Manipulation of Machine Learning Algorithms

Doctoral Consortium

Nicholas Bishop
University of Southampton
Southampton, UK
nb8g13@soton.ac.uk

ABSTRACT

As data becomes increasingly available, individuals, organisations and companies are increasingly applying machine learning algorithms to make decisions. In many cases, those decisions have a direct effect on those who provided the data to the decision maker. In other words, data providers often have a vested interest in the decisions made based on the data provided. Therefore, decision makers should anticipate that data providers may alter or change the data they provide in order to achieve a preferential outcome. Such strategic behaviour is not adequately modelled by classical machine learning settings in the literature. As a result, new machine learning algorithms are required, which take into the account the incentives and capabilities of data providers when making decisions. This paper summarises a PhD project which attempts to address this problem in a number of contexts.

KEYWORDS

Machine Learning; Computational Social Choice

ACM Reference Format:
Nicholas Bishop. 2022. Manipulation of Machine Learning Algorithms: Doctoral Consortium. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022,* IFAAMAS, 3 pages.

1 INTRODUCTION

When supplying data, data providers are often implicitly invested in how said data will be used. A canonical example often cited in literature is the problem of email spam classification [6], in which an email service provider is tasked with identifying and removing spam emails. In this case, a significant portion of the data is provided by spammers, who hope to bypass the spam filter. Therefore, email service providers must be aware that the spam they receive in the future may not resemble the spam emails they used to design their spam filters. Spammers will inevitably alter their spam emails to bypass the filter in the near future.

Note that email spammers are adversarial in nature. That is, their goal is to submit data with the intention of fooling the algorithm. More generally, the field of adversarial machine learning focuses on defending machine learning algorithms against adversaries whose aim is to hinder performance. Observe that adversarial machine learning makes a worst-case assumption regarding the incentives of data providers. In many cases, such as spam classification, this worst-case approach is sensible, as email spammers are antagonistic by nature. However, in many real world settings, the incentives of data providers can be significantly more nuanced. In other words, the goals of a data provider may neither completely align with the goals of the decision maker, nor directly oppose them, but often lies somewhere in between.

One example of this is the fashion company Zara, who use regression to distribute a limited number of designer goods amongst their retail stores [7]. During this process, Zara queries their store managers and asks them how many goods they think they can sell. The salary of a store manager is directly tied to the revenue their store generates. That is, a store manager can increase their salary by increasing the revenue generated by their store. As supply is limited, Zara found that store managers often over-reported the number of goods they could sell, in order to increase their chances of securing the goods they *knew* they could sell. In this case, each store manager is motivated to maximise the revenue of their store, whilst Zara aims to distribute their goods in order to maximise their total revenue across all stores. Whilst store managers are not intending to harm Zara's overall revenue, their self-interested and strategic behaviour inevitably does.

Such examples illustrate that, in many cases, assuming data providers are adversarial is often unrealistic. By relaxing this assumption, it stands to reason that we can devise machine learning algorithms which have better performance. In what follows, we will examine a number of settings in which the incentives of data providers are modelled in a more nuanced manner. The first setting we will consider is an extension of the traditional linear regression setting in which data providers disagree with a decision maker on the correct labelling of data points. Meanwhile, the second setting we investigate is a repeated matching setting, in which assigned resources are blocked when in use.

2 STACKELBERG PREDICTION GAMES FOR LINEAR REGRESSION

Consider the following regression setting. A learner is tasked with selecting a linear predictor $w \in \mathbb{R}^n$ to assign labels to input data. Data is drawn from some underlying distribution by data providers in the form $(x, y, z) \in \mathbb{R}^{n+2}$. Here, $x \in \mathbb{R}^n$ represents the input vector sampled from the underlying distribution, $y \in \mathbb{R}$ indicates the label of interest to the learner, and $z \in \mathbb{R}$ indicates the label preferred by the data provider.

Before x is passed onto the learner, the data provider has the opportunity to modify the input data. More precisely, the data provider may submit a different input vector $\hat{x} \in \mathbb{R}^n$ to the learner in place of x. We assume that the data provider has full knowledge of the linear predictor chosen by the learner when making this modification. For changing x to $\hat{x}$, the data provider incurs a cost $c(x, \hat{x})$. The learner observes $\hat{x}$ and makes the prediction $\hat{y} = w^\top x$.

Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Online.

