POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Algorithmic Bayesian Persuasion under Uncertainty

Laurea Magistrale in Computer Science Engineering - Ingegneria Informatica

Author: Luca Tombesi

Advisor: Prof. Alberto Marchesi

Co-advisor: Martino Bernasconi, Matteo Castiglioni

Academic year: 2022-2023

---

1. Introduction

In recent years, the value of information, and consequently of data, has grown exponentially. Owned data can be utilized in numerous ways to enhance wealth. For instance, in a business context, data are crucial for evaluating performance and understanding what strategies are effective. In other context having an *informational advantage*, namely having access to information that others lack, can be a significant asset, which can be used to *persuade* the other. *Persuasion* is the ability of inducing another player to perform an action that is the better for the one who is persuading. This specific form of persuasion, which capitalizes on an informational advantage, is known as *Bayesian persuasion*, and it is rising a lot in popularity. The Bayesian Persuasion model is a game of two players, the first is called *Sender*, who wishes to persuade the other player called *Receiver* through a signaling scheme. Dughmi and Xu [2] affirm that "since the first model of *Bayesian persusion* was published by Kamenica [4], persuasion as a share of economic activity appears to be growing — a more recent estimate places the figure at 30%". In this thesis we study the Bayesian persuasion problem in the context where the Sender does not know the actual utilities of the Receiver but has an uncertainty (δ-*knowledge*) of them. In other words we relax the constraint that the Sender has to know exactly the Receiver's payoffs. Such scenarios are extremely common and can be caused by an incomplete knowledge of the Sender or by a sub-optimal behaviour of the Receiver. This studied is link with various works like the one done by Bernasconi et al. [1]. Finally, a similar setting in the Stackelberg Equilibrium framework was studied by Gan et al. [3].

1.1. Thesis objective

The Sender's uncertainty leads to the situation where the Receiver plays in an unpredictable manner from the Sender perspective. Consequently, the main purpose of this thesis is to find a robust solution that effectively withstands scenarios where the Receiver engages in sub-optimal decision-making to this newly defined problem, in order to overcome at best the lack of the Sender knowledge about the Receiver utilities. In this thesis, we deeply analyze the problem, and we show that the approach to find the optimal solution significantly diverges from the standard case. From this analysis we understand how to find the solution and we design an algorithm that effectively resolves the studied problem. We study the time complexity of the algorithm, we implement it and we conduct an experimental analysis in order to evaluate its usability and its weak points in various contexts.

2. Bayesian persuasion under uncertainty

In order to represent the concept of uncertainty in the Bayesian persuasion framework, we introduce a parameter $\delta > 0$, which indicates the uncertainty on the knowledge of the Sender. So, the δ value represents the quantity of uncertainty in the Receiver's payoffs. The Sender uncertainty could be caused by many factors like limited observations, which would lead the Sender to have incorrect Receiver payoffs values, or a bounded Receiver rationality, which would mean that, even though the Sender has the correct Receiver's payoffs, the Receiver plays sub-optimally. Consequently the Receiver could play committing a small error δ from the Sender perspective. In this problem setting the Sender objective is to find a signaling scheme $\phi \in \Phi^m$, which is robust to the uncertainty. Φ^m is the space of all the signaling schemes. For this reason we have to define a new best response set of the Receiver called *Best Response delta set* (BR_δ set).

