structure is displayed in Table 5, where the new parameter c' is the (expected) loss per faulty item accepted.

		Batch of m Items		
		All Acceptable	x Faulty	
		$p = \theta^m$	$p = \binom{m}{x} \theta^x (1-\theta)^{m-x}$	**Exp. Loss**
D's Decision	Accept, d_0	0	$(m-x)\, c'$	$m\, c'\, (1-\theta)$
	Reject, d_1	c	0	$c\, \theta^m$

Table 5: Defender's loss function - Scenario B.

The expected losses of both decisions are:

$$l_D(d_0) = E_\theta\left[m\, c'\, (1-\theta)\right] = m\, c'\, (1 - E_\theta\left[\theta\right]), \qquad l_D(d_1) = E_\theta\left[c\, \theta^m\right] = c\, E_\theta\left[\theta^m\right].$$

Then, the decision should be to accept the batch (d_0) if

$$m\, c'\, (1 - E_\theta\left[\theta\right]) \leq c\, E_\theta\left[\theta^m\right] \quad \Longleftrightarrow \quad \frac{E_\theta\left[\theta^m\right]}{m} \geq \frac{c'}{c}\, (1 - E_\theta\left[\theta\right]).$$

As before, since $E_\theta\left[\theta^m\right]$ decreases as m increases, there will be a threshold value m_B such that if $m > m_B$, the decision would be to reject the batch (d_1). In particular, with the posterior $\mathcal{B}e(\alpha+s, \beta+r-s)$ model for θ, the decision is to accept the batch if and only if

$$\frac{E_\theta\left[\theta^m\right]}{m} \geq \frac{c'}{c}\, \frac{\beta+r-s}{\alpha+\beta+r}.$$

Once again, we can make use of expression (5) to find m_B recursively.

4.2 Adversarial problem

We deal now with the adversarial version, considering only the loss in Section 4.1.1. As reflected in the BAID in Figure 5, we face an attacker who may alter the received batch to confound the defender so as to reach some objectives.

The original batch X is influenced by parameters λ, which regulates the number m of items received, and θ, conditioning the quality of items. The attacker knows the size m of the batch before choosing his attack, possibly modifying the size of the final batch Y to n, which is observed by the defender before making her decision.

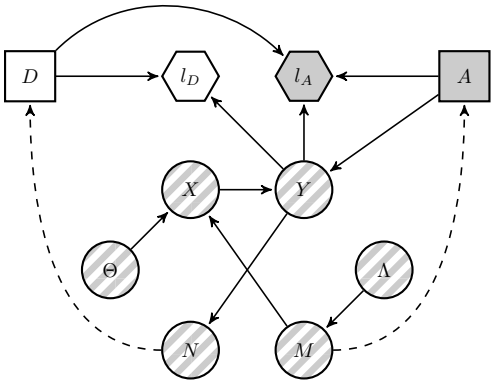
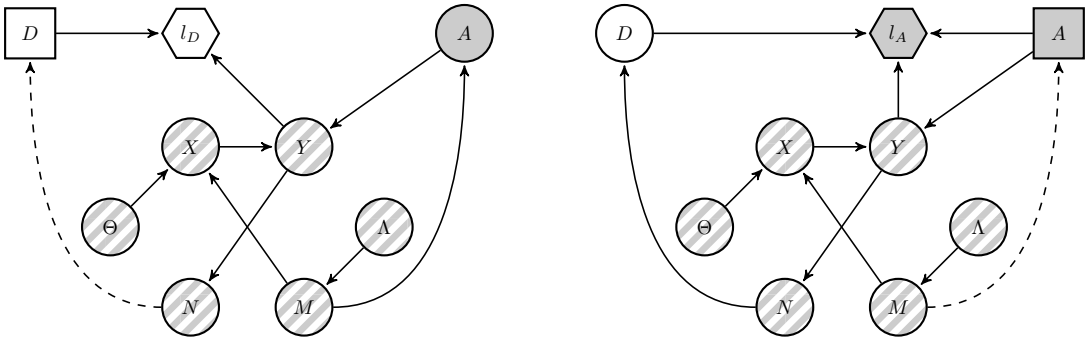


Figure 5: BAID for the adversarial batch acceptance problem.

The defender's and attacker's problems are, respectively, displayed in Figures 6a and 6b.



(a) Defender's problem (b) Attacker's problem

Figure 6: Adversarial batch acceptance problem.

