Deep Learning Based Casual Inference for Large-Scale Combinatorial Experiments: Theory and Empirical Evidence

Zikun Ye[1], Zhiqi Zhang[2], Dennis J. Zhang[2], Heng Zhang[3], Renyu Zhang[4]
[1] University of Illinois Urbana-Champaign, Urbana, IL
[2] Washington University in St. Louis, St. Louis, MO
[3] Arizona State University, Tempe, AZ
[4] The Chinese University of Hong Kong, Hong Kong, China
zikunye2@illinois.edu, z.zhiqi@wustl.edu, denniszhang@wustl.edu, hengzhang24@asu.edu, philipzhang@cuhk.edu.hk

Large-scale online platforms launch hundreds of randomized experiments (a.k.a. A/B tests) every day to iterate their operations and marketing strategies, while the combinations of these treatments are typically not exhaustively tested. It triggers an important question of both academic and practical interests: Without observing the outcomes of all treatment combinations, how to estimate the causal effect of any treatment combination and identify the optimal treatment combination? We develop a novel framework combining deep learning and doubly robust estimation to estimate the causal effect of any treatment combination for each user on the platform when observing only a small subset of treatment combinations. Our proposed framework (called debiased deep learning, DeDL) exploits Neyman orthogonality and combines interpretable and flexible structural layers in deep learning. We prove theoretically that this framework yields efficient, consistent and asymptotically normal estimators under mild assumptions, thus allowing for identifying the best treatment combination when only observing a few combinations. To empirically validate our method, we then collaborate with a large-scale video-sharing platform and implement our framework for three experiments involving three treatments where each combination of treatments is tested. When only observing a subset of treatment combinations, our DeDL approach significantly outperforms other benchmarks to accurately estimate and infer the average treatment effect (ATE) of any treatment combination, and to identify the optimal treatment combination.

Key words: Deep Learning, Double Machine Learning, Causal Inference, Field Experiments, Experimentation on Online Platforms

1. Introduction

Large-scale online platforms have penetrated billions of people's daily lives in various areas. As of January 2021, more than 53.6% of the world population (i.e., 4.2 billion people) are active social media users.[1] People connect with each other on social network platforms such as Facebook and TikTok, shop online on e-commerce platforms such as Amazon and Alibaba, and hail a ride on ride-sharing platforms such as Uber and Lyft, etc. Because of the tremendous values created by

[1] See https://datareportal.com/reports/digital-2021-global-overview-report.

these platforms, firms that develop and own such businesses are now worth more than 1 trillion US dollars. For example, in October 2022, the market value of Amazon was 1.2 trillion USD, that of Alphabet was 1.4 trillion USD, and that of Microsoft was 1.9 trillion USD. It is also estimated by the Committee on Judiciary of the USA in the Investigation of Competition in Digital Markets[2] that the total market value of platform-based tech firms will reach more than 30% of the annual global GDP within the next ten years.

Equipped with mountainous user data and advanced information technology, online platforms base their critical business decisions on advanced data analytics techniques. Of central importance are randomized experiments (a.k.a. A/B tests or field experiments; we use A/B tests and experiments interchangeably hereafter), which are widely considered the gold standard for causal inference and policy evaluation. Under an A/B test, a platform randomly assigns its users to different groups and applies a different treatment to users in each group. The controlled randomization enables the platform to credibly attribute the outcome differences of different user groups to the treatment effect of the strategies. Because of the online nature of their business and the vast user traffic, platforms can conveniently run A/B tests to evaluate and optimize their product design, pricing, and recommendation strategies (Kohavi et al. 2020). Usually, the analyst casts a policy change in these aspects as a treatment and compares it with the existing policy through an A/B test. Leading online platforms such as Facebook, Amazon, Google, and TikTok each run more than 10,000 online experiments annually, many of which engage millions of their users (Kohavi and Thomke 2017).

To quickly iterate its business operations, a large-scale online platform typically runs hundreds of A/B tests concurrently (see, e.g., Xiong et al. 2020). The sheer number of tests makes it difficult to test the joint effect of different treatments. In particular, due to the limited user traffic, a standard online experimentation method for the platform is the orthogonal traffic assignment design (Tang et al. 2010, Xiong et al. 2020): The treatment assignments of different individual A/B tests are independent. As a consequence, each user of the platform may be treated by a lot of A/B tests simultaneously. On the one hand, the orthogonal experiment design utilizes the user traffic of the platform more efficiently. Orthogonality ensures non-interference among experiments, so the platform gets a credible causal estimate for each treatment in each experiment. On the other hand, it largely ignores the joint effects caused by the combination of treatments in practice. It does not allow the platform managers to find the best combination of treatments for each user. In practice, platform managers typically assume the treatment effects of different A/B tests are linearly additive. Hence, the decision on whether and how to expand the traffic of one treatment,

[2] See https://judiciary.house.gov/uploadedfiles/competition_in_digital_markets.pdf?utm_campaign=4493-519.This link no longer works.

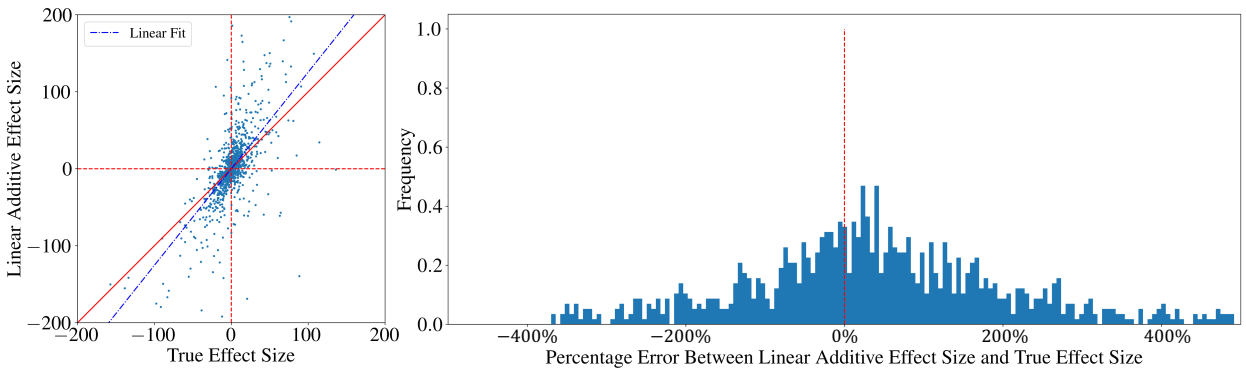


Figure 1 Heterogeneous Response to Two Experiments

for example, to all platform users, is irrespective of other concurrent experiments. Such a decision paradigm is particularly prevalent due to organizational reasons. For example, different stakeholders in a firm (e.g., the machine learning (ML) engineers and product managers) often manage their own set of A/B tests, and there can be little communication and coordination. The combined effects of multiple experiments are largely treated in a simple manner with the linear additivity assumption made.

Combining different treatments can create synergistic or antagonistic effects, depending on how different treatments interact with each other. It is usually difficult to predict which one is in effect without a formal test. In the worst case, two treatments that both benefit the platform can in fact hurt the platform instead once combined. To empirically illustrate that the interactions between different treatments should not be ignored, we collaborate with a large-scale online video-sharing platform (referred to as Platform O hereafter). We plot in Figure 1 the relationships between the treatment effects of two concurrent treatments (more institutional details will be provided in Section 4). We observed the causal effects of all four treatment combinations (2×2) in this example because the experiments are run under the full factorial design. To investigate the heterogeneous responses to the treatments with respect to user covariates, we divide the experimented users into $1,254$ subgroups based on their pre-treatment covariates, including gender, age, location, and degree of activeness. Each observation in Figure 1 represents one such subgroup. Figure 1(a) plots, for each subgroup, the true observed effect size in the experiment as well as the calculated effect size if we assume these two effects are linearly additive. It clearly reveals the *substantial gap* from the ground-truth to simply adopt the linear addition rule in policy evaluation with multiple experiments. Figure 1(b) illustrates that different user groups have drastically diverse responses toward the treatments, some with increasing marginal returns/losses and others with decreasing marginal returns/losses.

Given these observations, in this paper, our main research question is: *When conducting multiple A/B tests and only observing a small subset of treatment combinations, how to estimate and infer*

the causal effect of all treatment combinations and how to identify the optimal treatment combination? As discussed earlier, the most commonly adopted approach to solving this problem is to run individual experiments independently and infer the combined treatment effect by assuming linear additivity of treatment effects, an assumption not supported by our data and sensibly questionable in practice. An alternative solution is the factorial experiment design, which directly tests the causal effect of *each* treatment combination (Box et al. 1978, Wu and Hamada 2011, Dasgupta et al. 2015). However, for full factorial design, the user traffic required to obtain reliable estimation and inference results grows exponentially in the number of treatments, making this approach infeasible for a large-scale platform that runs hundreds of experiments concurrently. Another tempting approach might be to predict the outcome of each user under each treatment combination via an end-to-end ML model such as a deep neural network (DNN) in which one incorporates treatments as inputs. Such an approach is generally not amenable to the inference of causal effects. The inherent bias due to regulation or overfitting leads to an insufficient convergence to the true causal effects (e.g., average treatment effect, ATE) and, consequently, undesired statistical properties, which hinders effective causal inference. (see, e.g., Chernozhukov et al. 2018).

Consequently, we propose a novel statistical framework (called debiased deep learning, DeDL) that combines deep learning (DL) and double/debiased machine learning (DML) to estimate the causal effect of any treatment combination when only observing a subset of combinations. The framework yields asymptotically normal (therefore naturally $\sqrt{n}$-consistent) estimators, thus allowing for valid inference. In addition to its theoretical elegance, our DeDL framework is further validated by its implementation for three real A/B tests on Platform O involving millions of users. Our empirical results suggest that the DeDL approach substantially outperforms the linear-regression- and deep-learning-based benchmarks to accurately estimate and infer the ATE of any treatment combination, and to correctly identify the best combination. Synthetic data further demonstrate the robust performance of DeDL under model misspecification.

We seek to make both theoretical and empirical contributions in this paper, which are summarized as follows.

A Novel Causal Inference Framework for Multiple A/B tests with Theoretical Guarantee. As our main contribution, we provide a novel framework for researchers and practitioners to analyze the treatment effects of concurrent experiments and identify the optimal treatment combination. Our post-experiment causal inference framework leverages a combination of DL and DML to infer the ATE of any treatment combination. More specifically, we propose a DNN with an *interpretable model layer*, which explicitly captures the interactions between different experiments. Our framework also exploits Neyman orthogonality and generates asymptotically normal estimators, which makes statistical inference possible. Transparent and easy-to-check sufficient conditions

for the validity of our framework are derived. In particular, we emphasize that, compared with the full factorial design that requires the user traffic to increase exponentially in the number of individual treatments, the user traffic necessary for estimation and inference by our framework only scales linearly due to the novel model layer in the DNN succinctly capturing the interactions between individual treatments.

Empirical Validation of Our Framework. To demonstrate the practicality of DeDL, we implement the framework for a multiple-experiment setting on Platform O. The unique experimental architecture of these experiments enables us to observe the ground-truth causal effects of any treatment combination, thus allowing for comparing different approaches benchmarked against the ground truth and quantifying their estimation errors. We compare our DeDL approach with a set of linear-regression- and DL-based benchmarks. Importantly, we show that our framework yields much more accurate estimation for the ATE and more precise identification of the optimal treatment combination. Through comprehensive synthetic data in further simulation studies, we demonstrate the robustness of our DeDL framework in the presence of large biases in DNN training and a misspecified model layer.

To the best of our knowledge, this is the first work to validate the practical strength of the theoretically elegant DML-based methods through large-scale field experiments (N >2,000,000) where the unconfoundedness assumption is satisfied. Our discussion shed light on several issues on the practical side when applying not only our framework but also many other popular DML-based methods together with deep neural nets, such as when to expect the debias term adds value to inference and how to use cross-validation errors in the training stage to test if the model structure is misspecified. In addition, our DeDL framework can be readily applied to the analysis of individual-level treatment effects, continuous treatment levels, and observational data under unconfoundedness. Therefore, we believe that our study has the potential to inspire future researchers and practitioners to utilize the combined method of DL and DML for causal inference.

Organization of the paper. The rest of this paper is organized as follows. In Section 2, we position our paper in the relevant literature. In Section 3, we present our debiased deep learning framework to infer treatment effects and identify the optimal treatment combination. Section 4 applies the framework to analyze real-world experiments. In Section 5, we conduct comprehensive synthetic experiments to demonstrate the robust performance of our proposed framework. Section 6 concludes the paper. All proofs are relegated to the Appendix.

2. Literature Review

In the next, we review several streams of literature closely connected to our work.

The theory and applications of DML. The proposed DeDL framework stems from the recent advances in the semiparametric estimation and inference, the DML method in particular (e.g.,

Chernozhukov et al. 2018). Combining ML with Neyman orthogonality, the DML method performs remarkably well in estimating the parameters of interest in the presence of regularization and/or overfitting biases to estimate the nuance parameter(s). In particular, Farrell et al. (2021) establish novel nonasymptotic high-probability bounds for nuance parameter estimation with deep feedforward neural nets. Chiang et al. (2022) extend the DML framework by proposing a multiway cross-fitting algorithm suitable for multiway clustering sampled data such as panel data. Chernozhukov et al. (2022) develop an automatic DML framework using Lasso to learn the debiased term that often presents in the influence function directly from data. Combining DML with optimization further promotes its theoretical development in an operations context. For example, Qi et al. (2022) propose a personalized pricing algorithm by maximizing the expected revenue estimated using DML. We contribute to this literature by adapting the framework of Farrell et al. (2020) for capturing individual heterogeneity, tailoring it for the multiple-experiment setting, and pinpointing the technical conditions required for deriving valid estimators. Furthermore, we derive the estimator for the best-arm identification problem in this setting.

The DML framework has been extensively applied in many empirical settings for causal inference. For example, Knaus (2020) employs DML to evaluate the effectiveness of four labor programs in Switzerland. Dube et al. (2020) utilize DML to obtain debiased estimators for the effects of reward for MTurk workers on project duration to investigate monopsony in online labor markets. Farbmacher et al. (2022) apply DML combined with causal mediation analysis to study the effect of health insurance coverage on general health. Fan et al. (2022) explore the causal effect of maternal smoking on the birth weight of newborn babies via the DML estimator. Leveraging hundreds of experiments on Facebook, Gordon et al. (2022) find that DML implemented with observational data under the selection of ads for users may have substantial biases from the ground truth. While all applications beyond Gordon et al. (2022) use observational data, our research provides the performance the DML method with the data from a set of large-scale field experiments such that the fundamental unconfoundedness assumption is guaranteed. Contrary to the conclusion form Gordon et al. (2022) that DML estimates are far from experimental results, we show that our debiased estimators are accurate and valid for inference, significantly outperforming other benchmarks.

Estimation and inference with multiple experiments. Conventionally, researchers examine multiple-experiment settings through the lens of factorial design (i.e., full or fractional factorial designs). Interested readers are referred to Box et al. (1978) and Wu and Hamada (2011) for the detailed discussions of these classical approaches. Recent works (e.g., Dasgupta et al. 2015, Pashley and Bind 2019) marry such design strategies with the potential outcome framework (Imbens and Rubin 2015) for the study of causal inference. However, factorial design is hardly applicable to large-scale A/B testing platforms, where the number of experiments m can potentially be hundreds

or even thousands. It is next to impossible to obtain the 2^m treatment groups as required by the full-factorial design. Even with the fractional factorial design, the sheer number of treatments implies that practically one can only test $O(m)$ treatment combinations, suggesting only $O(m)$ direct or interaction effects are identifiable. The vast majority of the effects are, however, aliased away. Therefore, factorial design methods are rarely employed on large-scale A/B testing platforms. Proposing a new strategy to deal with the inference problem in such settings, our work applies the DML framework to the empirical analysis in the multiple-experiment setting and requires looser identification conditions. With an appropriately specified form of the response function to the treatment for each individual, we only need to observe $m+2$ treatment combinations for the treatment effect inference of all 2^m combinations. We also apply this framework to a real multiple-experiment setting on Platform O and show the empirical success of the framework in this setting.

Causal inference and its applications to online platforms. Causal inference has long been a central topic in many fields, such as economics, psychology, medical science, marketing, and operations (e.g., see Angrist and Pischke 2009, Wooldridge 2010). Recent advances in ML and high-dimensional statistics have enabled substantial development in this area. To name a few, Xie and Aurisset (2016) and Guo et al. (2021) propose variance reduction techniques that use covariates to adjust estimators and obtain more precise ones with fewer data. Goli et al. (2022) propose a theoretical framework to overcome the bias due to the interference in a ranking experiment on travel websites. Kallus et al. (2018) use matrix factorization and bound the estimation errors for average treatment effects to reduce the noise and measurement error in covariates. Lee and Shen (2018) estimate the winner's curse bias and use it to correct the final estimator for treatment effects. Athey et al. (2018) propose a two-stage approximate residual balancing algorithm to eliminate the bias in estimators obtained through sparse linear models. Arkhangelsky et al. (2021) propose a synthetic difference-in-differences estimator to deal with panel data, which possesses unbiasedness and consistency under regularity assumptions. Zhang and Politis (2022) improve the ridge regression estimator by adding a correction part to debias the original estimator. They use a wild bootstrap algorithm to construct a confidence interval. An influential school of works combine ML methods with causal inference (Athey and Imbens 2016, Wager and Athey 2018). Among this literature, as discussed above, DML (Chernozhukov et al. 2018, Farrell et al. 2020) has received much attention.

In particular, our paper speaks to the applications of causal inference to online platforms. With a large amount of data available on online platforms, works in this area have proliferated in recent years. On the empirical side, field experiments on large-scale platforms enable causal inference in a variety of business settings (e.g., Burtch et al. 2015, Cheung et al. 2017, Edelman et al. 2017, Zhang et al. 2020, Zeng et al. 2022). On the theoretical side, researchers propose innovative methods to

overcome challenges arising from online platforms, such as two-sided randomization (e.g., Ye et al. 2022, Nandy et al. 2021, Johari et al. 2022), sequential experiments (e.g., Xiong et al. 2022, Ju et al. 2019, Song and Sun 2021), and block randomization (Candogan et al. 2021). Whereas this literature typically focuses on the single-experiment setting, we study the inference problem with multiple experiments.

3. De-biased Deep Learning (DeDL) Framework

In Section 3.1, we first introduce the DL framework built upon Farrell et al. (2020) to study the estimation and inference of treatment effects in our multiple experiments setting.[3] Leveraging the semiparametric influence function derived via the pathwise derivative method (see Newey 1994, Farrell et al. 2020), we then demonstrate how we estimate our model in Section 3.2 and how to construct valid estimators to identify the best treatment combination in Section 3.3.

3.1. Structured Deep Learning Model

We first present the DL-based inference framework for multiple experiments in a large platform. There are m concurrent field experiments on the platform, each with binary treatment levels, represented by $\boldsymbol{T} \in \{0,1\}^m$.[4] Without loss of generality, we focus on the binary treatment case, which is a common practice for A/B tests on large-scale platforms, but our framework can be readily extended to continuous and discrete treatment levels. The platform can observe the individual-level response to the treatment $Y \in \mathbb{R}$,[5] along with the individual-level pre-treatment covariates $\boldsymbol{X} \in \mathbb{R}^{d_{\boldsymbol{X}}}$ and treatment level $\boldsymbol{T}$. The treatment assignment mechanism is denoted by the conditional distribution $\nu(\cdot\,|\,\cdot)$, i.e., $\nu(\boldsymbol{t}\,|\,\boldsymbol{x}) = \mathbb{P}[\boldsymbol{T}=\boldsymbol{t}\,|\,\boldsymbol{X}=\boldsymbol{x}]$ for any $\boldsymbol{t} \in \{0,1\}^m$ given any $\boldsymbol{x}$.

Following Farrell et al. (2020), we assume the data generating process (DGP) has the semi-parametric form

$$\mathbb{E}[Y|\boldsymbol{X}=\boldsymbol{x},\boldsymbol{T}=\boldsymbol{t}] = G(\boldsymbol{\theta}^*(\boldsymbol{x}),\boldsymbol{t}), \tag{1}$$

where $G(\cdot,\cdot)$ is the known link function, and $\boldsymbol{\theta}^*(\cdot): \mathbb{R}^{d_{\boldsymbol{X}}} \mapsto \mathbb{R}^{d_{\boldsymbol{\theta}}}$ are the unknown nuisance parameters as functions of covariates $\boldsymbol{x}$. In particular, $\boldsymbol{\theta}^*(\cdot)$ characterizes the heterogeneity in outcomes, and we shall predict them by ML models such as DNNs. The pre-specified link function $G(\cdot,\cdot)$ allows for the flexibility and interpretability of the relationship between the outcome Y and the treatment

[3] We use "multiple experiments", "multiple treatments", and "combinatorial experiments" interchangeably.

[4] *On notations:* Throughout the paper, vectors, and matrices are in boldface. Vectors are written as column vectors, and $\boldsymbol{v}'$ represents the transpose of vector $\boldsymbol{v}$. Random variables are represented by capital letters and their realizations by lower-case letters. The L_2 norm of function $f(\cdot)$ is defined as $||f(\boldsymbol{x})||_{L_2(\boldsymbol{X})} := \mathbb{E}[f(\boldsymbol{X})^2]^{1/2}$. We use $\mathbb{E}_n$ to denote the sample average and $\boldsymbol{M} \succ 0$ to denote that matrix $\boldsymbol{M}$ is positive definite.

[5] For expositional ease, we focus on the one-dimensional outcome setting throughout this paper. In practice, online platforms could be interested in multiple outcome metrics (e.g., the number of active users and revenue); and the extension of our framework to the case where $\boldsymbol{Y} \in \mathbb{R}^{d_{\boldsymbol{Y}}}$ $(d_{\boldsymbol{Y}} > 1)$ is straightforward.

combination $\boldsymbol{t}$. For example, if the link function is linear, i.e., $\mathbb{E}[Y|\boldsymbol{X}=\boldsymbol{x},\boldsymbol{T}=\boldsymbol{t}]=\boldsymbol{\theta}^*(\boldsymbol{x})'\boldsymbol{t}$, the effect of any treatment combination equals the linear summation of each individual treatment effect therein. Combining the interpretability of the link function and the generalizability of ML, our framework not only provides practitioners with accurate inferences for the experimental outcome but also delineates the interactions between different treatments.

We apply our framework to address the following two essential questions of both academic and practical values. (a) What is the ATE for each treatment combination? (b) Which treatment combination is the most valuable for the platform (i.e., with the highest ATE)? The second question is also referred to as the best-arm identification problem in the literature (Gabillon et al. 2012, Lattimore et al. 2016). Our framework involves a two-stage procedure. In the first *training* stage, we adopt DL to obtain a consistent estimator of the unknown parameter $\boldsymbol{\theta}^*(\cdot)$, which is denoted by $\hat{\boldsymbol{\theta}}(\cdot)$. In the second *estimation & inference* stage, based on the trained parameter $\hat{\boldsymbol{\theta}}(\cdot)$, we construct asymptotically normal estimators for the quantities of managerial interest (e.g., ATE), thus yielding valid inferences.

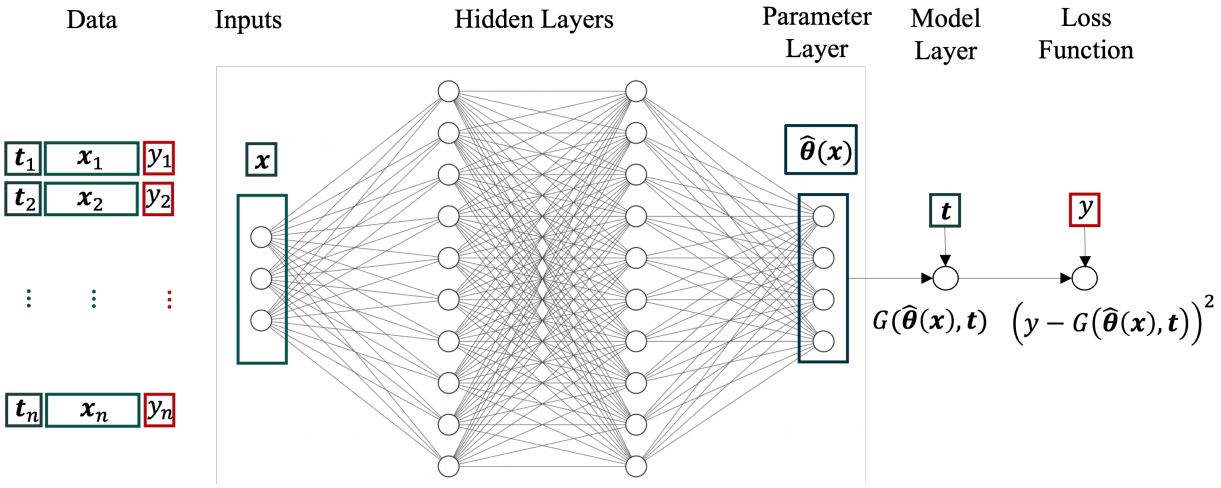


Figure 2 Illustration of Deep Neural Network with the Structured Model Layer

Inspired by our DGP (1), in the *training* stage, we use DNNs to approximate the unknown parameter $\boldsymbol{\theta}^*(\cdot)$. We then add a model layer (i.e., link function) on top of the DNNs to connect the outcome with the estimated parameters $\hat{\boldsymbol{\theta}}(\cdot)$ and treatment level $\boldsymbol{t}$. Figure 2 illustrates the whole model from $\boldsymbol{X}$ and $\boldsymbol{T}$ to Y. We use $\{(y_i,\boldsymbol{x}_i',\check{\boldsymbol{t}}_i')':1\le i\le n\}$ to denote the observed data, and use the check symbol over the treatment $\check{\boldsymbol{t}}$ to represent the realized treatment level in experiments. Note that DGP (1) suggests that the true parameter functions solve $\boldsymbol{\theta}^*(\cdot)\in\arg\min_{\boldsymbol{\theta}(\cdot)}\mathbb{E}[(Y-G(\boldsymbol{\theta}(\boldsymbol{X}),\boldsymbol{T}))^2]$. Consequently, we use the mean squared error, denoted by $\ell(y,\boldsymbol{t},\boldsymbol{\theta}(\boldsymbol{x}))=(y-G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}))^2$ as the loss

function used to train the estimator $\hat{\boldsymbol{\theta}}(\cdot)$. The estimators of $\boldsymbol{\theta}^*(\cdot)$ can be obtained by minimizing the empirical loss on the training data set

$$\hat{\boldsymbol{\theta}}(\cdot) := \underset{\boldsymbol{\theta}(\cdot)\in\mathcal{F}_{\text{DNN}}}{\arg\min} \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \check{\boldsymbol{t}}_i, \boldsymbol{\theta}(\boldsymbol{x}_i)) := \frac{1}{n}\sum_{i=1}^{n} (y_i - G(\boldsymbol{\theta}(\boldsymbol{x}_i), \check{\boldsymbol{t}}_i))^2, \tag{2}$$

where $\mathcal{F}_{\text{DNN}}$ is a pre-specified class of (deep) fully-connected neural nets. We will detail the choice of the (known) link function $G(\cdot,\cdot)$ and the training process of $\hat{\boldsymbol{\theta}}(\cdot)$ in Section 3.2 below.

In the *estimation and inference* stage, the key is to construct estimators for (a) the ATE of any treatment combination and (b) the improvement in ATE for the identified best arm over any other treatment combination. Formally, we define

$$\mu(\boldsymbol{t}) := \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t})] - \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t}_0)] \tag{3}$$

as the ground-truth ATE of treatment combination $\boldsymbol{t} \in \{0,1\}^m$, where $\boldsymbol{t}_0 = \mathbf{0}$ represents the control condition without any treatments. Denote $\hat{\mu}(\boldsymbol{t})$ as our proposed estimator for $\mu(\boldsymbol{t})$, we show that our framework provides $\sqrt{n}$-consistent, asymptotically normal, and semiparametric efficient estimators $\hat{\mu}(\cdot)$.

We denote $\boldsymbol{t}^*$ as the best treatment, the treatment with the largest value of $\mu(\boldsymbol{t})$, and let $\hat{\boldsymbol{t}}^* := \arg\max_{\boldsymbol{t}\in\{0,1\}^m} \hat{\mu}(\boldsymbol{t})$ be our proposed estimator for $\boldsymbol{t}^*$. We note that the ATE increment of the true best arm over any treatment combination is the difference between their (ground truth) ATEs,

$$\tau(\boldsymbol{t}) := \mu(\boldsymbol{t}^*) - \mu(\boldsymbol{t}),\ \boldsymbol{t} \in \{0,1\}^m, \tag{4}$$

which quantifies the loss of using $\boldsymbol{t}$ instead of $\boldsymbol{t}^*$. Notice that identifying the best arm is equivalent to identifying the ATE increment of each arm. Therefore, let $\hat{\tau}(\boldsymbol{t}) := \hat{\mu}(\hat{\boldsymbol{t}}^*) - \hat{\mu}(\boldsymbol{t})$ be an estimator of $\tau(\boldsymbol{t})$ for each $\boldsymbol{t} \in \{0,1\}^m$. We show that our proposed estimator $\hat{\tau}(\cdot)$ is also $\sqrt{n}$-consistent and asymptotically normal, and the empirical best-arm agrees with the true best-arm with probability approaching one.

3.2. Training Stage

In this subsection, we aim to theoretically show that, the empirical loss minimization estimator $\hat{\boldsymbol{\theta}}(\cdot)$ defined by (2) converges to the unknown parameter $\boldsymbol{\theta}^*(\cdot)$ sufficiently fast for valid inference in the second stage under reasonable regularity assumptions and a properly chosen link function $G(\cdot,\cdot)$ that well fits our multiple A/B tests setting. Throughout this paper, we make a regularity assumption commonly used in the DNN estimation literature (e.g., Yarotsky 2017, Farrell et al. 2020), i.e., Assumption 3 in Appendix A.1 It requires that the unknown ground-truth parameter functions $\boldsymbol{\theta}^*(\cdot)$ are uniformly bounded and sufficiently smooth.

We capture the richness of individual responses to different treatments with the non-parametric function $\boldsymbol{\theta}^*(\boldsymbol{X})$, while given $\boldsymbol{\theta}^*(\boldsymbol{X})$ we essentially assume that the individual outcomes are fully structured and described by $G(\cdot,\cdot)$. The choice of the link function should, on the one hand, reflect the economic nature of the multiple A/B tests business context and, on the other hand, be accompanied by associated treatment assignment mechanisms to ensure the identifiability and convergence of the estimates $\hat{\boldsymbol{\theta}}(\cdot)$. We propose the following link functions with clear economic interpretations.

ASSUMPTION 1 **(Link Functions)**. *We consider the following link functions* $G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})$ *where* $\boldsymbol{\theta}(\cdot):\mathbb{R}^{d_{\boldsymbol{X}}}\mapsto\mathbb{R}^{d_{\boldsymbol{\theta}}}$.

(a) Multiplicative Form. $G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})=\theta_0(\boldsymbol{x})(1+\theta_1(\boldsymbol{x})t_1)\ldots(1+\theta_m(\boldsymbol{x})t_m)$, *where* $\mu\le\theta_0(\boldsymbol{x})\le M$, *and* $\mu\le 1+\theta_k(\boldsymbol{x})\le M$, $k=1,\ldots,m$, *uniformly in* $\boldsymbol{x}$, *for some* $M>\mu>0$.

