Designing non-truthful mechanisms

Term paper for Algorithmic Economics

Disha Shur

Abstract

Auction design have recently leveraged deep learning network to learn and further produce good mechanisms. A common assumption while learning such mechanisms is the access to truthful bids. Considering most real world auction, that would enforce a serious limitation on the data one can use to train such models. In this work, I have tried to model non-truthfulness via regret in such a way that a neural network can be effectively trained on data from previous non-truthful auctions as well, which is more abundantly available than the truthful ones.

1 Introduction

The theory of designing optimal auctions analytically depends heavily on the revolutionary work in [7]. While it has been used to find the revenue maximizing auctions for specific cases, finding such auctions for a general case of items and buyers have proven to be difficult. An easier way out was offered by the data generated from random auctions carried out in the real-world and neural networks that could analyse that data and produce rules for optimal auctions for the future. The first work in this direction, to the author's knowledge, was in [2], while the authors have attempted to use their neural network model, RegretNET, to produce optimal auctions for a variety of items and bidders cases. Since then, many other models ([8],[11],[10],[1],[9]) addressing different aspects and approach to designing an optimal mechanism have been proposed. A common assumption in all of these works is the accessible data on past auctions consist of truthful bids. Most real-world auctions follow the English auction format; thus, rendering the truthful bids assumption invalid. The reason designing truthful auctions have received large attention is the revelation principle, stated here for convenience.

Theorem 1. *(Lemma 1 [7]) Given any feasible auction mechanism, there exists an equivalent feasible direct revelation mechanism which gives to the seller and all bidders the same expected utilities as in the given mechanism.*

As a consequence, auctions are first designed in the environment supporting direct revelation so as to be approximated later to a real world auction when some of the restrictions associated with the direct revelation principle can be relaxed. The problem with such relaxations is the revenue and welfare generated are also reduced according to the approximations introduced in the conditions of the mechanism. In order to ensure that the mechanism performs well in the real world, the respective conditions need to be considered when designing them. In most real-world auctions, the bidders do not tend to reveal their true valuations. As a consequence, the dominant strategy incentive compatible (DSIC) auctions generated by the neural network, under the enforced assumptions, are not as effective in the real world. Through this work, I have tried to address one of those assumptions.

1.1 Related work

There have been several work since [2] that try to make the neural network robust to assumptions enforced while designing them for a smooth transition to the real-world. The work in [10] tries to relax the dependence on the knowledge of the distributions from where the bidders draw their valuations. The assumption of the knowledge of distributions is one of the core assumptions in every mechanism attempted to be designed from past auctions. The methods used in [10] have attempted to design a truthful auction without assuming the knowledge of the distribution of valuations of the bidders. The problem addressed in [11] is that of modelling incentive compatibility. The choice of the bidder is determined by another network that takes as input the mechanism generated by the main network and outputs the bidders decision. This process leads to the exactness in incentive compatibility in the mechanism thus generated. Modelling a mechanism as an adversarial learning problem can lead to a long term utility for the bidder as described in described in [8]. When an auctioneer conducts multiple rounds to gather the bidders' valuation distribution, the bidder can choose to strategically reveal a bid that increases her long term utility and possibly rendering the auctioneer's mechanism non-optimal. [8] models this situation as a game and shows way in which mechanisms developed by a neural network assuming dependence on bidders' valuation distribution, as in [2], can be rendered non-optimal. Further, there has been no way to judge how efficient the auctions generated by neural networks actually are except on the basis of their performance in the known auction settings. For different settings of items and bidders, where the optimal auction is not known, there is no way to verify how strategyproof the generated auction is. [1] aims at explicitly characterizing the auctions generated through deep learning for its incentive-compatibility to bidder strategies using techniques from mixed integer programming. Specifically, they introduce modifications to the output functions in RegretNET as well as the mathematical functions used to compare the outputs of the neural network and bounds the regret, and hence the incentive to bid truthfully, a bidder might face through the auction generated by the neural network.

In all of the works described above, the focus of the neural network is to design a truthful mechanism. Truthful mechanisms are lucrative to analyze because of conditions introduced by DSIC assumptions. Although theoretically simple for certain cases, these mechanisms are practically ineffective. We discuss the problems associated with the methods described above in the next section.

1.2 Problem and Solution

Why I want to relax the truthful bids assumption Why is designing non-truthful mechanism important This is the part where you talk about what you did.