We gradually study three possible attack strategies S1, S2 and S3, identifying the attacker's decision variables, how the item arrival process changes, the attacker's loss function and the solution. The final number of items in a batch will be n, with x acceptable items and $m - x$ faulty ones, which we shall call outer faults (O-faults). The remaining $n - m$ items correspond to faulty items introduced by the attacker, which will be called A-faults. The attacker's loss will be smaller (greater benefit) if the defender accepts an A-fault rather than an O-fault.

4.2.1 S1: A-fault injection

Under this strategy, the attacker injects y_1 of his faulty items. The data received by the defender includes x acceptable items, $m - x$ O-faults and y_1 A-faults. The attacker needs to decide y_1, which is random to the defender. As announced in Section 4.1, λ will be relevant here, since it provides information about m.

Suppose first that the defender knows the distribution of $Y_1 \,|\, m$, which we designate $p_D(y_1 \,|\, m)$, describing her beliefs about how many faulty items will be included by the attacker if the original batch size is m. The loss structure for the defender is as in Table 6, where the probability of having a final batch size of $n = m + y_1$ items, given λ, is

$$p_1(n \,|\, \lambda) = \sum_{i=0}^{n} p_D(m = i \,|\, \lambda)\, p_D(y_1 = n - i \,|\, m = i),$$

reflecting the possible initial sizes of the batch and the included faulty items. The probability that all those items are acceptable ($x = m$ and $y_1 = 0$) is

$$q_1(n \,|\, \lambda) = \frac{p_D(m = n \,|\, \lambda)\, p_D(y_1 = 0 \,|\, m = n)}{p_1(n \,|\, \lambda)}\, \theta^n,$$

which indicates that the only combination for an acceptable final batch is having n initial acceptable items ($x = m = n$) and no faulty items included ($y_1 = 0$).

		Final Batch of n Items		
		All Acceptable	Some Faulty	
		$p = q_1(n \,|\, \lambda)$	$p = 1 - q_1(n \,|\, \lambda)$	**Exp. Loss**
D's Decision	Accept, d_0	0	1	$1 - q_1(n \,|\, \lambda)$
	Reject, d_1	c	0	$c\, q_1(n \,|\, \lambda)$

Table 6: Defender's loss function - Strategy S1.

The expected losses of decisions d_0 (accept) and d_1 (reject) are, respectively:

$$l_D(d_0) = 1 - E_\theta \left[E_\lambda \left[q_1(n \,|\, \lambda) \right] \right], \qquad l_D(d_1) = c\, E_\theta \left[E_\lambda \left[q_1(n \,|\, \lambda) \right] \right].$$

Then, the rule is to accept the batch (d_0) if

$$E_\theta \left[E_\lambda \left[q_1(n \,|\, \lambda) \right] \right] \geq \frac{1}{1 + c},$$

whose evaluation would typically require simulation.

We provide now an ARA procedure to estimate the crucial quantities $p_D(y_1 \,|\, m)$ and, thus, $q_1(n \,|\, \lambda)$. To do so, we consider the attacker's loss function reflected in Table 7, which depends on the batch composition and the decision made by the defender, as well as on the attacker's decision. We have that $x \in \{0, 1, \ldots, m\}$ and $y_1 \in \{0, 1, \ldots\}$, where x and y_1 are the amount of acceptable items and injected A-faults, respectively. The involved parameters are the expected gain h due to each O-fault, the expected gain g due to each A-fault and the unitary cost f of injecting A-faults.

		Final Batch Composition		
		Acceptable	O-Fault	A-Fault
		x	$m-x$	y_1
D's Decision	Accept, d_0	0	$-h$	$f-g$
	Reject, d_1	0	0	f

Table 7: Attacker's loss per item - Strategy S1.

Given that the attacker chooses y_1, his losses associated to both defender's decisions are:

$$l_A(d_0, y_1) = -h\,(m-x) + (f-g)\,y_1, \qquad l_A(d_1, y_1) = f\,y_1.$$

Knowing the original batch size m, the attacker selects y_1 to minimize his expected loss, which is

$$\begin{aligned}
\psi_A(y_1 \,|\, m) &= p_A(d_0 \,|\, m+y_1) \int \left(\sum_{x=0}^{m} p_A(x \,|\, m, \theta)\, l_A(d_0, y_1) \right) p_A(\theta)\, \mathrm{d}\theta \\
&\quad + (1 - p_A(d_0 \,|\, m+y_1))\, l_A(d_1, y_1) \\
&= y_1\,(f - g\, p_A(d_0 \,|\, m+y_1)) \\
&\quad - h\, p_A(d_0 \,|\, m+y_1) \int \left(\sum_{x=0}^{m} p_A(x \,|\, m, \theta)\,(m-x) \right) p_A(\theta)\, \mathrm{d}\theta,
\end{aligned}$$

where $p_A(d_0 \,|\, m+y_1)$ reflects the attacker's beliefs about the defender's decision being to accept the batch (d_0), given that she perceives the batch size to be $n = m + y_1$.