2.1. The BR_δ set

The new Best Response set that we define the Best Response delta set (BR_δ), is an extension of the concept of Best Response set. Namely given a δ value and a posterior probability ξ, we obtain a subset of the action set $[n]$, called $BR_\delta(\xi)$, where Θ is the set of all the states of nature, u_r and u_s are the Receiver and Sender payoff matrices. Then it is defined as $BR_\delta(\xi)=\{a' \in [n] / \sum_{\theta\in\Theta} \xi_\theta u_r(\theta, a') > \max_{a\in[n]} \sum_{\theta\in\Theta} \xi_\theta u_r(\theta, a) - \delta\}$. This new set includes not only the action that we perceive as the one with maximum utility for the Receiver, but also the other actions, which have for the Sender, a similar Receiver utility to her maximum. Our analysis aims at minimizing the Receiver sub-optimality by finding a robust solution. To this aim we define a new set that, starting from the $BR_\delta(\xi)$ set, finds the action that is worst for the Sender. Then given the $BR_\delta(\xi)$ set, we define the function $B_\delta(\xi)$, that given as input the posterior probability returns the action a, that corresponds to the action from the $BR_\delta(\xi)$ set with the minimum utility for the Sender: $B_\delta(\xi) = \min_{a\in BR_\delta} \sum_{\theta\in\Theta} \xi(\theta) u_s(a, \theta)$. Then the optimal signaling scheme $\phi*$, robust to the uncertainty, needs to induce the BR_δ set, which maximizes the minimum Sender utility. Indeed this solution guarantees to the Sender the maximum expected utility when the Receiver plays in the worst possible manner for him.

2.2. Non convexity of the BR_δ set

We show that the newly defined set is not convex. Namely, we show that $\forall\xi', \xi''$ where $BR_\delta(\xi') = BR_\delta(\xi'')$, $\forall\xi$ with $\xi = \alpha\xi' + (1-\alpha)\xi''$, where $\alpha \in (0, 1)$, then $BR_\delta(\xi) \neq BR_\delta(\xi') = BR_\delta(\xi'')$. This can be proven by the following example. We consider the case in which we have: δ=1; Three states of nature θ_1, θ_2 and θ_3; Three actions a_1, a_2 and a_3; Two posteriors ξ' and ξ". ξ' is (0.4,0.3,0.3) and ξ" is (0.3,0.4,0.3). The utilities for the Sender and the Receiver are written in the Table(1). In this case we have $BR_\delta(\xi')$=(a_1,a_2)

	θ_1	θ_2	θ_3
a_1	5,3	5,3	0,1
a_2	3,2	5,3	0,4
a_3	0,0	0,0	5,0

Table 1: Example of non convexity of the $BR_\delta(\xi)$ set

and $BR_\delta(\xi'')$=(a_1,a_2). By picking $\alpha = 0.5$ we have that $\xi = \alpha\xi' + (1-\alpha)\xi" = (0.35, 0.35, 0.3)$ and so the $BR_\delta(\xi)$ set would be (a_1,a_2,a_3), which causes the non convexity.
This negative result, however, lead the problem to not being solvable in polynomial time, which means that the problem is NP-Hard unless P$\neq$NP. Then in order to find a solution we have to create a new convex set, starting from the BR_δ set, composed by convex sub-regions.

2.3. Finding an optimal signaling scheme

To find the optimal signaling scheme $\phi*$, we firstly redefine the BR_δ set, depending on the signaling scheme ϕ, where $[n]$ is the set of actions, Θ is the set of states of nature and μ is the prior distribution. Then: $BR_\delta(\phi) = \{a' \in [n] / \sum_{\theta\in\Theta} \mu_\theta \phi_\theta u_r(\theta, a') > \max_{a\in[n]} \sum_{\theta\in\Theta} \mu_\theta \phi_\theta u_r(\theta, a) - \delta\}$. Therefore, From this set we create a new convex set, composed by convex sub-regions. Each sub-region is characterized by a set of actions A, which is a subset of $[n]$ and the action with the maximum Receiver utility called $a^*\in A$. Then for any possible set of action A$\in$ 2^n and any $a^*\in A$: $\chi_{(A,a*)}=\{\phi \in \Phi^m | \ BR_\delta(\phi) = A, \ u_r(\phi, a*) > u_r(\phi, a) \ \forall a \in A\}$. We have that $U_{a*\in A, A\in 2^n}\chi_{(A,a*)} = \Phi^m$. Indeed any $\phi \in \Phi^m$ must induce some Receiver δ-optimal response set A and an optimal Receiver response a^*. Then we can affirm that $\chi_{(A,a*)}$ is a convex set, because it is a polytope. Indeed it is described by linear constraints, which are linear in the variable ϕ, where $\phi = (\phi_1, ..., \phi_m)$. The constraints describing the polytope are the following:

