Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

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

Zikun Ye, Zhiqi Zhang, Dennis J. Zhang, Heng Zhang, Renyu Zhang

To cite this article:
Zikun Ye, Zhiqi Zhang, Dennis J. Zhang, Heng Zhang, Renyu Zhang (2025) Deep Learning-Based Causal Inference for Large-Scale Combinatorial Experiments: Theory and Empirical Evidence. Management Science

Published online in Articles in Advance 15 Oct 2025

. https://doi.org/10.1287/mnsc.2024.04625

https://pubsonline.informs.org/journal/mnsc

MANAGEMENT SCIENCE
Articles in Advance, pp. 1–24
ISSN 0025-1909 (print), ISSN 1526-5501 (online)

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

Zikun Ye,[a] Zhiqi Zhang,[b] Dennis J. Zhang,[b] Heng Zhang,[c] Renyu Zhang[d,*]

[a] Michael G. Foster School of Business, University of Washington, Seattle, Washington 98195; [b] Olin Business School, Washington University in St. Louis, St. Louis, Missouri 63130; [c] W. P. Carey School of Business, Arizona State University, Tempe, Arizona 85287; [d] Chinese University of Hong Kong Business School, The Chinese University of Hong Kong, Hong Kong, China

*Corresponding author

Contact: zikunye@uw.edu, https://orcid.org/0000-0001-9914-7966 (ZY); z.zhiqi@wustl.edu, https://orcid.org/0009-0005-4566-8148 (ZZ); denniszhang@wustl.edu, https://orcid.org/0000-0002-4544-775X (DJZ); hengzhang24@asu.edu, https://orcid.org/0000-0002-6105-6994 (HZ); philipzhang@cuhk.edu.hk, https://orcid.org/0000-0003-0284-164X (RZ)

Received: January 23, 2024
Revised: December 5, 2024
Accepted: March 2, 2025
Published Online in Articles in Advance: October 15, 2025

https://doi.org/10.1287/mnsc.2024.04625

Abstract. Large-scale online platforms launch hundreds of randomized experiments (also known as A/B tests) every day to iterate their operations and marketing strategies. The combinations of these treatments are typically not exhaustively tested, which triggers an important question of both academic and practical interest. Without observing the outcomes of all treatment combinations, how does one 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 show theoretically that this framework yields efficient, consistent, and asymptotically normal estimators under mild assumptions, thus allowing for identifying the best treatment combination when observing only a few combinations. To empirically validate our method, we collaborated with a large-scale video-sharing platform and implemented our framework for three experiments involving three treatments, where each combination of treatments is tested. When observing only a subset of treatment combinations, our DeDL approach significantly outperforms other benchmarks to accurately estimate and infer the average treatment effect of any treatment combination and to identify the optimal treatment combination.

History: Accepted by Vivek Farias, data science.
Funding: R. Zhang is grateful for financial support from the Hong Kong Research Grants Council General Research Fund [Grants 14502722, 14503224, and 14504123].
Supplemental Material: The online appendix and data files are available at https://doi.org/10.1287/mnsc.2024.04625.

Keywords: deep learning • double machine learning • causal inference • field experiments • experimentation on online platforms

1. Introduction

Large-scale online platforms have penetrated the daily lives of billions of people. As of January 2021, more than 53% of the world population (about 4.2 billion people) is active social media users.[1] People connect 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. These platforms create tremendous value; the firms that develop and own such businesses are now worth more than U.S. dollar (USD) 2 trillion. For example, in November 2024, the market value of Amazon was USD 2.1 trillion, that of Alphabet was USD 2.0 trillion, and that of Microsoft was USD 3.1 trillion. McKinsey & Company estimates that the total market value of platform-based tech firms will reach more than 30% of the annual global gross economic outcome within the next 10 years.[2]

Equipped with mountains of user data and advanced information technology, online platforms base their critical business decisions on data analytics techniques. Of central importance are randomized experiments (also known as 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 their 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, because of 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 numerous A/B tests simultaneously. On the one hand, the orthogonal experiment design utilizes the user traffic of the platform more efficiently. Orthogonality ensures noninterference 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 that 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, for example, to all platform users is irrespective of other concurrent experiments. Such a decision paradigm is particularly prevalent because of 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 often, there is little coordination. The combined effects of multiple experiments are largely treated in a simple manner with the linear additivity assumption made.

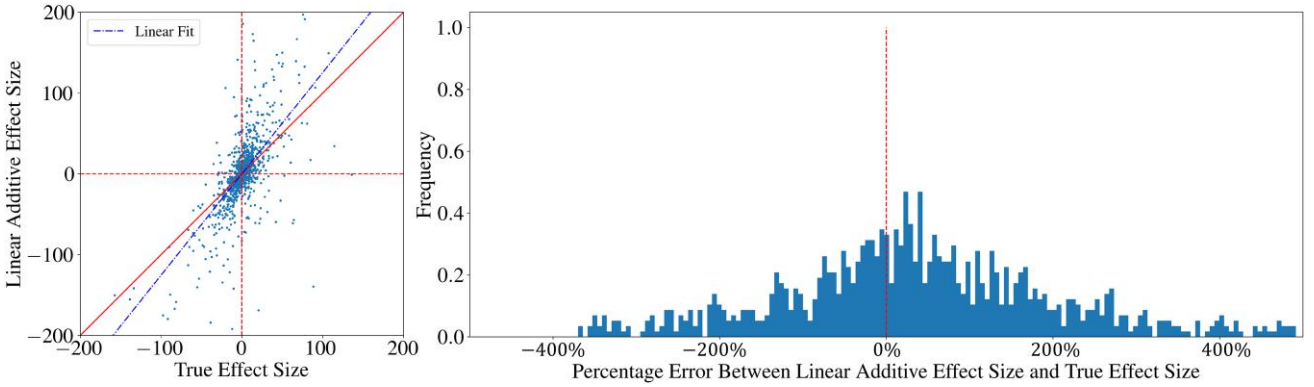
Combining different treatments can create synergistic or antagonistic effects depending on how different treatments interact with each other. It is usually difficult without a formal test to predict which one is in effect. In the worst case, two treatments that both benefit the platform can in fact hurt the platform 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 pretreatment covariates, including gender, age, location, and degree of activeness. Each observation in Figure 1 represents one such subgroup. The left panel of Figure 1 plots, for each subgroup, the true observed effect size in the experiment as well as the calculated effect size if we assume that these two effects are linearly additive. It clearly reveals the *substantial gap* from the ground truth to simply adopt the linear addition (LA) rule in policy evaluation with multiple experiments. The right panel of Figure 1 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 questions are as follows. *When conducting multiple A/B tests and observing only a small subset of treatment combinations, how does one estimate and infer the causal effect of all treatment combinations, and how does one 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 online 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 because of regulation or overfitting leads to an insufficient convergence to the true causal effects (i.e., average treatment effect (ATE)) and consequently, undesired statistical properties, which hinder effective inference (see, e.g., Chernozhukov et al. 2018).