Since we lack information about the attacker's probabilities and loss function, we model our uncertainty over them through random probabilities and losses (F, G, H, $P_A(d_0 \,|\, n), P_A(x \,|\, m, \theta), P_A(\theta)$), and look for the random optimal attack $Y_1^*(m)$ defined through:

$$\arg\min_{y_1} \left\{ \begin{array}{l} y_1\,(F - G\, P_A(d_0 \,|\, m+y_1)) \\ \\ - H\, P_A(d_0 \,|\, m+y_1) \displaystyle\int \left(\sum_{x=0}^{m} P_A(x \,|\, m, \theta)\,(m-x) \right) P_A(\theta)\, \mathrm{d}\theta \end{array} \right. .$$

Then, we would estimate

$$\hat{p}_D(y_1 \,|\, m) = P(y_1^*(m) = y_1) \approx \#\{Y_{1k}^*(m) = y_1\}/K,$$

where $\{Y_{1k}^*(m)\}_{k=1}^K$ would be a sample of size K from $Y_1^*(m)$, obtained by drawing from the involved components and computing the corresponding optimal amount of injected faulty items.

Regarding the attacker's random probabilities and losses, typical assumptions would be:

- The gains and costs could be uniformly distributed: $F \sim \mathcal{U}(f_1, f_2)$, $G \sim \mathcal{U}(g_1, g_2)$ and $H \sim \mathcal{U}(h_1, h_2)$.
- $P_A(d_0 \,|\, n)$ could be modeled through a uniform distribution, although this might require further recursion, if deeper strategic thinking is considered, as discussed in Section 2.1.
- Due to its specificity, $P_A(x \,|\, m, \theta)$ could actually be regarded as a Binomial distribution $\mathcal{B}in(m, \theta)$, i.e. not a random distribution.
- $P_A(\theta)$ could be a Dirichlet process with a Beta distribution base $\mathcal{B}e(\alpha+s, \beta+r-s)$ and concentration parameter ρ.

4.2.2 S2: Item modification

Under this strategy, the attacker modifies y_2 of the original items into faults of his. The data received by the defender includes $x - y_2^0$ acceptable items, $m - x - y_2^1$ O-faults and y_2 A-faults, where y_2^0 and y_2^1 verify $y_2^0 + y_2^1 = y_2$ and $0 \leq y_2^0 \leq x$, $0 \leq y_2^1 \leq m - x$. The attacker does not distinguish the type of items he changes and needs to decide y_2, which is random to the defender.

To start with, suppose that the defender knows the distribution $p_D(y_2 \,|\, m)$ of $Y_2 \,|\, m$, describing her beliefs about how many items will be modified by the attacker if the original batch size is m. The loss structure for the defender is as in Table 6, replacing $q_1(n \,|\, \lambda)$ by $q_2(n)$, defined as follows. First, the probability of having a final batch with $n = m$ items, given λ, is

$$p_2(n \,|\, \lambda) = p_D(m = n \,|\, \lambda),$$

reflecting the only possible initial size of the batch and the included faulty items. Then, the probability that all those items are acceptable ($x = m$ and $y_2 = 0$) is

$$q_2(n) = p_D(y_2 = 0 \,|\, m = n)\, \theta^n,$$

which indicates that the only combination for an acceptable final batch is having n initial acceptable items ($x = m = n$) and no faulty items included ($y_2 = 0$). In this case, knowing λ would be irrelevant for the batch configuration, since the initial and final batch sizes coincide.

The expected losses of both decisions are, respectively:

$$l_D(d_0) = 1 - E_\theta\left[q_2(n)\right], \qquad l_D(d_1) = c\, E_\theta\left[q_2(n)\right].$$

These may be simplified to:

$$l_D(d_0) = 1 - p_D(y_2 = 0 \,|\, m = n)\, E_\theta\left[\theta^n\right], \qquad l_D(d_1) = c\, p_D(y_2 = 0 \,|\, m = n)\, E_\theta\left[\theta^n\right].$$

The rule is to accept the batch (d_0) if

$$p_D(y_2 = 0 \,|\, m = n)\, E_\theta\left[\theta^n\right] \geq \frac{1}{1+c}.$$

We provide now an ARA procedure to estimate $Y_2 \,|\, m$ and, thus, the crucial quantity $p_D(y_2 = 0 \,|\, m)$. To do so, we consider the attacker's loss function reflected in Table 8,

which depends on the batch composition and the decision made by the defender (and the attacker's decision). It holds that $x \in \{0, 1, \ldots, m\}$ and $y_2 = y_2^0 + y_2^1 \in \{0, 1, \ldots, m\}$, where x and y_2 are, respectively, the amount of initial acceptable items and modified items. The new parameter f' is the cost of changing one item to make it faulty.