(C1) $\sum_\theta \mu_\theta \phi_\theta u_\theta(a*) \geq \sum_\theta \mu_\theta \phi_\theta u_\theta(a') + \delta$, $a^*\in A$, $\forall a' \notin A$

(C2) $\sum_\theta \mu_\theta \phi_\theta u_\theta(a*) \geq \sum_\theta \mu_\theta \phi_\theta u_\theta(a)$, $\forall a \in A$, $a^*\in A$

(C3) $\sum_\theta \mu_\theta \phi_\theta u_\theta(a*) \leq \sum_\theta \mu_\theta \phi_\theta u_\theta(a) - \delta$, $\forall a \in A$

The constraints C1 and C2 derive directly from the definition of $BR_\delta(\phi) = A$, while constraint C3 is derived by the other part of the definition of $\chi_{(A,a*)}$. The same convex polytope can be defined in the posterior space. The linear constraints C1–C3 become:

(P1) $\sum_\theta \xi_\theta u_\theta(a*) \geq \sum_\theta \xi_\theta u_\theta(a') + \delta$, $a^*\in A$, $\forall a' \notin A$

(P2) $\sum_\theta \xi_\theta u_\theta(a*) \geq \sum_\theta \xi_\theta u_\theta(a)$, $\forall a \in A$, $a^*\in A$

(P3) $\sum_\theta \xi_\theta u_\theta(a*) \leq \sum_\theta \xi_\theta u_\theta(a) + \delta$, $\forall a \in A$

To find the optimal Sender signaling scheme $\phi*$, we have designed an algorithm that solves multiple linear programs. In figure (1) we report a short version of the pseudo-code of the algorithm, which can be viewed in its complete form in the thesis work. The algorithm correctness relies on the fact that it always exists, starting from any signaling scheme $\phi \in \Phi^m$, a set A and an action a^*. The algorithm for any possible sub-region identified by the tuple $(A, a*)$ enumerates all possible vertices of the polytope in the posterior space, breaking ties in favour of the Sender. Then it finds the optimal signaling scheme for the subset of actions A called ϕ_A. This is done by using a Linear Program, which finds a feasible probability distribution, that represents the probability of the signaling scheme to induce the posteriors, that maximizes the Sender utility in the vertices of the polytope. Finally picks the optimal signaling scheme $\phi*$ as the one that brings the maximum Sender utility among all the signaling schemes found. Algorithm(1) always returns a valid signaling scheme $\phi*$.

Algorithm 1 Compute the optimal signaling scheme for the Sender

---

Input: Receiver utilities u_r, Sender utilities u_s, parameter $\delta > 0$ and the prior distribution μ.
Output: optimal signaling scheme $\phi*$ for the Sender.
phi=[]
for any non-empty $A \subseteq [n]$ **do**
  v=[]
  for $a \in A$ **do**
    $a^*=a$
    Append ξ and u_s of all the vertices of the polytope derived by the following constraints to the set of vertices v:
    $\sum_{\theta\in\Theta} \xi_\theta u_{r,\theta}(a*) \geq \sum_{\theta\in\Theta} \xi_\theta u_{r,\theta}(a') + \delta$, $a' \notin A$
    $\sum_{\theta\in\Theta} \xi_\theta u_{r,\theta}(a*) \leq \sum_{\theta\in\Theta} \xi_\theta u_{r,\theta}(a) - \delta$, $a \in A$
    $\sum_{\theta\in\Theta} \xi_\theta u_{r,\theta}(a*) \geq \sum_{\theta\in\Theta} \xi_\theta u_{r,\theta}(a)$, $a' \in A$
    $\xi \in \Delta^m$
  end for
  If the same posteriors are present in the set, then leave in the set
  only the one with maximum utility for the Sender.
  Solve the following LP, to obtain the signaling scheme of A.
  maximize: $utot = \sum_{k\in vertices} x_k v_k(u_s)$
  subject to:
  $\sum_{k\in vertices} x_A(k) v_k(\xi_\theta) = \mu_\theta$, $\theta \in \Theta$
  $x_k \geq 0$
  $\sum_{k\in vertices} x_k = 1$
  if a feasible distribution is found **then**
    $\phi_A(\theta, s) = \frac{x_k v_k(\theta)}{\mu_\theta}$, for all θ in Θ
    phi.append$(\phi_A, utot)$
  end if