(b) Standard Sigmoid Form. $G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})=a/\big(1+\exp(-(\theta_0(\boldsymbol{x})+\theta_1(\boldsymbol{x})t_1+\cdots+\theta_m(\boldsymbol{x})t_m))\big)+b$, *where* $a\neq 0$ *and* b *are known constants.*

(c) Generalized Sigmoid Form I. $G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})=\theta_{m+1}(\boldsymbol{x})/\big(1+\exp(-(\theta_1(\boldsymbol{x})t_1+\cdots+\theta_m(\boldsymbol{x})t_m))\big)$.

(d) Generalized Sigmoid Form II. $G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})=\theta_{m+1}(\boldsymbol{x})/\big(1+\exp(-(\theta_0(\boldsymbol{x})+\theta_1(\boldsymbol{x})t_1+\cdots+\theta_m(\boldsymbol{x})t_m))\big)$.

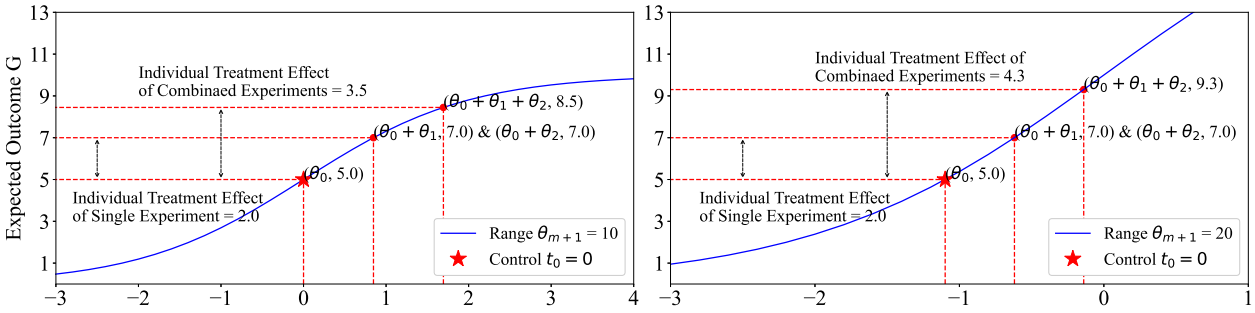


Figure 3 Illustration of Generalized Sigmoid Form II with Two Types Platform Users: Marginal Decreasing (Left) and Marginal Increasing (Right)

All four link functions in Assumption 1 capture the heterogeneity with respect to different covariates $\boldsymbol{x}$. The link function of the *Multiplicative Form* (Assumption 1(a)) assumes multiplicative *relative effect size* for different individual treatments, e.g., if each of two treatments increases the effect by 10%, then combined treatment increases by $(1+10\%)(1+10\%)-1=21\%$. However, *Multiplicative Form* can only characterize the increasing marginal effect due to its global convexity.

The link function of the sigmoid forms (Assumption 1(b), (c), and (d)) leverages the convex-concave structure of the sigmoid function, thus capturing both increasing and decreasing marginal effects at the individual level. As a result, it is able to capture both marginal increasing and

decreasing effects in ATE. Comparing three sigmoid link functions, one can notice that the *Standard Sigmoid Form* (Assumption 1(b)) and the *Generalized Sigmoid Form I* (Assumption 1(c)) are special cases of the *Generalized Sigmoid Form II* (Assumption 1(d)).

We illustrate the co-existence of decreasing and increasing marginal effects with the *Generalized Sigmoid Form II* through an example. We consider two experiments ($\boldsymbol{t}_i \in \{0,1\}^2$) and two user types as shown in Figure 3, where the y-axis represents an individual's average outcome $\mathbb{E}[Y_i|\boldsymbol{x}_i, \boldsymbol{t}_i]$ follows the *Generalized Sigmoid Form II* (Assumption 1(d)). Under the control condition (i.e., $\boldsymbol{t} = \boldsymbol{t}_0 = (0,0)'$), the expected outcome of both user types is 5 (i.e., $\mathbb{E}[Y_i|\boldsymbol{x}_i, \boldsymbol{t} = (0,0)'] = 5$ for all i), whereas the treatment effect of each experiment on each user type is 2 (i.e., $\mathbb{E}[Y_i|\boldsymbol{x}_i, \boldsymbol{t} = (1,0)'] = \mathbb{E}[Y_i|\boldsymbol{x}_i, \boldsymbol{t} = (0,1)'] = 7$ for all i). For the first (resp. second) user type, $\theta_{m+1} = 10$ (resp. $\theta_{m+1} = 20$). Straightforward calculation implies that $\mathbb{E}[Y_i|\boldsymbol{x}_i, \boldsymbol{t} = (1,1)'] = 8.5$ (i.e., the ATE is $8.5 - 5 = 3.5 < 2 + 2 = 4$, suggesting the decreasing marginal effects) for the first user type and $\mathbb{E}[Y_i|\boldsymbol{x}_i, \boldsymbol{t} = (1,1)'] = 9.3$ for the second user type (i.e., the ATE is $9.3 - 5 = 4.3 > 4$, suggesting the increasing marginal effects). Therefore, if the first type of users takes more (resp. less) than 36% of the entire population, the platform will have a decreasing (resp. an increasing) marginal ATE.

The *Standard Sigmoid Form* with known constants a and b may be restrictive, and cannot model the different outcome ranges for different individuals. The *Generalized Sigmoid Forms I and II* resolve this issue by incorporating the parameter $\theta_{m+1}(\boldsymbol{x})$, allowing for flexible and heterogeneous ranges of outcomes. Hence, we adopt the link function of *Generalized Sigmoid Form II* in our empirical study.

With our proposed link functions, we are ready to show the identifiability and convergence rate of the DNN in our framework. For identifiability, we further need the assumption that the treatment assignment mechanism is sufficiently "regular" (see Assumption 4 in Appendix A.1). For convergence, the additional assumption on the data observations being i.i.d. and bounded, and the nonparametric function $\boldsymbol{\theta}^*(\boldsymbol{x})$ being sufficiently smooth is also imposed (see Assumption 3 in Appendix A.1). Formally, we have the following proposition, the proof of which is relegated to Appendix A.3.

Proposition 1 (Identifiability and Convergence). *The following statements hold.*

(a) Under Assumptions 1 and 4 (in Appendix A.1), the parameter function $\boldsymbol{\theta}^*(\boldsymbol{x})$ can be nonparametrically identified in DGP (1).

(b) Under Assumptions 1, 3 (in Appendix A.1), and 4, if the structured DNN as illustrated in Figure 2 has width $H = O(n^{d_{\boldsymbol{X}}/2(p+d_{\boldsymbol{X}})} \log^2 n)$ and depth $L = O(\log n)$, there exists a positive constant C which depends on the fixed quantities in Assumption 3, such that with probability at least $1 - \exp(n^{-d_{\boldsymbol{X}}/(p+d_{\boldsymbol{X}})} \log^8 n)$, it holds

$$\|\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k^*\|_{L_2(\boldsymbol{X})}^2 \leq C\Big\{n^{-\frac{p}{p+d_{\boldsymbol{X}}}} \log^8 n + \frac{\log\log n}{n}\Big\},$$

for each $k \in [d_{\boldsymbol{\theta}}]$ *when* n *large enough.*

The key step to prove Proposition 1 is translating the convergence of DNN estimation on outcomes Y into that of the parameter function estimates $\hat{\boldsymbol{\theta}}(\cdot)$ under the treatment assignment mechanism sufficient for identification. The convergence rate given by Proposition 1 may not be optimal (see Farrell et al. 2021), but sufficiently fast for the subsequent inference in our setting. We also remark that, consistent with the DL estimation literature, the optimization errors are ignored in the *training* stage to obtain $\hat{\boldsymbol{\theta}}(\cdot)$ in our theoretical analysis. Another important implication of Proposition 1 is that, for our link functions in Assumption 1, it suffices to observe $m+2$ treatment combinations (see Assumption 4 in Appendix A.3 for details), which is orders of magnitude smaller than 2^m, to ensure the identifiability and sufficiently fast convergence. In other words, suppose there are 10 different treatments and in turn $2^{10} = 1024$ possible treatment combinations, our framework only needs to observe $10+2 = 12$ combinations to estimate the parameter function $\hat{\boldsymbol{\theta}}(\cdot)$ with sufficient convergence, which is only $12/1024 = 1.2\%$ of the total possible combinations.

3.3. Estimation and Inference Stage

Next, we leverage our nonparametric estimates $\hat{\boldsymbol{\theta}}(\boldsymbol{x})$ obtained through DNNs to construct asymptotically normal estimators for $\mu(\cdot)$ and $\tau(\cdot)$. We define the *advantage function* of the treatment combination $\boldsymbol{t}^1 \in \{0,1\}^m$ over the treatment combination $\boldsymbol{t}^2 \in \{0,1\}^m$ as

$$H(\boldsymbol{x}, \boldsymbol{\theta}(\boldsymbol{x}); \boldsymbol{t}^1, \boldsymbol{t}^2) := G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}^1) - G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}^2).$$

Thus, the ATE of any treatment combination $\boldsymbol{t} \in \{0,1\}^m$ can be rewritten as

$$\mu(\boldsymbol{t}) = \mathbb{E}[H(\boldsymbol{x}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0)].$$

Analogously, the ATE increment of the identified best-arm $\hat{\boldsymbol{t}}^*$ over any treatment combination $\boldsymbol{t}$ can be written as

$$\tau(\boldsymbol{t}) := \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{x}), \hat{\boldsymbol{t}}^*)] - \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{x}), \boldsymbol{t})] = \mathbb{E}[H(\boldsymbol{x}, \boldsymbol{\theta}^*(\boldsymbol{x}); \hat{\boldsymbol{t}}^*, \boldsymbol{t})].$$

Hence, it suffices to test the 1-sided hypothesis $\tau(\boldsymbol{t}) \geq 0$ for all $\boldsymbol{t}$. In this section, we focus on ATE discussion, whereas the detailed discussion of best-arm identification is relegated to Appendix A.7.

Note that we cannot directly use the plug-in estimator $\hat{\mu}(\boldsymbol{t}) = \frac{1}{n}\sum_{i=1}^{n} H(\boldsymbol{x}_i, \hat{\boldsymbol{\theta}}(\boldsymbol{x}_i); \boldsymbol{t}, \boldsymbol{t}_0)$ for ATE estimation, which is generally not asymptotically normal due to the potential bias of ML models. To solve this issue, our inference is mainly built upon the semi-parametric technique — influence function (a.k.a. Neyman orthogonal score), which implies the first-order insensitive to perturbations in the nuisance parameters. We refer interested readers to Newey (1994) and Subsection 2.2.5 in Chernozhukov et al. (2018) for more discussions of influence functions and Neyman orthogonality. Similar to other work using influence function in semi-parametric statistics, we make the following assumption:

ASSUMPTION 2. *For all $\boldsymbol{t} \in \{0,1\}^m$, the following conditions hold uniformly with respect to all $\boldsymbol{x}$: (i) The DGP (1) holds; (ii) $\boldsymbol{\Lambda}(\boldsymbol{x}) := 2\mathbb{E}[G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{T})G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{T})'|\boldsymbol{X}=\boldsymbol{x}]$ is invertible with bounded inverse, where $G_{\boldsymbol{\theta}}(\cdot,\boldsymbol{t})$ is the gradient of $G(\cdot,\boldsymbol{t})$ with respect to $\boldsymbol{\theta}$; and (iii) The ATE $\mu(\boldsymbol{t})$ is identified and pathwise differentiable.*

We remark that Assumption 2 is standard and not restrictive in the semiparametric statistics literature. The invertibility of $\boldsymbol{\Lambda}(\boldsymbol{x})$ is commonly assumed for deriving the influence function. As shown in Appendix A.5, this condition can be translated into a lenient one under the *Generalized Sigmoid Form II*. Finally, the identification of $\mu(\boldsymbol{t})$ immediately follows Proposition 1(a) and the pathwise differentiability of $\mu(\boldsymbol{t})$ is also standard.

PROPOSITION 2. *Suppose Assumption 2 holds. Then, the influence function for $\mu(\boldsymbol{t})$ is $\psi(\boldsymbol{z},\boldsymbol{\theta},\boldsymbol{\Lambda};\boldsymbol{t},\boldsymbol{t}_0)-\mu(\boldsymbol{t})$ with,*

$$\psi(\boldsymbol{z},\boldsymbol{\theta},\boldsymbol{\Lambda};\boldsymbol{t},\boldsymbol{t}_0)=H(\boldsymbol{x},\boldsymbol{\theta}(\boldsymbol{x});\boldsymbol{t},\boldsymbol{t}_0)-H_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{\theta}(\boldsymbol{x});\boldsymbol{t},\boldsymbol{t}_0)'\boldsymbol{\Lambda}(\boldsymbol{x})^{-1}\ell_{\boldsymbol{\theta}}(y,\check{\boldsymbol{t}},\boldsymbol{\theta}(\boldsymbol{x})), \tag{5}$$

where $\boldsymbol{\Lambda}(\boldsymbol{x}) := 2\mathbb{E}[G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{T})G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{T})'|\boldsymbol{X}=\boldsymbol{x}]$, $G_{\boldsymbol{\theta}}$ is the gradient of G with respect to $\boldsymbol{\theta}$, $H_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{\theta}(\boldsymbol{x});\boldsymbol{t},\boldsymbol{t}_0) := G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})-G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}_0)$ is the gradient of $H(\boldsymbol{x},\cdot;\boldsymbol{t},\boldsymbol{t}_0)$ with respect to $\boldsymbol{\theta}$ and $\ell_{\boldsymbol{\theta}}(y,\check{\boldsymbol{t}},\boldsymbol{\theta}(\boldsymbol{x})) := 2G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\check{\boldsymbol{t}})(G(\boldsymbol{\theta}(\boldsymbol{x}),\check{\boldsymbol{t}})-y)$ is the gradient of $\ell(y,\check{\boldsymbol{t}},\cdot)$ with respect to $\boldsymbol{\theta}$.

The influence function defined by Eqn. (5) contains a plug-in term $H(\boldsymbol{x},\boldsymbol{\theta}(\boldsymbol{x});\boldsymbol{t},\boldsymbol{t}_0)$ and a debiasing term $-H_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{\theta}(\boldsymbol{x});\boldsymbol{t},\boldsymbol{t}_0)'\boldsymbol{\Lambda}(\boldsymbol{x})^{-1}\ell_{\boldsymbol{\theta}}(y,\check{\boldsymbol{t}},\boldsymbol{\theta}(\boldsymbol{x}))$. Therefore, we call our framework de-biased deep learning (DeDL). In particular, the debiasing term is the multiplication of three gradients, which is easy to compute with the known treatment assignment mechanism $\nu(\cdot\,|\,\boldsymbol{x})$.

Based on this influence function and the cross-fitting technique (e.g., Chernozhukov et al. 2018), we can construct estimators as illustrated in Algorithm 1. We refer interested readers to Appendix A.6 for the details of constructing the estimators $\hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$ by Eqn. (23), $\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\mu)$ by Eqn. (24), and the confidence interval $\widehat{\mathcal{CI}}_{\mathsf{DeDL}}(\boldsymbol{t};\mu)$ by Eqn. (25).

We now present the following Proposition 3 on the asymptotic normality, which follows from Chernozhukov et al. (2018).

PROPOSITION 3. *Suppose $\{\boldsymbol{z}_i : i=1,\ldots,n\}$ is a random sample that obeys Assumption 2 and $\hat{\boldsymbol{\Lambda}}_s(\boldsymbol{x}_i)$ is invertible uniformly for $\boldsymbol{x}$. Furthermore, we assume for all subsamples $s=1,2,\ldots,S$, the estimators obey $\|\hat{\theta}_{sk}-\theta^*_{sk}\|_{L_2(\boldsymbol{X})}=o(n^{-1/4})$, $k\in\{1,\ldots,d_{\boldsymbol{\theta}}\}$, which holds under the assumptions and regularity conditions (for the structured DNN) of Proposition 1(b).*

(a) For any treatment level $\boldsymbol{t}\in\{0,1\}^m$,

$$\sqrt{n}\big(\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\mu)\big)^{-1/2}(\hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})-\mu(\boldsymbol{t}))\to_d\mathcal{N}(0,1),$$

Algorithm 1 DeDL Framework

1: (Cross-fitting) Split data samples into S non-overlapping folds $\mathcal{S}_s,\ s=1,\ldots,S$.
2: (Training) For each fold s, use the complement of $\mathcal{S}_s$ to train DNN to get $\hat{\boldsymbol{\theta}}_s(\cdot)$ based on (2), and compute $\hat{\boldsymbol{\Lambda}}_s(\cdot)=2\mathbb{E}[G_{\boldsymbol{\theta}}(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}),\boldsymbol{T})G_{\boldsymbol{\theta}}(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}),\boldsymbol{T})'|\boldsymbol{X}=\boldsymbol{x}]$.
3: (ATE Estimation and Inference) For each $\boldsymbol{t}\in\{0,1\}^m$, leverage the influence function ψ and use data $\mathcal{S}$ to construct the ATE estimator $\hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$ and variance estimator $\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\mu)$. Conduct ATE inference based on $\hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$ and $\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\mu)$.
4: (Best-arm Identification) Find empirical best arm $\hat{\boldsymbol{t}}^*:=\arg\max_{\boldsymbol{t}}\hat{\mu}(\boldsymbol{t})$. Similarly, use influence function ψ and cross-fitting to construct estimators $\hat{\tau}_{\mathsf{DeDL}}(\boldsymbol{t})$ and $\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\tau)$ (see Appendix A.7) for the inference on best arm identification.

(b) Furthermore suppose the best arm $\boldsymbol{t}^*:=\arg\max_{\boldsymbol{t}\in\{0,1\}^m}\mu(\boldsymbol{t})$ *is unique. We have* $\hat{\boldsymbol{t}}^*=\boldsymbol{t}^*$ *with probability approaching one as the sample size goes to infinity, and for any treatment level* $\boldsymbol{t}\in\{0,1\}^m$*,*

$$\sqrt{n}\big(\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\tau)\big)^{-1/2}(\hat{\tau}_{\mathsf{DeDL}}(\boldsymbol{t})-\tau(\boldsymbol{t}))\to_d\mathcal{N}(0,1).$$

Under reasonable assumptions, Proposition 1(b) validates the convergence rate requirement of $\hat{\boldsymbol{\theta}}$, $\|\hat{\theta}_{sk}-\theta^*_{sk}\|_{L_2(\boldsymbol{X})}=o(n^{-1/4})$, to establish the normality of the proposed estimators $\hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$ and $\hat{\tau}_{\mathsf{DeDL}}(\boldsymbol{t})$. The formal proof of Proposition 3 can be found in Appendix A.7. Importantly, the probability of failing to identify the true best arm vanishes as the sample size grows large. Therefore, Proposition 3 establishes the valid inference for ATE and best-arm identification under our DeDL framework. For the rest of this paper, we validate this framework with both experimental and synthetic data and demonstrate its superior performance over commonly used benchmarks.

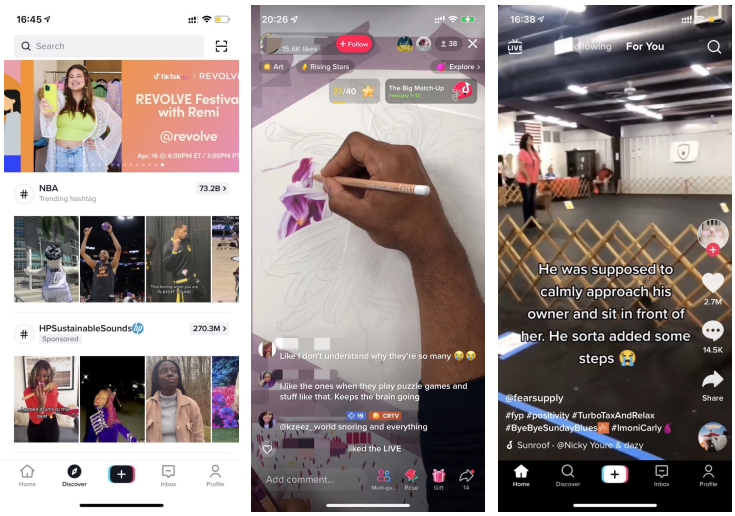
4. Application to Field Experiment Data

In this section, we conduct field experiments to test our theory. We apply our DeDL framework to the experimental data from Platform O. The empirical results highlight that, in the presence of unobserved treatment combinations, our approach more accurately estimates the ATE of any treatment combination than the commonly used benchmarks, and more efficiently identifies the optimal combination.

4.1. Field Setting, Experiments, and Data

To empirically validate our proposed framework, we collaborate with Platform O, which features interactive short videos. Platform O is one of the largest short-video platforms that serves billions of users globally every day. For Platform O, its users (referred to as "she" hereafter) may view the short videos on different product pages. In our empirical analysis, we focus on three main pages of Platform O. To better illustrate, we refer to similar pages on Tiktok: (i) the Discover Page (DP),

(ii) the Live Page (LP), and (iii) the For You Page (FYP) (See Figure 4.). On the DP, the platform generates trendy hashtags and videos based on users' preferences. On the LP, users are exposed to live streams. On the FYP, the platform recommends the best-performing videos (measured by, e.g., total click-throughs, total watch time duration, like rate, and forward rate, etc.) that fit each user's idiosyncratic interest. Users of Platform O can easily switch to any of these pages at any time they are using the platform.



(a) Discover Page (DP) (b) Live Page (LP) (c) For You Page (FYP)

Figure 4 An Illustration of Three Pages

Like all other large-scale UGC platforms such as Facebook and TikTok, Platform O simultaneously runs hundreds of A/B tests every day to evaluate and optimize its product designs and recommendation algorithms. For most of these A/B tests, the platform's main objective is to improve user engagement, which can be well approximated by the amount of screen time a user spends on the platform per day. Each experiment is randomized based on a distinct hash function of user ids which ensures that the treatment assignment mechanisms from any two experiments are independent. In this paper, we focus on a unique set of three A/B tests or treatments, each of which examines the effect of a major adjustment to the video recommendation algorithm on one of three main pages of Platform O (i.e., DP, LP, and FYP).[6] There is one unique feature of this set of experiments while the other sets of experiments on Platform O do not have: since these three experiments focus on the same population, the outcomes of the users under all $2^3 = 8$ possible treatment combinations are all *observable*. The main reason that the algorithm team in Platform

[6] For the sake of simplicity, we refer the changes of several parameters in recommendation algorithm as major adjustment. In practice, the major adjustment could be adding weights for videos created by popular authors, increasing exposure of live streams viewed by nearby users, changing the degree of diversity of videos in users' feeds, and so on.

O decided to test all three experiments on the same population is precise that they want to understand how much money they have left on the table by running each experiment independently and not finding the best combination.

With this unique setting, we can use this set of experiments to quantify the ground-truth ATE of any treatment combination, thus greatly facilitating our analysis to provide convincing validations of our DeDL framework against commonly adopted benchmarks. Throughout our empirical analysis, we use a three-dimensional binary vector $\boldsymbol{T} \in \mathcal{T} := \{0,1\}^3$ to represent the (random) treatment combination applied to a user, where the first component refers to whether the user is treated on DP, the second refers to whether she is treated in on LP, and the third refers to whether she is treated on FYP. Denote $\boldsymbol{t} \in \mathcal{T}$ as the realization of $\boldsymbol{T}$. For example, a treatment vector $\{1,0,0\}$ indicates that the user has received treatment on DP but is in the control group on LP and FYP.

This set of three experiments targeted $4,449,470$ users in total between January 10, 2021 and February 1, 2021. Throughout our analysis, we use the total screen time of all three pages per day for each user as the outcome variable, consistent with the platform's primary objective to boost user engagement. To fully leverage the power of our DeDL framework, we have also collected the pre-treatment covariate data for the users targeted by the experiments. The covariates adopted in our analysis include 16 discrete variables (such as gender, frequent residence area, age range and the user's activeness degree) and 10 continuous variables (such as the video-watching duration on each page per day in the 10 days right before the experiments). Table 5 (in Appendix B.1) gives a detailed description of all the covariates used in our analysis.

For each experiment, any targeted user, regardless of her pre-treatment covariates, was independently and randomly assigned to the treatment group (i.e., the new algorithm on the respective page was applied) with the probability 0.6 and to the control group (i.e., the baseline algorithm was applied) with the probability 0.4 in each experiment. Therefore, because all treatment assignments are orthogonal, each targeted user with covariates $\boldsymbol{X} = \boldsymbol{x}$ was assigned to the treatment combination $\boldsymbol{t} \in \{0,1\}^3$ with probability

$$\nu(\boldsymbol{t}|\boldsymbol{x}) = \mathbb{P}[\boldsymbol{T} = \boldsymbol{t}|\boldsymbol{X} = \boldsymbol{x}] = \prod_{k=1}^{3} \Big(0.4 \cdot \mathbb{I}[t_k = 0] + 0.6 \cdot \mathbb{I}[t_k = 1]\Big).$$

The ATE under treatment combination $\boldsymbol{t}$ is simply the expected outcome y under $\boldsymbol{t}$ compared to that under $\boldsymbol{t}_0 = (0,0,0)'$ over the same population $\boldsymbol{X}$. For a fair comparison of ATEs over different treatment combinations, we need to keep the user covariates similarly distributed under different treatment combinations in $\mathcal{T}$. Hence, we adopt stratified sampling to randomly select $2,066,606$ users from those who are targeted by all three experiments.[7] Specifically, based on the moments

[7] Although in principle our randomization should guarantee balanced covariates, because there are many covariates and some covariates' distributions have long tails, the covariates are not in fact perfectly balanced across all 8

and quantiles of the covariate distribution, we partition the covariate space into 69,111 strata, and then randomly sample the same number of users whose covariates lie within the stratum for each treatment combination. After the stratified sampling, we construct a new dataset that has about 258,325 users under each treatment combination (see Column (3) of Table 2) and hence 2,066,606 users in total. Hereafter, we call the data sample after stratified sampling the *stratified sample*, and all the empirical analysis from now on is performed on the stratified sample. For the stratified sample, it can be seen that $\mathbb{P}[T_k = 1] = \mathbb{P}[T_k = 0] = 0.5$ $(k = 1, 2, 3)$, independently distributed for different A/B tests. We detail the exact procedure of our stratified sampling in Appendix B.2.

To confirm the success of randomization among our stratified sample of users, we compare users under different treatment combinations in their gender, activeness on the platform, frequent residence area, pre-experiment active days, pre-experiment screen time of DP, LP and FYP, and pre-experiment app usage duration. As we can see from Table 1, the seven treatment combinations have similar proportions of male users, high-active users, and users from the south as the baseline combination $\boldsymbol{t}_0 = (0, 0, 0)'$. Moreover, the summary statistics of the covariates during 10 days prior to the experiments further assure that there is no significant difference between the average active days, average page screen time, and average app usage duration of the users under seven treatment combinations and those under baseline combination (all p-values > 0.05). Given the balanced user demographic and pre-experiment behavior covariates under different treatment combinations in our stratified sample, the difference between the outcome variables under different treatment combinations should be attributed to the experimental interventions, i.e., the implementation of new algorithms on different pages of Platform O. Furthermore, the randomization check reported in Table 1 also demonstrates that the covariate distributions are fairly similar with respect to different treatment combinations for the stratified sample, confirming that our DeDL framework can be applied with validity.

Table 2 documents the ground-truth ATE of all treatment combinations on the total screen time of all three pages per day benchmarked against the case where the baseline algorithm is applied in all three pages (i.e., $\mu(\boldsymbol{t})$ for all $\boldsymbol{t} \in \mathcal{T}$). To protect the sensitive data of Platform O, we only report the relative ATEs (see Columns (1) of Table 2). We emphasize that the orthogonal deployment of these three experiments enables us to observe the ground-truth ATE of *all* seven treatment combinations and, thus, to provide ground truth ATEs for us to validate our DeDL framework.

In order to validate our DeDL framework, we need to assume that some treatment conditions are unobserved and use our framework to recover these "unobserved" conditions and compared

conditions. This leads to an inconsistency to (3) and the theoretical discussion (Proposition 2) if one directly use sample means as the ground truth to validate the treatment effects. One solution would be to redefine treatment effect and re-derive a rather complex new influence function which admit different covariate distributions, but we opt for a simpler approach, the stratified sampling.

Table 1 Randomization Check

	Treatment Combination	$(0,0,0)'$	$(0,0,1)'$	$(0,1,0)'$	$(1,0,0)'$	$(1,1,1)'$	$(1,1,0)'$	$(1,0,1)'$	$(0,1,1)'$
User Demographics	Proportion of Male Users	60.51%	60.63% (0.41)	60.31% (0.13)	60.49% (0.85)	60.40% (0.40)	60.36% (0.26)	60.30% (0.11)	60.31% (0.15)
	Proportion of High-Active Users	29.67%	29.65% (0.88)	29.60% (0.55)	29.73% (0.62)	29.77% (0.44)	29.65% (0.85)	29.80% (0.34)	29.66% (0.91)
	Proportion of Users from the South	39.83%	39.64% (0.16)	39.90% (0.63)	39.80% (0.88)	39.85% (0.90)	39.68% (0.26)	39.69% (0.29)	39.74% (0.48)
User Behaviors 10 Days Prior to the Experiments	Average Active Days per User	6.2517	6.2466 (0.68)	6.2564 (0.71)	6.2489 (0.82)	6.2450 (0.59)	6.2494 (0.85)	6.2577 (0.63)	6.2575 (0.64)
	Average DP Screen Time per User	0.0173	0.0144 (0.31)	0.0151 (0.45)	0.0152 (0.46)	0.0140 (0.25)	0.0145 (0.34)	0.0152 (0.46)	0.0157 (0.58)
	Average LP Screen Time per User	0.0126	0.0091 (0.21)	0.0146 (0.49)	0.0104 (0.42)	0.0136 (0.75)	0.0145 (0.52)	0.0103 (0.42)	0.0159 (0.25)
	Average FYP Screen Time per User	0.0048	0.0022 (0.36)	0.0033 (0.62)	0.0044 (0.91)	0.0010 (0.18)	0.0033 (0.61)	0.0015 (0.25)	0.0024 (0.41)
	Average App Usage Duration per User	0.0129	0.0111 (0.55)	0.0114 (0.62)	0.0114 (0.61)	0.0104 (0.39)	0.0103 (0.37)	0.0120 (0.76)	0.0117 (0.69)

Note: p-values of t-tests between the baseline combination (0,0,0)' and other treatment combinations are reported in the parentheses. To protect sensitive data, the reported metrics of all screen time and app usage duration are rescaled.

Table 2 Ground-Truth ATE and Best-Arm Identification of 8 Treatment Combinations

Treatment Combination	Relative Effect Size (1)	Observed or Not (2)	Number of Users (3)
(0, 0, 0)	0.000%	Observable	258,249
(0, 0, 1)	1.091%**	Observable	258,340
(0, 1, 0)	-0.267%	Observable	258,367
(1, 0, 0)	0.758%*	Observable	258,321
(1, 1, 1)	2.121%****	Observable	258,375
(1, 1, 0)	0.689%	Unobservable	258,480
(1, 0, 1)	2.299%****	Unobservable	258,305
(0, 1, 1)	1.387%***	Unobservable	258,172

Note: To protect the sensitive data, ATE are proportionally rescaled to relative effect size. The optimal treatment combination (i.e., best-arm) is $\boldsymbol{t}^* = (1,0,1)'$. *p<0.05; **p<0.01; ***p<0.001; ****p<0.0001.

our results with the ground truth. In practice, different engineer and product teams launch the individual experiments independently (most likely in an asynchronous and uncoordinated fashion), and the centralized platform manager runs a back-test to check the treatment effect of the combined experiment (i.e., $\boldsymbol{t} = (1,1,1)'$) at the end. Following this business practice, we assume that the outcomes are observable for the baseline case, the three individual experiments, and the combined experiment, and unobservable otherwise (see Column (2) of Table 2). We denote $\mathcal{T}_o := \{\boldsymbol{t} \in \mathcal{T} : t_1 + t_2 + t_3 = 0, 1, 3\}$ as the set of observable treatment combinations and $\mathcal{T}_u := \{\boldsymbol{t} \in \mathcal{T} : \boldsymbol{t} \notin \mathcal{T}_o\}$ as the set of unobservable treatment combinations. Table 2 shows that the ground-truth ATE of some unobserved treatment combination (e.g., $\boldsymbol{t} = (0,1,0)'$) is insignificant (at $\alpha = 0.05$).