The main goal of this paper is to develop a new theoretically sound and practically feasible method to estimate the causal effect of any treatment combination when observing the outcome of only a small subset of

Figure 1. (Color online) Heterogeneous Response to Two Experiments



Notes. In the left panel, the x axis represents the true combined treatment effect of each subgroup, and the y axis represents the calculated treatment effect of each subgroup under the linear addition assumption. The 45° red solid line represents the equivalence of two effects, and the blue dashed line represents the linear fit of these two effects. The distinction between the blue dashed line and the red solid line elucidates that linear addition does not accurately recover the actual cumulative effect of the two treatments. The right panel shows the percentage error between the linear additive effect size and the true effect size.

combinations. To this end, following the recent cutting-edge research of combining semiparametric statistics and deep learning (DL) for inference, namely double/debiased machine learning (DML), we propose a statistical framework (called debiased deep learning (DeDL)). We highlight the following contributions of our paper.

1.1. A DML-Based Modeling Framework for Multiple A/B Tests with Mild Identification Requirement and Valid Inference

We propose a DeDL framework with theoretical guarantees for researchers and practitioners to analyze treatment effects in concurrent experiments and identify the optimal treatment combination. Our framework, unlike factorial designs that require observing an exponential number of treatment combinations, requires observing only a linear number of treatment conditions. Our main theoretical contribution is modeling treatment effect interactions as a nonparametric function (i.e., DNN) of user characteristics and treatment vectors within the double-machine learning framework (Chernozhukov et al. 2018, Farrell et al. 2020). We advocate for using a *generalized sigmoid link function* to map semiparametrically the treatment vector and nuisance parameters, such as user characteristics, to the outcome. In Theorem 1, we establish a new approximation bound showing that the approximation error of our proposed link function combined with the nonparametric DNN for nuisance parameters decreases at a square-root rate with the number of unique user characteristics, regardless of the data generation process (DGP) for average treatment effects. Furthermore, we show that the abstract theoretical assumptions of the DML framework translate into easily verifiable and practically satisfied conditions in this context. With these conditions verified in practice, following Farrell et al. (2020), we demonstrate that our method produces asymptotically normal (and thus, naturally $\sqrt{n}$-consistent) estimators, enabling valid inference.

1.2. Empirical Validation of Our Framework

To demonstrate the practicality of our DeDL framework, we implement it in a large-scale multiple-experiment setting ($N > 2{,}000{,}000$) on Platform O. We persuaded the company to conduct a costly full-factorial design with three treatment conditions (i.e., $2^3 = 8$ treatment combinations), allowing us to observe the ground-truth causal effects of any treatment combination for comparison. We compare our DeDL approach against a set of benchmarks, including linear regression (LR) and DL-based methods. Our results show that DeDL provides significantly more accurate estimates of the ATE and more precise identification of the optimal treatment combination. To the best of our knowledge, this is the first paper to validate the practical effectiveness of theoretically elegant DML methods through large-scale field experiments in the absence of unobserved endogenous variables. Although recent literature (Gordon et al. 2023) highlights the limitations of the DML framework when omitted variables are present, we demonstrate its core strength: accurately approximating flexible treatment and outcome functions by showcasing its superior performance in the absence of omitted variables. This evidence is increasingly relevant as more empirical researchers adopt DML methods, especially given the difficulty of verifying key technical assumptions, such as the first-stage convergence rate $o(n^{-1/4})$ of the ML algorithm. Furthermore, 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, a misspecified model layer, and highlight the covariate imbalance issue.

1.3. A Practitioner's Guide to DML-Based Causal Inference with DNNs

Our work builds on the latest advances in the DML literature (Chernozhukov et al. 2018, Farrell et al. 2020). Although this literature is theoretically elegant and powerful, it leaves many practical details and challenges for model implementation unaddressed. Our third contribution is to provide practical guidance and hands-on summaries for using DML in causal inference with DNNs. On the theoretical side, in Section 3, we outline the key steps for establishing a robust framework of using DML in various settings, including selecting an appropriate link function, training the DNN model, verifying the identifiability and convergence of nuisance parameters, constructing influence functions, and performing crossfitting. On the practical implementation side, in Section 5, we provide critical checkpoints and guidance for successful empirical implementation of the DML algorithm. For example, we emphasize the importance of addressing covariate imbalance through stratified sampling and underscore the value of DNN training error as a key indicator for evaluating goodness of fit when estimating nuisance parameters in practice. Finally, we also conduct comprehensive synthetic experiments to demonstrate the robustness of our proposed methods under various conditions (see Section 6). With these detailed demonstrations, we aim to bridge the gap between theory and practice, offering researchers and practitioners a clear road map for applying DML with DNNs to causal inference problems effectively and reliably.

The rest of this paper is organized as follows. In Section 2, we review the relevant literature. In Section 3, we present our DeDL framework to infer treatment effects and identify the optimal treatment combination. In Section 4, we apply the framework to analyze real-world experiments. In Section 5, drawing from our own learning, we provide a detailed discussion of some crucial issues in practical implementation, aiming to guide similar applications in other empirical contexts. In Section 6, we conduct comprehensive synthetic experiments to demonstrate the robust performance of our proposed framework. Section 7 concludes. All proofs are relegated to the online appendix. All codes regarding Section 6 can be accessed in the GitHub repository.[3]

2. Literature Review

In this section, we review several streams of literature closely connected to our work.

2.1. The Theory and Applications of DML

The proposed DeDL framework stems from the recent advances in 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 nuisance parameter(s). In particular, Farrell et al. (2021) establish novel nonasymptotic high-probability bounds for nuisance parameter estimation with deep feed-forward neural nets. Chiang et al. (2022) extend the DML framework by proposing a multiway crossfitting 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 the context of an operation. For example, Tang et al. (2025) propose a personalized pricing algorithm by maximizing the expected revenue estimated using DML. We contribute to this literature by adapting the DML framework in the combinatorial experiment setting. Our DeDL framework is most related to Farrell et al. (2020), in which the authors extend the partial linear model in Chernozhukov et al. (2018) into a more general semiparametric form (a prespecified model layer on the top of neural networks) combining nuisance parameters and high-dimension treatments. Although their approach is broadly applicable, the generality of their framework leaves the identification requirements under specific settings ambiguous. Their method presupposes that the identification holds under the current observational data, enabling valid extrapolation on unobserved treatments, but this assumption is not directly applicable in our combinatorial experiment context. To address this gap, we tailor their method by explicitly identifying the easy-to-check technical conditions required for identification and the convergence of DNN for our application. The inference result of our ATE estimator constructed from the crossfitting follows Chernozhukov et al. (2018) straightforwardly. Furthermore, we derive the estimator and inference result for the best treatment identification, the technique of which could be extended to broader decision-making problems based on DML estimators.