		Final Batch Composition		
		Acceptable	O-Fault	A-Fault
		$x - y_2^0$	$m - x - y_2^1$	y_2
D's Decision	Accept, d_0	0	$-h$	$f' - g$
	Reject, d_1	0	0	f'

Table 8: Attacker's loss per item - Strategy S2.

The attacker's (expected) losses associated with both defender's decisions, when he chooses y_2, are:

$$l_A(d_0, y_2) = -h\,(m - x - E\left[y_2^1\right]) + (f' - g)\,y_2, \qquad l_A(d_1, y_2) = f'\,y_2.$$

Assuming that the attacker chooses the items randomly, so that $E\left[y_2^1\right] = y_2\,\frac{m-x}{m}$, then:

$$l_A(d_0, y_2) = -h\,(m - x)\,(1 - \tfrac{y_2}{m}) + (f' - g)\,y_2.$$

The problem faced by the attacker is to select y_2 so as to minimize his expected loss when the original batch size is m, which is

$$\begin{aligned} \psi_A(y_2 \,|\, m) &= p_A(d_0 \,|\, m) \int \left(\sum_{x=0}^{m} p_A(x \,|\, m, \theta)\, l_A(d_0, y_2) \right) p_A(\theta)\, \mathrm{d}\theta \\ &\quad + (1 - p_A(d_0 \,|\, m))\, l_A(d_1, y_2) \\ &= y_2\,(f' - g\,p_A(d_0 \,|\, m)) \\ &\quad - h\,(1 - \tfrac{y_2}{m})\,p_A(d_0 \,|\, m) \int \left(\sum_{x=0}^{m} p_A(x \,|\, m, \theta)\,(m - x) \right) p_A(\theta)\, \mathrm{d}\theta, \end{aligned} \tag{6}$$

with $p_A(d_0 \,|\, m)$ as in Section 4.2.1.

Since we lack the attacker's probabilities, as well as the parameters of his loss function, we assume uncertainty about them and look for the random optimal attack $Y_2^*(m)$ defined through:

$$\arg\min_{y_2} \left\{ \begin{array}{l} y_2\,(F' - G\,P_A(d_0 \,|\, m)) \\ \\ -H\,(1 - \frac{y_2}{m})\,P_A(d_0 \,|\, m) \displaystyle\int \left(\sum_{x=0}^{m} P_A(x \,|\, m, \theta)\,(m - x) \right) P_A(\theta)\, \mathrm{d}\theta \end{array} \right. ,$$

where B would be the distribution over cost b. Then, we would estimate

$$\hat{p}_D(y_2 = 0 \,|\, m) = P(y_2^*(m) = 0) \approx \#\{Y_{2k}^*(m) = 0\}/K,$$

where $\{Y_{2k}^*(m)\}_{k=1}^K$ is a sample from $Y_2^*(m)$, obtained by drawing from the involved components and computing the corresponding optimal amount of items modified to make them faulty. Note that, due to the linearity of the attacker's loss function (6), the random optimal attack will always be 0 or m depending on whether it is worth modifying items. Non-linear loss functions for the attacker would allow different attacks to take place.

Typical assumptions about the attacker's random probabilities and losses would be similar to those in Section 4.2.1. In particular, $B \sim \mathcal{U}(f_1', f_2')$.

4.2.3 S3: Combination of strategies S1 and S2

Under this strategy, the attacker adds y_1 faulty items and converts y_2 of the original items into faults of his. The data received by the defender consists of $x - y_2^0$ acceptable items, $m - x - y_2^1$ O-faults and $y_1 + y_2$ A-faults, where y_2^0 and y_2^1 are subject to the same restrictions from Section 4.2.2. The attacker needs to decide both y_1 and y_2, which are random to the defender.