4.2. DeDL Framework on Experimental Data

In this subsection, we present the key steps in applying our DeDL framework to estimate and infer the ATE of each treatment combination (i.e., $\mu(\boldsymbol{t}) = \mathbb{E}[H(\boldsymbol{X}, \boldsymbol{\theta}(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0)]$ defined by Eqn. (3) for

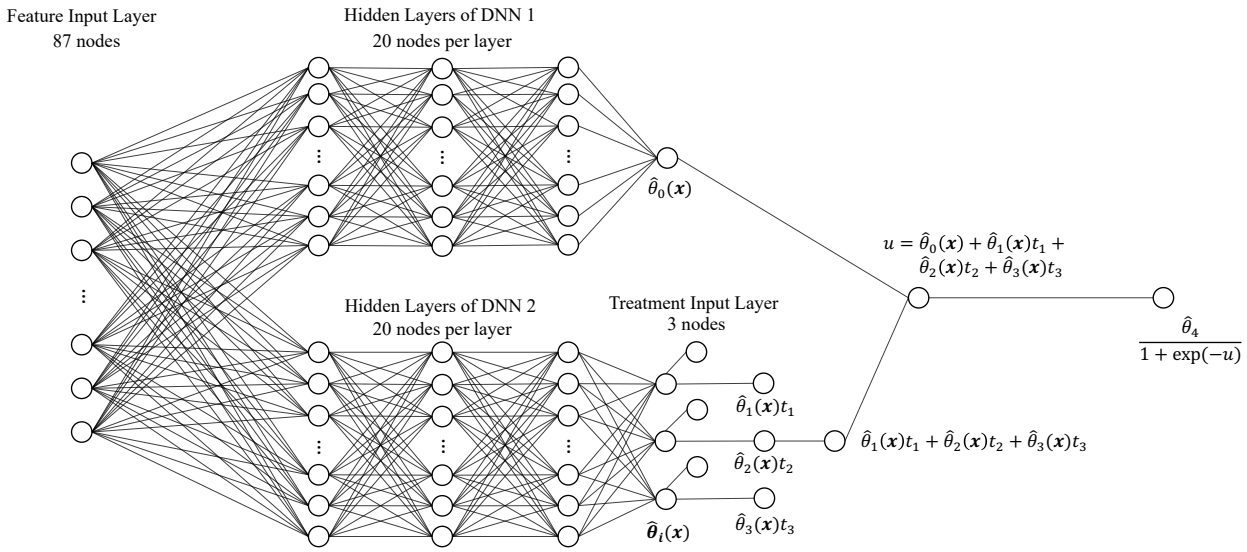


Figure 5 Structure of Deep Neural Network used in Empirical Analysis

all $\boldsymbol{t} \in \mathcal{T}$), whose ground-truth value is documented in Table 2 Column (1). The implementation details are provided in Appendix B.3. First, we consider the following model specification of DGP:

$$\mathbb{E}[Y|\boldsymbol{X}=\boldsymbol{x},\boldsymbol{T}=\boldsymbol{t}] = G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}) = \frac{\theta_4}{1+\exp(-(\theta_0(\boldsymbol{x})+\theta_1(\boldsymbol{x})t_1+\theta_2(\boldsymbol{x})t_2+\theta_3(\boldsymbol{x})t_3))}. \tag{6}$$

Note that the link function G, as a sigmoid function, can capture either "diminishing marginal return" or "increasing marginal return" of the experiments for different users, both of which we have observed in our data sample (see Figure 1). Here, we are using a simplified version of the *Generalized Sigmoid Form II* (Assumption 1(d)) to avoid overfitting. The parameter θ_4 can be thought as the maximum possible video-watching time of any user.

We use two DNNs with three hidden layers per network (20 nodes in each layer) to approximate the parameters $\theta_0(\cdot)$ and $\theta_k(\cdot)$, for $k \in \{1,2,3\}$, respectively. For each layer, ReLU function (i.e., $(x) = \max\{0,x\}$) is used as the activation function. We then concatenate the last layers of two DNNs, take the concatenation layer as the input of a sigmoid function layer, and add another linear layer to obtain $\hat{\theta}_4$, and finally pass all $\theta_k(\cdot)$ into a sigmoid layer to predict the outcome variable y. We illustrate our DNN architecture by Figure 5. Following traditional deep learning literature (some citation), we implement our DNN in Tensorflow, use the Adam optimizer (Kingma and Ba 2014) with stochastic gradient descent to train our DNN on the stratified sample, and employ the mean squared loss, $\ell(y,\boldsymbol{t},\boldsymbol{\theta}(\boldsymbol{x})) = (y - G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}))^2$, as our loss function. The detailed implementation of our training in Python can be found in Appendix B.3.

After obtaining the fitted estimator $\hat{\boldsymbol{\theta}}(\cdot)$, we estimate and infer the ATE of each treatment combination $\boldsymbol{t}$ using the stratified sample with our fitted influence function $\psi(\boldsymbol{z},\hat{\boldsymbol{\theta}},\hat{\boldsymbol{\Lambda}};\boldsymbol{t},\boldsymbol{t}_0)$ defined by Eqn. (5). More specifically, we apply the estimators $\hat{\mu}_{\mathsf{DeDL}}(\cdot)$ and $\widehat{\mathcal{CI}}_{\mathsf{DeDL}}(\cdot;\mu)$ defined by Eqn. (23)

and (25) to the stratified sample under each (observable or unobservable) treatment combination $\boldsymbol{t} \in \mathcal{T}$ to obtain the estimated value and confidence interval of $\mu(\boldsymbol{t})$. We remark that $\hat{\boldsymbol{\Lambda}}(\boldsymbol{x})$ as the estimator of $\boldsymbol{\Lambda}(\boldsymbol{x}) = \mathbb{E}[\ell_{\boldsymbol{\theta\theta}}(Y, \boldsymbol{T}, \boldsymbol{\theta}(\boldsymbol{X}))|\boldsymbol{X} = \boldsymbol{x}]$ can be directly evaluated once $\hat{\boldsymbol{\theta}}(\cdot)$ is obtained, because the distribution of the treatment combination $\boldsymbol{T}$ is known. See Appendix B.3 for details.

Finally, we apply our DeDL framework to identify the treatment combination with the highest ATE, $\boldsymbol{t}^* := \arg\max_{\boldsymbol{t}\in\mathcal{T}} \mu(\boldsymbol{t})$. More specifically, we identify the "best-arm" as $\hat{\boldsymbol{t}}^*_{\mathsf{DeDL}} := \arg\max_{\boldsymbol{t}\in\mathcal{T}} \hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$. Define $\hat{\mu}^*_{\mathsf{DeDL}} := \max_{\boldsymbol{t}\in\mathcal{T}} \hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$ as the ATE of the best-arm estimated by our DeDL framework. We apply the estimators $\hat{\tau}_{\mathsf{DeDL}}(\cdot) := \hat{\mu}^*_{\mathsf{DeDL}} - \hat{\mu}_{\mathsf{DeDL}}(\cdot)$ and $\widehat{\mathcal{CI}}_{\mathsf{DeDL}}(\cdot;\tau)$ defined by (26) and (29) to each treatment combination $\boldsymbol{t} \in \mathcal{T}$ and select the treatment combination(s) t^* with $\hat{\tau}_{\mathsf{DeDL}}(t^*)$ insignificant from 0.

4.3. Benchmarks

To evaluate the performance of our DeDL framework to (i) estimate and infer ATE and (ii) identify the optimal treatment combination, we consider four commonly used approaches as benchmarks: (a) the linear addition (LA) approach; (b) the linear regression (LR) approach; (c) the pure deep learning approach (PDL); and (d) the structured deep learning (SDL) approach. The implementation details of all four benchmarks are presented in Appendix C.

The LA approach assumes that ATE is linearly additive (i.e., $\mu(\boldsymbol{t}_1 + \boldsymbol{t}_2) = \mu(\boldsymbol{t}_1) + \mu(\boldsymbol{t}_2)$ for any $\boldsymbol{t}_1, \boldsymbol{t}_2 \in \mathcal{T}$) and, thus, predicts the ATE of an unobservable treatment combination using those of the observable individual experiments. This approach is intuitive, convenient, and scalable, thus widely adopted by most of the large-scale online platforms in practice such as Platform O. The standard error of an LA estimator for any treatment combination is estimated by assuming that the estimators for individual experiments are independent. For best-arm identification, the LA approach is equivalent to selecting all treatments that have a positive significant ATE.

Under the LR approach, we first predict the unobservable outcomes using a linear regression model trained on the observed sample by regressing the outcome y on $\boldsymbol{t}$ and $\boldsymbol{x}$. The estimation and inference of ATE for each treatment combination $\boldsymbol{t}$ is based on the pair-wise t-test between the outcome *predictions* under $\boldsymbol{t}$ and those under the baseline combination $\boldsymbol{t}_0$. The LR approach identifies the best arm by selecting the treatment combination with the highest ATE based on the *predicted* outcomes of all treatment combinations. We remark that both the LA and LR approaches inherently assume that the treatment effects are linearly additive and homogeneous.

The PDL approach employs a DNN with a similar structure as DeDL that predicts the outcome variable y as a function of both $\boldsymbol{x}$ and $\boldsymbol{t}$. Unlike DeDL that has concrete link function to describe the relationship of $\boldsymbol{t}$ to y conditional on $\boldsymbol{x}$, PDL treats both $\boldsymbol{x}$ and $\boldsymbol{t}$ as network inputs and in turn allows more flexible relationship from $\boldsymbol{t}$ and $\boldsymbol{x}$ to y. The PDL approach uses the same pair-wise

t-test as the LR approach on the *predicted outcomes* for inference of ATE. The identification of the best arm depends on the highest ATE among all treatment combinations. The PDL estimator fully leverages the predictive power of DNN (potentially more powerful than DeDL which assumes a concrete link function) but cannot use the influence function to debias the DL estimation as DeDL does. Therefore, the comparison between PDL and DeDL provides insights into the trade-off between the flexibility of DL models and the ability to construct influence function and debias.

The SDL approach is exactly the same as the DeDL approach with only one distinction: Unlike DeDL that uses the influence function to debias the estimation from DL, the SDL approach simply uses the prediction from the DL ($\hat{\mu}(\boldsymbol{t}) = \frac{1}{n}\sum_{i=1}^{n} H(\boldsymbol{x}_i, \hat{\boldsymbol{\theta}}(\boldsymbol{x}_i); \boldsymbol{t}, \boldsymbol{t}_0)$) as the estimator. Similar to the LR and PDL estimator, the SDL approach estimates and infers the ATE for each treatment combination $\boldsymbol{t}$ by running the pair-wise t-test on the *predicted outcomes*. Likewise, the optimal treatment combination is identified as the one with the highest ATE based on the predicted outcomes of all treatment combinations. The comparison between SDL and PDL reveals the trade-off between economic interpretations and predictive power, whereas that between SDL and DeDL highlights the value of bias correction in our framework.

For any approach $\pi \in \{\mathsf{LA}, \mathsf{LR}, \mathsf{PDL}, \mathsf{SDL}, \mathsf{DeDL}\}$, we use $\hat{\mu}_\pi(\boldsymbol{t})$ (resp., $\widehat{\mathcal{CI}}_\pi(\boldsymbol{t}; \mu)$) to denote the ATE estimator (resp., the confidence interval of $\hat{\mu}_\pi(\boldsymbol{t})$) generated by π for the treatment combination $\boldsymbol{t} \neq \boldsymbol{t}_0$. Likewise, we denote $\hat{\tau}_\pi(\boldsymbol{t})$ (resp., $\widehat{\mathcal{CI}}_\pi(\boldsymbol{t}; \tau)$) to denote the estimator for the ATE difference between the optimal arm $\boldsymbol{t}^*$ and the experiment combination $\boldsymbol{t} \neq \boldsymbol{t}_0$ (resp., the confidence interval for such ATE difference) generated by π.

4.4. Results on ATEs

We first compare the DeDL approach with the four benchmarks presented in Section 4.3 to estimate and infer the ATE of each "unobserved" treatment combination $\boldsymbol{t} \in \mathcal{T}^u$. As discussed above, the assignment mechanism of the three experiments on Platform O enables the observation the ground-truth ATE of *each* treatment combination, based on which we assume that three treatment conditions are not observed by the algorithm and evaluate the performance of all approaches on these unobserved conditions. In particular, we document the following performance metrics:

- Correct Direction Ratio (CDRu). For any estimation and inference approach π, we denote $\mathcal{T}^{cd}(\pi)$ as the set of all treatment combinations with *correct direction identification*, i.e., the treatment combinations $\boldsymbol{t} \in \mathcal{T}$ whose ground-truth ATE $\mu(\boldsymbol{t})$ have been correctly identified by π in terms of both (i) sign and (ii) statistical significance. Define $\mathcal{T}_u^{cd}(\pi) := \mathcal{T}^{cd}(\pi) \cap \mathcal{T}_u$ as the set of unobservable treatment combinations with correct direction identification for π. The CDR for unobservable treatment combinations (CDRu) of π as $\mathcal{CDR}_u(\pi) = \frac{|\mathcal{T}_u^{cd}(\pi)|}{|\mathcal{T}_u|} \times 100\%$.

Table 3 Comparison of Different Estimators of ATE

Estimator	Unobserved Treatment Combinations				Estimator	All Treatment Combinations			
	CDRu (1)	MAPEu (2)	MSEu (3)	MAEu (4)		CDR (5)	MAPE (6)	MSE (7)	MAE (8)
LA	2/3	30.06%	18.597	4.032	LA	7/8	12.02%	7.966	1.728
LR	2/3	4.90%	5.303	1.855	LR	7/8	17.37%	6.551	2.348
PDL	2/3	6.86%	4.876	1.838	PDL	6/8	14.76%	4.962	2.031
SDL	2/3	14.71%	5.623	2.271	SDL	6/8	14.03%	3.840	1.804
DeDL	3/3	1.75%	4.095	1.343	DeDL	8/8	3.07%	1.845	0.737

Note: The calculation of MSE, MSEu, MAE, and MAEu are based on the scaled outcome variable (see Table 2, Column (1)). MSE and MSEu are scaled by multiplying a constant. MAE and MAEu are scaled by multiplying another constant.

- Mean Absolute Percentage Error (MAPEu). The MAPE of any estimation and inference approach π is the average percentage error for unobserved treatment combinations with a significant ground-truth ATE. In other words, the MAPE of π for unobservable treatment combinations (MAPEu) is defined as $\mathcal{MAPE}_u(\pi) := \frac{1}{|\mathcal{T}_u^s|} \sum_{\boldsymbol{t} \in \mathcal{T}_u^s} \frac{|\mu(\boldsymbol{t}) - \hat{\mu}_\pi(\boldsymbol{t})|}{|\mu(\boldsymbol{t})|} \times 100\%$.
- Mean Squared Error (MSEu). The MSE of π for unobservable treatment combinations (MSEu) is defined as $\frac{1}{|\mathcal{T}_u|} \sum_{\boldsymbol{t} \in \mathcal{T}_u} (\mu(\boldsymbol{t}) - \hat{\mu}_\pi(\boldsymbol{t}))^2$.
- Mean Absolute Error (MAEu). The MAE of π for unobservable treatment combinations (MAEu) is defined analogously as $\frac{1}{|\mathcal{T}_u|} \sum_{\boldsymbol{t} \in \mathcal{T}_u} |\mu(\boldsymbol{t}) - \hat{\mu}_\pi(\boldsymbol{t})|$.

To give a clear picture of the comparisons among *all* 8 (observable and unobservable) treatment combinations and identify the best arm of them (see Section 4.5), we present the estimated treatment effects of all arms in Figure 6. We also calculate the above 4 metrics evaluated on both *unobserved* treatment combinations and *all* treatment combinations in Table 3.

Table 3 documents the comparison of our DeDL approach against the LA, LR, PDL and SDL benchmarks with respect to the 4 performance metrics described above. The punchline message of the empirical analysis is that our DeDL estimator substantially outperforms all 4 benchmarks with any of these performance metrics, providing evidence that the proposed framework to accurately estimate and infer the ATE of any treatment combination for multiple A/B tests on a large-scale online platform. More notably, we highlight that, to our best knowledge, this is the first rigorous empirical validation of the DML methodology in a practical setting with data from large-scale field experiments. As emphasized above, the unique concurrent orthogonal-design deployment of the three experiments on Platform O has made such validation possible by revealing the ground-truth ATE of each possible treatment combination. A key advantage of our DeDL approach to adopt DL in causal inference is its ability to capture the user-heterogeneity and non-linearity for the combined treatment effect of multiple A/B tests on a large-scale online platform, the observation of which has motivated this study (see Figure 1). As expected, such an advantage stems from the strong predictive power of deep neural networks.

As aforementioned, the comparison between our DeDL framework and the SDL estimator provides more insights into how bias correction (i.e., influence function) affects the final causal inference performance. To provide more insights, in Figure 6, we visualize these two approaches' ATE estimates and the 95%-confidence interval for each treatment combination (i.e., $\hat{\mu}_\pi(\boldsymbol{t})$ and $\widehat{\mathcal{CI}}_\pi(\boldsymbol{t};\mu)$ for $\pi \in \{\mathsf{SDL}, \mathsf{DeDL}\}$ and $\boldsymbol{t} \in \mathcal{T}$). Figure 6 shows that the bias correction can help causal inference in two significant ways: First, by correcting the bias due to the variabilities of the training data from the plug-in estimator, the DeDL approach is able to accurately identify the confidence interval of the ATEs while the SDL approach is always underestimating the standard error of ATE. This means that, without bias correction, the analysis leads to potentially more Type-I errors and higher false discovery rates for the platform. Second, DeDL provides a more accurate ATE estimate than SDL does, empirically confirming the advantage of the former for more accurate causal inference.

Moreover, the comparison between SDL and PDL in Table 3 shows that the pure DNN without assuming the link function can improve the performance of predicting the treatment effects (i.e., MAPEu reduced from 14.71% to 6.86%). This shows that, by using a concrete link function, we indeed sacrifice some prediction accuracy for the ability of deriving bias correction terms and making economic interpretation. However, since the performance of our DeDL estimator significantly outperform that of the PDL approach in all metrics, it shows that this sacrifice is justified in the context of causal inference. In other words, the benefit of assuming a concrete link function and use the corresponding bias correction will overweight the cost of not being able to describe a more flexible relationship between the treatment conditions and the outcome.

Last, we also compare the estimation MAPE of the DeDL estimator with SDL and LR estimators under increasing DNN training epochs in Figure 7. As is clear by Figure 7, both DeDL and SDL estimators yield smaller MAPE as DNN training loss is smaller (i.e., mean squared loss). We highlight that when DNN is poorly trained (i.e., within 100 epochs), the DeDL and SDL estimators have similar estimation MAPE, which are dominated by the LR estimator. In this case, we do not benefit from the debias term of DML. However, as the DNN converges, DeDL starts to show significant advantages in estimation accuracy compared to both SDL and LR estimators. This phenomenon is well-connected to the classical semi-parametric statistics literature, which requires the convergence rate of the estimated parameter functions $\boldsymbol{\theta}(\cdot)$ sufficiently fast to obtain asymptotically unbiased estimators of the treatment effects (e.g., Chernozhukov et al. 2018). Finally, Figure 7 is also translated into an important actionable insight when adopting our DeDL framework that the DNN training error can be a useful indicator for the quality of the second-stage estimation and the effectiveness of the debiasing via DML.

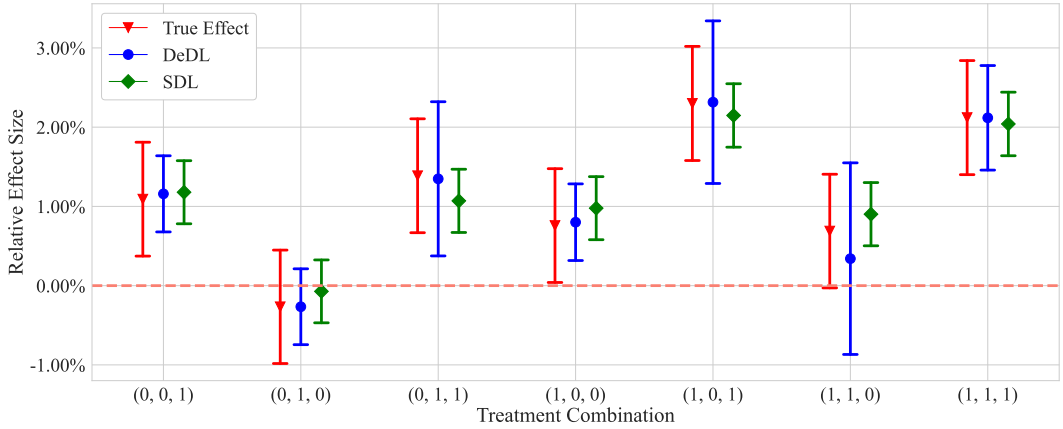


Figure 6 Detailed Comparisons between the SDL and DeDL Estimators. The bars represent confidence intervals.

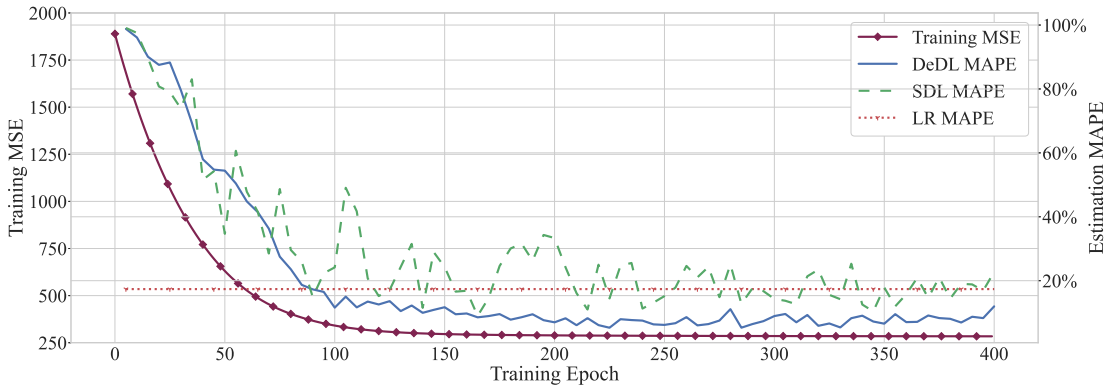


Figure 7 MAPE Comparison with DNN Training Epoch

4.5. Results on Best-Arm Identification

This subsection is devoted to applying our DeDL framework to identifying the optimal treatment combination with the highest ATE. By the ground-truth ATE estimates (Table 2 Column (1)), the "true" optimal treatment combination is $\boldsymbol{t}^* = (1, 0, 1)'$. We compare different estimators for best-arm identification in Table 4 and report the same set of performance metrics defined in Section 4.4. In particular, we focus on the comparison between $\tau(\boldsymbol{t})$ and $\hat{\tau}(\boldsymbol{t})$. Table 4 shows that that DeDL estimators significantly outperform the LA, LR, PDL, and SDL benchmarks with respect to all performance metrics in identifying the best treatment combination.

In summary, both Table 3 and 4 provide clear empirical evidence that our proposed DeDL estimator outperforms all other approaches in estimating the ATEs of unobserved conditions and identify the best treatment conditions. We highlight that this is among the first works to empirically investigate the accuracy of the DML framework to recover the ground-truth treatment effects

Table 4 Comparison of Different Estimators of Best-Arm Identification

Estimator	CDR (1)	MAPE (2)	MSE (3)	MAE (4)
LA	7/8	21.92%	15.539	3.091
LR	7/8	11.86%	3.232	1.727
PDL	7/8	12.83%	4.053	1.839
SDL	8/8	17.45%	7.186	2.442
DeDL	8/8	5.97%	1.995	0.780

Note: The calculation of MSE and MAE are based on the scaled outcome variable (see Table 2, Column (4)). MSE is scaled by multiplying a constant. MAE is scaled by multiplying another constant.

through large-scale field experiments. Our empirical evidence also sheds light on how crucial bias correction is in estimating the causal effects and provides insights into the trade-off between more flexible models and the ability to derive bias correction terms. Last but not least, we demonstrate that specifying and training a good ML model can be essential for second-stage inference. Indeed, we have carefully tailored the model structure, through the model layer of our DeDL framework, to the multiple-experiment problem, which leads to much-improved performances over an unstructured DL model. Consistent with the notorious difficulties of training DL models in the literature and practice, fine-tuning our model involves significant efforts.

5. Synthetic Experiments

In this section, we perform several synthetic experiments to gauge the robustness of our approach under different scenarios. We first validate our theory by varying the number of experiments $m \in \{4, 6, 8, 10\}$ in Section 5.1. Then, we investigate several factors that might impact the performance of DeDL estimators in practice. In Section 5.2, we test the performance of our DeDL ATE estimators with a potentially large bias of estimating $\hat{\boldsymbol{\theta}}(\cdot)$ and find that DeDL is fairly robust with moderate biases. We also systematically assess the performance of our method under model misspecification and shed light on how to test and select the link function in practice in Section 5.3. Furthermore, due to the page constraint, in Appendix D.4), we investigate a practical setting where the observed $\boldsymbol{X}$ distribution deviates from the population and discuss how to use the rebalancing method to get trustworthy estimates.

Simulation Setup. Throughout Sections 5.1 and 5.2, we assume that the link function G is correctly specified. Consistent with our empirical study in Section 4, we use the *Generalized Sigmoid Form II*, i.e., for each data point i, we have

$$y_i = \frac{\theta^*_{m+1}}{1 + \exp(\theta^*_0(\boldsymbol{x}_i) + \theta^*_1(\boldsymbol{x}_i)t_{i1} + \theta^*_2(\boldsymbol{x}_i)t_{i2} + \cdots + \theta^*_m(\boldsymbol{x}_i)t_{im})} + \epsilon_i, \tag{7}$$

where ϵ_i is the *i.i.d.* random noise with zero mean.

We assume that there are m concurrent field experiments, each with a binary treatment. We sample θ^*_{m+1} from the uniform distribution $\mathcal{U}(10, 20)$ throughout, while generating $\theta_j(\boldsymbol{x})$, $j = 0, 1, \ldots, m$,

differently in different subsequent sections. We generate data points $\boldsymbol{z}_i = (y_i, \boldsymbol{x}_i', \boldsymbol{t}_i')'$ as *i.i.d.* copies of the random vector $\boldsymbol{Z} = (Y, \boldsymbol{X}', \boldsymbol{T}')'$. The random perturbation ϵ follows a uniform distribution $\mathcal{U}(-0.05, 0.05)$. Without loss of generality, we generate covariates $\boldsymbol{x}$ as follows: (1) the dimension of covariates $\boldsymbol{X}$ satisfies $d_{\boldsymbol{X}} = 10$; (2) the different components of $\boldsymbol{X}$ are *i.i.d.* following the uniform distribution $\mathcal{U}(0,1)$. We remark that larger random perturbations, higher dimensional covariates and/or more complicated joint distributions of $\boldsymbol{X}$ can be easily incorporated into the simulation. We adopt the current setting for ease of model training and efficiency of experiments.

To ensure that the identifiability and sufficient curvature conditions (i.e., Assumption 4) are met, we adopt the following treatment assignment mechanism $\nu(\cdot\,|\,\cdot)$. We assume the independence between $\boldsymbol{X}$ and $\boldsymbol{T}$, whose distribution we denote as $\nu(\boldsymbol{t}) = \mathbb{P}[\boldsymbol{T} = \boldsymbol{t}]$. Furthermore, in the training stage, with equal probability, we randomly assign each experimental unit to one of $m+2$ different treatment combinations with equal probability, i.e., $\nu(\boldsymbol{t}) = 1/(m+2)$ if $\boldsymbol{t} \in \{\boldsymbol{t} \in \{0,1\}^m : \sum_{i=1}^m t_i = 0 \text{ or } 1\} \cup \{(1,1,0,\ldots,0)'\}$ and $\nu(\boldsymbol{t}) = 0$ for other treatment combinations. In other words, we assume the *partially observed outcome* setting with only $m+2$ observable treatment combinations while other treatment outcomes are masked for gauging the performance of estimators.

Our neural network structure to estimate $\theta_j^*(\boldsymbol{x}),\ j = 0, 1, \ldots, m$ is pre-specified as two-layer perceptrons with ReLU activation functions and 10 nodes in hidden layers in all experiments throughout this section. Because there are $m+2$ unknown parameters, which linearly scale with the number of experiments m, we generate $500m$ *i.i.d.* experimental data points $\boldsymbol{z}_i$ for the DNN training, and another independent $500m$ experimental data points for the inference stage. During the DNN training stage, we randomly split this data set with 70% for training and the rest 30% for cross-validation. We adopt the MSE loss function and Adam algorithm (Kingma and Ba 2014). In most experiments, we stop training when the loss on cross-validation data is less than a fixed threshold of 0.3. We empirically tested various thresholds and found that the gain from a smaller stopping threshold is marginal and, thus, pick this threshold based on the computational efficiency consideration. In the experiments with misspecified link functions and imbalanced data that will be discussed later, the cross-validation loss tends to increase. Hence, we adjust the threshold accordingly in such cases. We also experiment with popular training strategies such as dropout or regularizing the weights, but the gain is marginal so we do not include them in this discussion.

At the inference stage, we independently generate a data sample with the same size as the training data. To avoid the rare case where the empirical estimate $\hat{\boldsymbol{\Lambda}}(\boldsymbol{x})$ is not invertible (e.g., $\boldsymbol{\theta}_i(\boldsymbol{x}) = \boldsymbol{\theta}_j(\boldsymbol{x})$ for all $i, j \in \{1, 2, \ldots, m\}$), we add a small regularization to $\hat{\boldsymbol{\Lambda}}(\boldsymbol{x})$ so $(\hat{\boldsymbol{\Lambda}}(\boldsymbol{x}) + 0.0005\boldsymbol{I}_{d_{m+2}})^{-1}$ is well-defined. Similar regularization or trimming techniques based on the propensity score are quite common in practice for numerical stability.

To calculate the true ATE over the population, we use the sample average approximation with 10,000 independent samples. We also use the standard t-test with a significance level 0.05 to determine if the ATE of an experiment combination is statistically significant. To derive statistical metrics such as confidence intervals, all synthetic experiments are replicated 200 times.

5.1. Validation of the DeDL Estimator

In this subsection, we aim at empirically validating the theoretical results in Section 3 and further demonstrating the superior performance of our DeDL estimator in practice. Such experiments are necessary due to the gaps between theory and practice. In particular, there are two potential inconsistencies between the underlying theory (e.g., see Section 3) and practical settings. (a) The key theoretical result Proposition 1 is proved under the assumption that one obtains the estimator $\hat{\boldsymbol{\theta}}(\boldsymbol{x})$ by (almost) minimizing the empirical loss. However, the loss of DNN is difficult to optimize globally, especially given our novel structured architecture with a model layer. (b) The theoretical DNN width $O(n^{d_{\boldsymbol{X}}/2(p+d_{\boldsymbol{X}})}\log^2 n)$ and depth $O(\log n)$ required in Proposition 1 are clearly too large for practical applications. Practitioners often fine-tune these hyperparameters at a much smaller scale. Given these practical issues, we find it necessary to conduct numerical experiments in addition to the field experiment to demonstrate the performance of DeDL in general settings.