1. More non-truthful auctions Real world auctions are rife with non-truthful reporting. Consequently, auctions designed under DSIC assumptions do not perform optimally in the real world setup where the bidders have incentives to report non-truthfully.

There has been work towards making the neural network, aimed at producing DSIC auctions, robust towards bidders' strategies. However a big factor is that the data used for training the model may itself may be non-truthful (or $\epsilon-$truthful). The most common way to model this is the regret faced by the bidder. Consider the auction model in [2].

$$rgt_i = \frac{1}{L}\sum_{l=1}^{L}\max_{v'} u_i(v_i^{(l)}; (v_i', v_{-i}^{(l)})) - u_i(v_i^{(l)}; v^{(l)}) \tag{1}$$

The equation describes the regret faced by bidder i for every valuation profile, L, considered. The way it is implemented is that an optimization algorithm is given as input the valuation profiles from the data that goes through multiple iterations to update the following cost function,

$$\Delta_w C_\rho(w; \lambda^t) = -\frac{1}{B}\sum_l\sum_i \Delta_w p_i^{(w)}(v^{(l)}) + \sum_i\sum_l \lambda_i^t g_{l,i} + \rho\sum_i\sum_l rgt_i(w) g_{l,i} \tag{2}$$

where $g_{l,i}$ is the change in regret, modelled as follows,

$$g_{l,i} = \Delta_w \max_{v'} u_i(v_i^{(l)}; (v_i', v_{-i}^{(l)})) - u_i(v_i^{(l)}; v^{(l)}) = \Delta_w rgt_i \tag{3}$$

Note that according to equations (1), (2) and (3), the valuations v given as input are considered as true valuations in the sense that the model is designed assuming that the valuations are true. Consider the case where the valuations are false. The optimization problem in equation (2) uses the weights of the neural network from previous iteration to calculate the utility caused by possible false valuation. Here if the bid is already fake and utility due to $v_i^{(l)}$ is already high, then the change in regret $g_{l,i} = \nabla_w rgt_i$ would be negligible which, in turn, would reduce their significance in the overall update equation in (2). Thus, according to the given model in [2], if the valuation v is not a true one, it can end up that regret factor is not being included at all in the model.

The work in [9] tries to quantitatively compare the truthfulness of the mechanisms produced through neural networks. The incentive to bid non-truthful had been modelled using regret here as well. The paper uses the following proposition to model regret into their network,

Theorem 2. *Let $\mathcal{M}$ be an additive auction with 1 bidder and m items. Let P and R denote the expected revenue and regret, $P = E_{V \in D}[p(V)]$ and $R = E_{V \in D}[r(V)]$. There exists a mechanism $\mathcal{M}^*$ with expected revenue $P^* = (\sqrt{P} - \sqrt{R})^2$ and zero regret, $R^* = 0$*

Using the above proposition, in order to enforce truthfulness on the auction designed, their model is made to minimize the loss as follows,

$$\mathcal{L}_m(P, R) = -(\sqrt{P} - \sqrt{R}) + R \tag{4}$$

The regret R was computed by a separate optimization problem in [2], as shown in equation (1). In [9], they believe it to be a mapping from the true valuation v to any other value that maximizes the utility. This is treated as a different network module that accepts as input the true valuation and outputs the maximum utility possible through tuning the bids.

$$\mathcal{L}_r(w) = E_{V \in D}[\sum_{i=1}(u_i^{(w)}(v_i, (M(V)_i, V_{-i})))] \tag{5}$$

The objective of this module is equivalent to maximizing the expected value of utility cause by misreported valuation as shown in equation (5). The complete model works as the auctioneer module (the one with loss as described by equation (4)) outputs an auction format that is taken as input, along with the bids from the data, by the misreporter module and minimizes the negative of the loss described by equation (5) and outputs the misreports to the auctioneer module and the process goes on until desired convergence.

This paper takes as input the bids in the data and assumes them as the valuation profile. According to equation (5), the way the misreports are being mapped from the input values, if fake bids are provided to the module, the entire process will converge quickly because then the misreporters loss will already be low and the auctioneer module will not have to change its weight parameters many times. As a result, the output auction will have low truthfulness because essentially the misreporter module did not get to work at all since the fake bids had already made the utility value high.