Suppose first that the defender knows the joint distribution of $(Y_1, Y_2) \,|\, m$, which we designate $p_D(y_1, y_2 \,|\, m)$. The loss structure for the defender is as in Table 6, with $q_1(n \,|\, \lambda)$ replaced by $q_3(n \,|\, \lambda)$, defined as follows. First, the probability of having a final batch of $n = m + y_1$ items, given λ, is

$$p_3(n \,|\, \lambda) = p_1(n \,|\, \lambda),$$

reflecting the possible initial sizes of the batch and the included faulty items (as in Strategy S1). Then, the probability that all those items are acceptable ($x = m$ and $y_1 = y_2 = 0$) is

$$q_3(n \,|\, \lambda) = \frac{p_D(m = n \,|\, \lambda)\, p_D(y_1 = 0, y_2 = 0 \,|\, m = n)}{p_3(n \,|\, \lambda)}\, \theta^n, \tag{7}$$

which indicates that the only combination for an acceptable final batch is having n initial acceptable items ($x = m = n$) and no faulty items included ($y_1 = y_2 = 0$). As in Section 4.2.1, λ provides information about m.

The expected losses of decisions d_0 (accept) and d_1 (reject) are, respectively:

$$l_D(d_0) = 1 - E_\theta \left[E_\lambda \left[q_3(n \,|\, \lambda) \right] \right], \qquad l_D(d_1) = c\, E_\theta \left[E_\lambda \left[q_3(n \,|\, \lambda) \right] \right].$$

Then, the rule is to accept the batch (d_0) if

$$E_\theta \left[E_\lambda \left[q_3(n \,|\, \lambda) \right] \right] \geq \frac{1}{1 + c}, \tag{8}$$

which would require simulation to be ascertained as in Section 4.2.1 .

We provide now an ARA procedure to estimate the crucial quantities $p_D(y_1, y_2 \,|\, m)$ and, thus, $q_3(n \,|\, \lambda)$. To do so, we consider the attacker's loss function in Table 9, which depends on the batch composition and the decision made by the defender (and the

attacker's decision). It holds that $x \in \{0, 1, \ldots, m\}$, $y_1 \in \{0, 1, \ldots\}$ and $y_2 = y_2^0 + y_2^1 \in \{0, 1, \ldots, m\}$, where x, y_1 and y_2 are the amount of original acceptable items, injected A-faults and modified items, respectively.

		Final Batch Composition			
		Acceptable	O-Fault	A-Fault	
				Injected	*Modified*
		$x - y_2^0$	$m - x - y_2^1$	y_1	y_2
D's Decision	Accept, d_0	0	$-h$	$f - g$	$f' - g$
	Reject, d_1	0	0	f	f'

Table 9: Attacker's loss per item - Strategy S3.

As in Section 4.2.2, we assume the attacker chooses the items randomly, so that $E[y_2^1] = y_2 \frac{m-x}{m}$. Then, the attacker's (expected) losses associated with both defender's decisions when the attacker chooses (y_1, y_2) are:

$$l_A(d_0, y_2) = -h\,(m - x)\,(1 - \tfrac{y_2}{m}) + (f - g)\,y_1 + (f' - g)\,y_2, \qquad l_A(d_1, y_2) = f\,y_1 + f'\,y_2.$$

The attacker chooses (y_1, y_2) to minimize his expected loss when the original size of the batch is m, which is

$$\begin{aligned}
\psi_A(y_1, y_2 \,|\, m) &= p_A(d_0 \,|\, m + y_1) \int \left(\sum_{x=0}^{m} p_A(x \,|\, m, \theta)\, l_A(d_0, y_1, y_2) \right) p_A(\theta)\, \mathrm{d}\theta \\
&\quad + (1 - p_A(d_0 \,|\, m + y_1))\, l_A(d_1, y_1, y_2) \\
&= y_1\,(f - g\,p_A(d_0 \,|\, m + y_1)) + y_2\,(f' - g\,p_A(d_0 \,|\, m + y_1)) \\
&\quad - h\,(1 - \tfrac{y_2}{m})\,p_A(d_0 \,|\, m + y_1) \int \left(\sum_{x=0}^{m} p_A(x \,|\, m, \theta)\,(m - x) \right) p_A(\theta)\, \mathrm{d}\theta,
\end{aligned}$$

with $p_A(d_0 \,|\, m + y_1)$ as in Section 4.2.1.