The DML is not solely the subject of extensive theoretical research. The DML has been extensively applied in many empirical settings for heterogeneous treatment effects. For example, Knaus (2022) 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 rewards for Amazon Mechanical Turk (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. (2023) find that DML implemented with observational data under the selection of ads for users may have substantial biases from the ground truth. Although all applications beyond Gordon et al. (2023) use observational data, our research is the first to provide evidence on the performance of 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. (2023) that DML estimates are far from experimental results, we show that our debiased estimators are accurate and valid for inference, significantly outperforming other benchmarks. Our documented strong empirical validation and comprehensive discussions of the DML method can benefit both researchers and practitioners.

2.2. 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 detailed discussions of these classical approaches. Recent works (e.g., Dasgupta et al. 2015, Pashley and Bind 2023) 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 one can only practically test $O(m)$ treatment combinations, suggesting that 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.

2.3. Causal Inference and Its Applications to Online Platforms

Causal inference has long been a central topic in many fields, including economics, psychology, medical science, marketing, and operations (e.g., see Angrist and Pischke 2009 and 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. Farias et al. (2021) compute debiased estimators of treatment effects under general intervention patterns, which subsume the synthetic control paradigm. Goli et al. (2024) propose a theoretical framework to overcome the bias because of the interference in a ranking experiment on travel websites. Kallus et al. (2018) use matrix factorization and bound the estimation errors for ATEs 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 combines 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. Furthermore, operations researchers also apply optimization techniques, such as robust optimization (e.g., see Lim et al. 2006), to study the estimation in causal inference (e.g., Bertsimas et al. 2022).

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 online 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. 2023). On the theoretical side, researchers propose innovative methods to overcome challenges arising from online platforms, such as two-sided randomization (e.g., Nandy et al. 2021, Johari et al. 2022, Ye et al. 2023), sequential experiments (e.g., Song and Sun 2021, Bojinov et al. 2023, Xiong et al. 2023), and block randomization (Candogan et al. 2024). Whereas this literature

typically focuses on the single-experiment setting, we study the inference problem with multiple experiments.

3. Debiased Deep Learning Framework

In this section, we introduce the DL-based framework following Chernozhukov et al. (2018) and Farrell et al. (2020) for estimating and inferring treatment effects in our multiple-experiments setting.[4] As discussed in Section 1, one may view our work as a "playbook" for implementing DNN-backed DML for causal inference in broader settings, and in this section, we outline the theoretical foundations using multiple treatment effect estimation as an example. Whereas the procedural details are specifically tailored to address our research questions in the multiple-experiment setting, they underscore the critical elements that are broadly essential. For researchers and practitioners interested in other causal inference problems using DML, following a similar procedure is expected to yield the construction of desired estimators. We also emphasize that this section focuses on the theoretical side, and the guidelines for practical implementation are deferred to Section 5.

Our framework comprises three steps, starting with determining the setting and specifying the data generation process as detailed in Sections 3.1 and 3.2. Training the DL model is the second critical step, which requires careful design of a specialized model layer that enables accurate approximation of any ATE function with finite samples (Section 3.3). In the third step, we develop $\sqrt{n}$-consistent DML-type estimators for inference and optimal treatment combination identification (Section 3.4).

3.1. The Setup

Building on the recent advances in ML-powered causal inference, we consider the DL-based inference framework for multiple experiments on a large platform. There are m concurrent field experiments on the platform, each with binary treatment levels, represented by $\boldsymbol{T} \in \{0,1\}^m$.[5] Without loss of generality, we focus on the binary treatment case, which is a common practice for A/B tests on large-scale online 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 $Y \in \mathbb{R}$[6] along with the individual-level pretreatment covariates $\boldsymbol{X} \in \mathbb{R}^{d_X}$ 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 $\boldsymbol{x}$).

In this setting, we 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)? We refer to the second question as the best treatment identification problem.

3.2. Structured Deep Learning: Specify the Data Generation Process

In the first step, following Farrell et al. (2020), we postulate that the DGP has the semiparametric form

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

given any $\boldsymbol{x}$ and $\boldsymbol{t} \in \{0,1\}^m$. $G(\cdot,\cdot)$ is the known link function, and $\boldsymbol{\theta}^*(\cdot) : \mathbb{R}^{d_X} \longmapsto \mathbb{R}^{d_\theta}$ are the unknown nuisance parameters as functions of covariates $\boldsymbol{x}$.[7] In particular, $\boldsymbol{\theta}^*(\cdot)$ characterizes the heterogeneity in outcomes, and we shall predict them by ML models, such as DNNs. The prespecified link function $G(\cdot,\cdot)$ allows for the flexibility and interpretability of the relationship between the outcome Y and the treatment combination $\boldsymbol{t}$. For example, if the link function is linear in $\boldsymbol{t}$ but with heterogeneous coefficients (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 sum 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 treatments.

Before we explain the link function in detail, several remarks are in order. First, with slight adjustments, the DGP can be extended to general DML settings for causal inference. Specifically, treatments can still be represented by $\boldsymbol{t}$, accommodating multiple treatment levels or even continuous treatments depending on the application. The researcher selects the G function that fits the context, such as a logistic function for a binary outcome (Farrell et al. 2020) or a partially linear specification (Chernozhukov et al. 2018). In some cases, where the economic interpretation of the G function is well understood, this selection process is straightforward; in others, it requires careful consideration. Our subsequent discussion in this section offers an illustrative example. In particular, the choice of G function needs to balance the modeling power (see Sections 3.2.1 and 3.2.2) and nice statistical properties to reduce the difficulty of estimating nuisance parameters (see Proposition 1).