Looking closely at these drawbacks, one can estimate the problem is in the modelling of the regret factor through the input bids, which maybe the true or false valuations. Thus, a robust model would be the one that is capable of working with non-truthful bids and generating auctions that ensure optimal revenue and welfare even when bidders do not have the incentive to bid truthfully. Non-truthful mechanisms offer an attractive solution. Moreover, non-truthful mechanisms have shown robustness to the valuation distribution of the bidders. In all the methods described above, a common assumption is that the the auctioneers know the distributions that the bidders draw their valuation from. It has been shown in [4] that truthful mechanisms are not always the optimal auctions and that the non-truthful ones are more resilient with respect to the bidders' valuation distribution. Thus, we will try to use the model introduced in [9] to model an optimal non-truthful mechanism.

2 Model

It has been established that with a specified number of sample complexity, it is possible to design optimal non-truthful auctions that have revenue and welfare close to an optimal truthful auction. Stating the conditions from [6] for designing optimal non-truthful mechanism,

Theorem 3. *With $m_{design} = p_{design}(n, 1/\epsilon)$ design-time samples of profiles of Bayes-Nash equilibrium bids from any mechanism in the family, parameters of the designed mechanism can be selected.*

The reason the models described above are not able to work with misreported valuations is because the regret is being generated as dependent on the valuations. Consequently, when the values are misreported, the regret is false as well.

2.1 Regret is a part of utility

An intuitive way to think about regret is as a part of utility. When an auctioneer runs multiple rounds of an auction and releases the information on largest and the second largest bids after each run, each bidder's utility, for the sold item, will change according to the released information. [5] offers a way to model regret as a part of utility as follows,

$$u_i(v_i, b_i|b_w, b_s) = \begin{cases} v_i - p_i - h(b_i - b_s) & \text{if i wins} \\ -g(v_i - b_w) & \text{if i loses} \end{cases} \tag{6}$$

In the above equation, b_w is the highest bid, b_s is the second highest bid and $h(.)$ and $g(.)$ are the regret functions. The belief behind this is when the bids are taken from a truthful mechanism, the concept of reducing regret makes sense in terms of the utility. But if the profile belongs to a non-truthful mechanism, regret would refer to overbidding that is the money left on table by the winner, also known as the winner regret or underbidding that is when an agent bids less than what their true valuation allows, also known as the loser regret. Accordingly, the regret functions need to be such that they have the following properties.

1. Non-decreasing: $h(.)$, the winner regret, will increase with the difference between its own bid and the 2nd highest bid as it will be leaving more money on the table. The loser regret, $g(.)$, will be higher according to how much lower the winning bid is than its own valuation.

2. Non-negative: This is because of the way they have been modelled into the utility function.

3. Differentiable: This is necessary to calculate the gradient of the utility function.

4. Trivial: $h(x) = 0$ for $x = 0$ because if there is only a small difference between the highest bid and second highest bid, the bidder's regret will be small (we argue with small and not 0 because then then winner would have been chosen randomly and the entire setup will break; hence, we assume that for some $\epsilon > 0$, $b_w - b_s \geq \epsilon$). For the second highest bidder, if it learns that the winning bid was above its valuation, it will have nothing to regret for.

Based on the modified utility function, an optimal revenue maximizing auction can be designed as the one that maximizes the price an auctioneer accumulated at the end of an auction as well as induces the lowest regret which would mean, with the utility function defined as above, the maximal utility. Thus solution would attempt to maximize both the utility and the revenue, or to solve the following objective function,

$$\min_{p,a} -[P|U] \tag{7}$$

The above system is just the maximizer of the augmented value of the price, hence revenue, and the utility achieved by the auction.

The new neural network can be designed as a simple multi-layer perceptron that takes as input the bids and tries to map then into a payment and auction profile that results in equation (7).

Notice that the model does not assume a non-truthful bid to be a truth valuation anywhere unlike the previous models discussed. Consequently, the modelled regret would not falter even if the inputs are untruthful valuations.

3 Conclusion and Open problems

There has been scarce interest in design and analysis of non-truthful mechanisms even with their proven advantages over truthful mechanisms. I have tried to develop a neural model to develop an optimal non-truthful mechanism so that it can consider non-truthful bids (not just the true valuation) to design the mechanism.