To generate the functions $\boldsymbol{\theta}^*(\boldsymbol{x})$ in this subsection, we first define the coefficient matrix $\boldsymbol{A} \in \mathbb{R}^{(m+1)\times d_{\boldsymbol{X}}}$ with each component independently drawn from the uniform distribution $\mathcal{U}(-0.5, 0.5)$ and write the j the row of $\boldsymbol{A}$ as row vector $\boldsymbol{A}_{[j]}$. Then, we let $\theta_j^*(\boldsymbol{x}) = (\boldsymbol{A}_{[j+1]}\boldsymbol{x})^3,\ j = 0, 2, \ldots, m$. As mentioned, the parameter $\boldsymbol{\theta}_{m+1}^*$ is randomly generated from the uniform distribution $\mathcal{U}(10, 20)$. To facilitate numerical experiments, we choose a relatively simple structure for the neural network as discussed previously. However, more complex function $\boldsymbol{\theta}^*(\boldsymbol{x})$ can be readily incorporated and tested in our synthetic experiments, and DNN training related hyperparameters, e.g., width, depth, training algorithm, etc., should be fine tuned to accommodate in such cases.

In Figure 8(a), we evaluate the performance of different estimators including LA, LR, PDL, SDL and DeDL with varying numbers of A/B tests, i.e., $m \in \{4, 6, 8, 10\}$. We train these estimators with partially observed outcomes as discussed above. In Appendix D.1, we report complete simulation results in Table 8 and Table 9, along with a more detailed discussion. The performance of MAE is highly correlated with other performance metrics, so to keep the discussion simple, we mainly report and visualize the performance comparison under MAE along with the 95% confidence interval, shown in Figure 8. The confidence interval is computed using the 200 instances for each parameter combination. One can observe from panel (a) that DeDL has the best performance under all $m \in \{4, 6, 8, 10\}$. Increased values of m lead to quick degradation of the performance of LA and LR. Such simple models relying on linear extrapolation are unable to capture the rich treatment effects, but the performance of SDL and DeDL are relatively stable.

Figure 8(b) displays the performance of different variants of PDL estimators under different m values. For a better visualization, we report the performance of PDL estimators with a different scale on the y-axis than that in (a). Among these different PDL estimators, we use subscripts s (small) and l (large) to represent different widths of neural nets, with 10 and 40 hidden nodes for all three hidden layers, respectively. All DNNs have three linear layers followed by ReLU activation layers. The subscripts p and a represent the training samples generated from *partially observed treatments* and *all* treatments, respectively. For a fair comparison to SDL and DeDL estimators, we focus on PDL_s_p with partially observed treatments and a similar DNN size to SDL. We observe that, with partially observed combinations, both PDL_s_p and PDL_l_p perform poorly, mainly driven by the bad performance under the unobserved treatments. Increased network size does not help much. However, when we incorporate data from all treatment combinations for training, estimators PDL_s_a and PDL_l_a obtain comparable performance with the best estimator DeDL. Therefore, a structured DNN model allows us to capture the interaction of experiments in practical settings, while the model flexibility in PDL only helps in unrealistic scenarios where a large proportion of experimental combinations are observed. We refer interested readers to Appendix D.1 for more detailed comparisons, discussions, and additional simulations of incorporating regularizers into PDL.

Because all performance metrics show similar patterns under different m values, in all experiments in subsequent sections, we maintain $m = 4$ for computation efficiency. Also, due to the inferior performance of PDL with partially observed treatments, we do not report its performance in the following sections.

5.2. Robustness to DNN Convergence Rate

As pointed out by the semi-parametric estimation literature (e.g., Chernozhukov et al. 2018), in practice, it is sometimes questionable whether DNN estimators $\hat{\boldsymbol{\theta}}(\cdot)$ can achieve the $o(n^{-1/4})$ convergence rate required for the inference stage. To systematically illustrate the performance of the debiasing technique in learning the treatment effects in combinatorial experiments, we artificially control the biases in DNN estimators to evaluate the impact of such biases in the second-stage inference. Specifically, we use an approach with a similar spirit to Chernozhukov et al. (2018) by constructing $\hat{\boldsymbol{\theta}}(\boldsymbol{x})$ with manually controlled biases. First, parameter functions are defined as $\theta^*_{j-1}(\boldsymbol{x}) = \boldsymbol{A}_{[j+1]}\boldsymbol{x}$, for $j = 0, 1, \ldots, m$, where each component in the coefficient matrix $\boldsymbol{A} \in \mathbb{R}^{(m+1)\times d_{\boldsymbol{X}}}$ is generated under independent and uniform distribution $\mathcal{U}(-0.5, 0.5)$. The parameter $\boldsymbol{\theta}^*_{m+1}$ is randomly generated from the uniform distribution $\mathcal{U}(10, 20)$. Next, instead of training a DNN for estimation, we manually set the biased estimator $\hat{\theta}_j(\boldsymbol{x}) = (1 + err_j)\theta^*_j(\boldsymbol{x}),\ j = 0, 1, \ldots, m+1$, where all $err_j,\ j = 0, 1, \ldots, m+1$ terms independently follow the uniform distribution $\mathcal{U}(-\delta, \delta)$. We set different levels of the bias range coefficient $\delta \in \{0.1, 0.2, 0.3\}$ to investigate the effectiveness of DeDL with

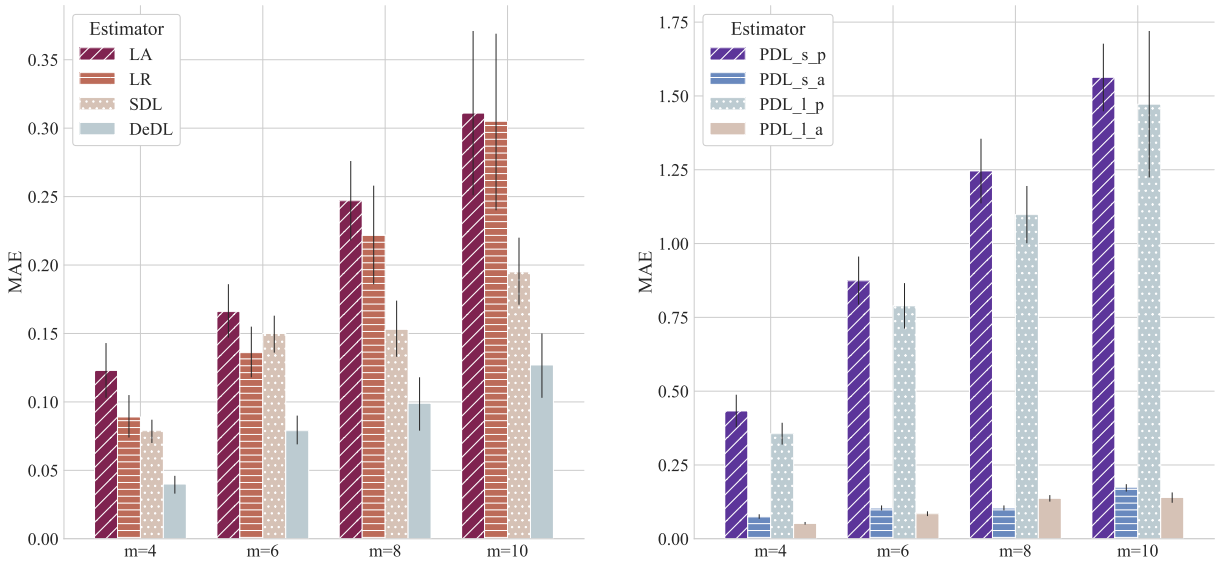


Figure 8 MAE comparison among estimators under the increasing number of experiments m values. Panel (a) shows the performance of LA,LR,SDL, and DeDL. Panel (b) presents the performance of PDLs with different network size specifications and partially/fully observed data.

different levels of biases the estimators $\hat{\boldsymbol{\theta}}$. In the following discussion, we focus on the MAE performance metric, and we refer interested readers to Appendix D.2 for a more detailed comparison and discussion.

As documented in Figure 9, DeDL has much smaller MAEs than SDL, implying a significant performance improvement with adding de-bias term. In all settings, DeDL performs better than LA and LR despite that SDL may generate MAEs much larger than LA and LR, in particular when $\delta = 0.3$. This shows that the DeDL estimator is fairly robust with moderate biases in $\hat{\boldsymbol{\theta}}$. However, we also point out that when the bias from training is large, debiasing may not improve the performance, as we illustrated in Figure 7 in Section 4. In our simulation, we have a similar observation that, when δ increases to 1.0, DeDL may perform worse than SDL under the MAPE metric. In such cases, it is critical to improve DeDL by training a better DNN model, for example, through using a larger network.

5.3. Link Function Misspecification

In this subsection, we first investigate how a misspecified link function impair the effectiveness of our debiased estimator. Indeed, a key assumption of our framework is that the link function G is correctly specified. In practice, however, it can be challenging to select the best link function. On the positive side, as we will discuss, one may test the efficacy of the link function by checking the cross-validation errors in the DNN training stage.

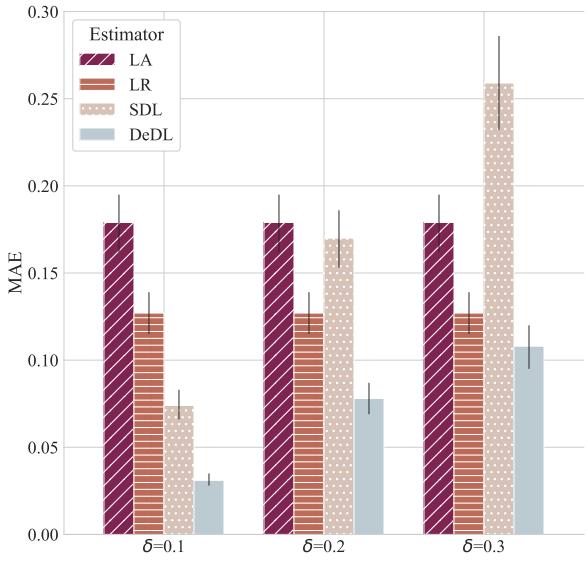


Figure 9 **MAE Comparison Under Decreasing DNN Convergence Rate**

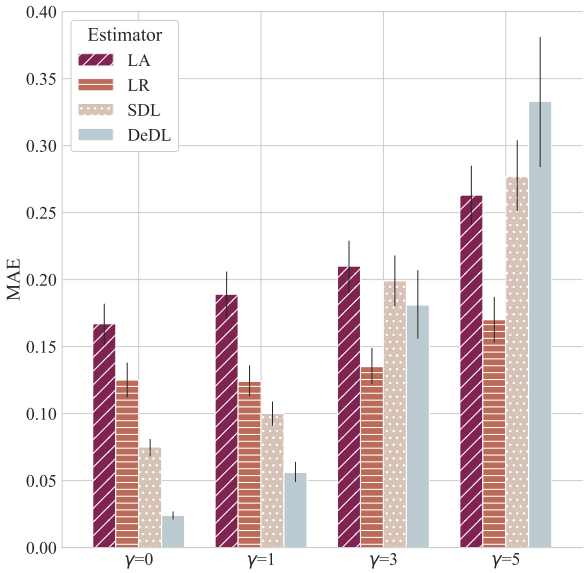


Figure 10 **MAE Comparison Under Link Function Misspecification**

For this subsection, we adjust our true DGP as,

$$y_i = \frac{\theta^*_{m+1}}{1+\exp(\theta^*_0(\boldsymbol{x}_i)+\theta^*_1(\boldsymbol{x}_i)t_{i1}+\cdots+\theta^*_m(\boldsymbol{x}_i)t_{im})} + \gamma\Big(\beta^*_0(\boldsymbol{x}_i)+\beta^*_1(\boldsymbol{x}_i)t_{i1}+\cdots+\beta^*_m(\boldsymbol{x}_i)t_{im}\Big)+\epsilon_i, \tag{8}$$

where parameter functions are defined as $\theta^*_{j-1}(\boldsymbol{x}) = \boldsymbol{A}_{[j+1]}\boldsymbol{x}$, and $\beta^*_j(\boldsymbol{x}) = \boldsymbol{B}_{[j]}\boldsymbol{x}$, for $j = 0, 1, \ldots, m$. The parameter $\gamma \geq 0$ captures to what extent the true link function deviates from the Generalized Sigmoid Form II, which we still adopt in our DeDL framework for estimating the treatment effect. The larger the γ, the more misspecified our model is. We generate both the coefficient matrices $\boldsymbol{A} \in \mathbb{R}^{(m+1)\times d_{\boldsymbol{X}}}$ and $\boldsymbol{B} \in \mathbb{R}^{(m+1)\times d_{\boldsymbol{X}}}$ with independent entries and each component follows the uniform distribution $\mathcal{U}(-0.5, 0.5)$. The parameter $\boldsymbol{\theta}^*_{m+1}$ is again randomly generated from the uniform distribution $\mathcal{U}(10, 20)$.

We run synthetic experiments under different levels of model misspecification $\gamma \in \{0, 1, 3, 5\}$. To illustrate how the link function is skewed, we plot in Figure 11 the histograms of outcome y in Eqn. (8). The x-axis represents the experimental outcome y among the whole population in all 2^m treatment combinations with equal probability. Observe that when γ is large, e.g., $\gamma = 5$, the bias from using the sigmoid link function to approximate the experimental outcome is likely to be significant. For example, positive θ_{m+1} cannot capture the negative y.

Here we plot the comparison between different methods under model misspecification in Figure 10, while the complete simulation results are reported in Appendix D.3. This comparison reals that when the link function is more specified (i.e., larger γ), SDL and DeDL estimators perform

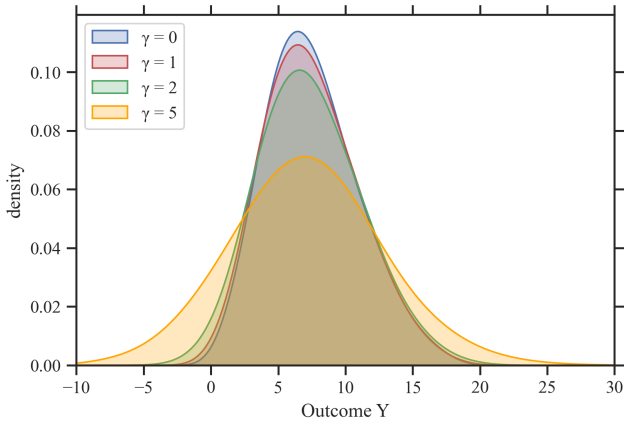


Figure 11 Experimental Outcome of Misspecified Model

substantially worse, whereas the performances of the LA and LR estimators are relatively stable. Indeed, the LA and LR estimators do not face any significant increases in MAPE as γ gets larger. Also, when γ is increased to 5, DeDL performs worse than SDL, implying that debiasing via Neyman orthogonality hurts the performance when the link function is not correctly specified.

In practice, however, it is difficult, if not impossible, to verify the true link function. Fortunately, one may detect link function misspecification through large in-sample validation errors. In other words, we can resort to checking the cross-validation errors in the DNN training stage. To shed light on this point, we report the in-sample validation errors from our experiments. Specifically, we compare the errors induced by pure DNN without a sigmoid link function (a three-layer perceptron in our case) and structured DNN (a two-layer perceptron followed by a link function layer), respectively. Note that pure DNN takes treatment level $\tilde{\boldsymbol{t}}$ together with covariates $\boldsymbol{x}$ as inputs to the first linear layer. In contrast, the structured DNN only takes covariates $\boldsymbol{x}$ to the first linear layer and uses treatment level $\tilde{\boldsymbol{t}}$ in the final link function layer. For a fair comparison, we set the pure DNN structure with a similar width and depth, i.e., $(d_{\boldsymbol{x}}+d_{\boldsymbol{t}}+1)$–10–ReLU–10–ReLU–10–ReLU–1. Equipped with higher flexibility, pure DNN can better approximate individual responses in the partially observed treatment combinations, if indeed the link function is severely misspecified. Hence, we can use the pure DNN as a benchmark for the in-sample comparison to check whether the link function is reasonable. Adopting the same Adam training algorithm and training samples, we obtain the following 95% $\mathcal{CI}$s of cross-validation mean squared errors under different misspecification levels: 1) $[0.044, 0.053]$, $[0.086, 0.112]$, $[0.116, 0.159]$, $[0.142, 0.197]$ for pure DNN under $\gamma = 0, 1, 3, 5$ respectively; 2) $[0.012, 0.014]$, $[0.019, 0.026]$, $[0.076, 0.106]$, $[0.215, 0.279]$ for structured DNN under $\gamma = 0, 1, 3, 5$ respectively. One can observe that when $\gamma \in \{0, 1, 5\}$, the structured DNN has a smaller or comparable in-sample loss than the pure DNN, which indicates the reasonable performance of the generalized sigmoid link function to approximate the outcome variable. However, when the misspecification level $\gamma = 5$, the in-sample error from the generalized sigmoid form grows larger

that of the pure DNN. Correspondingly, in Figure 11, we observe severe performance degradation for both SDL and DeDL and a significant negative impact from the debiase term. In this case, we suggest experimenting with a different link function in the first DNN training stage and/or not using the debiased estimator. More generally, as long as the in-sample loss of the structured DNN is on par with that of a pure DNN with similar depth and width, we recommend adopting this structured DNN with debiasing.

6. Conclusion

In this paper, we leverage the DeDL framework to infer treatment effects for concurrent experiments and identify the best treatment combination. We show the superior performance of our method using data from three A/B tests on a large-scale platform. We also demonstrate the robustness of the DeDL method through synthetic experiments. Our framework can also be applied to analyze the individual heterogeneity of treatment effects with observational data under unconfoundedness.

We close our paper by discussing several future directions. First, for multi-level discrete and continuous treatments, though our link functions can still be applied, researchers can design more flexible link functions that better fit the richer treatment assignment mechanisms for the identification and convergence of parameter functions. Second, combining the current framework with classical causal inference methods such as instrumental variables and difference-in-differences could lead to some interesting future work.

References

Angrist JD, Pischke JS (2009) *Mostly harmless econometrics: An empiricist's companion* (Princeton university press).

Arkhangelsky D, Athey S, Hirshberg DA, Imbens GW, Wager S (2021) Synthetic difference-in-differences. *American Economic Review* 111(12):4088–4118.

Athey S, Imbens G (2016) Recursive partitioning for heterogeneous causal effects. *Proceedings of the National Academy of Sciences* 113(27):7353–7360.

Athey S, Imbens GW, Wager S (2018) Approximate residual balancing: debiased inference of average treatment effects in high dimensions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 80(4):597–623.

Box GE, Hunter WH, Hunter S, et al. (1978) *Statistics for experimenters*, volume 664 (John Wiley and sons New York).

Burtch G, Ghose A, Wattal S (2015) The hidden cost of accommodating crowdfunder privacy preferences: A randomized field experiment. *Management Science* 61(5):949–962.

Candogan O, Chen C, Niazadeh R (2021) Correlated cluster-based randomized experiments: Robust variance minimization. *Chicago Booth Research Paper* (21-17).

Chernozhukov V, Chetverikov D, Demirer M, Duflo E, Hansen C, Newey W, Robins J (2018) Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal* 21(1):C1–C68.

Chernozhukov V, Newey WK, Singh R (2022) Automatic debiased machine learning of causal and structural effects. *Econometrica* 90(3):967–1027.

Cheung WC, Simchi-Levi D, Wang H (2017) Dynamic pricing and demand learning with limited price experimentation. *Operations Research* 65(6):1722–1731.

Chiang HD, Kato K, Ma Y, Sasaki Y (2022) Multiway cluster robust double/debiased machine learning. *Journal of Business & Economic Statistics* 40(3):1046–1056.

Dasgupta T, Pillai NS, Rubin DB (2015) Causal inference from 2k factorial designs by using potential outcomes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 77(4):727–753.

Dube A, Jacobs J, Naidu S, Suri S (2020) Monopsony in online labor markets. *American Economic Review: Insights* 2(1):33–46.

Edelman B, Luca M, Svirsky D (2017) Racial discrimination in the sharing economy: Evidence from a field experiment. *American economic journal: applied economics* 9(2):1–22.

Fan Q, Hsu YC, Lieli RP, Zhang Y (2022) Estimation of conditional average treatment effects with high-dimensional data. *Journal of Business & Economic Statistics* 40(1):313–327.

Farbmacher H, Huber M, Lafférs L, Langen H, Spindler M (2022) Causal mediation analysis with double machine learning. *The Econometrics Journal* 25(2):277–300.

Farrell MH, Liang T, Misra S (2020) Deep learning for individual heterogeneity: an automatic inference framework. *arXiv preprint arXiv:2010.14694* .

Farrell MH, Liang T, Misra S (2021) Deep neural networks for estimation and inference. *Econometrica* 89(1):181–213.

Gabillon V, Ghavamzadeh M, Lazaric A (2012) Best arm identification: A unified approach to fixed budget and fixed confidence. *Advances in Neural Information Processing Systems* 25.

Goli A, Lambrecht A, Yoganarasimhan H (2022) A bias correction approach for interference in ranking experiments. *Available at SSRN 4021266* .

Gordon BR, Moakler R, Zettelmeyer F (2022) Close enough? a large-scale exploration of non-experimental approaches to advertising measurement. *arXiv preprint arXiv:2201.07055* .

Guo Y, Coey D, Konutgan M, Li W, Schoener C, Goldman M (2021) Machine learning for variance reduction in online experiments. *Advances in Neural Information Processing Systems* 34:8637–8648.

Imbens GW, Rubin DB (2015) *Causal inference in statistics, social, and biomedical sciences* (Cambridge University Press).

Johari R, Li H, Liskovich I, Weintraub GY (2022) Experimental design in two-sided platforms: An analysis of bias. *Management Science* 68(10):7069–7089.

Ju N, Hu D, Henderson A, Hong L (2019) A sequential test for selecting the better variant: Online a/b testing, adaptive allocation, and continuous monitoring. *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 492–500.

Kallus N, Mao X, Udell M (2018) Causal inference with noisy and missing covariates via matrix factorization. *Advances in neural information processing systems* 31.

Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Knaus MC (2020) Double machine learning based program evaluation under unconfoundedness. *arXiv preprint arXiv:2003.03191* .

Kohavi R, Tang D, Xu Y (2020) *Trustworthy online controlled experiments: A practical guide to a/b testing* (Cambridge, UK: Cambridge University Press).

Kohavi R, Thomke S (2017) The surprising power of online experiments. *Harvard business review* 95(5):74–82.

Lattimore F, Lattimore T, Reid MD (2016) Causal bandits: Learning good interventions via causal inference. *Advances in Neural Information Processing Systems* 29.

Lee MR, Shen M (2018) Winner's curse: Bias estimation for total effects of features in online controlled experiments. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 491–499.

Li X, Ding P, Rubin DB (2020) Rerandomization in 2^k factorial experiments. *The Annals of Statistics* 48(1):43–63.

Morgan KL, Rubin DB (2012) Rerandomization to improve covariate balance in experiments. *The Annals of Statistics* 40(2):1263–1282.

Nandy P, Venugopalan D, Lo C, Chatterjee S (2021) A/b testing for recommender systems in a two-sided marketplace. *Advances in Neural Information Processing Systems* 34:6466–6477.

Newey WK (1994) The asymptotic variance of semiparametric estimators. *Econometrica: Journal of the Econometric Society* 1349–1382.

Pashley NE, Bind MAC (2019) Causal inference for multiple treatments using fractional factorial designs. *arXiv preprint arXiv:1905.07596* .

Qi Z, Tang J, Fang E, Shi C (2022) Offline personalized pricing with censored demand. *Available at SSRN* .

Song Y, Sun T (2021) Ensembling experiments to optimize interventions along customer journey: A reinforcement learning approach. *Available at SSRN 3939073* .

Tang D, Agarwal A, O'Brien D, Meyer M (2010) Overlapping experiment infrastructure: More, better, faster experimentation. *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 17–26.

Wager S, Athey S (2018) Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association* 113(523):1228–1242.

Wooldridge JM (2010) *Econometric analysis of cross section and panel data* (MIT press).

Wu CJ, Hamada MS (2011) *Experiments: planning, analysis, and optimization* (John Wiley & Sons).

Xie H, Aurisset J (2016) Improving the sensitivity of online controlled experiments: Case studies at netflix. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 645–654.

Xiong R, Chin A, Taylor S, Athey S (2022) Bias-variance tradeoffs for designing simultaneous temporal experiments .

Xiong T, Wang Y, Zheng S (2020) Orthogonal traffic assignment in online overlapping a/b tests. Technical report, EasyChair.

Yarotsky D (2017) Error bounds for approximations with deep relu networks. *Neural Networks* 94:103–114.

Ye Z, Zhang DJ, Zhang H, Zhang R, Chen X, Xu Z (2022) Cold start to improve market thickness on online advertising platforms: Data-driven algorithms and field experiments. *Management Science* forthcoming.

Zeng Z, Dai H, Zhang D, Zhang H, Zhang RP, Xu Z, Shen ZJM (2022) The impact of social nudges on user-generated content for social network platforms. *Management Science* forthcoming.

Zhang DJ, Dai H, Dong L, Qi F, Zhang N, Liu X, Liu Z, Yang J (2020) The long-term and spillover effects of price promotions on retailing platforms: Evidence from a large randomized experiment on alibaba. *Management Science* 66(6):2589–2609.

Zhang Y, Politis DN (2022) Ridge regression revisited: Debiasing, thresholding and bootstrap. *The Annals of Statistics* 50(3):1401–1422.

Online Appendices

Appendix A: Technical Details

A.1. Regularity Assumptions

The following regularity assumption is made throughout the theoretical analysis of this paper.

ASSUMPTION 3. *(a). $\boldsymbol{z}_i = (y_i, \boldsymbol{x}_i', \check{\boldsymbol{t}}_i')'$, $1 \le i \le n$, are i.i.d. copies from the population random variables $\boldsymbol{Z} = (Y, \boldsymbol{X}', \boldsymbol{T}')' \in \mathcal{Y} \times [-1,1]^{d_{\boldsymbol{X}}} \times \{0,1\}^m$, where $\mathcal{Y}$ is the bounded support of the outcome Y.*

(b). The parameter function $\boldsymbol{\theta}^*(\boldsymbol{x})$ is uniformly bounded. Furthermore, $\theta_k^*(\boldsymbol{x}) \in W^{p,\infty}([-1,1]^{d_{\boldsymbol{X}}})$, $k = 1, 2, \ldots, d_{\boldsymbol{\theta}}$, where for positive integers p, define the Hölder ball $W^{p,\infty}([-1,1]^{d_{\boldsymbol{X}}})$ of functions $h : \mathbb{R}^{d_{\boldsymbol{X}}} \mapsto \mathbb{R}$ with smoothness $p \in \mathbb{N}_+$ as,

$$W^{p,\infty}([-1,1]^{d_{\boldsymbol{X}}}) := \Big\{ h : \max_{\boldsymbol{r}, |\boldsymbol{r}| < p} \operatorname*{ess\,sup}_{\boldsymbol{v} \in [-1,1]^{d_{\boldsymbol{X}}}} D^{\boldsymbol{r}} h(\boldsymbol{v}) \le 1 \Big\},$$

where $\boldsymbol{r} = (r_1, \ldots, r_{d_{\boldsymbol{X}}})$, $|\boldsymbol{r}| = r_1 + \cdots + r_{d_{\boldsymbol{X}}}$ and $D^{\boldsymbol{r}} h$ is the weak derivative.

We remark that Assumption 3(a) implies that the DGP is uniformly bounded whereas Assumption 3(b) ensures that the ground-truth parameter functions are uniformly bounded, and enjoy sufficient smoothness so they can be accurately approximated by DNNs. The smoothness assumptions (see, also, Assumption 2 in Farrell et al. 2020) are critical to deriving the sufficiently fast convergence rate of the estimator $\hat{\boldsymbol{\theta}}(\cdot)$.

We also make the following assumption throughout our analysis to ensure the identifiability and sufficient convergence rate. Let $\boldsymbol{t}(S) = (t_1(S), \ldots, t_m(S))' \in \{0,1\}^m$ denote the treatment assignment such that $t_i(S) = \mathbf{1}\{i \in S\}$, where $S \subset \{1, 2, \ldots, m\}$, and define $\tilde{\boldsymbol{T}} := (1, \boldsymbol{T}')'$.

ASSUMPTION 4. *(a) Uniformly for all $\boldsymbol{x}$, $\mathbb{E}[\tilde{\boldsymbol{T}}\tilde{\boldsymbol{T}}' | \boldsymbol{X} = \boldsymbol{x}] \succ 0$.*

(b) Any of the following conditions hold:

(i) $G(\cdot,\cdot)$ is of the Multiplicative Form;

(ii) $G(\cdot,\cdot)$ is of the Standard Sigmoid Form;

(iii) $G(\cdot,\cdot)$ is of the Generalized Sigmoid Form I, $\mathbb{E}[\boldsymbol{T}\boldsymbol{T}' | \boldsymbol{X} = \boldsymbol{x}] \succ 0$, $|\theta_{m+1}^*(\boldsymbol{x})| > 0$, and $\nu(\boldsymbol{t}(\emptyset)|\boldsymbol{x}) > 0$;

(iv) $G(\cdot,\cdot)$ is of the Generalized Sigmoid Form II, $|\theta_{m+1}^*(\boldsymbol{x})| > 0$, and there exists a triplet (i, S_1, S_2) ($i \in \{1, \ldots, m\}$, $S_1, S_2 \subset \{1, 2, \ldots, m\}$) such that $i \notin S_1$, $i \notin S_2$, $S_1 \neq S_2$, and $\nu(\boldsymbol{t}(S_1 \cup \{i\})|\boldsymbol{x}) \cdot \nu(\boldsymbol{t}(S_1)|\boldsymbol{x}) \cdot \nu(\boldsymbol{t}(S_2 \cup \{i\})|\boldsymbol{x}) \cdot \nu(\boldsymbol{t}(S_2)|\boldsymbol{x}) > 0$, and $|G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2)) - G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1))| > 0$.

We remark that Assumption 4(a) is satisfied as long as all m individual treatment conditions and the fully control condition are assigned with positive probability, i.e., $\nu((0,0,\ldots,0)'|\boldsymbol{x}) > 0$, $\nu((1,0,\ldots,0)'|\boldsymbol{x}) > 0$, $\nu((0,1,\ldots,0)'|\boldsymbol{x}) > 0, \ldots,$ $\nu((0,0,\ldots,1)'|\boldsymbol{x}) > 0$, uniformly for all $\boldsymbol{x}$.

A.2. Influence Function

The following Lemma formally states a generic result regarding the influence function, which proves useful to derive the influence function in our setting (Proposition 2). The derivation of this influence function result is already presented in Farrell et al. (2020). We reproduce the derivation of influence function for completeness. This derivation follows the pathwise derivative method originated in Newey (1994).