end for
$\phi* = max_{utot}(phi)$
return$(\phi*)$

2.4. Algorithm complexity

To find the optimal solution $\phi*$, Algorithm(1) has to find all the possible robust signaling scheme and then has to pick the optimal one. In order to find a feasible robust signaling scheme, given a possible $BR_\delta(\xi)$ set called A in the proof, the algorithm has to iterate for every action $a \in A$. The algorithm computes all the vertices with a time complexity of $O(n^3)$. After the algorithm finds all the possible vertices, it solves an LP in order to find a feasible probability distribution, which maximizes the Sender utility. The LP is solved in $O(mn)$. If a feasible distribution is found the algorithm computes the signaling scheme ϕ_A, where A indicates the signaling scheme of the partition A. Then given a partition A, the computation of the signaling scheme ϕ_A has a time complexity of $O(mn^4)$, which is polynomial. However, the algorithm has to iterate this process for every possible partition derived by the set of actions $[n]$. Consequently the algorithm iterates for every possible subset of A. This process implies a complexity of $O(2^n)$, which is exponential. Finally the time complexity of the algorithm is O($2^n poly(m, n)$).

3. Experimental results

We implemented our algorithm in Python and we tested it by conducting two types of experiments. The first focused on identifying and analyzing disparities in the CPU time and assessing the feasibility across various settings. The second experiment concentrated specifically on the δ parameter, aiming to illustrate how this parameter impacts the expected utility for the Sender. We used CPLEX as a solver for the LP.

3.1. Execution time

We computed the execution time of our algorithm across multiple settings. From now on we name the execution time as CPU time.

Test suite and parameters. The settings were based on two parameters: the number of state of natures m and the number of actions n, with both parameters limited to a maximum value of 7. We then generated a range of problem settings, encompassing every possible combination of actions and states of nature, varying from 2 to 7. This resulted in the analysis of 36 distinct problem settings. For each setting, in order to find a good approximation of

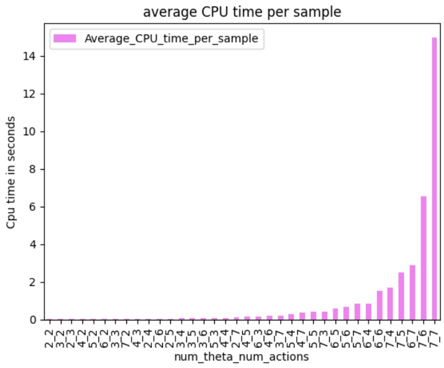

Figure 1: CPU time of each phase

the CPU time, we created 20 different samples. Each sample contained all necessary inputs for the algorithm's operation. Specifically we set the δ value at 0.1 and randomly generated the prior and utility matrices for both the Sender and the Receiver. The utility value for a specific state of nature and a specific action was fixed in a range from 0 to 1. All the time measures were done using the python library *time* and the code was executed using the *Google colab* CPU.