Since we lack the attacker's probabilities and parameters of his loss function, we assume uncertainty about them and look for the random optimal attack $(Y_1^*, Y_2^*)(m)$ defined through:

$$\arg\min_{y_1, y_2} \left\{ \begin{array}{l} y_1\,(F - G\,P_A(d_0 \,|\, m + y_1)) + y_2\,(F' - G\,P_A(d_0 \,|\, m + y_1)) \\ - H\,(1 - \tfrac{y_2}{m})\,P_A(d_0 \,|\, m + y_1) \displaystyle\int \left(\sum_{x=0}^{m} P_A(x \,|\, m, \theta)\,(m - x) \right) P_A(\theta)\, \mathrm{d}\theta \end{array} \right. .$$

Then, we would estimate

$$\hat{p}_D(y_1, y_2 \,|\, m) = P(y_1^*(m) = y_1, y_2^*(m) = y_2) \approx \#\{Y_{1k}^*(m) = y_1, Y_{2k}^*(m) = y_2\}/K,$$

where $\{(Y_{1k}^*, Y_{2k}^*)(m)\}_{k=1}^K$ is a sample of size K from $(Y_1^*, Y_2^*)(m)$, obtained by drawing from the involved components and computing the optimal amounts of injected faulty items and items changed to make them faulty.

Finally, we would make assumptions similar to those in Sections 4.2.1 and 4.2.2, concerning the attacker's random probabilities and losses.

4.3 Batch acceptance: A numerical example

As an illustration of the batch acceptance model, this section provides a numerical example of the analysis in Section 4.2 considering strategy S3. Concerning the defender's problem, following assumptions in Section 4.1, the elements involved are:

- The rate λ of original incoming items. The prior over λ will be a $\mathcal{G}a(5,1)$ distribution; i.e. we expect the average size of the original batch to be of 5 items.
- The probability θ that an item is acceptable. The prior over θ will be a $\mathcal{B}e(9,1)$ distribution; i.e. we expect the average probability of an item's acceptability to be 0.9.
- The (expected) opportunity costs c associated with rejecting a batch with all acceptable items. We will assume that $c = 0.9$.

With regard to the attacker's problem, in accordance with assumptions included in Section 4.2, suppose the following assessments are made:

- The gains and costs will be uniformly distributed as: $F \sim \mathcal{U}(0.25, 0.5)$, $F' \sim \mathcal{U}(0.3, 0.6)$, $G \sim \mathcal{U}(0.8, 1)$ and $H \sim \mathcal{U}(0, 0.25)$. Two relevant assumptions are being made: first, on average, injecting A-faults involves less effort for the attacker than modifying items to A-faults as he has broader control over the process; second, the expected gain due to A-faults is greater than that due to O-faults as he may better design them to fulfill his objectives.
- $P_A(d_0 \,|\, n)$ will be modeled through a uniform distribution dependent on the final batch size n. To avoid further recursion, we assume that the attacker relates it to the defender's non-adversarial version of the problem in Section 4.1. He could consider her accepting the batch with probability $E_\theta[\theta^n]$ in terms of its original expected acceptability. Additionally, he could weigh that probability by 0.5, admitting that the defender might presume him to be manipulating every other batch. In this manner, we estimate

$$E\left[P_A(d_0 \,|\, n)\right] = \frac{E_\theta[\theta^n]}{2} = \frac{1}{2}\prod_{k=0}^{n-1}\frac{9+k}{10+k} = \frac{9}{18+2n},$$

  making use of the defender's prior over θ and expression (5). In order to allow some uncertainty, and assuming that $P_A(d_0 \,|\, n) > P_A(d_0 \,|\, n+1)$ for any value of n, we finally adopt

$$P_A(d_0 \,|\, n) \sim \mathcal{U}\left(\frac{9}{19+2n}, \frac{10+n}{9+n}\,\frac{9}{19+2n}\right).$$

- Due to its specificity, $P_A(x \,|\, m, \theta)$ will be considered to coincide with the $\mathcal{B}in(m, \theta)$ distribution determined by the defender.
- $P_A(\theta)$ will be a Dirichlet process with a Beta distribution base $\mathcal{B}e(9, 1)$ and concentration parameter $\rho = 100$.

As a result of the previous assessments, we may estimate the attack probabilities for each original batch size m as follows:

Algorithm 2 Batch S3: Numerical example - Simulating the attacker's problem

Data: Original batch size m; number of iterations K; upper bound for the amount of injected items $\overline{Y}_1$.

Set $p(y_1, y_2) = 0, \quad y_1 = 0, \ldots, \overline{Y}_1,\ y_2 = 0, m.$

For $k = 1$ **to** K

Generate $f_k \sim \mathcal{U}(0.25, 0.50)$.

Generate $f'_k \sim \mathcal{U}(0.30, 0.60)$.

Generate $g_k \sim \mathcal{U}(0.80, 1.00)$.

Generate $h_k \sim \mathcal{U}(0.00, 0.25)$.

Generate distribution $p_A^k(\theta) \sim \mathcal{D}ir\mathcal{P}(\mathcal{B}e(9, 1), 100)$.