LEMMA 1 **(Theorem 2 in Farrell et al. (2020))**. *For all $\boldsymbol{t} \in \{0,1\}^m$, suppose the following conditions hold uniformly for $\boldsymbol{x}$. (i) Equation (1) holds and identifies $\boldsymbol{\theta}^*(\cdot)$. (ii) $\boldsymbol{\Lambda}(x) := \mathbb{E}[\ell_{\boldsymbol{\theta}\boldsymbol{\theta}}(Y, \boldsymbol{T}, \boldsymbol{\theta}(\boldsymbol{x}))|\boldsymbol{X} = \boldsymbol{x}]$ is invertible with uniformly bounded inverse. (iii) Parameter $\mu(\boldsymbol{t})$ is identified, pathwise differentiable, and $H(\cdot)$ is thrice continuously differentiable in $\boldsymbol{\theta}$. (iv) $H(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t}, \boldsymbol{t}_0)$ and $\ell_{\boldsymbol{\theta}}(Y, \boldsymbol{T}, \boldsymbol{\theta}^*(\boldsymbol{X}))$ possess $q > 4$ finite absolute moments and positive variance. Then for the treatment effect $\mu(\boldsymbol{t})$, the Neyman orthogonal score is $\psi(\boldsymbol{z}, \boldsymbol{\theta}, \boldsymbol{\Lambda}; \boldsymbol{t}, \boldsymbol{t}_0) - \mu(\boldsymbol{t})$, where*

$$\psi(\boldsymbol{z}, \boldsymbol{\theta}, \boldsymbol{\Lambda}; \boldsymbol{t}, \boldsymbol{t}_0) = H(\boldsymbol{x}, \boldsymbol{\theta}(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0) - H_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{\theta}(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0)' \boldsymbol{\Lambda}(\boldsymbol{x})^{-1} \ell_{\boldsymbol{\theta}}(y, \check{\boldsymbol{t}}, \boldsymbol{\theta}(\boldsymbol{x})), \tag{9}$$

where $\ell_{\boldsymbol{\theta}}, H_{\boldsymbol{\theta}}$ are $d_{\boldsymbol{\theta}}$-dimensional vectors of first order derivatives, and $\ell_{\boldsymbol{\theta}\boldsymbol{\theta}}$ is the $d_{\boldsymbol{\theta}} \times d_{\boldsymbol{\theta}}$ Hessian matrix of ℓ, with $\{k_1, k_2\}$ element defined by $\partial^2 \ell / \partial \theta_{k_1} \partial \theta_{k_2}$.

Proof. Following Newey (1994), we assume that the parametric model is indexed by η, so the data distribution is expressed as F_η, and true models are indexed by $\eta = 0$. Correspondingly,

$$\mu(\eta) = \int H(\boldsymbol{x}, \boldsymbol{\theta}(\boldsymbol{x}; \eta); \boldsymbol{t}, \boldsymbol{t}_0) f(\boldsymbol{x}; \eta) dx,$$

where $f(\cdot)$ is the density of $\boldsymbol{x}$ regarding an appropriate dominating measure. When evaluating at $\eta = 0$, the dependence on η is omitted, e.g., $\boldsymbol{\theta}(\boldsymbol{x}; 0)$ is simplified as $\boldsymbol{\theta}^*(\boldsymbol{x})$.

For the (true) score $S(y, \boldsymbol{x}, \boldsymbol{t}) = f'_\eta(y, \boldsymbol{x}, \boldsymbol{t}; \eta) / f(y, \boldsymbol{x}, \boldsymbol{t}; \eta)|_{\eta=0}$ of the parametric model, we need to find a function $\psi(y, \boldsymbol{x}, \boldsymbol{t})$ such that

$$\left.\frac{\partial \mu(\eta)}{\partial \eta}\right|_{\eta=0} = \mathbb{E}[\psi(y, \boldsymbol{X}, \boldsymbol{T}) S(y, \boldsymbol{X}, \boldsymbol{T}).$$

According to Newey (1994), the function $\psi(\cdot)$ is then the uniquely defined influence function, given the technical conditions that we have assumed.

Note that

$$\begin{aligned} \left.\frac{\partial \mu(\eta)}{\partial \eta}\right|_{\eta=0} &= \left.\frac{\partial}{\partial \eta} \int H(\boldsymbol{x}, \boldsymbol{\theta}(\boldsymbol{x}; \eta); \boldsymbol{t}, \boldsymbol{t}_0) f(\boldsymbol{x}; \eta) d\boldsymbol{x}\right|_{\eta=0} \\ &= \int H(\boldsymbol{x}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0) \frac{\partial f(\boldsymbol{x}; \eta)}{\partial \eta} d\boldsymbol{x} + \int \left(H_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0)' \boldsymbol{\theta}_\eta(\boldsymbol{x}) \right) f(\boldsymbol{x}) d\boldsymbol{x} \end{aligned} \tag{10}$$

where $\boldsymbol{\theta}_\eta(\boldsymbol{x}) = \boldsymbol{\theta}'_\eta(\boldsymbol{x}; \eta)|_{\eta=0}$ is the $d_{\boldsymbol{\theta}}$ dimensional vector gradient of $\boldsymbol{\theta}(\boldsymbol{x}; \eta)$ with respect to η, evaluated at $\eta = 0$, and $H_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0)$ is the $d_{\boldsymbol{\theta}}$ dimensional gradient vector of H with respect to $\boldsymbol{\theta}$, evaluated at $\eta = 0$.

Next, we will show both terms in equation (10) can be written as expectations of products with score function. The following standard fact of the score function is used,

$$S(y, \boldsymbol{x}, \boldsymbol{t}) = S(y, \boldsymbol{t}|\boldsymbol{x}) + S(\boldsymbol{x})$$

Then the first term of (10) becomes

$$\begin{aligned} \int H(\boldsymbol{x}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0) \frac{\partial f(\boldsymbol{x}; \eta)}{\partial \eta} d\boldsymbol{x} &= \mathbb{E}[H(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0) S(\boldsymbol{X})] \\ &= \mathbb{E}[H(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0) S(Y, \boldsymbol{X}, \boldsymbol{T})], \end{aligned}$$

where the first equality follows $S(\boldsymbol{x}) f(\boldsymbol{x}) = \partial f(\boldsymbol{x}; \eta) / \partial \eta|_{\eta=0}$ and the second equality follows the fact $\mathbb{E}[H(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0) S(Y, \boldsymbol{T}|\boldsymbol{X})] = \mathbb{E}[H(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0) \mathbb{E}[S(Y, \boldsymbol{T}|\boldsymbol{X})] \,|\, X] = 0$. Note that this term is the standard plug-in term in the influence function.

Notice that from the assumption $\mathbb{E}[y \mid \boldsymbol{X}=\boldsymbol{x}, \boldsymbol{T}=\boldsymbol{t}] = G(\boldsymbol{\theta}^*(\boldsymbol{x}), \boldsymbol{t})$ and the fact the first-order condition for square loss minimization holds for any η, we can get the following identity,

$$\mathbb{E}_\eta\Big[\ell_{\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}(\boldsymbol{X};\eta)) \,\Big|\, \boldsymbol{X}=\boldsymbol{x}\Big] \equiv 0.$$

Making this explicit gives the form

$$\int \ell_{\boldsymbol{\theta}}(\boldsymbol{z}, \boldsymbol{\theta}(\boldsymbol{x};\eta)) f_{y,\boldsymbol{t}\,|\,\boldsymbol{x}}(y,\boldsymbol{t};\eta \,|\, \boldsymbol{x}) \equiv 0.$$

Differentiating this identity with respect to η and using the chain rule, and evaluating at $\eta = 0$, we obtain

$$\mathbb{E}\Big[\ell_{\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) S(Y, \boldsymbol{T} \,|\, \boldsymbol{X}) \,|\, \boldsymbol{X}=\boldsymbol{x}\Big] + \mathbb{E}\Big[\ell_{\boldsymbol{\theta}\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) \boldsymbol{\theta}_\eta(\boldsymbol{X}) \,|\, \boldsymbol{X}=\boldsymbol{x}\Big] = 0, \tag{11}$$

where $S(Y, \boldsymbol{T} \,|\, \boldsymbol{X})$ is conditional score and obtained by $S(Y, \boldsymbol{t} \,|\, \boldsymbol{x}) f_{y,\boldsymbol{t}\,|\,\boldsymbol{x}}(y, \boldsymbol{t} \,|\, \boldsymbol{x}) = \partial f_{y,\boldsymbol{t}\,|\,\boldsymbol{x}}(y,\boldsymbol{t};\eta \,|\, \boldsymbol{x}) / \partial \eta \,|_{\eta=0}$. Rearranging (11) and using the fact $\boldsymbol{\theta}$ is the function of $\boldsymbol{x}$, we can get the following equality,

$$\mathbb{E}\Big[\ell_{\boldsymbol{\theta}\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) \,|\, \boldsymbol{X}=\boldsymbol{x}\Big] \boldsymbol{\theta}_\eta(\boldsymbol{x}) = -\mathbb{E}\Big[\ell_{\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) S(Y, \boldsymbol{T} \,|\, \boldsymbol{X}) \,|\, \boldsymbol{X}=\boldsymbol{x}\Big].$$

By definition, $\boldsymbol{\Lambda}(\boldsymbol{x}) := \mathbb{E}\Big[\ell_{\boldsymbol{\theta}\boldsymbol{\theta}}(\boldsymbol{z}, \boldsymbol{\theta}^*(\boldsymbol{x})) \,|\, \boldsymbol{X}=\boldsymbol{x}\Big]$ and we have assumed that $\boldsymbol{\Lambda}(\boldsymbol{x})$ is nonsingular, we can solve the equation to get

$$\begin{aligned}
\boldsymbol{\theta}_\eta(\boldsymbol{x}) &= -\mathbb{E}\Big[\ell_{\boldsymbol{\theta}\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) \,|\, \boldsymbol{X}=\boldsymbol{x}\Big]^{-1} \mathbb{E}\Big[\ell_{\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) S(Y, \boldsymbol{T} \,|\, \boldsymbol{X}) \,|\, \boldsymbol{X}=\boldsymbol{x}\Big] \\
&= \mathbb{E}\Big[\boldsymbol{\Lambda}(\boldsymbol{x})^{-1} \ell_{\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) S(Y, \boldsymbol{T} \,|\, \boldsymbol{X}) \,|\, \boldsymbol{X}=\boldsymbol{x}\Big].
\end{aligned}$$

Insert the above equality into (10) and use the iterated expectations. Then we rewrite the second term as,

$$\begin{aligned}
\int \Big(H_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0)' \boldsymbol{\theta}_\eta(\boldsymbol{x})\Big) f(\boldsymbol{x}) d\boldsymbol{x} &= -\mathbb{E}\Big[H_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0)' \mathbb{E}\big[\boldsymbol{\Lambda}(\boldsymbol{X})^{-1} \ell_{\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) S(Y, \boldsymbol{T} \,|\, \boldsymbol{X}) \,|\, \boldsymbol{X}\big]\Big] \\
&= -\mathbb{E}\Big[\mathbb{E}\big[H_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0)' \boldsymbol{\Lambda}(\boldsymbol{X})^{-1} \ell_{\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) S(Y, \boldsymbol{T} \,|\, \boldsymbol{X}) \,|\, \boldsymbol{X}\big]\Big] \\
&= -\mathbb{E}\Big[H_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0)' \boldsymbol{\Lambda}(\boldsymbol{X})^{-1} \ell_{\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) S(Y, \boldsymbol{T} \,|\, \boldsymbol{X})\Big].
\end{aligned}$$

Because the first order condition holds conditionally, we have,

$$\mathbb{E}\Big[H_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0)' \boldsymbol{\Lambda}(\boldsymbol{X})^{-1} \ell_{\boldsymbol{\theta}}(\boldsymbol{z}, \boldsymbol{\theta}^*(\boldsymbol{X})) S(\boldsymbol{X})\Big] = \mathbb{E}\Big[H_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0)' \boldsymbol{\Lambda}(\boldsymbol{X})^{-1} \mathbb{E}[\ell_{\boldsymbol{\theta}}(\boldsymbol{Z}, \boldsymbol{\theta}^*(\boldsymbol{X})) \,|\, \boldsymbol{X}] S(\boldsymbol{X})\Big],$$

which evaluates to zero. By applying $S(y, \boldsymbol{x}, \boldsymbol{t}) = S(y, \boldsymbol{t} \,|\, \boldsymbol{x}) + S(\boldsymbol{x})$ and using the above equality, we can further write the second term in (10) as,

$$-\mathbb{E}\Big[H_{\boldsymbol{\theta}}(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0)' \boldsymbol{\Lambda}(\boldsymbol{X})^{-1} \ell_{\boldsymbol{\theta}}(\boldsymbol{z}, \boldsymbol{\theta}^*(\boldsymbol{X})) S(Y, \boldsymbol{T}, \boldsymbol{X})\Big]$$

Thus, we can finally write equation (10) as the following form,

$$\frac{\partial \mu(\eta)}{\partial \eta} \,|_{\eta=0} = \int H(\boldsymbol{x}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0) - H_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{\theta}^*(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0)' \boldsymbol{\Lambda}(\boldsymbol{x})^{-1} \ell_{\boldsymbol{\theta}}(\boldsymbol{z}, \boldsymbol{\theta}^*(\boldsymbol{x})) f(\boldsymbol{x}) d\boldsymbol{x}.$$

This concludes the proof. □

A.3. Proof of Proposition 1

We first present without proof a key convergence result inherited from Farrell et al. (2020).

LEMMA 2 **(Theorem 1 in Farrell et al. (2020))**. *Suppose Assumption 3, and the following regularity assumptions hold,*

(a). (NONPARAMETRIC IDENTIFIABILITY) *The parameter function $\boldsymbol{\theta}^*(\boldsymbol{x})$ can be nonparametrically identified in DGP (1).*

(b). (LIPSCHITZ CONTINUITY) *There exists a positive constant C_ℓ such that, for any $\boldsymbol{\theta}(\cdot)$, $\tilde{\boldsymbol{\theta}}(\cdot)$ and $\boldsymbol{x}$,*

$$|\,\ell(y,\boldsymbol{t},\boldsymbol{\theta}(\boldsymbol{x}))-\ell(y,\boldsymbol{t},\tilde{\boldsymbol{\theta}}(\boldsymbol{x}))|\le C_\ell\|\boldsymbol{\theta}(\boldsymbol{x})-\tilde{\boldsymbol{\theta}}(\boldsymbol{x})\|_2, \tag{12}$$

(c). (SUFFICIENT CURVATURE) *There exist positive constants c_1 and c_2 such that, for any $\boldsymbol{\theta}(\cdot)\in\mathcal{F}_{DNN}$,*

$$c_1\mathbb{E}[\|\boldsymbol{\theta}(\boldsymbol{X})-\boldsymbol{\theta}^*(\boldsymbol{X})\|_2^2]\le\mathbb{E}[\ell(Y,\boldsymbol{T},\boldsymbol{\theta}(\boldsymbol{X}))]-\mathbb{E}[\ell(Y,\boldsymbol{T},\boldsymbol{\theta}^*(\boldsymbol{X}))]\le c_2\mathbb{E}[\|\boldsymbol{\theta}(\boldsymbol{X})-\boldsymbol{\theta}^*(\boldsymbol{X})\|_2^2]. \tag{13}$$

With the structured DNN of width $H=O(n^{\frac{d_{\boldsymbol{X}}}{2(p+d_{\boldsymbol{X}})}}\log^2 n)$ and depth $L=O(\log n)$ as illustrated in Figure 2, there exists a constant C such that

$$\|\hat{\boldsymbol{\theta}}_k-\boldsymbol{\theta}_k^*\|^2_{L_2(\boldsymbol{X})}\le C\Big\{n^{-\frac{p}{p+d_{\boldsymbol{X}}}}\log^8 n+\frac{\log\log n}{n}\Big\}$$

and

$$\mathbb{E}_n\big[(\hat{\boldsymbol{\theta}}_k-\boldsymbol{\theta}_k^*)^2\big]\le C\Big\{n^{-\frac{p}{p+d_{\boldsymbol{X}}}}\log^8 n+\frac{\log\log n}{n}\Big\}$$

for n large enough with probability at least $1-\exp(n^{-\frac{d_{\boldsymbol{X}}}{p+d_{\boldsymbol{X}}}}\log^8 n)$, for $k=1,\ldots,d_{\boldsymbol{\theta}}$.

Assumptions in Lemma 2 are natural and common in the nonparametric M-estimation literature, which consists of three parts: (a) the nonparametric identifiability of $\boldsymbol{\theta}^*(\cdot)$, (b) the Lipschitz continuity of loss (i.e., Eqn. (12)), and (c) the sufficient curvature of loss (i.e., Eqn. (13)). Whereas the Lipschitz continuity condition is mild and easy to check in our setting, the identifiability of $\boldsymbol{\theta}^*(\cdot)$ and the sufficient curvature condition are non-trivial and should be verified carefully. In particular, the sufficient curvature condition (13) is usually implied by a proper choice of the link function $G(\cdot,\cdot)$ and the treatment assignment mechanism $\nu(\cdot|\cdot)$. This condition helps translate the convergence of outcomes Y into that of parameter functions $\boldsymbol{\theta}(\cdot)$.

To get the convergence results of parameter functions shown in Proposition 1, it is sufficient for us to verify Assumption 3 and the assumptions in Lemma 2 are satisfied. Notice that Assumption 3 is essentially the mild regularity assumption on the data-generating process. And the Lipschitz condition in Assumption (b) can be easily satisfied by for all our proposed link functions in Section 3.2 because all G functions are sufficiently smooth with bounded $\boldsymbol{X}$, $\boldsymbol{T}$, and $\boldsymbol{\theta}$. Because we use the squared loss function throughout the paper, the second "$\le$" of curvature conditions in Assumption (c) is satisfied for our link functions. It is sufficient for us to verify the identification assumption and first "$\le$" of curvature conditions in Assumption (c) for different forms of G functions to guarantee the convergence of structured DNNs.

To simplify the notation, we define the one-dimension sigmoid function as $S(x):=1/(1+\exp(-x))$. In the following, we give the proof for each proposed link function.

Standard Sigmoid Form In this part, we start with the standard sigmoid form,

$$G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}) := \frac{a}{1+\exp\Big(-(\theta_0(\boldsymbol{x})+\theta_1(\boldsymbol{x})t_1+\cdots+\theta_m(\boldsymbol{x})t_m)\Big)}+b. \tag{14}$$

where constant $a \neq 0,\ b$ are known constants, $\boldsymbol{\theta}' = (\theta_0, \theta_1, \ldots, \theta_m)$. This form with $a = 1,\ b = 0$ can be widely used for binary classification. To satisfy the sufficient curvature condition, we need the sufficient condition that matrix $\mathbb{E}[\tilde{\boldsymbol{T}}\tilde{\boldsymbol{T}}'|\boldsymbol{X} = \boldsymbol{x}]$ is invertible uniformly in $\boldsymbol{x}$, where $\tilde{\boldsymbol{T}}' = (1, \boldsymbol{T}')$. Note that because the sigmoid function is continuously invertible, it is equivalent to verify for linear link function $\boldsymbol{\theta}'\tilde{\boldsymbol{t}}$. The detailed proof of identification and curvature conditions are straightforward and can be found in Corollary 1 in Farrell et al. (2020).

Multiplicative Form

$$G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}) = \theta_0(\boldsymbol{x})\big(1+\theta_1(\boldsymbol{x})t_1\big)\ldots\big(1+\theta_m(\boldsymbol{x})t_m\big), \tag{15}$$

where $\mu \leq \theta_0(\boldsymbol{x}) \leq M$, and $\mu \leq 1+\theta_k(\boldsymbol{x}) \leq M,\ k = 1, \ldots, m$, uniformly in $\boldsymbol{x}$, for some $M > \mu > 0$. This uniformly positive condition guarantees the function form is well-defined. The proof of identification and curvature conditions can be verified as shown in the following.

$$\begin{aligned}\log G &= \log\theta_0(\boldsymbol{x})(1+\theta_1(\boldsymbol{x})t_1)\ldots(1+\theta_m(\boldsymbol{x})t_m)\\ &= \log\theta_0(\boldsymbol{x})+\log(1+\theta_1(\boldsymbol{x})t_1)+\cdots+\log(1+\theta_m(\boldsymbol{x})t_m)\\ &= \log\theta_0(\boldsymbol{x})+\log(1+\theta_1(\boldsymbol{x}))t_1+\cdots+\log(1+\theta_m(\boldsymbol{x}))t_m,\end{aligned}$$

which means the functional form G is equivalent to $\exp(a(\boldsymbol{x})+b_1(\boldsymbol{x})t_1+\cdots+b_m(\boldsymbol{x})t_m)$, with $a(\boldsymbol{x}) = \log\theta_0(\boldsymbol{x})$, and $b_k(\boldsymbol{x}) = \log(1+\theta_k(\boldsymbol{x}))$. Because the exponential function is monotone and smooth, to satisfy the sufficient curvature condition, we only need, $\mathbb{E}[\tilde{\boldsymbol{T}}\tilde{\boldsymbol{T}}'|\boldsymbol{X} = \boldsymbol{x}]$ are invertible uniformly in $\boldsymbol{x}$, where $\tilde{\boldsymbol{T}}' = (1, \boldsymbol{T}')$.

Generalized Sigmoid Form I

$$G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}) := \frac{\theta_{m+1}(\boldsymbol{x})}{1+\exp\Big(-(\theta_1(\boldsymbol{x})t_1+\cdots+\theta_m(\boldsymbol{x})t_m)\Big)}. \tag{16}$$

where $\theta_{m+1}(\boldsymbol{x}) \in \mathbb{R}$ can capture the range of outcome, and the sign of $\theta_i(\boldsymbol{x})$ represents the experiment $i = 1, \ldots, m$ has a positive or negative effect. To satisfy both identification and sufficient curvature, the condition that $\mathbb{E}[\boldsymbol{T}\boldsymbol{T}'|\boldsymbol{X} = \boldsymbol{x}]$ are invertible uniformly in $\boldsymbol{x}$ is not sufficient anymore.

In this case, we need the stronger assumption that uniformly for any $\boldsymbol{x}$, $\nu(\boldsymbol{0}|\boldsymbol{x}) = \mathbb{P}[\boldsymbol{t} = \boldsymbol{0}|\boldsymbol{x}] > 0$ and $\mathbb{E}[\boldsymbol{T}\boldsymbol{T}'|\boldsymbol{X} = \boldsymbol{x}]$ invertible. It is easy to verify the identification since under $\boldsymbol{t} = \boldsymbol{0}$, $\theta_0(\boldsymbol{x})$ is identified, and following the same argument with invertible $\mathbb{E}[\boldsymbol{T}\boldsymbol{T}'|\boldsymbol{X} = \boldsymbol{x}]$, we get identified $\boldsymbol{\theta}(x)$.

To prove the Proposition 1 under *Generalized Sigmoid Form I*, it is sufficient to show the identification of $\boldsymbol{\theta}^*(\boldsymbol{x})$ and the sufficient curvature property under the assumption $\nu(\boldsymbol{0}|\boldsymbol{x}) > 0$, $\mathbb{E}[\boldsymbol{T}\boldsymbol{T}'|\boldsymbol{X} = \boldsymbol{x}] \succ 0$, and $|\theta^*_{m+1}(\boldsymbol{x})| > 0$ uniformly in $\boldsymbol{x}$, i.e., Assumption 4 (a) and (b-iii).

Proof. For simplicity of discussion, we use notation $\boldsymbol{T}=(T_1,\ldots,T_m)'$, and $\boldsymbol{\theta}=(\theta_1,\ldots,\theta_m)'$.

First, we show that $\boldsymbol{\theta}^*(\boldsymbol{x})$ can be nonparametrically identified. Let $\mathbb{E}[Y|\boldsymbol{X}=\boldsymbol{x},\boldsymbol{T}=\boldsymbol{t}]=\hat{\theta}_{m+1}(\boldsymbol{x})S(\hat{\boldsymbol{\theta}}(\boldsymbol{x})'\boldsymbol{t})=\tilde{\theta}_{m+1}(\boldsymbol{x})S(\tilde{\boldsymbol{\theta}}(\boldsymbol{x})'\boldsymbol{t})$. Because $\nu(\boldsymbol{0}|\boldsymbol{x})=\mathbb{P}[\boldsymbol{T}=\boldsymbol{0}|\boldsymbol{X}=\boldsymbol{x}]>0$, and $S(0)\neq 0$, we get $\hat{\theta}_{m+1}(\boldsymbol{x})=\tilde{\theta}_{m+1}(\boldsymbol{x})$. Because $\theta^*_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}^*(\boldsymbol{x})'\boldsymbol{t})\neq 0$, and $\hat{\theta}_{m+1}(\boldsymbol{x})=\tilde{\theta}_{m+1}(\boldsymbol{x})$, we have $S(\hat{\boldsymbol{\theta}}(\boldsymbol{x})'\boldsymbol{t})=S(\tilde{\boldsymbol{\theta}}(\boldsymbol{x})'\boldsymbol{t})$. Note that S is continuously invertible. We derive the following result,

$$0=\mathbb{E}[(\hat{\boldsymbol{\theta}}'(x)\boldsymbol{t}-\tilde{\boldsymbol{\theta}}'(x)\boldsymbol{t})^2|\boldsymbol{x}]=(\hat{\boldsymbol{\theta}}'(x)-\tilde{\boldsymbol{\theta}}'(x))\mathbb{E}[\boldsymbol{T}\boldsymbol{T}'|\boldsymbol{x}](\hat{\boldsymbol{\theta}}(x)-\tilde{\boldsymbol{\theta}}(x)).$$

Since $\mathbb{E}[\boldsymbol{T}\boldsymbol{T}'|\ \boldsymbol{X}=\boldsymbol{x}]\succ 0$, we have $\hat{\boldsymbol{\theta}}(x)=\tilde{\boldsymbol{\theta}}(x)$.

Next, we show the sufficient curvature condition. Due to the squared loss function $\mathbb{E}[\ell(Y,\boldsymbol{T},\boldsymbol{\theta}(\boldsymbol{X}))]-\mathbb{E}[\ell(Y,\boldsymbol{T},\boldsymbol{\theta}^*(\boldsymbol{X}))]=\mathbb{E}[(G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})-G(\boldsymbol{\theta}^*(\boldsymbol{x}),\boldsymbol{t}))^2]=\mathbb{E}[(\theta_{m+1}(\boldsymbol{X})S(\boldsymbol{\theta}(\boldsymbol{X})'\boldsymbol{T})-\theta^*_{m+1}(\boldsymbol{X})S(\boldsymbol{\theta}^*(\boldsymbol{X})'\boldsymbol{T}))^2]$, its is equivalent to show that there exist a constant $c>0$ such that $c\mathbb{E}[\sum_{i=1}^{m+1}(\theta_i(\boldsymbol{X})-\theta^*_i(\boldsymbol{X}))^2]\leq\mathbb{E}[(\theta_{m+1}(\boldsymbol{X})S(\boldsymbol{\theta}(\boldsymbol{X})'\boldsymbol{T})-\theta^*_{m+1}(\boldsymbol{X})S(\boldsymbol{\theta}^*(\boldsymbol{X})'\boldsymbol{T}))^2]$.

Because the condition $\mathbb{P}[\boldsymbol{T}=\boldsymbol{0}|\ \boldsymbol{X}=\boldsymbol{x}]>0,\ \forall\boldsymbol{x}$ and $S(0)\neq 0$, there exits a constant $c_1>0$, such that,

$$\mathbb{E}[(\theta_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})-\theta^*_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}^*(\boldsymbol{x})'\boldsymbol{t}))^2]\geq c_1\mathbb{E}[(\theta_{m+1}(\boldsymbol{x})-\theta^*_{m+1}(\boldsymbol{x}))^2]. \tag{17}$$

With the condition $\mathbb{E}[\boldsymbol{T}\boldsymbol{T}'|\boldsymbol{X}=\boldsymbol{x}]\succ 0$, we can show that there exists a constant $c_2>0$, the following inequality holds,

$$\mathbb{E}[(S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})-S(\boldsymbol{\theta}^*(\boldsymbol{x})'\boldsymbol{t}))^2]\geq c_2\mathbb{E}[\sum_{i=1}^{m}(\theta_i(\boldsymbol{x})-\theta^*_i(\boldsymbol{x}))^2]. \tag{18}$$

Follow the below decomposition from the triangle inequality,

$$\begin{aligned}
&|\theta^*_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}^*(\boldsymbol{x})'\boldsymbol{t})-\theta_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})|\\
\geq&|\theta^*_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}^*(\boldsymbol{x})'\boldsymbol{t})-\theta^*_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})|-|\theta^*_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})-\theta_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})|\\
=&|\theta^*_{m+1}(\boldsymbol{x})||S(\boldsymbol{\theta}^*(\boldsymbol{x})'\boldsymbol{t})-S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})|-|\theta^*_{m+1}(\boldsymbol{x})-\theta_{m+1}(\boldsymbol{x})|S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})
\end{aligned}$$

With the boundedness condition of $\theta^*_{m+1}(x)$, and $S(\cdot)$, and taking the full expectation to both sides, it holds that there exists a constant $M_1>0$ such that

$$\mathbb{E}[(S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})-S(\boldsymbol{\theta}^*(\boldsymbol{x})'\boldsymbol{t}))^2]\leq M_1(\mathbb{E}[(\theta_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}(\boldsymbol{x})'\boldsymbol{t})-\theta^*_{m+1}(\boldsymbol{x})S(\boldsymbol{\theta}^*(\boldsymbol{x})'\boldsymbol{t}))^2]+\mathbb{E}[(\theta^*_{m+1}(\boldsymbol{x})-\theta_{m+1}(\boldsymbol{x}))^2]). \tag{19}$$

Putting the above three inequalities (17), (18), and (19) together, we finish the proof.

□

Generalized Sigmoid Form II

In this part, we consider the following parametric form of the conditional mean model

$$G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}):=\frac{\theta_{m+1}(\boldsymbol{x})}{1+\exp\Big(-(\theta_0(\boldsymbol{x})+\theta_1(\boldsymbol{x})t_1+\cdots+\theta_m(\boldsymbol{x})t_m)\Big)}. \tag{20}$$

Let $\boldsymbol{t}(S)$ denote the treatment assignment such that $t_i(S)=1$ if and only if $i\in S$, where $S\subset\{1,2,\ldots,m\}$. We restate the following sufficient conditions mentioned in Assumption 4 (a) and (b-iv).

- The matrix $\mathbb{E}[\tilde{\boldsymbol{T}}\tilde{\boldsymbol{T}}'|\boldsymbol{X}=\boldsymbol{x}]\succ 0$, where $\tilde{\boldsymbol{T}}'=(1,\boldsymbol{T}')$.