Second, it should be noted that in our setting, an alternative, more straightforward method would involve building a pure neural network without any parametric structure (i.e., DNN($\boldsymbol{x}, \boldsymbol{t}$)), which however, does not ensure valid inferences and is prone to overfitting the observed combinations. Consequently, a standard pure DNN may not work well because of its potentially poor out-of-sample performance, even with meticulous regularization. We demonstrate in Online Appendix D that standard regularizations, such as dropout layer and Lasso regularization, are ineffective for our work. In essence, the G function that we utilize

is a variant of a regularized DNN that has been specifically tailored for our application, drawing upon prior knowledge of interaction patterns. Given that a suitably predefined G function is critical in our application, our study focuses primarily on selecting an appropriate G beyond Farrell et al. (2020), with theoretical underpinnings, strong approximation capabilities, and easy-to-validate and easy-to-satisfy identification conditions. Regarding estimation and inference, the procedures that we adopt are straightforward and align with the standard DML (Chernozhukov et al. 2018).

Third, here we also give a brief overview of subsequent theoretical development; having specified the DGP, our framework involves the following two stages. 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 and 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.

3.2.1. Choose the Link Function. As discussed, we capture the richness of individual heterogeneity with the nonparametric function $\boldsymbol{\theta}^*(\cdot)$, whereas given $\boldsymbol{\theta}^*(\cdot)$, we essentially assume that the individual outcomes are fully structured and described by $G(\cdot,\cdot)$. In our setting, the link functions G may take many different forms. For example, the most straightforward choice of G is the linear function of $\boldsymbol{t}$ with heterogeneous coefficients (i.e., $G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}) = \theta_0(\boldsymbol{x}) + \theta_1(\boldsymbol{x})t_1 + \cdots + \theta_m(\boldsymbol{x})t_m$). Although this simple linear form only requires $m+1$ linear independent $(1,\boldsymbol{t}')$ observed treatment vectors for identification, it clearly fails to capture the treatment interactions. Another extreme is $G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}) = \theta_0(\boldsymbol{x}) + \sum_{i=1}^{m}\theta_i(\boldsymbol{x})t_i + \sum_{i\neq j}\theta_{ij}(\boldsymbol{x})t_it_j + \cdots + \theta_{12\cdots m}(\boldsymbol{x})t_1t_2\cdots t_m$, which contains all heterogeneous high-order interaction terms. Although this is the most accurate link function form for our application, it also requires the strongest identification condition (i.e., nonzero assignment probability for all treatment combinations, which is infeasible in practice). Thus, 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 associated with the proper treatment assignment mechanisms to ensure the identifiability and convergence of the estimates $\hat{\boldsymbol{\theta}}(\cdot)$. To enhance the depth of discussion, we propose the following concrete link functions with clear economic interpretations.

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

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

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

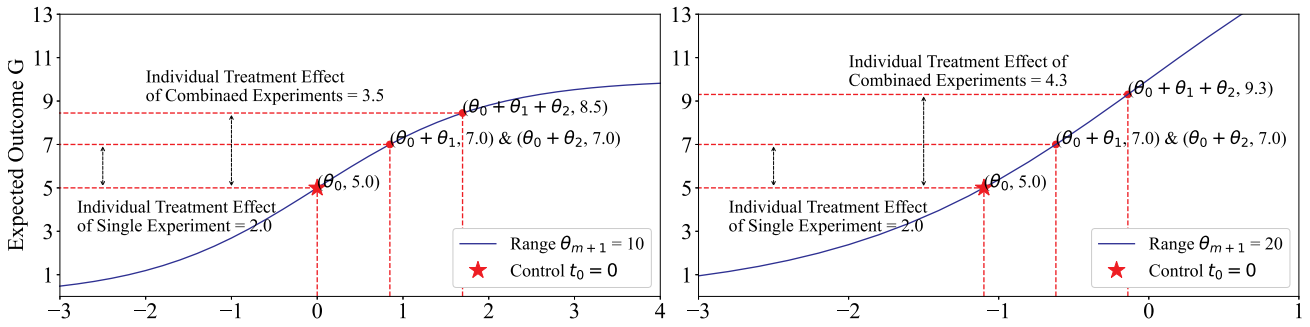
d. *Generalized sigmoid form II.* $G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}) = \theta_{m+1}(\boldsymbol{x})/(1+\exp(-(\theta_0(\boldsymbol{x}) + \theta_1(\boldsymbol{x})t_1 + \cdots + \theta_m(\boldsymbol{x})t_m)))$.

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, the *multiplicative form* can characterize only the increasing marginal effect because of 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. We illustrate the coexistence 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 2, where the y axis represents that 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 five (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 two (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 (second) user type, $\theta_{m+1}=10$ ($\theta_{m+1}=20$). Straightforward calculation implies that $\mathbb{E}[Y_i|\boldsymbol{X}_i,\boldsymbol{T}=(1,1)']=8.5$ (thus, the individual treatment effect 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 (thus, the individual treatment effect is $9.3-5=4.3>4$, suggesting the increasing marginal effects). Therefore, if the first type of users takes more (less) than 36% of the entire population, the platform will have a decreasing (an increasing) marginal ATE.

The *standard sigmoid form* with known constants a and b may be restrictive and cannot model the different outcome ranges across individuals. The *generalized sigmoid forms I* and *II* resolve this issue by incorporating the parameter $\theta_{m+1}(\boldsymbol{x})$. 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)). Hence, we adopt the link function of the *generalized sigmoid form II* in our empirical study.

Figure 2. (Color online) Illustration of Generalized Sigmoid Form II with Two Types of Platform Users: Marginal Decreasing (Left Panel) and Marginal Increasing (Right Panel)



3.2.2. Justify the Link Function: Universal Approximation.

To better illustrate the expressive power of our proposed *generalized sigmoid form II* to capture ATEs, we present a counterintuitive example in Table 1, where the ATE of the individual treatments has a different sign from that of the combined treatment. There are two heterogeneous individuals and two individual experiments. We assume that the individual response follows our *generalized sigmoid form II* with the specific parameters detailed in Table 1. It is apparent that in this example, the ATEs for the first and second individual treatments are −0.04 and −0.06, respectively. However, the combined treatment effect has a different sign (i.e., 0.11). As a structured sigmoid function with only $O(m)$ parameters, our individual-level link function seems restrictive at the first glance, but the ATE averaging over the whole population can be quite flexible.

Furthermore, to highlight that our proposed link function is indeed of practical interest, we develop the following bound on the approximation power of the *generalized sigmoid form II* in Theorem 1, which shows that this link function can be used to approximate arbitrary ATEs with a provable finite-sample error bound. This result echoes the well-established expressive capabilities of the sigmoid function as evidenced in both the theoretical and empirical studies of neural networks. This result confirms that the use of sigmoid function adds great versatility to our model. Interested readers may refer to the proof of the theorem for details in Online Appendix A.