For $y_1 = 0$ **to** $\overline{Y}_1$

Generate $\pi_A^{0,k}(y_1) \sim \mathcal{U}\left(\dfrac{9}{19 + 2m + 2y_1}, \dfrac{10 + m + y_1}{9 + m + y_1}\, \dfrac{9}{19 + 2m + 2y_1}\right)$.

$$\psi_A^k(y_1, 0) = y_1 \left(f_k - g_k\, \pi_A^{0,k}(y_1)\right) - h_k\, \pi_A^{0,k}(y_1) \int \left(\sum_{x=0}^{m} \binom{m}{x} \theta^x\, (1-\theta)^{m-x}\, (m - x)\right) p_A^k(\theta)\, \mathrm{d}\theta.$$

$$\psi_A^k(y_1, m) = y_1 \left(f_k - g_k\, \pi_A^{0,k}(y_1)\right) + m \left(f'_k - g_k\, \pi_A^{0,k}(y_1)\right).$$

End For

Find $(y_1^*, y_2^*) = \underset{y_1 \in \{0, \ldots, \overline{Y}_1\},\, y_2 \in \{0, m\}}{\arg\min} \psi_A^k(y_1, y_2)$.

Set $p(y_1^*, y_2^*) = p_(y_1^*, y_2^*) + 1$.

End For

Set $\hat{p}_D(y_1^*, y_2^*) = p(y_1^*, y_2^*)/K, \quad y_1 = 0, \ldots, \overline{Y}_1,\ y_2 = 0, m.$

Table 10 reflects an application of the previous scheme with $K = 500$ (sufficient for illustrative purposes as the process is computationally intensive due to the need to sample from the Dirichlet process) and $\overline{Y}_1 = 5$ leading to the estimates of $\hat{p}_D(y_1, y_2 \,|\, m)$ in Table 10 with an original batch size of $m = 0, 1, \ldots, 8$.

		Original Batch Size - m								
		0	1	2	3	4	5	6	7	8
Attack - (y_1, y_2)	$(0,0)$	0.392	0.340	0.558	0.720	0.834	0.914	0.976	0.994	1.000
	$(1,0)$	0.300	0.190	0.150	0.140	0.106	0.070	0.020	0.006	0.000
	$(2,0)$	0.200	0.142	0.110	0.040	0.030	0.012	0.004	0.000	0.000
	$(3,0)$	0.082	0.052	0.026	0.006	0.004	0.000	0.000	0.000	0.000
	$(4,0)$	0.020	0.016	0.004	0.000	0.000	0.000	0.000	0.000	0.000
	$(5,0)$	0.006	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	$(0,m)$	-	0.218	0.134	0.090	0.026	0.004	0.000	0.000	0.000
	$(1,m)$	-	0.030	0.016	0.004	0.000	0.000	0.000	0.000	0.000
	$(2,m)$	-	0.010	0.002	0.000	0.000	0.000	0.000	0.000	0.000
	$(3,m)$	-	0.002	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	$(4,m)$	-	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	$(5,m)$	-	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 10: Defender's assessment of $\hat{p}_D(y_1, y_2 \,|\, m)$ using ARA.

Plugging such values in (7), we may compute the expected probability that all items are acceptable when the observed size of the final batch is $n = 0, 1, \ldots, 8$. In Table 11, we provide those values and the defender's decision according to acceptance rule in expression (8), being the cutting value $1/(1+c) = 0.526$.

	Final Batch Size - n								
	0	1	2	3	4	5	6	7	8
$E_\theta\left[E_\lambda\left[q_3(n \,|\, \lambda)\right]\right]$	1.000	0.485	0.554	0.541	0.521	0.514	0.503	0.515	0.514
Accept, d_0	Yes	No	Yes	Yes	No	No	No	No	No

Table 11: Defender's decision given a final batch of n items.

The following general remarks may be extrapolated from Tables 10 and 11:

- When an empty batch is received ($n = 0$), the model behaves correctly and accepts the batch as we know that there are no faulty items.
- For smaller original batch sizes, the attacker is greater compelled to both inject and/or modify items as it is more likely that all original items are acceptable. This might cause the defender not to accept batches with a really small final size (in our case, $n = 1$).

- For bigger original sizes of the batch, the attacker is discouraged to intervene and thus avoid costs as it is more likely that some original items are already faulty. This might cause the defender to accept batches with a medium final size (in our case, $n = 2, 3$).
- When a sufficiently large batch is received (in our case, starting with $n = 4$), the defender will not accept the batch as she will expect the original batch to include faulty items.