- There exists a triplet $i \in \{1,\dots,m\}$, S_1, $S_2 \subset \{1,2,\dots,m\}$ such that $i \notin S_1$, $i \notin S_2$, $S_1 \neq S_2$, and

$$\nu(\boldsymbol{t}(S_1 \cup \{i\})|\boldsymbol{x}) \cdot \nu(\boldsymbol{t}(S_1)|\boldsymbol{x}) \cdot \nu(\boldsymbol{t}(S_2 \cup \{i\})|\boldsymbol{x}) \cdot \nu(\boldsymbol{t}(S_2)|\boldsymbol{x}) > 0,$$

  and

$$|G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2)) - G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1))| > 0.$$

- $|\theta_{m+1}(\boldsymbol{x})| > 0$

We argue that those assumptions are sufficient to guarantee identification and sufficient curvature near the truth, i.e., Proposition 1 under *Generalized Sigmoid Form II*

Proof. To flesh out the analysis, we notice that

$$\frac{\theta_{m+1}(\boldsymbol{x})}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t})} - 1 = \exp\Big(- (\theta_0(\boldsymbol{x}) + \theta_1(\boldsymbol{x})t_1 + \cdots + \theta_m(\boldsymbol{x})t_m)\Big).$$

As a result, we it must be that

$$\left(\frac{\theta_{m+1}(\boldsymbol{x})}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\}))} - 1\right) \Big/ \left(\frac{\theta_{m+1}(\boldsymbol{x})}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1))} - 1\right) = \left(\frac{\theta_{m+1}(\boldsymbol{x})}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\}))} - 1\right) \Big/ \left(\frac{\theta_{m+1}(\boldsymbol{x})}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2))} - 1\right),$$

since both the left-hand side and right hand side equal $\exp(-\theta_i(\boldsymbol{x})t_i)$. After the basic operation, this identity can be rewritten as

$$\begin{aligned}\theta_{m+1}(\boldsymbol{x}) \cdot \Bigg[&\left(\frac{1}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2))} - \frac{1}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1))}\right) \cdot \theta_{m+1}(\boldsymbol{x}) \\ &- \frac{1}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\}))} - \frac{1}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2))} + \frac{1}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\}))} + \frac{1}{G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1))}\Bigg] = 0\end{aligned}$$

Because we assume $\theta_{m+1}(\boldsymbol{x}) \neq 0$, this equation admits a unique solution of $\theta_{m+1}(\boldsymbol{x})$. This implies that $\theta_{m+1}(\boldsymbol{x})$ is identified under our assumption. Indeed, suppose that there exist $\tilde{\theta}_0(\boldsymbol{x}), \tilde{\theta}_1(\boldsymbol{x}), \dots, \tilde{\theta}_{m+1}(\boldsymbol{x})$ such that

$$G(\tilde{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{T}) = G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{T}),$$

for all given the realization $\boldsymbol{x}$ and $\boldsymbol{t}$. Then,

$$\mathbb{P}\Big[\boldsymbol{T} = \boldsymbol{t}(S_1 \cup \{i\})\Big|\boldsymbol{X}\Big] \cdot \mathbb{P}\Big[\boldsymbol{T} = \boldsymbol{t}(S_1)\Big|\boldsymbol{X}\Big] \cdot \mathbb{P}\Big[\boldsymbol{T} = \boldsymbol{t}(S_2 \cup \{i\})\Big|\boldsymbol{X}\Big] \cdot \mathbb{P}\Big[\boldsymbol{T} = \boldsymbol{t}(S_2)\Big|\boldsymbol{X}\Big] > 0,$$

implies $G(\tilde{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{T}) = G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{T})$ for $\boldsymbol{T} = \boldsymbol{t}(S_1 \cup \{i\})$, $\boldsymbol{T} = \boldsymbol{t}(S_1)$, $\boldsymbol{T} = \boldsymbol{t}(S_2 \cup \{i\})$ and $\boldsymbol{T} = \boldsymbol{t}(S_2)$. Then, the argument above implies that $\theta_{m+1}(\boldsymbol{x}) = \tilde{\theta}_{m+1}(\boldsymbol{x})$. The rank condition $\mathbb{E}[\tilde{\boldsymbol{T}}\tilde{\boldsymbol{T}}'|\boldsymbol{X} = \boldsymbol{x}] \succ 0$ implies that all other parameters are identified as well given the unique $\theta_{m+1}(\boldsymbol{x})$, because it follows from $\theta_{m+1}(\boldsymbol{x}) = \tilde{\theta}_{m+1}(\boldsymbol{x})$ that

$$\theta_0(\boldsymbol{x}) + \theta_1(\boldsymbol{x})T_1 + \cdots + \theta_m(\boldsymbol{x})T_m = \tilde{\theta}_0(\boldsymbol{x}) + \tilde{\theta}_1(\boldsymbol{x})T_1 + \cdots + \tilde{\theta}_m(\boldsymbol{x})T_m$$

for all given the realization $\boldsymbol{x}$ and $\boldsymbol{t}$, since the sigmoid function is monotone. This contradicts the rank condition unless $\theta_i(\boldsymbol{x}) = \tilde{\theta}_i(\boldsymbol{x})$ for all $i = 0, \dots, m$. To summarize, it must be that $\tilde{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{\theta}(\boldsymbol{x})$, uniformly for all realized $\boldsymbol{X} = \boldsymbol{x}$.

Notice that the above proof of identification also holds in the $\boldsymbol{X}$-a.e. sense if we relax assumptions by only requiring those conditions in $\boldsymbol{X}$-a.e. sense. However, to satisfy the sufficient curvature condition near the truth, assumptions in a.e. sense is not sufficient, and we do require it to hold uniformly for all $\boldsymbol{X} = \boldsymbol{x}$.

Next, we give the proof of the sufficient curvature condition that there exists a constant $c_1 > 0$ such that

$$c_1 \mathbb{E}[\|\boldsymbol{\theta}(\boldsymbol{X}) - \boldsymbol{\theta}^*(\boldsymbol{X})\|_2^2] \leq \mathbb{E}[(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{T}) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{T}))^2].$$

To carry out the analysis of the sufficient curvature condition, we first notice that the assumption,

$$\mathbb{P}\Big[\boldsymbol{T} = \boldsymbol{t}(S_1 \cup \{i\})\Big|\boldsymbol{X}\Big] \cdot \mathbb{P}\Big[\boldsymbol{T} = \boldsymbol{t}(S_1)\Big|\boldsymbol{X}\Big] \cdot \mathbb{P}\Big[\boldsymbol{T} = \boldsymbol{t}(S_2 \cup \{i\})\Big|\boldsymbol{X}\Big] \cdot \mathbb{P}\Big[\boldsymbol{T} = \boldsymbol{t}(S_2)\Big|\boldsymbol{X}\Big] > 0,$$

implies there exists a constant c such that,

$$\begin{aligned}
c\mathbb{E}\Big[\big(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{t}(S_1 \cup \{i\})) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t}(S_1 \cup \{i\}))\big)^2\Big] &\leq \mathbb{E}\Big[(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{T}) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{T}))^2\Big] \\
c\mathbb{E}\Big[\big(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{t}(S_1)) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t}(S_1))\big)^2\Big] &\leq \mathbb{E}\Big[(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{T}) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{T}))^2\Big] \\
c\mathbb{E}\Big[\big(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{t}(S_2 \cup \{i\})) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t}(S_2 \cup \{i\}))\big)^2\Big] &\leq \mathbb{E}\Big[(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{T}) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{T}))^2\Big] \\
c\mathbb{E}\Big[\big(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{t}(S_2)) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t}(S_2))\big)^2\Big] &\leq \mathbb{E}\Big[(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{T}) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{T}))^2\Big].
\end{aligned}$$

Note that the right-hand side value of the above inequalities characterizes the DNN estimation error at the first stage and is expressed as a function in the number of samples n, see Lemma 2. Under the same reasoning as an identification proof, we can express $\theta_{m+1}(\boldsymbol{x})$ as,

$$\begin{aligned}
&\Big(G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\})) + G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\})) \\
&-G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\})) - G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\}))\Big) \\
&\Big/\Big(G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1)) - G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\}))G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2))\Big).
\end{aligned}$$

Because the denominator is uniformly bounded away from 0 by Assumption and $\theta_{m+1}(\boldsymbol{x})$ is expressed as a bounded and smooth function in $G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1 \cup \{i\}))$, $G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2 \cup \{i\}))$, $G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_1))$, $G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}(S_2))$, the errors of which are bounded almost surely in $\boldsymbol{x}$. We can conclude that $\mathbb{E}[|\theta_{m+1}(\boldsymbol{X}) - \theta^*_{m+1}(\boldsymbol{X})|^2] \leq O\Big(\mathbb{E}\Big[(G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{T}) - G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{T}))^2\Big]\Big)$.

We spread out the following decomposition,

$$\begin{aligned}
&\Big|G(\boldsymbol{\theta}^*(\boldsymbol{x}), \boldsymbol{t}) - G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t})\Big| \\
=&\Big|\theta^*_{m+1}(\boldsymbol{x})/\big(1 + \exp\big(-(\theta^*_0(\boldsymbol{x}) + \cdots + \theta^*_m(\boldsymbol{x})t_m)\big)\big) - \theta_{m+1}(\boldsymbol{x})/\big(1 + \exp\big(-(\theta_0(\boldsymbol{x}) + \cdots + \theta_m(\boldsymbol{x})t_m)\big)\big)\Big| \\
\geq&\Big|\theta^*_{m+1}(\boldsymbol{x})/\big(1 + \exp\big(-(\theta^*_0(\boldsymbol{x}) + \cdots + \theta^*_m(\boldsymbol{x})t_m)\big)\big) - \theta^*_{m+1}(\boldsymbol{x})/\big(1 + \exp\big(-(\theta_0(\boldsymbol{x}) + \cdots + \theta_m(\boldsymbol{x})t_m)\big)\big)\Big| \\
&- \Big|\theta^*_{m+1}(\boldsymbol{x})/\big(1 + \exp\big(-(\theta_0(\boldsymbol{x}) + \cdots + \theta_m(\boldsymbol{x})t_m)\big)\big) - \theta_{m+1}(\boldsymbol{x})/\big(1 + \exp\big(-(\theta_0(\boldsymbol{x}) + \cdots + \theta_m(\boldsymbol{x})t_m)\big)\big)\Big| \\
=&\theta^*_{m+1}(\boldsymbol{x}) \cdot \Big|1/\big(1 + \exp\big(-(\theta_0(\boldsymbol{x}) + \cdots + \theta_m(\boldsymbol{x})t_m)\big)\big) - 1/\big(1 + \exp\big(-(\theta^*_0(\boldsymbol{x}) + \cdots + \theta^*_m(\boldsymbol{x})t_m)\big)\big)\Big| \\
&- \Big|\theta_{m+1}(\boldsymbol{x}) - \theta^*_{m+1}(\boldsymbol{x})\Big| \cdot \Big(1/\big(1 + \exp\big(-(\theta_0(\boldsymbol{x}) + \cdots + \theta_m(\boldsymbol{x})t_m)\big)\big)\Big).
\end{aligned}$$

By taking the full expectation to both sides and rearranging terms, we can find a bounded constant M such that the following inequality holds due to the boundness of the sigmoid function and $\theta^*(\boldsymbol{x})$,

$$\begin{aligned}
&\mathbb{E}\Big[\Big(1\Big/\Big(1 + \exp\big(-(\theta_0(\boldsymbol{X}) + \theta_1(\boldsymbol{X})T_1 + \cdots + \theta_m(\boldsymbol{X})T_m)\big)\Big) - 1\Big/\Big(1 + \exp\big(-(\theta^*_0(\boldsymbol{X}) + \theta^*_1(\boldsymbol{X})T_1 + \cdots + \theta^*_m(\boldsymbol{X})T_m)\big)\Big)\Big)^2\Big] \\
&\qquad \leq M\mathbb{E}\Big[\big(G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{T}) - G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{T})\big)^2 + \big(\theta^*_{m+1}(\boldsymbol{X}) - \theta_{m+1}(\boldsymbol{X})\big)^2\Big].
\end{aligned} \tag{21}$$

Notice that by the rank condition $\mathbb{E}[\tilde{\boldsymbol{T}}\tilde{\boldsymbol{T}}'|\boldsymbol{X}] \succ 0$, there must exist a constant c_2 such that,

$$\mathbb{E}\Big[\Big(1\Big/\Big(1+\exp\big(-(\theta_0(\boldsymbol{X})+\theta_1(\boldsymbol{X})T_1+\cdots+\theta_m(\boldsymbol{X})T_m)\big)\Big)-1\Big/\Big(1+\exp\big(-(\theta_0^*(\boldsymbol{X})+\theta_1^*(\boldsymbol{X})T_1+\cdots+\theta_m^*(\boldsymbol{X})T_m)\big)\Big)\Big)^2\Big] \geq c_2\mathbb{E}\Big[\sum_{i=0}^{m}\big(\theta_i(\boldsymbol{X})-\theta_i^*(\boldsymbol{X})\big)^2\Big]. \tag{22}$$

Combining inequalities (21), (22), and $\mathbb{E}[|\theta_{m+1}(\boldsymbol{X})-\theta_{m+1}^*(\boldsymbol{X})|^2] \leq O\Big(\mathbb{E}\Big[(G(\boldsymbol{\theta}(\boldsymbol{X}),\boldsymbol{T})-G(\boldsymbol{\theta}^*(\boldsymbol{X}),\boldsymbol{T}))^2\Big]\Big)$, we finish the proof by showing the fact that,

$$\mathbb{E}\Big[\sum_{i=0}^{m+1}\big(\theta_i(\boldsymbol{X})-\theta_i^*(\boldsymbol{X})\big)^2\Big] \leq O\Big(\mathbb{E}\Big[(G(\boldsymbol{\theta}(\boldsymbol{X}),\boldsymbol{T})-G(\boldsymbol{\theta}^*(\boldsymbol{X}),\boldsymbol{T}))^2\Big]\Big),$$

i.e., there exist a constant c_1 such that $c_1\mathbb{E}[\|\boldsymbol{\theta}(\boldsymbol{X})-\boldsymbol{\theta}^*(\boldsymbol{X})\|_2^2] \leq \mathbb{E}\Big[(G(\boldsymbol{\theta}(\boldsymbol{X}),\boldsymbol{T})-G(\boldsymbol{\theta}^*(\boldsymbol{X}),\boldsymbol{T}))^2\Big]$. □

A.4. Proof of Proposition 2

Proof. The results directly follow from Theorem 2 of Farrell et al. (2020), which we also replicate in Lemma 1 in Appendix A.2. We now verify the assumptions of Lemma 1.

First, we derive the first order derivative $\ell_{\boldsymbol{\theta}}$ and Hessian matrix $\ell_{\boldsymbol{\theta\theta}}$. With the loss function $\ell(y,\boldsymbol{t},\boldsymbol{\theta}(\boldsymbol{x})) = (y-G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}))^2$, we can compute to get,

$$\begin{aligned}
&\ell_{\boldsymbol{\theta}} = (\partial\ell/\partial\theta_1,\ldots,\partial\ell/\partial\theta_{d_{\boldsymbol{\theta}}})' = 2(G-y)G_{\boldsymbol{\theta}},\\
&\ell_{\boldsymbol{\theta\theta}} = 2G_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}' + 2G_{\boldsymbol{\theta\theta}}(G-y),\\
&\boldsymbol{\Lambda}(\boldsymbol{x}) = \mathbb{E}[\ell_{\boldsymbol{\theta\theta}}|\boldsymbol{x}] = 2\mathbb{E}_{\boldsymbol{T}|\boldsymbol{x}}[G_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}'|\boldsymbol{x}] + 2\mathbb{E}_{\boldsymbol{T}|\boldsymbol{x}}\Big[G_{\boldsymbol{\theta\theta}}\mathbb{E}_{y|\boldsymbol{t},\boldsymbol{x}}[(G-y)|\boldsymbol{t},\boldsymbol{x}]|\boldsymbol{x}\Big] = 2\mathbb{E}[G_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}'|\boldsymbol{x}].
\end{aligned}$$

The first three conditions of Lemma 1 follow from Assumptions 2(i), (ii) and (iii). Because the link function G satisfies the smoothness and boundedness conditions of the fourth condition of Lemma 1, Proposition 2 follows immediately from Lemma 1.

□

A.5. Verification of Assumption 2(ii)

In the following, we illustrate that under our treatment assignment mechanism in Assumption 4, Assumption 2(ii), i.e., $\boldsymbol{\Lambda}(\boldsymbol{x}) \succ 0$ (i.e., vectors $\{G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})\}_{\boldsymbol{t}}$ are non-degenerate) is easy to satisfy and can be translated to the very lenient condition. Without loss of generality, we take *Generalized Sigmoid Form II* for illustration.

By definition, we have $\boldsymbol{\Lambda}(\boldsymbol{x}) = 2\mathbb{E}[G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{T})G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{T})'|\boldsymbol{X}=\boldsymbol{x}]$, where

$$G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})' = \Big(\frac{\theta_{m+1}(\boldsymbol{x})\exp(-(\theta_0(\boldsymbol{x})+\cdots+\theta_m(\boldsymbol{x})t_m))}{(1+\exp(-(\theta_0(\boldsymbol{x})+\cdots+\theta_m(\boldsymbol{x})t_m)))^2}(1,t_1,\ldots,t_m),\ \frac{1}{1+\exp(-(\theta_0(\boldsymbol{x})+\cdots+\theta_m(\boldsymbol{x})t_m))}\Big).$$

Thus, to verify $\boldsymbol{\Lambda}(\boldsymbol{x})$ is invertible, it suffices to show that the matrix constructed by vectors $\{G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t})\}_{\boldsymbol{t}}$ has full rank $m+2$. For ease of exposition, we drop the dependence on $\boldsymbol{x}$ without confusion because all conditions should hold uniformly in $\boldsymbol{x}$.

In the following, we consider a common treatment assignment as a special mechanism of Assumption 4*(a),(b-iv)* for illustration (all other assignment rules can be translated into similar lenient conditions). Specifically, to satisfy $\mathbb{E}[\tilde{\boldsymbol{T}}\tilde{\boldsymbol{T}}'|\boldsymbol{X}=\boldsymbol{x}] \succ 0$, we consider the assignment $\nu(\boldsymbol{t}(\emptyset)|\boldsymbol{x}) > 0$, $\nu(\boldsymbol{t}(\{1\})|\boldsymbol{x}) > 0$, $\nu(\boldsymbol{t}(\{2\})|\boldsymbol{x}) > 0,\ldots,\nu(\boldsymbol{t}(\{m\})|\boldsymbol{x}) > 0$, together with the overlapping assignment $\nu(\boldsymbol{t}(S_1)|\boldsymbol{x}) > 0$ (WLOG, choose S_1 from one of single treatment conditions, fix $S_1=\{1\}$), $\nu(\boldsymbol{t}(S_1\cup\{i\})|\boldsymbol{x}) > 0$, $\nu(\boldsymbol{t}(S_2)|\boldsymbol{x}) > 0$, and $\nu(\boldsymbol{t}(S_2\cup\{i\})|\boldsymbol{x}) > 0$.

Let $e_0 = \exp(-\theta_0)$, $e_i = \exp(-\theta_0 - \theta_i)$, for $i = 1, 2, \ldots, m$, and $e_{m+1} = \exp(-\boldsymbol{\theta}' \boldsymbol{t}(S_1 \cup \{i\}))$, then the rank can be written as,

$$rank(\boldsymbol{\Lambda}) \geq rank \begin{pmatrix} \frac{\theta_{m+1} e_0}{1+e_0}(1,0,0,0,\ldots,0) & \frac{1}{1+e_0} \\ \frac{\theta_{m+1} e_1}{1+e_1}(1,1,0,0,\ldots,0) & \frac{1}{1+e_1} \\ \frac{\theta_{m+1} e_2}{1+e_2}(1,0,1,0,\ldots,0) & \frac{1}{1+e_2} \\ \frac{\theta_{m+1} e_3}{1+e_3}(1,0,0,1,\ldots,0) & \frac{1}{1+e_3} \\ \vdots & \vdots \\ \frac{\theta_{m+1} e_m}{1+e_m}(1,0,0,0,\ldots,1) & \frac{1}{1+e_m} \\ \frac{\theta_{m+1} e_{m+1}}{1+e_{m+1}}(1, \boldsymbol{t}(S_1 \cup \{i\})') & \frac{1}{1+e_{m+1}} \end{pmatrix} = rank \begin{pmatrix} \frac{e_0}{1+e_0}(1,0,0,0,\ldots,0) & \frac{1}{1+e_0} \\ \frac{e_1}{1+e_1}(1,1,0,0,\ldots,0) & \frac{1}{1+e_1} \\ \frac{e_2}{1+e_2}(1,0,1,0,\ldots,0) & \frac{1}{1+e_2} \\ \frac{e_3}{1+e_3}(1,0,0,1,\ldots,0) & \frac{1}{1+e_3} \\ \vdots & \vdots \\ \frac{e_m}{1+e_m}(1,0,0,0,\ldots,1) & \frac{1}{1+e_m} \\ \frac{e_{m+1}}{1+e_{m+1}}(1, \boldsymbol{t}(S_1 \cup \{i\})') & \frac{1}{1+e_{m+1}} \end{pmatrix}$$

$$= rank \begin{pmatrix} 1\ 0\ 0 & 0 & \ldots\ 0 & \frac{1}{e_0} \\ 1\ 1\ 0 & 0 & \ldots\ 0 & \frac{1}{e_1} \\ 1\ 0\ 1 & 0 & \ldots\ 0 & \frac{1}{e_2} \\ 1\ 0\ 0 & 1 & \ldots\ 0 & \frac{1}{e_3} \\ \vdots & & & \vdots\ \ \vdots \\ 1\ 0\ 0 & 0 & \ldots\ 1 & \frac{1}{e_m} \\ 1 & \boldsymbol{t}(S_1 \cup \{i\})' & & \frac{1}{e_{m+1}} \end{pmatrix} = rank \begin{pmatrix} 1\ 0\ 0\ 0\ \ldots\ 0 & \frac{1}{e_0} \\ 0\ 1\ 0\ 0\ \ldots\ 0 & \frac{1}{e_1} - \frac{1}{e_0} \\ 0\ 0\ 1\ 0\ \ldots\ 0 & \frac{1}{e_2} - \frac{1}{e_0} \\ 0\ 0\ 0\ 1\ \ldots\ 0 & \frac{1}{e_3} - \frac{1}{e_0} \\ \vdots & \vdots \\ 0\ 0\ 0\ 0\ \ldots\ 1 & \frac{1}{e_m} - \frac{1}{e_0} \\ 0\ 0\ 0\ 0\ \ldots\ 0 & \frac{1}{e_{m+1}} - \frac{1}{e_0} - (\frac{1}{e_1} - \frac{1}{e_0}) - (\frac{1}{e_i} - \frac{1}{e_0}) \end{pmatrix}.$$

The inequality follows from the fact the right-hand side matrix is only expanded by partial of vectors $\{G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t})\}_{\boldsymbol{t}}$. The first equation follows from $\theta_{m+1} \neq 0$. The second equation follows from $1 + e_i \neq 0$ and $e_i \neq 0$, for all $i = 1, 2, \ldots, m$. The third equation follows subtraction operations among rows. To guarantee the full rank $rank(\boldsymbol{\Lambda}) = m + 2$, one sufficient condition is $\frac{1}{e_{m+1}} - \frac{1}{e_1} - \frac{1}{e_i} + \frac{1}{e_0} \neq 0$, i.e., bottom right entry is non-zero, which is a very weak condition.

For other assignment mechanisms in Assumption 4, one can also translate the invertibility of $\boldsymbol{\Lambda}$ into a very lenient condition. We omit the details for brevity.

A.6. ATE Estimator Construction via Cross-fitting

In the following, we introduce the sample-splitting/cross-fitting and estimation procedure from Chernozhukov et al. (2018). Note that in our setting with the known distribution of $\boldsymbol{T}$, we only require the two-way splitting. But for unknown $\boldsymbol{T}$ distribution, three-way splitting is required, with an additional portion of samples used for obtaining $\hat{\boldsymbol{\Lambda}}(\boldsymbol{x})$.

One can split the data samples $\{1, 2, \ldots, n\}$ into S non-overlapping copies $\mathcal{S}_s \subset \{1, 2, \ldots, n\},\ s = 1, 2 \ldots, S$ with the cardinality $|\mathcal{S}_s|$ being proportionally to the sample size n, and let $\mathcal{S}_s^c$ be the complement of $\mathcal{S}_s$. First, we use $\mathcal{S}^c$ to get estimators $\hat{\boldsymbol{\theta}}_s(\cdot)$ of parameters $\boldsymbol{\theta}^*(\cdot)$, and compute $\hat{\boldsymbol{\Lambda}}_s(\cdot)$ given the estimators $\hat{\boldsymbol{\theta}}_s(\cdot)$ and distribution of $\boldsymbol{T}$. Then, one can use the other samples to construct an estimator of $\mu(\boldsymbol{t})$, for any $\boldsymbol{t} \in \{0, 1\}^m$ as,

$$\hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t}) = \frac{1}{S} \hat{\mu}_s(\boldsymbol{t}), \qquad \hat{\mu}_s(\boldsymbol{t}) = \frac{1}{|\mathcal{S}_s|} \sum_{i \in \mathcal{S}_s} \psi(\boldsymbol{z}_i, \hat{\boldsymbol{\theta}}_s(\boldsymbol{x}_i), \hat{\boldsymbol{\Lambda}}_s(\boldsymbol{x}_i); \boldsymbol{t}, \boldsymbol{t}_0). \tag{23}$$

Similarly, the variance estimator can be constructed as,

$$\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t}; \mu) = \frac{1}{S} \hat{\Psi}_s(\boldsymbol{t}), \qquad \hat{\Psi}_s(\boldsymbol{t}) = \frac{1}{|\mathcal{S}_s|} \sum_{i \in \mathcal{S}_s} \Big(\psi(\boldsymbol{z}_i, \hat{\boldsymbol{\theta}}_s(\boldsymbol{x}_i), \hat{\boldsymbol{\Lambda}}_s(\boldsymbol{x}_i); \boldsymbol{t}, \boldsymbol{t}_0) - \hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t}) \Big)^2. \tag{24}$$

The asymptotic normality in Proposition 3*(a)* directly follows from Chernozhukov et al. (2018), and detailed proof can also be found in Farrell et al. (2020).

Therefore, the $(1-\alpha)$-confidence interval of $\hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$ is given by

$$\widehat{\mathcal{CI}}_{\mathsf{DeDL}}(\boldsymbol{t};\mu) = \left[\hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t}) - \frac{1}{\sqrt{n}}\cdot\Phi^{-1}\left(1-\frac{\alpha}{2}\right)\cdot\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\mu), \hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t}) + \frac{1}{\sqrt{n}}\cdot\Phi^{-1}\left(1-\frac{\alpha}{2}\right)\cdot\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\mu)\right], \quad (25)$$

where $\Phi^{-1}(\cdot)$ is the inverse cumulative distribution function of a standard normal random variable.

A.7. Estimation and Inference for Best-Arm Identification

After getting asymptotically normal estimators of average treatment effect $\hat{\mu}(\boldsymbol{t})$ for all experiment combinations $\boldsymbol{t}\in\{0,1\}^m$, the next step is identifying the best experiment combination, which is defined as $\boldsymbol{t}^* := \arg\max_{\boldsymbol{t}\in\{0,1\}^m}\mu(\boldsymbol{t})$.

Following the common practice, one can search for the best treatment combination by searching for the highest ATE estimation. Formally, we define the empirical best treatment level as $\hat{\boldsymbol{t}}^* := \arg\max_{\boldsymbol{t}\in\{0,1\}^m}\hat{\mu}(\boldsymbol{t})$. The remaining job is verifying whether $\hat{\boldsymbol{t}}^*$ is the best treatment level with significant improvements over all other treatment levels, i.e., we do the one-side hypothesis test on whether the average treatment effect of $\hat{\boldsymbol{t}}^*$ is significantly better than other treatments,

$$H_0 : \tau(\boldsymbol{t}) > 0, \text{for all } \boldsymbol{t}\in\{0,1\}^m\backslash\{\hat{\boldsymbol{t}}^*\},$$

where

$$\tau(\boldsymbol{t}) := \mu(\hat{\boldsymbol{t}}^*) - \mu(\boldsymbol{t}) = \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{x}),\hat{\boldsymbol{t}}^*)] - \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{x}),\boldsymbol{t})],$$

is the improvement of ATE of empirically best $\hat{\boldsymbol{t}}^*$ over the treatment level $\boldsymbol{t}\in\{0,1\}^m$. Notice that the $\tau(\boldsymbol{t})$ can be rewritten as,

$$\tau(\boldsymbol{t}) = \mathbb{E}[H(\boldsymbol{x},\boldsymbol{\theta}^*(\boldsymbol{x});\hat{\boldsymbol{t}}^*,\boldsymbol{t})],$$

which is similar to the expression $\mu(\boldsymbol{t}) = \mathbb{E}[H(\boldsymbol{x},\boldsymbol{\theta}^*(\boldsymbol{x});\boldsymbol{t},\boldsymbol{t}_0)]$ simply by changing the inputs of advantage function H.

Applying the similar influence function (9) developed in Section 3.3, i.e., $\psi(\boldsymbol{z},\boldsymbol{\theta},\boldsymbol{\Lambda};\hat{\boldsymbol{t}}^*,\boldsymbol{t}) = H(\boldsymbol{x},\boldsymbol{\theta}(\boldsymbol{x});\hat{\boldsymbol{t}}^*,\boldsymbol{t}) - H_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{\theta}(\boldsymbol{x});\hat{\boldsymbol{t}}^*,\boldsymbol{t})'\boldsymbol{\Lambda}(\boldsymbol{x})^{-1}\ell_{\boldsymbol{\theta}}(y,\check{\boldsymbol{t}},\boldsymbol{\theta}(\boldsymbol{x}))$ and the two-way splitting procedure, we construct the estimators

$$\hat{\tau}_{\mathsf{DeDL}}(\boldsymbol{t}) := \hat{\mu}_{\mathsf{DeDL}}(\hat{\boldsymbol{t}}^*_{\mathsf{DeDL}}) - \hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t}), \quad (26)$$

and variance estimate $\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\tau)$ for $\tau_{\mathsf{DeDL}}(\boldsymbol{t})$.

Proof of Proposition 3

The asymptotic normality in Proposition 3*(a)* directly follows from Chernozhukov et al. (2018), and detailed proof can also be found in Farrell et al. (2020). For Part (b), however, because the empirically optimal arm $\hat{\boldsymbol{t}}^* := \arg\max_{\boldsymbol{t}\in\{0,1\}^m}\hat{\mu}(\boldsymbol{t})$ depends on the samples used for training and inference for ATEs, the proof in Chernozhukov et al. (2018), Farrell et al. (2020) cannot be directly extended for the inference of best-arm identification, i.e., asymptotic normality of $\hat{\tau}_{\mathsf{DeDL}}(\boldsymbol{t})$. This is also the major challenge in the proof of Proposition 3(b) below.

Proof. To tackle the challenge brought by the independence of empirically optimal arm $\hat{\boldsymbol{t}}^*$ on the randomness of samples. We introduce an imaginary but valid inference function based on the true optimal $\boldsymbol{t}^*$, which is fixed as a prior and does not depend on the randomness of samples, i.e., $\psi(\boldsymbol{z},\boldsymbol{\theta},\boldsymbol{\Lambda};\boldsymbol{t}^*,\boldsymbol{t}) - (\mu(\boldsymbol{t}^*) - \mu(\boldsymbol{t}))$. Note that this introduced influence function only virtually exists because one cannot know $\boldsymbol{t}^*$ in advance and is only for proof purposes.

Given the functions $\psi(\boldsymbol{z},\boldsymbol{\theta},\boldsymbol{\Lambda};\boldsymbol{t}^*,\boldsymbol{t})$ and $\psi(\boldsymbol{z},\boldsymbol{\theta},\boldsymbol{\Lambda};\hat{\boldsymbol{t}}^*,\boldsymbol{t})$, we construct the following estimands to simplify the exposition and drop out the subscript DeDL, i.e., $\mu(\boldsymbol{t})$ to represent $\mu_{\mathsf{DeDL}}(\boldsymbol{t})$ without confusion.

$$\hat{\zeta}(\boldsymbol{t}) := \hat{\mu}(\boldsymbol{t}^*) - \hat{\mu}(\boldsymbol{t}), \qquad \zeta(\boldsymbol{t}) := \mu(\boldsymbol{t}^*) - \mu(\boldsymbol{t}),$$

$$\hat{\tau}(\boldsymbol{t}) := \hat{\mu}(\hat{\boldsymbol{t}}^*) - \hat{\mu}(\boldsymbol{t}), \qquad \tau(\boldsymbol{t}) := \mu(\boldsymbol{t}^*) - \mu(\boldsymbol{t}).$$

Similarly, one can construct the variance estimate $\hat{\Psi}(\boldsymbol{t};\zeta)$ of the virtual $\zeta(\boldsymbol{t})$.