Table 1. Counterintuitive Example Under Generalized Sigmoid Form II

Treatment combination $\boldsymbol{t}'$	HTE of individual 1	HTE of individual 2	ATE
(1, 0)	0.92	−1.00	−0.04
(0, 1)	0.49	−0.60	−0.06
(1, 1)	1.27	−1.05	0.11

Notes. Parameter $\boldsymbol{\theta}' = (\theta_0, \theta_1, \theta_2, \theta_3)$ for the first individual is $(0, 1, 0.5, 4)$, and $\boldsymbol{\theta}'$ for the second individual is $(-1, -3, -1, 4)$. HTE is the heterogeneous effect.

Theorem 1 (Universal Approximation of Generalized Sigmoid Link Function). *Given a population that contains $K \geq 1$ distinct feature vectors $\{\boldsymbol{x}_\ell \in \mathbb{R}^{d_X} : \ell = 1, \ldots, K\}$ such that $\boldsymbol{X} = \boldsymbol{x}_\ell$ with probability $p(\ell) > 0$ for all $\ell = 1, \ldots, K$, where $\sum_{\ell=1}^{K} p(\ell) = 1$, and any outcome function $f(\boldsymbol{t}) : \{0,1\}^m \mapsto \mathbb{R}$, there exists a feature mapping $\{\boldsymbol{\theta}(\boldsymbol{x}_\ell) : \ell = 1, \ldots, K\}$ such that*

$$\left(\frac{1}{2^m} \sum_{\boldsymbol{t} \in \{0,1\}^m} (f(\boldsymbol{t}) - \mathbb{E}_X[G(\boldsymbol{\theta}(\boldsymbol{X}), \boldsymbol{t})])^2 \right)^{\frac{1}{2}} \lesssim \frac{1}{\sqrt{K}},$$

where $G(\cdot, \cdot)$ is the generalized sigmoid form II of individual response $G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}) = \theta_{m+1}(\boldsymbol{x})/(1 + \exp(-(\theta_0(\boldsymbol{x}) + \theta_1(\boldsymbol{x})\, t_1 + \cdots + \theta_m(\boldsymbol{x}) t_m)))$.[8]

We comment on several observations. First, the omitted constant in Theorem 1 depends on the function $f(\cdot)$ and m and is independent of K. This result highlights how the approximation power of our proposed individual-level sigmoid model (i.e., *generalized sigmoid form II*) improves as the number of distinct feature vectors K increases; it proves that one can use this link function to approximately construct any ATE function and captures arbitrary interactions among individual treatments, with an error of order $\mathcal{O}(K^{-1/2})$. We interpret K as the number of individuals with distinct features, suggesting that an increase in the number of distinct individual profiles in the population enhances the potential for a more precise approximation of the ATE function.

Second, this theoretical development is motivated by the classical idea of approximating continuous functions using sigmoid activation units in neural networks (e.g., see Hornik et al. 1989). Specifically, the result is deeply rooted in the classical mathematical theory of Fourier transformation. In our setting, where treatments are modeled as Boolean variables, the Fourier transformation of $f(\boldsymbol{t})$ can be expressed using indicator

functions, which can be effectively approximated by sigmoid functions given sufficient flexibility in the feature space. This underscores the modeling power of our link function; we are not aware of similar approximation results with other link functions, such as multiplicative or simple linear ones.

Third, we acknowledge that such an approximation result does not ensure the practical realizability of the nuisance parameters $\boldsymbol{\theta}(\cdot)$ proven to exist by Theorem 1, nor does it assure a precise estimate of the ATE. This limitation may be attributed to several factors. The first is that the link function $G(\cdot,\cdot)$ can indeed be misspecified. The model misspecification issue is further investigated and discussed in our synthetic experiments in Section 6. Second, as is well documented in the computer science literature (e.g., Adcock and Dexter 2021), there exists a large gap between the theoretical approximation ability of DNNs and their actual performance, which may be resolved by designing better DNN architectures and training algorithms. The third factor is that the nuisance parameters may not be identifiable under the assignment mechanism of the A/B tests, especially when the focus is too narrow on specific treatment combinations without sufficient explorations. We discuss the detailed identification conditions of our proposed link functions, including the *generalized sigmoid form II* following Proposition 1.

Finally, a closer examination of the proof of Theorem 1 reveals the deliberate and intricate design of our *generalized sigmoid form II* compared with the *standard sigmoid form* and the *standard sigmoid form I*. Incorporating a feature-dependent scale parameter, $\theta_{m+1}(\boldsymbol{x})$, and intercept, $\theta_0(\boldsymbol{x})$, appears crucial for realizing the expressive power emphasized in Theorem 1. Moreover, although we could generalize the link function to include more terms, such as $G(\boldsymbol{\theta}(\boldsymbol{x}),\boldsymbol{t}) = \theta_{m+1}(\boldsymbol{x})/(1+\exp(-(\theta_0(\boldsymbol{x}) + \theta_1(\boldsymbol{x})t_1 + \cdots + \theta_m(\boldsymbol{x})t_m))) + \theta_{m+2}(\boldsymbol{x})$, the proof also indicates that this additional complexity does not deliver significantly better approximations of the ATEs. From an empirical point of view, such additional complexity also makes identifying the nuisance parameters more challenging, which may not be justified by the increased challenge in identification.

3.3. Training Stage

In the second step (the *training* stage), we use DNNs to approximate the unknown parameter $\boldsymbol{\theta}^*(\cdot)$ motivated by our DGP (1). We add a model layer to represent the 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 3 illustrates the whole model from $\boldsymbol{X}$ and $\boldsymbol{T}$ to Y (see also Farrell et al. 2020, figure 2). We use $\{(y_i,\boldsymbol{x}_i',\check{\boldsymbol{t}}_i')' : 1 \le i \le n\}$ to denote the realization of random vector $(Y,\boldsymbol{X}',\boldsymbol{T}')'$ (i.e., the observed data from experiments). We use the check symbol over the treatment $\check{\boldsymbol{t}}$ to represent the realized treatment level in experiments to avoid confusion by the notation $\boldsymbol{t} \in \{0,1\}^m$ representing any target treatment in ATE as defined in (3).