Results. The chart in Figure 1 describes the relationship between the CPU time and the dimension of the problem. As we can see the time needed grows in an exponential manner, aligning with our theoretical expectations. However, our data stated that solving a problem with seven states of nature and seven actions takes, on average, only 14 seconds. This means that even though the problem is NP-hard, it can still be used to solve problems with a relatively large setting.

We further analyzed the CPU time of individual phases of the algorithm to understand the source of the exponential increase in the CPU time, highlighted by the Figure 2.

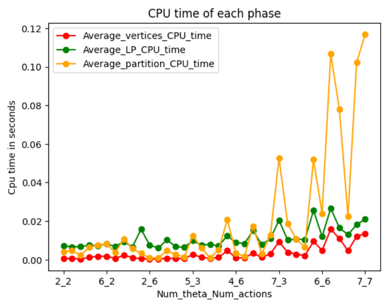

Figure 2: CPU time of each phase

The red line represents the average time for vertices calculation, the green for the linear programming (LP) calculation, and the orange for the execution of a partition. As the problem's complexity increases, all three metrics rise. However, the first two exhibit polynomial growth, while the exponential increase is primarily due to the partition execution, represented by the orange line. This empirical finding corroborates the theoretical discussion about the algorithm's complexity, pinpointing the analysis of all possible partitions as the main cause of its exponential nature.

Finally we uncovered which of the two parameters impacted more in the CPU time. From the

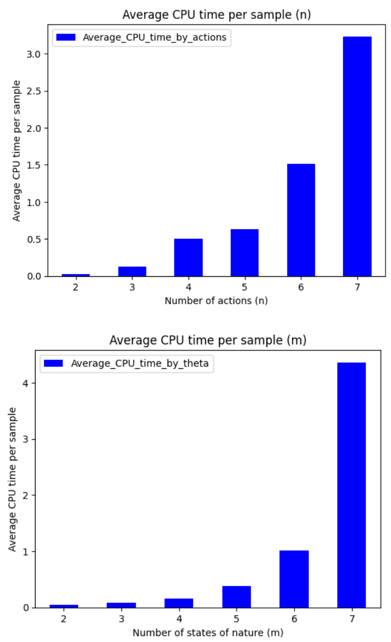

Figure 3: **UP**: Relationship between the average execution time and n. **BOTTOM**: Relationship between the average execution time and n

charts of Figure (3) it can be observed that the chart related to the actions has a trend, which is dominated by the chart related to the states of nature. Indeed we have that from point to point the increase in values is bigger for the parameter m rather than for the parameter n and so the number of states of nature have a bigger impact on the CPU time rather than the number of actions.

3.2. Uncertainty

The influence of the δ parameter is strongly related to the problem setting. Indeed the u_r payoff matrix plays a big role in this aspect. Indeed when uncertainty is high relative to payoff values, the δ value has a more significant impact on the Sender's strategy than in scenarios with a lower δ value. A key observation is that the presence of uncertainty, rather than its magnitude, primarily affects the Sender's utility. Indeed if there is uncertainty the Sender has to find a robust signaling scheme rather than a signaling scheme that simply maximizes his expected utility, which changes completely the approach to the problem. Therefore, our analysis focuses on cases where the utility values in the u_r payoff matrix is such that the utility values are relatively uniform, allowing the uncertainty factor to introduce greater variability into the Sender's signaling scheme and, consequently, his expected utility.

Test suite and parameters. Like the first experiment, we set the the maximum values for the number of states of nature m and actions n to 7. Then we proceeded to create a setting for all the possible combinations of states of nature and actions, generating 17 samples for each scenario. However, the creation of samples was done differently. Unlike the first experiment, we standardized the prior distribution and utility matrices across all settings, with utility values ranging from 0 (minimum) to 0.5 (maximum). While, for the δ value, we selected a different value for every sample, ranging from 0.1 to 0.9.