The learner incurs loss $\ell_1(\hat{y}, y)$, whilst the data provider incurs loss $\ell_2(\hat{y}, z)$. The goal of the learner is to select the linear predictor which minimises their expected loss according to the underlying distribution from which the data provider gathers their samples. Similarly, the data provider aims to minimise their loss for each data point, whilst trading off against the cost they pay for modification. More precisely, the learner seeks to minimise $\mathbb{E}[\ell_1(\hat{y}, y)]$. On the other hand, given w and a data point (x, y, z), the data provider chooses an $\hat{x}$ which minimises $\ell_2(\hat{y}, y) + c(\hat{x}, x)$.

Note that this setting is incredibly expressive when it comes to modelling the preferences of data providers. We place no assumption on the labels z assigned to each data point. As a result, this setting is very flexible and can model a wide range of strategic incentives. In fact, this setting is an instance of the Stackelberg prediction game model introduced by [4].

To aid the learner, we assume that they posses a clean, unmodified sample, $\{x_i, y_i, z_i\}_{i=1}^m$from the distribution of interest which they can use for the purposs of training. In practice, such a training set may be obtained by querying data providers. An intuitive approach for the learner is to select the linear predictor which minimises their empirical risk. That is, the linear predictor which solves the following minimisation problem:

$$\begin{aligned} \min_{w} \quad & \sum_{i=1}^{m} \ell_1(w^\top \hat{x}_i, y_i) \\ \text{s.t.} \quad & \hat{x}_i = \operatorname*{argmin}_{\tilde{x}} \ \ell_2(w^\top \tilde{x}_i, z_i) + c(x_i, \tilde{x}_i) \end{aligned}$$

However, this approach poses many issues from an optimisation perspective. Thus, it is worth asking: is empirical risk minimisation even computationally tractable in this setting? In [3], we answer in the affirmative, and show that there exists an efficient polynomial time algorithm based on semidefinite programming. In short, this algorithm consists of bisection search, where each iteration requires solving a semidefinite program (SDP). [10] improve further on this result and shows that, using matrix congruence, only one SDP needs to be solved.

However, a significant number of open questions remain, Are there similar algorithms for similar settings? For example, does an efficient algorithm exist for a similar setting, where linear regression is replaced by the task of support vector regression, or by logistic regression? Additionally, what statistical guarantees can we make regarding the empirical risk minimisation approach outlined above? For the traditional supervised learning setting, empirical risk minimisation is justified using results from statistical learning theory. Do similar theoretical results hold for strategic setting above? Statistical guarantees for empirical risk minimisation based approaches in similar settings have been found [9, 11], so it seems likely the answer to this question is yes.

As we have already mentioned, this setting is fairly general. Can we do better when we know more information regarding the strategic incentives of the data providers? For example, when $z = y$, this setting resembles the performative prediction setting studied by [8]. One may expect to find faster algorithms in this case. Similarly, the learner may know that the incentives of data providers have some additional structure which could be exploited. What assumptions on the preferences of data providers leads to more tractable algorithms?

3 MATCHING WITH BLOCKING

In the previous section, we extended a traditional machine learning setting (linear regression) to incorporate strategic data providers. In this section, we study a problem with roots in mechanism design. Namely we study the problem of repeated one-sided matching.

Consider a sequential matching problem which takes place over T time steps. At each time step a central mechanism must match a set of agents to a set of services. Each agent holds cardinal preferences over the services. That is, when an agent i is assigned service j, it receives a utility of $u_{ij} \in \mathbb{R}$. Additionally, when an agent i is assigned a service j, the service is blocked for the next $d_{i,j}$ time steps and cannot be matched to any agent. Blocking is useful for modelling settings with scarce or depleted resources that may become unavailable after use. For example, consider the problem of matching freelnace contractors to companies. When a contractor is assigned to a company on a given date, they are unavailable for the duration of the contract. Blocking was first studied in the context of multi-armed bandits by [1].

The goal of the central mechanism is to construct a feasible sequence of matchings which maximises the social welfare. That is, to find a matching which maximises $\sum_{t=1}^{T} \sum_i u_{i,i(t)}$, where $i(t)$ denotes the service assigned to agent i at time step t.