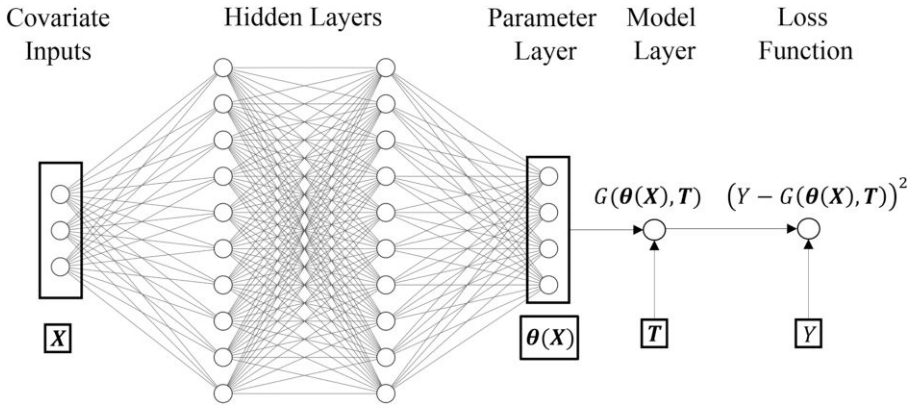
3.3.1. Specify the Loss Function. The loss function measures the quality of the estimated nuisance parameter and plays an important role in the training stage. Depending on the application, different loss function can be used. For example, the maximum log likelihood can used in case of binary outcomes. As a general guideline, a certain smoothness assumption is needed (e.g., Farrell et al. 2020, assumption 1) to obtain a theoretical guarantee (e.g., Proposition 1 in our setting). Our 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 squared error denoted by $\ell(y_i,\check{\boldsymbol{t}}_i',\boldsymbol{\theta}(\boldsymbol{x}_i)) = (y_i - G(\boldsymbol{\theta}(\boldsymbol{x}_i),\check{\boldsymbol{t}}_i'))^2$ as the per-observation 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

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

where $\mathcal{F}_{DNN}$ is the set of fully connected neural nets with bounded outputs. Note that we use DNNs to approximate $\boldsymbol{\theta}(\boldsymbol{x})$ for two reasons. (1) DNNs are easier to engineer with a general link function G, and one can leverage off-the-shelf packages (e.g., PyTorch and TensorFlow) to train models at scale. (2) DNNs have better approximation and prediction power in general compared with other ML models. Our framework can be readily applied to other ML models to get $\hat{\boldsymbol{\theta}}(\boldsymbol{x})$ as long as the convergence rate is upper bounded by $o(n^{-1/4})$.

3.3.2. Identifiability and Convergence of Feature Mappings. With our proposed link functions, we are ready to show the identifiability and convergence rate of the DNN in our framework. The proof strategy largely builds on the approach in Farrell et al. (2020). Our primary theoretical contribution here is the introduction of practical, primitive conditions in our setting that ensure the identifiability of our model and facilitate the verification of the convergence conditions outlined in Farrell et al. (2020). Using multiple-experiment treatment effect estimation as a test bed, we demonstrate the type of analysis required to provide theoretical guarantees for the DML framework. Specifically, for identifiability, we further assume that the treatment assignment mechanism is sufficiently "regular" (see Assumption 4 in Online Appendix A.2). For convergence, the additional assumptions on the data observations being i.i.d.

Figure 3. Illustration of the Deep Neural Network with the Structured Model Layer



and bounded and $\boldsymbol{\theta}^*(\boldsymbol{x})$ being sufficiently smooth are also imposed (see Assumption 3 in Online Appendix A.2). Formally, we have the following proposition, the proof of which is relegated to Online Appendix A.4.

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

a. *Under Assumption 1 and Assumption 4 in Online Appendix A.2, the parameter function* $\boldsymbol{\theta}^*(\boldsymbol{x})$ *can be nonparametrically identified in DGP (1).*

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

$$\|\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k^*\|_{L_2(\mathbf{X})}^2 \lesssim n^{-\frac{p}{p+d_X}}\log^8 n + \frac{\log\log n}{n}$$

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

The key step to proving 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 tight (see Farrell et al. 2021), but it is sufficiently fast for the subsequent inference in our setting if $p > d_X$.

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 Online Appendix A.2 for details), which are orders of magnitude smaller than 2^m, to ensure the identifiability and sufficiently fast convergence. In other words, suppose that there are 10 different treatments and in turn, $2^{10} = 1024$ possible treatment combinations. Our framework needs to observe only $10 + 2 = 12$ 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. It should be noted that the requirement for $m + 2$ combinations is not arbitrary. Take the *generalized sigmoid form II* for example. We first stipulate that $\mathbb{E}[\tilde{\boldsymbol{T}}\tilde{\boldsymbol{T}}'|\boldsymbol{X}] > 0$ almost uniformly everywhere; that is, its smallest eigenvalue is uniformly lower bounded away from zero across all $\boldsymbol{X}$, except on a set of zero probability. Specifically, this condition is readily met under a common and lenient treatment assignment rule. Each of the m individual treatment conditions as well as the full-control condition is assigned with a positive probability. In other words, we assign $\boldsymbol{t} = (0,0,\ldots,0)'$ and $(1,0,\ldots,0),(0,1,\ldots,0)',\ldots,(0,0,\ldots,1)$, each with a positive probability. This assumption is relatively mild. The $m + 1$ linearly independent combinations $(1,\boldsymbol{t}')$ integrate seamlessly because the link function includes a linear component $\theta_0(\boldsymbol{x}) + \theta_1(\boldsymbol{x})t_1 + \cdots + \theta_m(\boldsymbol{x})t_{m+1}$. Beyond these $m + 1$ combinations, we note that with a total of $m + 2$ unknown parameters in the *generalized sigmoid form II*, at least one additional treatment combination should be assigned with a positive probability for identification. For example, if two conditions $\boldsymbol{t} = (1,0,0,0)'$ and $(0,1,0,0)'$ are already assigned when $m = 4$, only one additional overlapping condition $(1,1,0,0)'$ is required to identify the nuisance parameter $\theta_{m+1}(\boldsymbol{x})$ in the numerator. Such modest requirement for identification stems from the well-structured nature of the proposed link function. More generally, proposing a more complex link function would entail a more demanding identification challenge.