5 Discussion

We have provided an ARA framework to deal with the AHT problem. In this way, symmetric losses and strong common knowledge assumptions typical of non-cooperative game theory in adversarial signal processing, adversarial classification and adversarial machine learning are avoided. We have assumed that we were supporting an agent who essentially needs to ascertain which of several hypotheses holds, based on observations from a source that may be perturbed by another agent with some purpose. In doing this, the agent has to forecast the action of the adversary and then find her optimal alternatives. We focused on testing two simple hypothesis but the framework may be extended to other types of hypothesis tests.

Multiple attacker cases in the AHT problem are also of interest. An ARA perspective would support the defender versus all of them. In this case, we would need to differentiate possibilities in which attackers are completely independent or partially or totally coordinated or are such that their attacks influence somehow each other. It could also be the case that there are several defenders, possibly cooperating but with different observations of the data flow.

An illustrative application in relation with batch acceptance has been studied. We have assumed that the defender observes the size of the batch, but this might not be the case (e.g. when screening containers at international ports). When the defender has no information about the batch size other than her previous experience, we could think of a multi-stage version of the model proposed in Section 4. New strategies for the attacker such as the injection of (apparently) acceptable items to confound the defender could then be considered. It could also be the case that besides the batch size, the defender observes additional features of the items and this information would be incorporated to the testing problem. Other loss functions could be explored as well, including that in Section 4.1.2.

Finally, further applications may be found in the context of, for example, adversarial signal processing, such as in Electronic Warfare (EW) where pulse/signal environment is generally very complex with many different radars transmitting simultaneously. Time interval between two pulses emitted by a threat radar is defined as a Pulse Repetition Interval (PRI). PRI tracking is an important problem in naval EW applications because knowledge of the PRI is used to defend ships against radar-guided missiles. The signals received may be jammed by hostile radars and this results in missing pulses due to reduced sensitivity of the receiver, see Hock and Soyer (2006) for an introduction.

Acknowledgements

The work of DRI is supported by the Spanish Ministry of Economy and Innovation program MTM2014-56949-C3-1-R and the AXA-ICMAT Chair on Adversarial Risk Analysis. DRI and FR acknowledge the support from the ESF-COST Action IS1304 on Expert Judgement. Besides, JGO's research is financed by the Spanish Ministry of Economy and Competitiveness under FPI SO grant agreement BES-2015-072892. This work has also been partially supported by the Spanish Ministry of Economy and Competitiveness, through the "Severo Ochoa" Programme for Centres of Excellence in R&D (SEV-2015-0554).

References

D. Banks, J. Ríos and D. Ríos Insua. *Adversarial Risk Analysis (2016).* CRC Press, Boca Raton, FL, 2015.

M. Barni and F. Pérez-González. Coping with the enemy: Advances in adversary-aware signal processing. *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing*: 8682–8686, 2013.

M. Barni and B. Tondi. Binary hypothesis testing game with training data. *IEEE Transactions on Information Theory*, 60(8): 4848–4866, 2014.

J. O. Berger. Could Fisher, Jeffreys and Neyman have agreed on testing? *Statistical Science*, 18(1): 1–32, 2003.

J. O. Berger and T. Sellke. Testing a point null hypothesis: The irreconcilability of p values and evidence. *Journal of the American Statistical Association*, 87(397): 112–122, 1987.

N. Dalvi, P. Domingos, S. Sanghai and D. Verma. Adversarial classification. *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*: 99–108, 2004.

S. French and D. Ríos Insua. *Kendall's Library of Statistics 9: Statistical Decision Theory (2000).* Wiley, New York, NY, 2000.

S. Hargreaves-Heap and Y. Varoufakis. *Game Theory: A Critical Introduction (2004).* Routledge, New York, NY, 1995.

M. Hock and R. Soyer. A Bayesian approach to signal analysis of pulse trains. *Bayesian Monitoring, Control and Optimization*, CRC Press, New York, NY: 215–243, 2006.

D. Koller and B. Milch. Multi-agent influence diagrams for representing and solving games. *Games and Economic Behavior*, 45(1): 181–221, 2003.

J. Ríos and D. Ríos Insua. Adversarial risk analysis for counterterrorism modeling. *Risk Analysis*, 32(5): 894–915, 2012.

D. Ríos Insua, J. Ríos and D. Banks. Adversarial risk analysis. *Journal of the American Statistical Association*, 104(486): 841–854, 2009.

J. D. Tygar. Adversarial machine learning. *IEEE Internet Computing*, 15(5): 4–6, 2011.

A. Wald. Statistical decision functions. *Springer Series in Statistics - Breakthroughs in Statistics (Vol. 1, 1992)*, Springer New York, New York, NY: 342–357, 1950.