To aid the central mechanism, each agent is required to submit a list of ordinal preferences over the available services. We assume each agent submits a list of ordinal preferences with the aim of maximising their own utility (i.e. the sum of their payoffs across all time steps). Our goal is to find a matching mechanism which is truthful and attains the highest social welfare possible. By truthful we simply mean that each agent is incentivised to report the ordinal preferences induced by its underlying cardinal preferences.

One may expect a simple algorithm to work in this setting. For example, consider an algorithm which takes any truthful and efficient algorithm for one-shot one-sided ordinal matching, such as random serial dictatorship (RSD), and applies it on every time step. Without blocking, such an algorithm is optimal and truthful. However, blocking creates dependencies between time steps, and as such, a simple algorithm such as the one described above is neither truthful or efficient in the presence of blocking.

In [2], we propose an alternate method for constructing a sequential matching mechanism from one-shot ordinal matching mechanisms. We show that this approach approximates the optimal sequence of matchings, and satisfies a notion of approximate truthfulness, in the sense that it has a bounded incentive ratio [5].

This work raises a number of interesting questions. First of all, is this the best we can do? Is there a mechanism which is *truthful* and achieves a meaningful approximation of the optimal matching sequence? Additionally, observe that blocking is just one of many ways of specifying the evolution of each agent's preferences over the time horizon in a way that depends on the actions of the central mechanism. More generally, can we find effective sequential mechanisms for preferences that evolve depending on the action taken by the central mechanism?

REFERENCES

[1] Soumya Basu, Rajat Sen, Sujay Sanghavi, and Sanjay Shakkottai. 2019. Blocking Bandits. In *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.),

Vol. 32. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2019/file/88fee0421317424e4469f33a48f50cb0-Paper.pdf

[2] Nicholas Bishop, Hau Chan, Debmalya Mandal, and Long Tran-Thanh. 2022. Sequential Blocked Matching. In *36th AAAI Conference on Artificial Intelligence*.

[3] Nicholas Bishop, Long Tran-Thanh, and Enrico Gerding. 2020. Optimal Learning from Verified Training Data. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 9520–9529. https://proceedings.neurips.cc/paper/2020/file/6c1e55ec7c43dc51a37472ddcbd756fb-Paper.pdf

[4] Michael Brückner and Tobias Scheffer. 2011. Stackelberg Games for Adversarial Prediction Problems. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Diego, California, USA) *(KDD '11)*. Association for Computing Machinery, New York, NY, USA, 547–555. https://doi.org/10.1145/2020408.2020495

[5] Ning Chen, Xiaotie Deng, Hongyang Zhang, and Jie Zhang. 2012. Incentive Ratios of Fisher Markets. In *Proceedings of the 39th International Colloquium Conference on Automata, Languages, and Programming - Volume Part II* (Warwick, UK) *(ICALP'12)*. Springer-Verlag, Berlin, Heidelberg, 464–475. https://doi.org/10.1007/978-3-642-31585-5_42

[6] Nilesh Dalvi, Pedro Domingos, Mausam, Sumit Sanghai, and Deepak Verma. 2004. Adversarial Classification. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Seattle, WA, USA) *(KDD '04)*. Association for Computing Machinery, New York, NY, USA, 99–108. https://doi.org/10.1145/1014052.1014066

[7] Ofer Dekel, Felix Fischer, and Ariel D. Procaccia. 2010. Incentive compatible regression learning. *J. Comput. System Sci.* 76, 8 (2010), 759–777. https://doi.org/10.1016/j.jcss.2010.03.003

[8] Juan Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. 2020. Performative Prediction. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*, Hal Daumé III and Aarti Singh (Eds.). PMLR, 7599–7609. https://proceedings.mlr.press/v119/perdomo20a.html

[9] Ravi Sundaram, Anil Vullikanti, Haifeng Xu, and Fan Yao. 2021. PAC-Learning for Strategic Classification. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 9978–9988. https://proceedings.mlr.press/v139/sundaram21a.html

[10] Jiali Wang, He Chen, Rujun Jiang, Xudong Li, and Zihao Li. 2021. Fast Algorithms for Stackelberg Prediction Game with Least Squares Loss. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 10708–10716. https://proceedings.mlr.press/v139/wang21d.html

[11] Hanrui Zhang and Vincent Conitzer. 2021. Incentive-Aware PAC Learning. *Proceedings of the AAAI Conference on Artificial Intelligence* 35, 6 (May 2021), 5797–5804. https://ojs.aaai.org/index.php/AAAI/article/view/16726