3.4. Estimation and Inference Stage

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 treatment over any other treatment combination. We first 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 ground-truth ATE of any treatment combination $\boldsymbol{t} \in \{0,1\}^m$ can be written as

$$\begin{aligned}\mu(\boldsymbol{t}) &= \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t})] - \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t}_0)] \\ &= \mathbb{E}[H(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}, \boldsymbol{t}_0)]. \end{aligned} \tag{3}$$

Denote $\hat{\mu}(\boldsymbol{t})$ as our proposed estimator for $\mu(\boldsymbol{t})$; then, we show that $\hat{\mu}(\cdot)$ is asymptotically normal and semiparametric efficient. Analogously, the ATE increment of the best treatment $\boldsymbol{t}^*$ over any $\boldsymbol{t} \in \{0,1\}^m$ is written as

$$\begin{aligned}\tau(\boldsymbol{t}) &:= \mu(\boldsymbol{t}^*) - \mu(\boldsymbol{t}) = \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t}^*)] - \mathbb{E}[G(\boldsymbol{\theta}^*(\boldsymbol{X}), \boldsymbol{t})] \\ &= \mathbb{E}[H(\boldsymbol{X}, \boldsymbol{\theta}^*(\boldsymbol{X}); \boldsymbol{t}^*, \boldsymbol{t})]. \end{aligned} \tag{4}$$

Notice that identifying the best treatment boils down to identifying the ATE increment of each treatment combination. For the validity of inference of identified best treatment, we also test the one-sided hypothesis $\tau(\boldsymbol{t}) \geq 0$ for all $\boldsymbol{t}$. Letting $\hat{\tau}(\boldsymbol{t}) := \hat{\mu}(\hat{\boldsymbol{t}}^*) - \hat{\mu}(\boldsymbol{t})$ be an estimator of $\tau(\boldsymbol{t})$ for each $\boldsymbol{t} \in \{0,1\}^m$, we can also show that our proposed estimator $\hat{\tau}(\cdot)$ is also $\sqrt{n}$ consistent, and the empirical best treatment agrees with the true best treatment with probability approaching one. We mention that for brevity, this subsection focuses on ATE estimation and inference discussion, whereas the detailed discussion of the best treatment identification is relegated to Online Appendix A.8.

3.4.1. Construct the Influence Function. One 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; this estimator is generally not $\sqrt{n}$ consistent because of the large bias of ML models. Intuitively, the issue with this naïve estimator is that the error of the nuisance parameter is accounted for. To solve this, the inference with DML is mainly built upon the semiparametric technique—influence function (also known as the Neyman orthogonal score)—which implies that the first order is insensitive to perturbations in the nuisance parameters. We refer interested readers to Newey (1994) and Chernozhukov et al. (2018, section 2.2.5) for more discussion of influence functions and Neyman orthogonality. Similar to other works using influence function in semiparametric 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* $G_{\boldsymbol{\theta}}(\boldsymbol{\theta}, \boldsymbol{t})$ *is the gradient of* $G(\boldsymbol{\theta}, \boldsymbol{t})$ *with respect to* $\boldsymbol{\theta}$; *and* (iii) *the ATE* $\mu(\boldsymbol{t})$ *is identified and path-wise differentiable*.

We remark that Assumption 2 imposes several regularity conditions, which are standard and not restrictive in the literature on semiparametric statistics. The invertibility of $\boldsymbol{\Lambda}(\boldsymbol{x})$ is commonly assumed for deriving the influence function, and it can be easily satisfied in our setting. As shown in Online Appendix A.6, this invertibility 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 path-wise differentiability of $\mu(\boldsymbol{t})$ is a standard regularity condition. Such assumptions should be considered as standard in applications of DML. They facilitate the construction of the influence function, whose idea is attributed to Farrell et al. (2020), and we adapt it to our context.

Proposition 2 (Influence Function). *Suppose Assumptions 1 and 2, Assumption 3 in Online Appendix A.2, and Assumption 4 in Online Appendix A.2 hold; 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*

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

where $\boldsymbol{z} = (y, \boldsymbol{x}', \check{\boldsymbol{t}}')'$ *is observed data,* $\boldsymbol{\Lambda}(\boldsymbol{x}) := 2\mathbb{E}[G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{T}) G_{\boldsymbol{\theta}}(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{T})' | \boldsymbol{X} = \boldsymbol{x}]$, $G_{\boldsymbol{\theta}}$ *is 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 *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 gradient of* ℓ *with respect to* $\boldsymbol{\theta}$.

The influence function defined by (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 the framework *debiased deep learning*. The computation of $\boldsymbol{\Lambda}(\boldsymbol{x})$ is straightforward under the known treatment assignment mechanism $\nu(\cdot|\boldsymbol{x})$.

3.4.2. Complete the Framework: Asymptotic Normality. Based on this influence function and the crossfitting technique (e.g., Chernozhukov et al. 2018, Farrell et al. 2020), we can construct estimators as illustrated in Algorithm 1. We refer interested readers to the Online Appendix for the details of constructing the estimators $\hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$ by Equation (33) in Online Appendix A.7 and $\hat{\Psi}_{\mathsf{DeDL}}(\boldsymbol{t}; \mu)$ by Equation (34) in Online Appendix A.7 and the confidence interval $\widehat{\mathcal{CI}}_{\mathsf{DeDL}}(\boldsymbol{t}; \mu)$ by Equation (35) in Online Appendix A.7.

Algorithm 1 (Implementing DeDL with Crossfitting)

1: (Crossfitting) Split data samples into S nonoverlapping 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

$\hat{\mu}_{\text{DeDL}}(\boldsymbol{t})$ and variance estimator $\hat{\Psi}_{\text{DeDL}}(\boldsymbol{t};\mu)$. Conduct ATE inference based on $\hat{\mu}_{\text{DeDL}}(\boldsymbol{t})$ and $\hat{\Psi}_{\text{DeDL}}(\boldsymbol{t};\mu)$.

4: (Best treatment identification) Find empirical best treatment $\hat{\boldsymbol{t}}^* := \arg\max_{\boldsymbol{t}} \hat{\mu}(\boldsymbol{t})$. Similarly, use influence function ψ and crossfitting to construct estimators $\hat{\tau}_{\text{DeDL}}(\boldsymbol{t})$ and $\hat{\Psi}_{\text{DeDL}}(\boldsymbol{t};\tau)$ (see Online Appendix A.8) for the inference on the best treatment identification.

We now present Theorem 2 on the asymptotic normality.

Theorem 2 (Asymptotic Normality). *Suppose Assumptions 1 and 2, Assumption 3 in Online Appendix A.2, and Assumption 4 in Online Appendix A.2 hold and $\hat{\boldsymbol{\Lambda}}_s(\boldsymbol{x}_i)$ is uniformly invertible. Furthermore, we assume for all subsamples $s = 1, 2, \ldots, S$ that the estimators obey $\|\hat{\theta}_{sk} - \theta_k^*\|_{L_2(X)} = o(n^{-1/4})$, $k \in \{1, \ldots, d_{\boldsymbol{\theta}}\}$, which holds under the assumptions and regularity conditions (for the structured DNN) of Proposition 1(b).*

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

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

b. *Furthermore, suppose the best treatment $\boldsymbol{t}^* := \arg\max_{\boldsymbol{t}\in\{0,1\}^m} \mu(\boldsymbol{t})$ 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 $\boldsymbol{t} \in \{0,1\}^m$,*

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

The formal proof of Theorem 2 can be found in Online Appendix A.8. Importantly, the probability of failing to identify the true best treatment vanishes as the sample size grows large. Therefore, Theorem 2 establishes the valid inference for ATE and best treatment identification under our DeDL framework. For the rest of this paper, we validate this framework with both experimental and synthetic data, and we demonstrate its superior performance over commonly used benchmarks.