First, we can easily establish the asymptotic normality of $\hat{\zeta}(\boldsymbol{t})$ following Theorem 3 in Farrell et al. (2020),

$$\sqrt{n}\big(\hat{\Psi}(\boldsymbol{t};\zeta)\big)^{-1/2}(\hat{\zeta}(\boldsymbol{t}) - \zeta(\boldsymbol{t})) = \sum_{i=1}^{n}\big(\hat{\Psi}(\boldsymbol{t};\zeta)\big)^{-1/2}\big(\psi(\boldsymbol{z}_i,\boldsymbol{\theta}^*(\boldsymbol{x}_i),\boldsymbol{\Lambda}(\boldsymbol{x}_i),\boldsymbol{t}^*,\boldsymbol{t}) - \zeta(\boldsymbol{t})\big)/\sqrt{n} + o_p(1) \to_d \mathcal{N}(0,1). \tag{27}$$

Next, we aim at showing with probability going to 1, the optimal treatment combination is correctly identified i.e., $\lim_{n\to\infty}\mathbb{P}[\hat{\boldsymbol{t}}^* = \boldsymbol{t}^*] = 1$, with a little abuse use of notation because $\hat{\boldsymbol{t}}^*$ depends on sample size n, and can be written as $\hat{\boldsymbol{t}}_n^*$ more rigorously. Since it does not cause any confusion, we keep using $\hat{\boldsymbol{t}}^*$ for simplicity.

Note that we have $\hat{\mu}(\boldsymbol{t}^*) \to_p \mu(\boldsymbol{t}^*)$, as well as $\hat{\mu}(\boldsymbol{t}) \to_p \mu(\boldsymbol{t})$ for any $\boldsymbol{t} \in \{0,1\}^m$ $\boldsymbol{t}^*$. So we have $\big(\hat{\mu}(\boldsymbol{t}^*), \hat{\mu}(\boldsymbol{t})\big) \to_p \big(\mu(\boldsymbol{t}^*), \mu(\boldsymbol{t})\big)$. Then by continuous mapping theorem (max operator is L_1 norm), we have,

$$\max\Big(|\hat{\mu}(\boldsymbol{t}^*) - \mu(\boldsymbol{t}^*)|, |\hat{\mu}(\boldsymbol{t}) - \mu(\boldsymbol{t})|\Big) \to_p 0,$$

which means with probability going to 1,

$$\max\Big(|\hat{\mu}(\boldsymbol{t}^*) - \mu(\boldsymbol{t}^*)|, |\hat{\mu}(\boldsymbol{t}) - \mu(\boldsymbol{t})|\Big) \le \frac{\mu(\boldsymbol{t}^*) - \mu(\boldsymbol{t})}{4}, \tag{28}$$

due to the assumption of the unique optimal $\boldsymbol{t}^*$.

Furthermore, Eqn. (28) implies,

$$\hat{\mu}(\boldsymbol{t}^*) - \hat{\mu}(\boldsymbol{t}) = [\mu(\boldsymbol{t}^*) - (\mu(\boldsymbol{t}^*) - \hat{\mu}(\boldsymbol{t}^*))] - [\mu(\boldsymbol{t}) - (\mu(\boldsymbol{t}) - \hat{\mu}(\boldsymbol{t}))] \ge \frac{\mu(\boldsymbol{t}^*) - \mu(\boldsymbol{t})}{2} > 0.$$

Therefore, with the probability going to 1, we have $\hat{\mu}(\boldsymbol{t}^*) > \hat{\mu}(\boldsymbol{t})$. Taking union bound over $(2^m - 1)$ treatment combinations, we have $\hat{\mu}(\boldsymbol{t}^*) > \max_{\boldsymbol{t}\neq\boldsymbol{t}^*}\hat{\mu}(\boldsymbol{t})$, which means with the probability going to 1, $\hat{\boldsymbol{t}}^* = \arg\max_{\boldsymbol{t}}\hat{\mu}(\boldsymbol{t}) = \boldsymbol{t}^*$.

Note that $\hat{\boldsymbol{t}}^* = \boldsymbol{t}^*$ simply implies that $\sqrt{n}\big(\hat{\Psi}(\boldsymbol{t};\zeta)\big)^{-1/2}(\hat{\zeta}(\boldsymbol{t}) - \zeta(\boldsymbol{t})) - \sqrt{n}\big(\hat{\Psi}(\boldsymbol{t};\tau)\big)^{-1/2}(\hat{\tau}(\boldsymbol{t}) - \tau(\boldsymbol{t})) = 0$ by the construction. Then, we have that for any $\varepsilon > 0$, $\lim_{n\to\infty}\mathbb{P}[|\sqrt{n}\big(\hat{\Psi}(\boldsymbol{t};\zeta)\big)^{-1/2}(\hat{\zeta}(\boldsymbol{t}) - \zeta(\boldsymbol{t})) - \sqrt{n}\big(\hat{\Psi}(\boldsymbol{t};\tau)\big)^{-1/2}(\hat{\tau}(\boldsymbol{t}) - \tau(\boldsymbol{t}))| < \varepsilon] = 1$.

Combing the fact $\sqrt{n}\big(\hat{\Psi}(\boldsymbol{t};\zeta)\big)^{-1/2}(\hat{\zeta}(\boldsymbol{t}) - \zeta(\boldsymbol{t})) - \sqrt{n}\big(\hat{\Psi}(\boldsymbol{t};\tau)\big)^{-1/2}(\hat{\tau}(\boldsymbol{t}) - \tau(\boldsymbol{t})) \to_p 0$ with the asymptotic normality result (27), by Slutsky's theorem, we have,

$$\sqrt{n}\big(\hat{\Psi}(\boldsymbol{t};\tau)\big)^{-1/2}(\hat{\tau}(\boldsymbol{t}) - \tau(\boldsymbol{t})) = \sum_{i=1}^{n}\big(\hat{\Psi}(\boldsymbol{t};\tau)\big)^{-1/2}\big(\psi(\boldsymbol{z}_i,\boldsymbol{\theta}^*(\boldsymbol{x}_i),\boldsymbol{\Lambda}(\boldsymbol{x}_i),\boldsymbol{t}^*,\boldsymbol{t}) - \tau(\boldsymbol{t})\big)/\sqrt{n} + o_p(1) \to_d \mathcal{N}(0,1),$$

which finishes the proof. □

Given the asymptotic normality, the $(1-\alpha)$-confidence interval for $\hat{\tau}_{\mathsf{DeDL}}(\boldsymbol{t})$ is given by

$$\widehat{\mathcal{CI}}_{\mathsf{DeDL}}(\boldsymbol{t};\tau) = \left[\hat{\tau}_{\mathsf{DeDL}}(\boldsymbol{t}) - \frac{1}{\sqrt{n}}\cdot\Phi^{-1}\left(1-\frac{\alpha}{2}\right)\cdot\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\tau), \hat{\tau}_{\mathsf{DeDL}}(\boldsymbol{t}) + \frac{1}{\sqrt{n}}\cdot\Phi^{-1}\left(1-\frac{\alpha}{2}\right)\cdot\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t};\tau)\right]. \tag{29}$$

Appendix B: Empirical Analysis with Field Experiment Data

B.1. User Covariates Used in Section 4

Table 5 present all the user covariates used in our empirical analysis in Section 4.

Table 5 User Covariates Used in the DeDL Framework

	Variable	Description
Discrete Variables	Age Range	The age range of user: young, mid-age, senior, old, or unknown
	Gender	The gender of user: male, female, or unknown
	Operating System	The OS of user's device: Android, IOS, or other
	Product Version	The version used: lite version, express version, or normal version
	Feed Model	The preferred mode of user: video stream, video cover stream, or unknown
	Phone Price Range	The price range of user's device: luxury, expensive, affordable, or unknown
	User Activeness Degree	The activeness of user: high-active, middle-active, low-active, or new user
	User Active-Deepness Degree	The active-deepness of user: deep-active, low-active, or new user
	Number of Followers Interval	Interval of user's followers: <10, 10 - 10k, 10k - 100k, >100k
	Number of Followers Range	Range of user's followers: amateur, medium producer, or premium producer
	Number of Mutual Followers Interval	Interval of user's friends: <10, 10 - 10k, 10k - 100k, >100k
	Number of Mutual Followers Range	Range of user's friends: low-sociable, middle-sociable, or high-sociable
	Frequent Residence Area	The region area in which the user is frequent on the platform: South, North, or unknown
	Frequent Residence City Level	The level of city in which the user is frequent on the platform: large city, big city, medium city, small city, or unknown
	Frequent Residence City Type	The type of city in which the user is frequent on the platform: city, town, rural, or unknown
	Active Engagement City Level	The level of city in which the user is always active: large city, big city, medium city, small city, or unknown
Continuous Variables	Average App Usage Duration	User's average usage duration on platform per day
	Average Video Watching Time	User's average time on watching videos on platform per day
	Average Live Watching Time	User's average time on watching live on platform per day
	Average DP Video Watching Time	User's average time on watching videos on Discover Page per day
	Average LP Video Watching Time	User's average time on watching videos on Live Page per day
	Average FYP Video Watching Time	User's average time on watching videos on For You Page per day
	Average FP Video Watching Time	User's average time on watching videos on Following Page per day
	Average DP Screen Time	User's average time on Discover Page per day
	Average LP Screen Time	User's average time on Live Page per day
	Average FP Screen Time	User's average time on Following Page per day

B.2. Stratified Sampling

A three-step stratified sampling is employed to keep users' covariates balanced with respect to eight different treatment combinations. Firstly, we categorize 10 continuous variables in Table 5 by their quantile intervals $[0\%, 25\%)$, $[25\%, 50\%)$, $[50\%, 75\%)$, and $[75\%, 100\%]$. Specifically, we assign 1, 2, 3, and 4 as new values for values in each quantile interval respectively for each continuous variable. After all the variables are discretized, we proceed to divide the population into subpopulations according to imbalanced covariates. To check for diversely distributed covariates, we utilize a pairwise T-test between baseline combination $(0, 0, 0)$ and 7 treatment combinations for covariates in Table 5. Among 26 covariates, 16 discrete covariates show no significant difference between baseline combination $(0, 0, 0)$ and seven treatment combinations while 10 continuous variables are imbalanced distributed. Therefore, we divide users into 69,111 strata by the value of imbalanced covariates. Namely, users are grouped if they have the same value in all imbalanced covariates. Last, we perform random sampling. In each stratum, we set the minimum number of users among all treatment combinations as the target size and then randomly sample the target size of individuals in each

combination as the stratified sample. Therefore, the stratified sample has a similar number of users in each treatment combination, and we can assume the treatment assignment mechanism is $\mathbb{P}[T_i = 1] = \mathbb{P}[T_i = 0] = 0.5$ $(i = 1, 2, 3)$.

B.3. Implementation Details of DeDL

In this section, we present the implementation details of using DeDL to estimate ATE. We use the 4-fold cross-fitting proposed in Chernozhukov et al. (2018) to obtain estimators. We randomly partition the observed data of five observed treatment combinations with 1,291,652 data points into four folds $(I_s)_{s=1}^4$ such that each fold has approximately 322,913 data points. For each fold index s, we follow the steps described below to obtain an estimated ATE for each treatment combination. First, we use the other three folds $(I_i)_{i\in\{1,2,3,4\}\setminus s}$ as training data to fit into structured DNN (see Figure 5) and get an estimator of the unknown parameters $\hat{\boldsymbol{\theta}}_s(\cdot)$. Then, for each treatment combination $\boldsymbol{t}$, we use fold (I_s) to construct the estimator $\hat{\mu}_s(\hat{\boldsymbol{t}})$ following Equation 23. Specifically, for each treatment combination $\boldsymbol{t} \in \mathcal{T}$, we use the trained structured DNN to predict non-debiased potential outcome defined as $H(\boldsymbol{x}, \hat{\theta}_s(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0) := G(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}), \boldsymbol{t}) - G(\hat{\boldsymbol{\theta}}_{\boldsymbol{s}}(\boldsymbol{x}), \boldsymbol{t}_0)$ given user covariates $\boldsymbol{x}$ in fold (I_s). We continue substituting $\boldsymbol{x}$ in fold (I_s) to the unknown parameters $\hat{\boldsymbol{\theta}}_s(\cdot)$ obtained by three other folds to calculate the debiased term as the right part of the right side in Equation 9 so that we have $\psi(\boldsymbol{z}, \hat{\boldsymbol{\theta}}_s, \hat{\boldsymbol{\Lambda}}_s; \boldsymbol{t}, \boldsymbol{t}_0)$ for each data point $\boldsymbol{z}$ in fold (I_s). By averaging $\psi(\boldsymbol{z}, \hat{\boldsymbol{\theta}}_s, \hat{\boldsymbol{\Lambda}}_s; \boldsymbol{t}, \boldsymbol{t}_0)$, we obtain estimated ATE of treatment combination $\boldsymbol{t}$ in fold index s as $\hat{\mu}_s(\boldsymbol{t})$. Results of estimated ATE and 95%-confidence interval for each treatment combination in each fold are presented in the first four rows in each section in Table 6. After 4-fold cross-fitting, we aggregate $\hat{\mu}_s(\hat{\boldsymbol{t}})$ by taking their average value as the final estimator $\hat{\mu}(\hat{\boldsymbol{t}})$ for each treatment combination $\boldsymbol{t}$. We report the results of the final estimator in the last row of each section in Table 6.

The estimators for best-arm identification are obtained through similar implementation procedures. Each fold will generate its estimators for best-arm $\tau(\boldsymbol{t})$ regarding treatment combination $\boldsymbol{t}$ for $\boldsymbol{t} \in \mathcal{T}$. Aggregating the estimators from 4 folds generates the DeDL estimator for best-arm identification.

Appendix C: Implementation Details of Benchmarks

Linear Addition (LA) Estimator

To obtain LA estimator for each treatment level, we use ATE of five observed treatment combinations to calculate ATE of unobserved treatment combinations. For observed treatment combinations, their estimated ATE and significant level are the same as the ground truth. For unobserved treatment combinations, we use the linear addition of ATE of individual observed treatment combinations as the estimated ATE. Furthermore, we estimate the standard error of estimated ATE for unobserved treatment combinations by assuming the estimators for individual experiments are independent. We report the LA estimators in the first row in each section of treatment combination in Table 7. The top four treatment combinations are observable and therefore LA estimator yields zero error (see column (5-7) in Table 7). The estimators for ATE of the bottom three treatment combinations use the ATE of individual treatment combinations, i.e. $(0, 0, 1)$, $(0, 1, 0)$, and $(1, 0, 0)$ to calculate the final results $\hat{\mu}_{\boldsymbol{t}^*} = \hat{\mu}_{\boldsymbol{t}_1+\boldsymbol{t}_2} = \mu(\boldsymbol{t}_1) + \mu(\boldsymbol{t}_2)$. The standard deviation of the estimator follows $\hat{\sigma}(\hat{\mu}_{\boldsymbol{t}^*}) = \sqrt{\sigma(\mu(\boldsymbol{t}_1)) + \sigma(\mu(\boldsymbol{t}_2))}$.

Table 6 Detailed Results of 4-fold DeDL Estimators

Treatment Combination	Ground-Truth ATE (1)	Fold (2)	Estimated ATE (3)	95%-Confidence Interval for ATE Estimate (4)	APE (5)	SE (6)	AE (7)
(0, 0, 1)	1.091%**	1	0.974%	[0.484%, 1.464%]	10.76%	1.379	1.174
		2	1.277%	[0.780%, 1.774%]	17.01%	3.448	1.857
		3	1.337%	[0.895%, 1.780%]	22.55%	6.054	2.461
		4	1.045%	[0.552%, 1.538%]	4.22%	0.212	0.461
		Mean	1.158%	[0.678%, 1.639%]	6.14%	0.450	0.671
(0, 1, 0)	-0.267%	1	-0.117%	[−0.611%, 0.376%]	NA	2.242	1.174
		2	-0.358%	[−0.845%, 0.128%]	NA	0.835	0.914
		3	0.038%	[−0.399%, 0.475%]	NA	9.312	3.052
		4	-0.627%	[−1.125%, −0.129%]	NA	12.973	3.602
		Mean	-0.266%	[−0.745%, 0.212%]	NA	0.000	0.008
(1, 0, 0)	0.758%*	1	0.707%	[0.200%, 1.214%]	6.73%	0.260	0.510
		2	0.761%	[0.271%, 1.252%]	0.44%	0.000	0.033
		3	0.986%	[0.546%, 1.427%]	30.13%	5.216	2.284
		4	0.747%	[0.250%, 1.244%]	1.47%	0.012	0.111
		Mean	0.800%	[0.317%, 1.284%]	5.59%	0.180	0.424
(1, 1, 1)	2.121%****	1	2.457%	[1.785%, 3.128%]	15.84%	11.288	3.360
		2	2.304%	[1.630%, 2.978%]	8.63%	3.353	1.831
		3	1.254%	[0.630%, 1.878%]	40.85%	75.044	8.662
		4	2.457%	[1.788%, 3.127%]	15.88%	11.341	3.368
		Mean	2.118%	[1.458%, 2.778%]	0.12%	0.00	0.026
(1, 1, 0)	0.689%	1	-0.616%	[−2.801%, 1.568%]	NA	170.299	13.050
		2	0.900%	[0.259%, 1.541%]	NA	4.451	2.110
		3	0.640%	[0.136%, 1.271%]	NA	0.021	0.146
		4	0.376%	[−1.067%, 1.819%]	NA	9.766	3.125
		Mean	0.341%	[−0.868%, 1.550%]	NA	12.108	3.480
(1, 0, 1)	2.299%****	1	2.396%	[1.075%, 3.716%]	4.19%	0.926	0.962
		2	2.830%	[2.086%, 3.573%]	23.07%	28.129	5.304
		3	2.137%	[1.561%, 2.714%]	7.04%	2.619	1.619
		4	1.899%	[0.435%, 3.362%]	17.42%	16.048	4.006
		Mean	2.315%	[1.289%, 3.341%]	0.70%	0.026	0.160
(0, 1, 1)	1.387%***	1	0.702%	[−0.659%, 2.062%]	49.39%	46.919	6.850
		2	2.127%	[1.265%, 2.989%]	53.41%	54.849	7.406
		3	1.168%	[0.597%, 1.739%]	15.79%	4.792	2.189
		4	1.395%	[0.296%, 2.493%]	0.58%	0.00	0.080
		Mean	1.348%	[0.375%, 2.321%]	2.80%	0.151	0.388

Note: The calculation of APE, SE, and AE is based on the scaled outcome variable (see Column (1) of this table). The significance levels are encoded as *p<0.05; **p<0.01; ***p<0.001; ****p<0.0001. SE is scaled by multiplying a constant. AE is scaled by multiplying another constant.

Linear Regression (LR) Estimator

LR estimator uses the regression coefficients as the estimated ATE. The regression is defined as:

$$y = \beta_1 T_1 + \beta_2 T_2 + \beta_3 T_3 + \boldsymbol{\alpha}\boldsymbol{x},$$

where T_1, T_2, and T_3 denote for ExpT, ExpN, and ExpR, respectively, and $\boldsymbol{x}$ denotes for user's characteristics in Table B.1 after getting dummy variable. 1,291,652 data points in five observed treatment combinations are used to obtain the estimators for β_1, β_2, and β_3. Similar to LA estimators, LR estimators assume the linear addition of ATE of treatment combinations with respect to ATE of the individual experiment. There are two ways of utilizing coefficients obtained from linear regression. We can either use the $\hat{\beta}_i$ as the estimated ATE for individual experiment i then add the corresponding $\hat{\beta}_i$ of the individual experiment for ATE of treatment

combinations, or we can use the trained linear model to predict potential outcome y given user covariates $\boldsymbol{x}$ under treatment combination $\boldsymbol{t}$ for $\boldsymbol{t} \in \mathcal{T}$, and then use pair-wise t-test to obtain the final estimated ATE for each treatment combination. Both ways will yield the same estimators. To maintain the fairness of comparing the estimators of other benchmarks, a 4-fold crossing fitting is used here. Each time, only 75% of training data will be utilized to train the linear regression model. The average value of estimated ATEs is presented in the second row in each section of treatment combination in Table 7.

Pure Deep Learning (PDL) Estimator

4-fold cross-fitting is implemented to reduce the overfitting problem of PDL estimators. Similar to the implementation of DeDL approach in Appendix B.3, we evenly partition the observed data of five treatment combinations into four random folds. For each fold index, we use the data from three other folds as training data to approximate $y = G(\boldsymbol{x}, \boldsymbol{t})$ in a deep learning model. The structure of this deep learning model is a 3-layer DNN with 20 nodes in each layer. We call it "pure" deep learning model since it exerts zero constraints on the form of function G. The DNN in each fold is trained with Adam optimizer and mean squared error loss as loss function. We then use trained DNN to predict the potential outcomes under 8 treatment combinations $\boldsymbol{t}$ for $\boldsymbol{t} \in \mathcal{T}$ given user's covariates $\boldsymbol{x}$ in the fold and conduct the pair-wise t-test between treatment combination $\boldsymbol{t}$ and baseline combination $\boldsymbol{t}_0$ to obtain estimated ATE and its corresponding standard deviation. Taking the average value of estimated ATEs and squared error of the estimator in four folds provide the final PDL estimators and their squared errors. The third row in each section of treatment combination in Table 7 shows the estimated ATE of PDL approach.

Structured Deep Learning (SDL) Estimator

Following the same implementation procedure of 4-fold cross-fitting as DeDL approach, the SDL estimator differs from DeDL estimator in the score function. Instead of using the Neyman orthogonal score function $\psi(\boldsymbol{z}, \hat{\boldsymbol{\theta}}_s, \hat{\boldsymbol{\Lambda}}_s; \boldsymbol{t}, \boldsymbol{t}_0)$ with a debiased term, SDL estimator use the original non-debiased potential outcome which is defined as $H(\boldsymbol{x}, \hat{\theta}_s(\boldsymbol{x}); \boldsymbol{t}, \boldsymbol{t}_0) := G(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}), \boldsymbol{t}) - G(\hat{\boldsymbol{\theta}}_{\boldsymbol{s}}(\boldsymbol{x}), \boldsymbol{t}_0)$. To derive the variance of SDL estimator, we use the traditional pair-wise t-test such that $\widehat{Var}(\hat{\mu}_{\mathsf{SDL}}(\boldsymbol{t}^*)) = \frac{1}{n-1}\sum_{i=i}^{n}(G(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}), \boldsymbol{t}^*) - \overline{G(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}), \boldsymbol{t}^*)})^2 + \frac{1}{n-1}\sum_{i=i}^{n}(G(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}), \boldsymbol{t}_0) - \overline{G(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}), \boldsymbol{t}_0)})^2$. Note that this is an underestimation of the true variance as we neglect noise in $G(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}), \boldsymbol{t}^*)$. Detailed results of benchmark estimators are presented in the last row of each treatment combination section in Table 7.

Appendix D: Complete Results of Synthetic Experiments

D.1. Complete Results of Experiments in Subsection 5.1

We document the complete simulation results of Section 5.1 in Table 8 and Table 9. We evaluate the performance of estimators with an increasing number of A/B tests, i.e., $m \in \{4, 6, 8, 10\}$, and report corresponding results in Panel A to D respectively. The first column "Estimator" describes which estimator is tested out. The second column "CDR" shows the proportion of treatment combinations whose estimated ATE significance levels and signs are consistent with the true ATE. The third column "MAPE" gives the mean absolute percentage error of ATE estimates over all treatment combinations whose real average treatment effects are significant. In other words, we rule out insignificant treatment combinations when calculating MAPE.

Table 7 Detailed Results of Benchmark Estimators

Treatment Combination	Ground-Truth ATE (1)	Estimator (2)	Estimated ATE (3)	CD (4)	APE (5)	SE (6)	AE (7)
(0, 0, 1)	1.091%**	LA	1.091%	1	0.00%	0.000	0.000
		LR	1.329%	1	21.79%	5.657	2.379
		PDL	1.247%	1	14.24%	2.417	1.555
		SDL	1.179%	1	8.00%	0.763	0.873
(0, 1, 0)	-0.267%	LA	-0.267%	1	NA	0.000	0.000
		LR	-0.013%	1	NA	6.460	2.542
		PDL	-0.036%	0	NA	5.353	2.314
		SDL	-0.072%	0	NA	3.792	1.947
(1, 0, 0)	0.758%*	LA	0.758%	1	0.00%	0.000	0.000
		LR	1.079%	1	42.29%	10.028	3.206
		PDL	1.043%	1	37.60%	8.126	2.851
		SDL	0.978%	1	28.95%	4.816	2.195
(1, 1, 1)	2.121%****	LA	2.121%	1	0.00%	0.000	0.000
		LR	2.395%	1	12.95%	7.539	2.746
		PDL	2.326%	1	9.67%	4.209	2.052
		SDL	2.040%	1	3.78%	0.642	0.801
(1, 1, 0)	0.689%	LA	0.491%	1	NA	3.902	1.975
		LR	1.066%	0	NA	14.233	3.773
		PDL	1.030%	0	NA	11.625	3.410
		SDL	0.902%	0	NA	4.543	2.132
(1, 0, 1)	2.299%****	LA	1.850%	1	19.56%	2.023	4.498
		LR	2.408%	1	4.72%	1.178	1.085
		PDL	2.333%	1	1.46%	0.112	0.336
		SDL	2.148%	1	6.59%	2.297	1.516
(0, 1, 1)	1.387%***	LA	0.824%	0	40.56%	3.163	5.624
		LR	1.316%	1	5.08%	0.495	0.704
		PDL	1.217%	1	12.26%	2.890	1.700
		SDL	1.070%	1	22.84%	10.030	3.167

Note: The calculation of APE, SE, and AE is based on the scaled outcome variable (see Column (1) of this table). The significance levels are encoded as *p<0.05; **p<0.01; ***p<0.001; ****p<0.0001. SE is scaled by multiplying a constant. AE is scaled by multiplying another constant.

Otherwise, those insignificant treatment effects would result in close-to-zero value in the denominators for APE calculation, resulting in an undesired metric. Similarly, column "MSE" and column "MAE" represent squared error and absolute error, respectively. In column "MSE" and column "MAE", we do not exclude those insignificant combinations. Indeed, unlike MAPE, close-to-zero treatment effects do not cause a problem for MSE and MAE. As a result, MAE and MSE over all combinations can serve as a supplement to MAPE. However, to better understand the scales of MSE and MAE errors, we also report 95% confidence intervals of average absolute treatment effects over all combinations in table notes. Lastly, the column "BAI" represents the Best-Arm Identification result. For each replication of the experiment, we verify whether different methods can successfully identify the best treatment combination. The value in this column shows the proportion of replications in which the best combination is identified.

Table 8 Learning Estimator Validation Result

Panel A: Comparison of different estimators under $m = 4$					
Estimator	CDR	MAPE	MSE	MAE	BAI
LA	92.40% (91.02%, 93.78%)	22.64% (20.10%, 25.17%)	0.070 (0.023, 0.118)	0.123 (0.103, 0.143)	79.5% (73.9%, 85.1%)
LR	95.34% (94.46%, 96.23%)	15.30% (13.56%, 17.04%)	0.038 (0.0179, 0.0582)	0.089 (0.074, 0.105)	82.0% (76.6%, 87.3%)
PDL_s_p	88.59% (87.02%, 90.17%)	68.83% (60.76%, 76.89%)	0.867 (0.572, 1.163)	0.433 (0.378, 0.488)	56.0% (49.1%, 62.9%)
SDL	95.12% (94.17%, 96.08%)	16.12% (14.57%, 17.66%)	0.018 (0.011, 0.024)	0.079 (0.070, 0.087)	90.5% (86.4%, 94.6%)
DeDL	97.53% (96.90%, 98.15%)	7.20% (6.45%, 7.95%)	0.008 (0.004, 0.012)	0.040 (0.033, 0.046)	93.5% (90.1%, 96.9%)
Panel B: Comparison of different estimators under $m = 6$					
Estimator	CDR	MAPE	MSE	MAE	BAI
LA	92.05% (90.70%, 93.40%)	27.35% (23.81%, 30.89%)	0.080 (0.056, 0.105)	0.166 (0.147, 0.186)	73.0% (66.8%, 79.2%)
LR	94.88% (94.10%, 95.67%)	18.42% (16.61%, 20.24%)	0.073 (0.038, 0.108)	0.136 (0.118, 0.155)	75.5% (69.4%, 81.6%)
PDL_s_p	84.23% (82.54%, 85.93%)	111.70% (98.07%, 125.33%)	2.172 (1.736, 2.607)	0.874 (0.793, 0.956)	28.0% (21.7%, 34.3%)
SDL	93.07% (92.17%, 93.96%)	27.22% (25.16%, 29.29%)	0.051 (0.037, 0.064)	0.150 (0.136, 0.163)	68.0% (61.5%, 74.5%)
DeDL	95.28% (94.61%, 95.96%)	13.33% (11.99%, 14.68%)	0.023 (0.012, 0.033)	0.079 (0.069, 0.090)	87.0% (82.3%, 91.7%)
Panel C: Comparison of different estimators under $m = 8$					
Estimator	CDR	MAPE	MSE	MAE	BAI
LA	93.83% (92.74%, 94.92%)	28.37% (24.98%, 31.75%)	0.196 (0.135, 0.256)	0.247 (0.218, 0.276)	65.0% (58.3%, 71.7%)
LR	95.51% (94.87%, 96.14%)	22.49% (20.03%, 24.96%)	0.209 (0.128, 0.289)	0.222 (0.186, 0.258)	75.6% (69.2%, 82.0%)
PDL_s_p	78.89% (77.01%, 80.77%)	140.60% (128.28%, 152.91%)	3.825 (3.059, 4.591)	1.246 (1.136, 1.355)	11.5% (7.0%, 16.0%)
SDL	94.89% (94.25%, 95.53%)	22.18% (20.20%, 24.15%)	0.072 (0.025, 0.120)	0.153 (0.133, 0.174)	67.5% (61.0%, 74.0%)
DeDL	95.53% (94.92%, 96.15%)	13.49% (11.82%, 15.15%)	0.049 (0.007, 0.092)	0.099 (0.079, 0.118)	82.0% (76.6%, 87.4%)
Panel D: Comparison of different estimators under $m = 10$					
Estimator	CDR	MAPE	MSE	MAE	BAI
LA	95.01% (93.96%, 96.05%)	31.54% (24.23%, 38.84%)	0.326 (0.152, 0.500)	0.311 (0.251, 0.371)	58.8% (47.7%, 69.8%)
LR	95.51% (94.87%, 96.14%)	25.28% (21.73%, 28.83%)	0.359 (0.146, 0.572)	0.305 (0.240, 0.369)	62.0% (52.3%, 71.7%)
PDL_s_p	77.27% (75.50%, 79.03%)	163.85% (151.47%, 176.22%)	5.360 (4.527, 6.193)	1.563 (1.448, 1.677)	5.0% (2.0%, 8.0%)
SDL	94.27% (93.21%, 95.33%)	26.37% (22.97%, 29.78%)	0.077 (0.054, 0.100)	0.195 (0.171, 0.220)	56.2% (45.1%, 67.4%)
DeDL	94.54% (93.48%, 95.60%)	16.59% (13.39%, 19.78%)	0.046 (0.026, 0.067)	0.127 (0.103, 0.150)	78.8% (69.6%, 87.9%)

Note: All experiments are replicated 200 times with 95% $\mathcal{CI}$s reported in parentheses. 95% $\mathcal{CI}$s of average absolute treatment effects are $(0.68, 1.06)$, $(1.09, 1.30)$, $(1.45, 1.71)$, and $(1.54, 2.03)$ respectively in Panel A, Panel B, Panel C, and Panel D.