While preparing for this work, many more gaps were observed in the theory for optimal conditions in non-truthful auctions. The bids considered as input for all the neural network models were assumed to be, not only truthful valuations, but also from auctions that have attained their equilibrium allocation. This might not be true always which would further mean that the auctions generated using that training data are not optimal as well. The only definitive way of measuring optimal auctions are comparing it for a setting at which the optimal auction is known. Although [1] has furnished a way to analyze incentive compatibility in mechanisms produced by neural networks as in [2], there needs to be a more thorough analysis for the equilibrium conditions in a non-truthful auction. I believe an analysis and comparison of the models generated by these networks and the parameters employed for that particular run might help quantify the approximations going on in the model that resulted into those auctions.

Secondly, as with all other neural network outputs, its not easy to explain the mechanism that is being output by these networks. To better explain the working of these models regarding auction designing, it might be helpful to compare the allocation generated by these models with the impossibility results talked about in [3]. Specifically, it coins a permeability factor that characterizes the price of anarchy with the help of the allocation algorithm, by upper bounding the critical bids of the players by a permeability factor of the revenue achieved by the algorithm. Considering these two quantities when produced by the model might give new insights about auctions in the settings that were previously not explored due to analytical difficulties.

Thirdly, the sample complexity for non-truthful auctions calculated in [6] was based on the fact that the bids in the accessible data are in Bayes-Nash equilibrium and hence induce stationary equilibrium in the auction designed using those. Notice that although stationary equilibrium is a generalization of the Bayes-Nash equilibrium to mechanisms with sample access to previous runs of the same mechanism, they are not exactly the same. This part can cause error because of not having any quantifiable approximation between the Bayes-Nash equilibrium conditions and the stationary equilibrium conditions. Although the approximate fraction of the optimal welfare, attained by the mechanism thus generated, has been calculated, this approximation is based on the stationary equilibrium and not the Bayes-Nash equilibrium from which the bids are assumed to be sampled.

Therefore, we can say that the field of non-truthful mechanism design, although advantageous compared to truthful mechanisms, offer, inconsistencies and approximations, that need to be exactly solved.

References

[1] Michael Curry, Ping-yeh Chiang, Tom Goldstein, and John Dickerson. Certifying strategyproof auction networks. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 4987–4998. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/3465ab6e0c21086020e382f09a482ced-Paper.pdf.

[2] Paul Duetting, Zhe Feng, Harikrishna Narasimhan, David Parkes, and Sai Srivatsa Ravindranath. Optimal auctions through deep learning. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1706–1715. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/duetting19a.html.

[3] Paul Dütting and Thomas Kesselheim. Algorithms against anarchy: Understanding non-truthful mechanisms. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, EC '15, page 239–255, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450334105. doi: 10.1145/2764468.2764507. URL https://doi.org/10.1145/2764468.2764507.

[4] Yiding Feng and Jason D. Hartline. An end-to-end argument in mechanism design (prior-independent auctions for budgeted agents), 2018.

[5] Emel Filiz-Ozbay and Erkut Y. Ozbay. Auctions with anticipated regret: Theory and experiment. *American Economic Review*, 97(4):1407–1418, September 2007. doi: 10.1257/aer.97.4.1407. URL https://www.aeaweb.org/articles?id=10.1257/aer.97.4.1407.

[6] Jason Hartline and Samuel Taggart. Sample complexity for non-truthful mechanisms. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, EC '19, page 399–416, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450367929. doi: 10.1145/3328526.3329632. URL https://doi.org/10.1145/3328526.3329632.

[7] Roger B Myerson. Optimal auction design. *Mathematics of operations research*, 6(1):58–73, 1981.

[8] Thomas Nedelec, Jules Baudet, Vianney Perchet, and Noureddine El Karoui. Adversarial learning for revenue-maximizing auctions. In *20th International Conference on Autonomous Agents and Multiagent Systems*, Londres, United Kingdom, 2021. URL https://hal.archives-ouvertes.fr/hal-03089784.

[9] Jad Rahme, Samy Jelassi, and S. Matthew Weinberg. Auction learning as a two-player game. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YHdeAO61l6T.

[10] Daniel Reusche and Nicolás Della Penna. Towards prior-free approximately truthful one-shot auction learning via differential privacy, 2021.

[11] Weiran Shen, Pingzhong Tang, and Song Zuo. Automated mechanism design via neural networks. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '19, page 215–223, Richland, SC, 2019. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450363099.