Results. Throughout this experimentation we noted that when the δ value is comparable with the Receiver payoffs, then the difference in the Sender expected utility between the setting with 0.1 δ and the one with 0.9 δ is often present. Moreover, as expected, a bigger uncertainty does not bring a better utility value for the Sender, but at most it results in an equal utility value. We used the difference metric in order to evaluate how the δ value impacted in the result and so to understand how much is the distance between the maximum possible utility value and the minimum one in a robust context. Regarding the relationship between the δ and the number of actions and the relationship between the δ and the number of state of nature, as it can be noted by the charts of Figures (4 and 5), there is no direct correlation between changes in the Sender's expected utility and the two parameter values.

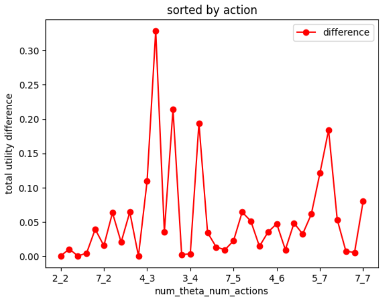

Figure 4: Difference in expected utility as the δ varies ordered by the number of actions

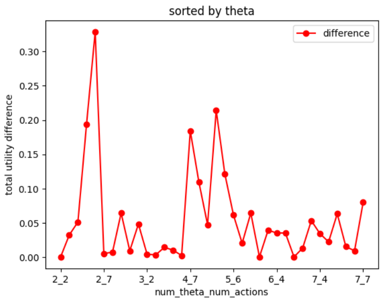

Figure 5: Difference in expected utility as the δ varies ordered by the number of states of nature

4. Conclusions

In this thesis we studied the Bayesian persuasion framework. We extended the existing knowledge regarding this type of problem, by exploring scenarios where the Sender faces uncertainty about the Receiver's behavior. To address this unpredictability we decided to face the problem in a robust manner to safeguard the Sender's expected utility. We modeled the problem with a focus on the worst-case scenario, where the Receiver invariably opts for the action most detrimental to the Sender. This is despite the fact that, from the Sender's perspective, the Receiver's choice is sub-optimal. This approach led to a novel definition of the Best Response set, and we demonstrated that this set is not convex. This finding resulted in the problem not being able to be solved in polynomial time. To find the solution, it is mandatory to divide the not convex set in all the possible convex partitions, which lead to an exponential time solution, unless $P = NP$. We then proceeded to analyze the usability of the algorithm through an in depth experimentation. The time experimentation lead to the conclusion that, even though the problem is NP-Hard, it can be a useful resource, in vast majority of cases, where the number of states of nature and of actions are limited. While the experimentation on the uncertainty lead to the conclusion that the main discrimen for the Sender expected utility is the presence or absence of uncertainty, because it would lead to a completely different approach to the problem. However, when the uncertainty factor is present and the Receiver payoffs esteems are close to the δ value, then the Sender expected utility could suffer from greater variability depending on the δ value.

References

[1] M. Bernasconi, M. Castiglioni, A. Marchesi, N. Gatti, and F. Trovò. Sequential information design: Learning to persuade in the dark. *Advances in Neural Information Processing Systems*, 35:15917–15928, 2022.

[2] S. Dughmi and H. Xu. Algorithmic bayesian persuasion. In D. Wichs and Y. Mansour, editors, *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 412–425. ACM, 2016. doi: 10.1145/2897518.2897583. URL `https://doi.org/10.1145/2897518.2897583`.

[3] J. Gan, M. Han, J. Wu, and H. Xu. Robust stackelberg equilibria. In K. Leyton-Brown, J. D. Hartline, and L. Samuelson, editors, *Proceedings of the 24th ACM Conference on Economics and Computation, EC 2023, London, United Kingdom, July 9-12, 2023*, page 735. ACM, 2023. doi: 10.1145/3580507.3597680. URL `https://doi.org/10.1145/3580507.3597680`.

[4] G. Kamenica. Bayesian persuasion. *American Economic association*, 2011. URL `https://www.aeaweb.org/articles?id=10.1257/aer.101.6.2590`.