Table 9 Performance of PDL Estimators

Panel A: Comparison of different estimators under $m = 4$					
Estimator	CDR	MAPE	MSE	MAE	BAI
PDL_s_p	88.59% (87.02%, 90.17%)	68.83% (60.76%, 76.89%)	0.867 (0.572, 1.163)	0.433 (0.378, 0.488)	56.0% (49.1%, 62.9%)
PDL_s_a	95.16% (94.27%, 96.04%)	16.01% (14.49%, 17.53%)	0.012 (0.009, 0.016)	0.075 (0.068, 0.083)	78.0% (72.2%, 83.8%)
PDL_l_p	89.12% (87.61%, 90.64%)	60.15% (53.09%, 67.20%)	0.493 (0.375, 0.610)	0.356 (0.319, 0.393)	56.0% (49.1%, 62.9%)
PDL_l_a	96.34% (95.57%, 97.12%)	12.55% (11.20%, 13.89%)	0.006 (0.004, 0.007)	0.052 (0.048, 0.057)	86.5% (81.7%, 91.3%)
Panel B: Comparison of different estimators under $m = 6$					
Estimator	CDR	MAPE	MSE	MAE	BAI
PDL_s_p	84.23% (82.54%, 85.93%)	111.70% (98.07%, 125.33%)	2.172 (1.736, 2.607)	0.874 (0.793, 0.956)	28.0% (21.7%, 34.3%)
PDL_s_a	95.04% (94.43%, 95.65%)	20.03% (18.39%, 21.67%)	0.021 (0.017, 0.024)	0.104 (0.096, 0.113)	70.5% (64.1%, 76.9%)
PDL_l_p	83.87% (82.33%, 85.40%)	111.37% (99.17%, 123.57%)	1.748 (1.357, 2.138)	0.789 (0.712, 0.866)	25.0% (18.9%, 31.1%)
PDL_l_a	95.73% (95.05%, 96.42%)	16.91% (14.93%, 18.89%)	0.015 (0.012, 0.019)	0.085 (0.077, 0.093)	68.0% (61.5%, 74.5%)
Panel C: Comparison of different estimators under $m = 8$					
Estimator	CDR	MAPE	MSE	MAE	BAI
PDL_s_p	78.89% (77.01%, 80.77%)	140.60% (128.28%, 152.91%)	3.825 (3.059, 4.591)	1.246 (1.136, 1.355)	11.5% (7.0%, 16.0%)
PDL_s_a	95.04% (94.43%, 95.65%)	20.03% (18.39%, 21.67%)	0.021 (0.017, 0.024)	0.104 (0.096, 0.113)	70.5% (64.1%, 76.9%)
PDL_l_p	80.80% (79.15%, 82.44%)	125.38% (114.46%, 136.29%)	3.059 (2.347, 3.770)	1.098 (1.001, 1.195)	10.0% (5.8%, 14.2%)
PDL_l_a	95.43% (94.90%, 95.95%)	20.24% (18.36%, 22.13%)	0.035 (0.028, 0.041)	0.137 (0.126, 0.148)	57.5% (50.6%, 64.4%)
Panel D: Comparison of different estimators under $m = 10$					
Estimator	CDR	MAPE	MSE	MAE	BAI
PDL_s_p	77.27% (75.50%, 79.03%)	163.85% (151.47%, 176.22%)	5.360 (4.527, 6.193)	1.563 (1.448, 1.677)	5.0% (2.0%, 8.0%)
PDL_s_a	95.14% (94.69%, 95.59%)	24.87% (22.64%, 27.11%)	0.052 (0.044, 0.061)	0.173 (0.160, 0.185)	43.0% (36.1%, 49.9%)
PDL_l_p	76.03% (73.04%, 79.01%)	159.58% (124.10%, 195.06%)	5.436 (3.533, 7.339)	1.472 (1.224, 1.720)	6.0% (1.2%, 10.8%)
PDL_l_a	95.67% (94.82%, 96.53%)	19.66% (17.00%, 22.31%)	0.033 (0.025, 0.041)	0.140 (0.122, 0.157)	56.0% (41.7%, 70.3%)

Note: All experiments are replicated 200 times with 95% $\mathcal{CI}$s reported in parentheses.

We first compare estimators within each panel in Table 8. We observe the following consistent patterns across all panels. First, DeDL outperforms (or no significant difference) LA, LR, PDL_s_p and SDL under all metrics, which validates our theory and provides strong evidence for the advantage of our method. In particular, DeDL increases the success rate of both CDR and BAI, and decreases MAPE, MSE, and MAE compared to SDL by a significant margin. This demonstrates the value of debiasing in the influence function (5). Second, generally, LA and LR perform worse than SDL, demonstrating the advantage of neural networks over the linear method, despite the fact that that SDL can still be asymptotically biased.

Comparing across different panels, we observe that the performance of all estimators becomes worse when m grows larger. In particular, the performance of BAI decreases fast due to the exponentially increased number of combinations. Even so, DeDL can still successfully identify the best arm among $1,024$ combinations with a relatively high probability of 78.8% when $m = 10$. We may remark on the underlying mechanisms behind these observed degenerating performances. On the one hand, intuitively, due to the sigmoid link function setup, when the number of field experiments gets larger, the treatment effect becomes more nonlinear. It creates difficulty for the LA estimator. Therefore, the performance of LA worsens due to its inherent lack of model richness. On the other hand, with the correct link function specification, one may expect that the performance of SDL and DeDL should be relatively stable because of their modeling power. However, since we fix the same DNN structure with a constant number of 10 hidden nodes across all experiments for a fair comparison with LA, the model complexity of neural networks is limited by design. We have verified offline that we can increase the number of hidden nodes when $m = 6, 8, 10$ to achieve a similar performance as $m = 4$. We skip the detailed discussion here due to the space limit. For this reason, we are confident that the difficulty in estimation and inference from an increased m can be mitigated by increasing the network structure complexity, so without loss of generality, in all experiments in the following subsections, we maintain $m = 4$ for computation efficiency.

We report the performance of PDL estimators in a separate Table 9 due to the bad performance. The subscripts s and l represent different widths of neural nets, with 10 and 40 hidden nodes for hidden layers, resulting in the DNN structure $(d_{\boldsymbol{x}} + d_{\boldsymbol{t}} + 1)$-10-ReLU-10-ReLU-10-ReLU-1 and $(d_{\boldsymbol{x}} + d_{\boldsymbol{t}} + 1)$-40-ReLU-40-ReLU-40-ReLU-1, respectively The subscripts p and a represent the training samples generated from partially observed treatments or all treatments. For a fair comparison to estimators in Table 8, we should focus on PDL_s_p with partially observed treatments and similar DNN size as SDL. The training stopping criteria is set the same as SDL with 0.3 validation loss threshold. Note that with partially observed combinations, both PDL_s_p and PDL_l_p perform worse than all other estimators under all metrics. This is due to the bad performance of the out-of-sample test under the unobserved treatments, rather than the bad approximation ability of DNN. Because one can notice that when we increase the DNN width from 10 to 40, the performance only slightly increases. One can also notice that PDL indeed has much better in-sample tests. Because when we incorporate data from all treatment combinations in the training, the resulting estimators PDL_s_a and PDL_l_a have comparable performance with the best estimator DeDL.

It is within the expectation that PDL with partially observed treatments has such a bad performance because it can only access the data generated by base treatment level, m single experiment data, and only one treatment level with interaction $(1, 1, 0, \ldots, 0)$. It means that it is almost impossible for PDL to learn interaction between different experiments. Unlike LR and SDL, which put the structured form of experimental interaction, PDL only aims at increasing the performance of in-sample outcome prediction, totally ignoring the out-of-sample performance. One may argue whether it is due to the over-fitting of PDL. To answer the question of why PDL performs badly in more detail, we do a further investigation. To simplify the discussion, we conduct an extra synthetic experiment with $m = 3$, and report the result in Figure 12. Each point in scatter plots represents the values of real ATE (x-axis) and predicted ATE (y-axis). From left to right, we visualize

the performance of LR, PDL-base (i.e., PDL_s_p), PDL-dropout (i.e., PDL_s_p with $p = 0.1$ dropout regularizer after each activation layer), and PDL-L1 (i.e., PDL_s_p with L_1 regularization loss over DNN parameters with fine-tuned weight 0.05). Four subfigures in the upper panel show the in-sample performance, while lower panel subfigures show the performance under unobserved treatment combinations. Notice that PDL-base has better in-sample performance than LR as the data points are more concentrated around true line $y = x$, but PDL-base has the worst out-of-sample performance. Although LR assumes linear extrapolation of treatment effects, in our example, most scatter points in the out-of-sample test are still well concentrated around $y = x$ except clear patterns of overestimate when absolute ATE increases. When applying regularizations, PDL-dropout and PDL-L1 have deteriorated in-sample performance with points less concentrated around the true line while improving out-of-sample performance. But we highlight that we generally do not know the out-of-sample ground-truth ATE in practice, which makes it difficult to guide the selection of regularizers. We also try other regularizers, e.g., early stopping, L_2 parameter weights, and smaller network sizes. It is still not comparable to LR. Due to the bad performance of PDL with partially observed treatments, we do not report its performance in the following simulations.

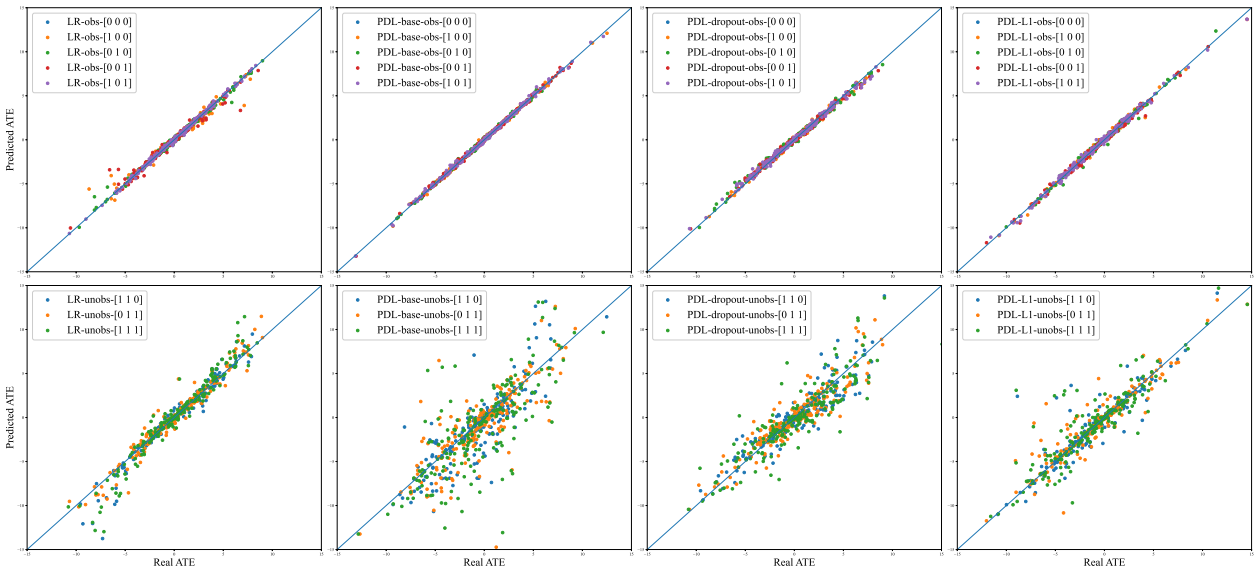


Figure 12 LR and PDL Estimator Comparison

D.2. Complete Results of Experiments in Subsection 5.2

The complete results from this set of experiments are documented in Table 11, the columns of which are the same as Table 8. Notice that estimators LA and LR have the same performance when decreasing the DNN convergence rate with increasing δ. Thus, we report their performance in Panel A separately. When comparing results within panels B, C, and D, we can observe that DeDL significantly improves the performance of SDL by decreasing MSE, and MAE, as well as increasing BAI, while CDR and MAPE are only marginally improved. With increasing δ, SDL and DeDL are getting worse under all performance metrics due to the larger bias. However, even with this being said, DeDL is still significantly better than LA and LR even when

Table 10 Performance of PDL with Regularizers

Estimator	CDR	MAPE	MSE	MAE	BAI
LA	93.56% (92.06%, 95.07%)	27.44% (22.60%, 32.29%)	0.705 (0.504, 0.905)	0.408 (0.359, 0.457)	89.0% (84.6%, 93.4%)
LR	95.50% (94.37%, 96.63%)	26.01% (20.56%, 31.45%)	0.654 (0.449, 0.859)	0.389 (0.335, 0.443)	91.5% (87.6%, 95.4%)
PDL-base	91.62% (90.22%, 93.03%)	52.68% (44.27%, 61.09%)	2.930 (2.104, 3.756)	0.762 (0.675, 0.849)	73.0% (66.8%, 79.2%)
PDL-dropout	90.81% (89.17%, 92.46%)	35.74% (31.43%, 40.04%)	0.967 (0.739, 1.195)	0.504 (0.453, 0.555)	63.5% (56.8%, 70.2%)
PDL-L1	92.00% (90.25%, 93.75%)	33.64% (27.96%, 39.32%)	1.274 (0.513, 2.035)	0.426 (0.354, 0.498)	84.5% (79.4%, 89.6%)

Note: All experiments are replicated 200 times under $m = 3$. 95% $\mathcal{CI}$s reported in parentheses.

$\delta = 0.3$ as shown in Panel D. In summary, DeDL is consistently better than other estimators even with large DNN estimation biases as long as the link function is not misspecified.

Table 11 DNN Convergence Violation Result

Panel A: Performance of Benchmarks					
Estimator	CDR	MAPE	MSE	MAE	BAI
LA	94.56% (93.69%, 95.44%)	19.93% (17.94%, 21.93%)	0.085 (0.066, 0.104)	0.179 (0.162, 0.195)	92.0% (88.2%, 95.8%)
LR	96.88% (96.21%, 97.54%)	13.59% (12.11%, 15.08%)	0.046 (0.036, 0.056)	0.127 (0.115, 0.139)	94.0% (90.7%, 97.3%)
Panel B: Comparison of different estimators under $\delta = 0.1$					
Estimator	CDR	MAPE	MSE	MAE	BAI
SDL	96.97% (96.32%, 97.62%)	4.98% (4.58%, 5.38%)	0.013 (0.010, 0.015)	0.074 (0.066, 0.083)	92.5% (88.8%, 96.2%)
DeDL	98.56% (98.12%, 99.01%)	5.04% (4.32%, 5.76%)	0.003 (0.002, 0.003)	0.031 (0.028, 0.035)	97.0% (94.6%, 99.4%)
Panel C: Comparison of different estimators under $\delta = 0.2$					
Estimator	CDR	MAPE	MSE	MAE	BAI
SDL	94.19% (93.19%, 95.19%)	11.06% (10.28%, 11.83%)	0.061 (0.049, 0.072)	0.170 (0.153, 0.186)	85.5% (80.6%, 90.4%)
DeDL	96.31% (95.50%, 97.12%)	11.10% (9.46%, 12.74%)	0.016 (0.012, 0.020)	0.078 (0.069, 0.087)	95.5% (92.6%, 98.4%)
Panel D: Comparison of different estimators under $\delta = 0.3$					
Estimator	CDR	MAPE	MSE	MAE	BAI
SDL	93.44% (92.48%, 94.39%)	15.53% (14.33%, 16.73%)	0.145 (0.118, 0.173)	0.259 (0.232, 0.286)	80.0% (74.4%, 85.6%)
DeDL	94.63% (93.74%, 95.51%)	12.78% (11.05%, 14.50%)	0.032 (0.024, 0.039)	0.108 (0.095, 0.120)	92.5% (88.8%, 96.2%)

Note: All simulations are replicated 200 times with 95% $\mathcal{CI}$s reported in parentheses. 95% $\mathcal{CI}$s of average absolute treatment effect is $(1.45, 1.65)$ in all Panels.

D.3. Complete Results of Experiments in Subsection 5.3

Table 12 shows the complete simulation results. Comparing across different panels representing different values of γ, one can observe that when the link function gets more misspecified, SDL and DeDL estimators get

worse under all metrics. While the MAPE, MSE, and MAE performances of LA and LR are not significantly deteriorated, considering the increased absolute treatment effects listed in table notes. When the model is not misspecified or marginally misspecified, i.e., $\gamma \in \{0,1,3\}$, SDL and DeDL work no worse than LA in all performance metrics. When $\gamma = 1$, DeDL can only marginally improve the performance of SDL for all metrics. However, when γ is increased to 3, we can observe that DeDL has worse MSE performance than SDL. Even worse, when γ is increased to 5, DeDL is not better than SDL under all performance metrics. In summary, we still recommend using DeDL when $\gamma \in \{0,1,3\}$.

D.4. Imbalanced Covariates

This set of simulations investigates a practical setting where the observed $\boldsymbol{X}$ distribution deviates from the population. We call this setting as *imbalanced covariates*, i.e., $\boldsymbol{X}$ covariates imbalance between any treatment level $\boldsymbol{t} \in \{0,1\}^m$ and population $\boldsymbol{X}$ distribution. The imbalanced covariates affect both training and inference stages, invalidating the DeDL estimators. We discuss how to do rebalancing to get precise and trustworthy estimates of ATEs.

We adopt exactly the same simulation setup to Section 5.2 except that the observed covariates follow a different distribution. More specifically, the last dimension of $\boldsymbol{x}$ follows the exponential distribution with rate $\lambda \in \{2.0, 1.0, 0.5\}$ instead of the uniform distribution $\mathcal{U}(0,1)$. Note that the ground truth ATEs are still calculated using the uniform distributed $\boldsymbol{x}$. We highlight that although this setup is relatively simple because the observed covariates uniformly deviate from the true distribution across all treatment combinations while in practice the observed covariates $\boldsymbol{x}$ may even follow different distributions under different $\boldsymbol{t}$, this simulation result still demonstrates the importance of rebalancing.

To reconcile the imbalanced covariates issue, one can do stratified sampling on sampled units to match the covariate distribution over the population, as we implemented in our empirical study. There are also many re-randomization techniques to improve covariate balance in experiments. We refer interested readers to Morgan and Rubin (2012), Li et al. (2020). But in this simulation study, to keep the discussion simple, we use the same stratified sampling procedure in our empirical study. Specifically, we focus on the imbalanced covariate dimension with exponential distribution. First, we do stratified sampling to only keep the data with $\boldsymbol{x}$ in the support $[0,1]^{d_{\boldsymbol{X}}}$. Then we do stratified sampling to make sure the imbalanced dimension of $\boldsymbol{x}$ is rebalanced in the sense that the sample sizes in $[0,1]^{d_{\boldsymbol{X}}-1} \times [0, 0.5)$ and $[0,1]^{d_{\boldsymbol{X}}-1} \times [0.5, 1]$ are the same. Note that by stratified sampling, we may discard some samples and sacrifice efficiency. We also conducted the stratified sampling with higher accuracy, e.g., the numbers of samples in five buckets with 0.2 bandwidth are the same.

We report the MAEs for different estimators with balanced and imbalanced covariates in Figure 13(a) and (b), respectively. Using the imbalanced covariates for both training and inference, Figure 13(a) reports the comparison results. The complete result is listed in Table 13. After the stratified sampling, we use the rebalanced covariates for both training and inference and show the result in Figure 13(b). $\lambda \in \{2.0, 1.0, 0.5\}$ indicates the growing imbalance level. The key observation is that when the data is too imbalanced with $\lambda \in \{1.0, 0.5\}$, DeDL is not precise. After the stratified sampling, de-biasing is still trustworthy, reducing MAE compared to SDL.

Table 12 Model Misspecification Result

Panel A: Comparison of different estimators under $\gamma = 0$					
Estimator	CDR	MAPE	MSE	MAE	BAI
LA	94.78% (93.80%, 95.76%)	20.69% (18.30%, 23.08%)	0.074 (0.056, 0.091)	0.167 (0.151, 0.182)	94.5% (91.3%, 97.7%)
LR	97.38% (96.81%, 97.94%)	12.81% (11.53%, 14.08%)	0.043 (0.033, 0.053)	0.125 (0.112, 0.138)	94.5% (91.3%, 97.7%)
SDL	97.71% (97.18%, 98.26%)	10.07% (8.92%, 11.21%)	0.014 (0.010, 0.018)	0.075 (0.068, 0.081)	96.5% (93.9%, 99.1%)
DeDL	99.28% (98.99%, 99.57%)	3.23% (2.68%, 3.78%)	0.002 (0.001, 0.003)	0.024 (0.021, 0.027)	99.0% (97.6%, 100.4%)
Panel B: Comparison of different estimators under $\gamma = 1$					
Estimator	CDR	MAPE	MSE	MAE	BAI
LA	95.56% (94.77%, 96.35%)	21.76% (19.29%, 24.23%)	0.091 (0.069, 0.112)	0.189 (0.171, 0.206)	87.5% (82.9%, 92.1%)
LR	97.38% (96.79%, 97.96%)	12.85% (11.45%, 14.24%)	0.042 (0.033, 0.052)	0.124 (0.113, 0.136)	88.0% (83.5%, 92.5%)
SDL	97.22% (96.57%, 97.86%)	11.55% (10.40%, 12.70%)	0.027 (0.019, 0.035)	0.100 (0.091, 0.109)	90.5% (86.4%, 94.6%)
DeDL	98.81% (98.33%, 99.29%)	5.67% (4.87%, 6.47%)	0.014 (0.008, 0.020)	0.056 (0.049, 0.064)	92.5% (88.8%, 96.2%)
Panel C: Comparison of different estimators under $\gamma = 3$					
Estimator	CDR	MAPE	MSE	MAE	BAI
LA	94.62% (93.71%, 95.54%)	18.07% (16.18%, 19.96%)	0.112 (0.075, 0.148)	0.210 (0.190, 0.229)	89.0% (84.6%, 93.4%)
LR	97.09% (96.46%, 97.73%)	12.10% (10.77%, 13.43%)	0.050 (0.036, 0.063)	0.135 (0.122, 0.149)	88.0% (83.5%, 92.5%)
SDL	97.00% (96.33%, 97.67%)	15.10% (13.54%, 16.66%)	0.123 (0.088, 0.159)	0.199 (0.180, 0.218)	90.5% (86.4%, 94.6%)
DeDL	97.50% (96.77%, 98.22%)	13.44% (10.87%, 16.00%)	0.151 (0.096, 0.207)	0.181 (0.156, 0.207)	92.5% (88.8%, 96.2%)
Panel D: Comparison of different estimators under $\gamma = 5$					
Estimator	Significance	MAPE	MSE	MAE	BAI
LA	95.34% (94.47%, 96.22%)	19.93% (17.12%, 22.74%)	0.155 (0.125, 0.185)	0.263 (0.241, 0.285)	88.0% (83.5%, 92.5%)
LR	96.97% (96.21%, 97.73%)	12.05% (10.28%, 13.81%)	0.077 (0.053, 0.102)	0.170 (0.153, 0.187)	92.5% (88.8%, 96.2%)
SDL	95.90% (95.09%, 96.72%)	17.24% (15.21%, 19.26%)	0.241 (0.183, 0.299)	0.277 (0.251, 0.304)	88.5% (84.0%, 93.0%)
DeDL	95.46% (94.43%, 96.50%)	23.20% (18.31%, 28.09%)	0.559 (0.375, 0.743)	0.333 (0.284, 0.381)	84.5% (79.4%, 89.6%)

Note: All simulations are replicated 200 times with 95% $\mathcal{CI}$s reported in parentheses. 95% $\mathcal{CI}$s of average absolute treatment effects are $(1.45, 1.65)$, $(1.52, 1.72)$, $(2.06, 2.36)$, and $(2.71, 3.07)$ respectively in Panel A, Panel B, Panel C, and Panel D.

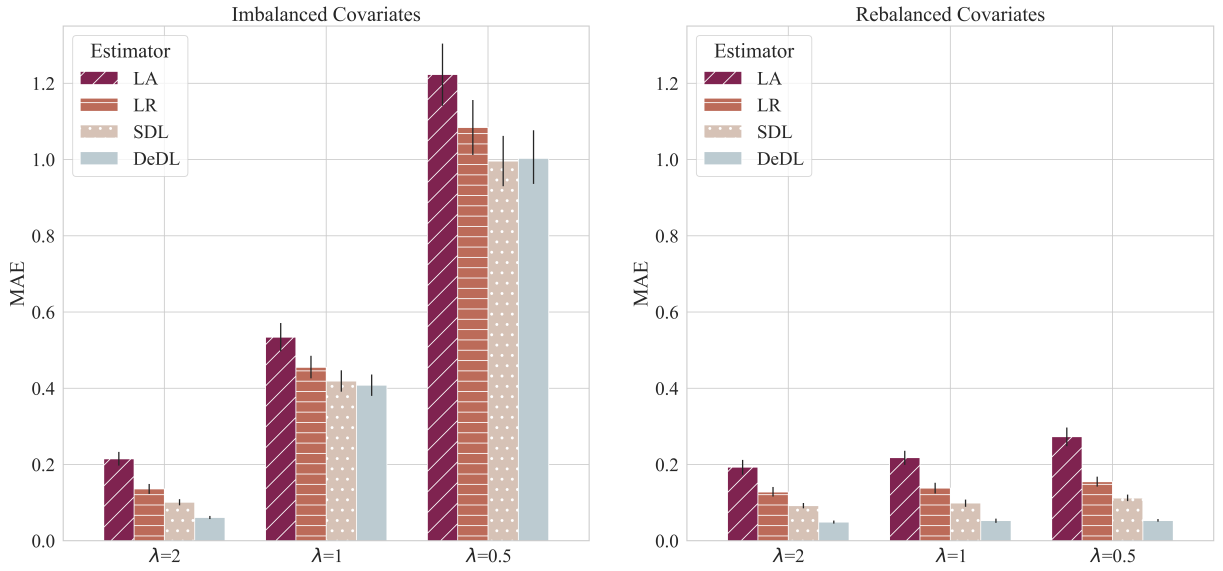


Figure 13 **MAE comparison among estimators under the imbalanced covariates setting. Panel (a) shows the performance of** LA,LR,SDL, **and** DeDL **before rebalancing. Panel (b) presents the performance after rebalancing.**

Table 13 Imbalanced Covariates Result

Estimator	CRD	MAPE	MSE	MAE	BAI
Panel A: Comparison of different estimators under $\lambda = 2.0$					
	Before Covariates Rebalancing				
LA	91.22% (89.99%, 92.45%)	32.03% (28.21%, 35.85%)	0.104 (0.081, 0.127)	0.215 (0.197, 0.233)	87.5% (82.9%, 92.1%)
LR	96.00% (95.32%, 96.68%)	18.92% (16.87%, 20.97%)	0.047 (0.036, 0.059)	0.136 (0.123, 0.149)	91.0% (87.0%, 95.0%)
SDL	96.12% (95.42%, 96.83%)	15.84% (14.09%, 17.60%)	0.024 (0.018, 0.031)	0.101 (0.093, 0.109)	95.0% (92.0%, 98.0%)
DeDL	96.44% (95.74%, 97.13%)	11.10% (9.71%, 12.48%)	0.007 (0.006, 0.008)	0.061 (0.057, 0.065)	95.0% (92.0%, 98.0%)
	After Covariates Rebalancing				
LA	93.94% (92.85%, 95.02%)	27.24% (23.10%, 31.38%)	0.102 (0.074, 0.129)	0.193 (0.173, 0.212)	92.0% (88.2%, 95.8%)
LR	96.44% (95.75%, 97.13%)	18.36% (15.69%, 21.03%)	0.044 (0.033, 0.056)	0.128 (0.116, 0.141)	93.5% (90.1%, 96.9%)
SDL	96.62% (95.95%, 97.30%)	14.47% (12.66%, 16.28%)	0.019 (0.015, 0.023)	0.092 (0.085, 0.099)	96.0% (93.3%, 98.7%)
DeDL	97.66% (97.11%, 98.20%)	7.86% (6.66%, 9.05%)	0.006 (0.004, 0.007)	0.049 (0.045, 0.053)	98.0% (96.0%, 100.0%)
Panel B: Comparison of different estimators under $\lambda = 1.0$					
	Before Covariates Rebalancing				
LA	87.84% (86.47%, 89.21%)	82.23% (73.27%, 91.18%)	0.519 (0.445, 0.593)	0.534 (0.497, 0.571)	71.5% (65.2%, 77.8%)
LR	90.44% (89.36%, 91.51%)	70.13% (62.66%, 77.60%)	0.371 (0.321, 0.420)	0.455 (0.426, 0.485)	72.0% (65.7%, 78.3%)
SDL	90.53% (89.41%, 91.66%)	71.28% (63.36%, 79.20%)	0.304 (0.260, 0.348)	0.419 (0.391, 0.447)	76.0% (70.0%, 82.0%)
DeDL	90.59% (89.51%, 91.68%)	68.73% (61.16%, 76.30%)	0.291 (0.248, 0.334)	0.408 (0.380, 0.436)	75.5% (69.5%, 81.5%)
	After Covariates Rebalancing				
LA	91.44% (90.29%, 92.58%)	32.66% (28.67%, 36.66%)	0.109 (0.086, 0.133)	0.218 (0.199, 0.236)	88.0% (83.5%, 92.5%)
LR	95.84% (95.17%, 96.52%)	20.03% (16.01%, 24.04%)	0.056 (0.036, 0.076)	0.138 (0.124, 0.152)	94.5% (91.3%, 97.7%)
SDL	96.00% (95.23%, 96.77%)	15.04% (12.57%, 17.51%)	0.027 (0.018, 0.036)	0.099 (0.089, 0.108)	96.0% (93.3%, 98.7%)
DeDL	96.47% (95.84%, 97.10%)	9.16% (7.23%, 11.08%)	0.008 (0.004, 0.012)	0.053 (0.047, 0.058)	96.0% (93.3%, 98.7%)
Panel C: Comparison of different estimators under $\lambda = 0.5$					
	Before Covariates Rebalancing				
LA	78.41% (76.31%, 80.51%)	186.87% (167.63%, 206.10%)	2.644 (2.289, 2.999)	1.223 (1.141, 1.304)	43.5% (36.6%, 50.4%)
LR	80.84% (78.92%, 82.77%)	167.21% (150.48%, 183.94%)	2.097 (1.811, 2.383)	1.084 (1.012, 1.156)	47.0% (40.0%, 54.0%)
SDL	81.50% (79.51%, 83.49%)	162.02% (144.89%, 179.15%)	1.737 (1.503, 1.972)	0.996 (0.930, 1.062)	51.0% (44.0%, 58.0%)
DeDL	81.34% (79.36%, 83.33%)	163.63% (146.21%, 181.06%)	1.767 (1.528, 2.005)	1.003 (0.936, 1.070)	51.5% (44.5%, 58.5%)
	After Covariates Rebalancing				
LA	88.78% (87.40%, 90.17%)	42.56% (37.80%, 47.32%)	0.158 (0.126, 0.191)	0.273 (0.248, 0.297)	84.0% (78.9%, 89.1%)
LR	94.78% (93.85%, 95.72%)	22.60% (20.11%, 25.09%)	0.055 (0.045, 0.065)	0.155 (0.142, 0.168)	88.5% (84.0%, 93.0%)
SDL	94.41% (93.49%, 95.32%)	19.69% (17.41%, 21.97%)	0.026 (0.022, 0.030)	0.112 (0.104, 0.121)	92.0% (88.2%, 95.8%)
DeDL	94.75% (93.87%, 95.63%)	9.27% (8.32%, 10.21%)	0.005 (0.005, 0.006)	0.053 (0.050, 0.057)	93.5% (90.1%, 96.9%)