As a concluding remark of this section, we summarize the key checkpoints for conducting DML-based causal inference using DNNs that are generically applicable to other settings. The first step involves specifying the DGP (Section 3.2). Particular attention should be given to designing the link function G (Section 3.2.1) to balance modeling power (Section 3.2.2) and the statistical properties that enhance the estimation of the nuisance parameters (see Proposition 1). In the second step, the DNN is trained using the available data and the link function as the top layer of the network (Section 3.3). Here, it is crucial to choose a loss function that possesses favorable theoretical properties for training (Section 3.3.1). If the link function G and the loss function are both appropriately specified, one can expect theoretical guarantees for the convergence of the nuisance parameters (Proposition 1). For robustness, crossfitting is commonly applied (Algorithm 1). This technique splits the data into overlapping folds, with each fold's complement used to train a separate DNN estimator of the nuisance parameters. In the third step, where treatment effect estimation and inference are performed (Section 3.4), influence functions are constructed to mitigate the bias from machine learning errors (Proposition 2). Using these influence functions and the estimated nuisance parameters from crossfitting, one can construct the treatment effect estimator and the associated confidence interval (Algorithm 1). If each step is carefully carried out, one can show that the final estimator of the treatment effect is asymptotically normal, which enables statistical inference (Theorem 2).

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 (and identifies the optimal combination) than the commonly used benchmarks.

4.1. Field Setting, Experiments, and Data

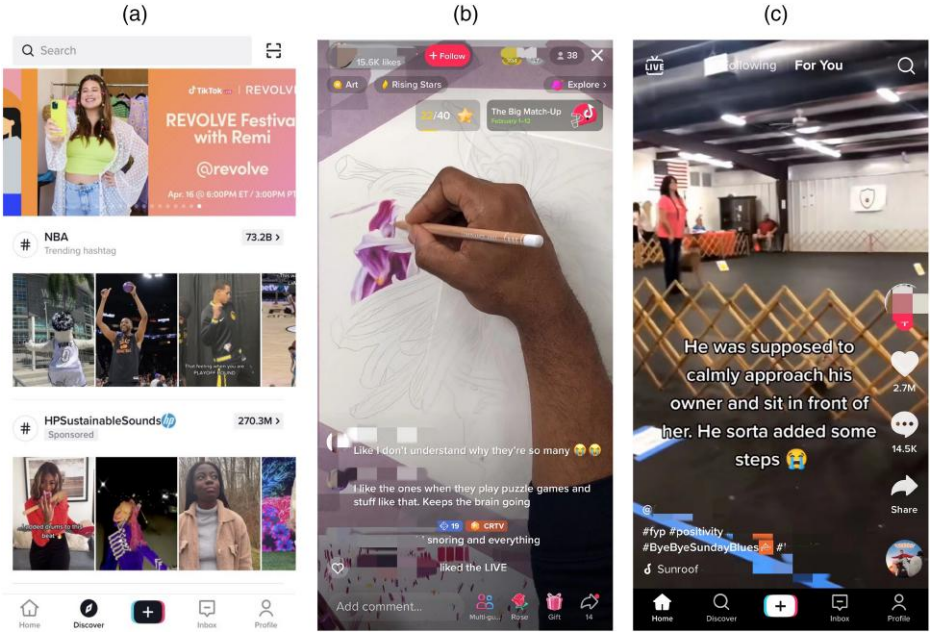
In this section, we introduce the setup of our empirical setting. This empirical application highlights a unique experiment setting that allows us to verify the success of our DeDL framework with *observable* ground truth. To our knowledge, this is the first large-scale empirical validation based on a large-scale field experiment.

4.1.1. Platform O and the Experimental Background.

To empirically validate our proposed framework, we collaborate with Platform O, which features interactive short videos. Platform O, one of the largest short-video platforms, serves billions of users globally every day. 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 panels (a)–(c), respectively, of 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) that fit each user's idiosyncratic interests. Users of Platform O can easily switch to any of these pages at any time that they are using the platform.

Figure 4. (Color online) An Illustration of the Three Pages



Notes. (a) Discover page. (b) Live page. (c) For you page.

Like all other large-scale UGC platforms, such as Facebook and TikTok, Platform O runs hundreds of A/B tests daily 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 that a user spends on the platform per day. Each experiment is randomized based on a distinct hash function of user identifications, 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).[9] This set of experiments has one unique feature that the other sets of experiments on Platform O do not have; because these three experiments were run 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 O decided to test all three experiments on the same population is precisely 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. Although we admit that three is the smallest number of individual treatments that could be interesting, we highlight that our orthogonal experiment data set is of large scale and high quality, delivering trustworthy empirical evidence on the performance of double machine learning in the real setting. To validate the applicability of our framework to a larger number of experiments, we conduct simulations with synthetic data for $m = 10$ in Online Appendix D.2. If the number of experiments is even larger (e.g., $m = 100$ or 1,000), we suggest first reducing m by filtering out insignificant treatments, which take a large proportion in practice. Another widely adopted practice is combining some similar experiments as one synthetic larger experiment.

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 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 $\boldsymbol{t} = (1,0,0)'$ 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 pretreatment 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 degree of activeness) and 10 continuous variables (such as the video-watching duration on each page per day in the 10 days right before the experiments). Table A1 in Online Appendix B.1 describes all of the covariates used in our analysis.

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

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

The ATE under treatment combination $\boldsymbol{t}$ is simply the expected outcome y under $\boldsymbol{t}$ compared with 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 balance the covariates observed in different treatment groups, which is explained in detail in Section 5.1.

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 report only the relative ATEs (see column (1) in 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.

To validate our DeDL framework, we assume that some treatment conditions are unobserved; we use our framework to recover these "unobserved" conditions, and we compare 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) in Table 2). We denote $\mathcal{T}_o := \{\boldsymbol{t} \in \mathcal{T} : t_1 + t_2 + t_3 \in \{0,1,3\}\}$ 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., $\boldsymbol{t} = (1,1,0)'$) 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., $\mu(\boldsymbol{t})$ defined by (3) for all $\boldsymbol{t} \in \mathcal{T}$), whose ground-truth value is documented in column (1) in Table 2. The implementation details are provided in Online Appendix D.5. First, we consider the following model specification of DGP:

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

The link function G as a sigmoid function can capture either "diminishing marginal return" or "increasing

Table 2. Ground-Truth ATE and Best Treatment Identification of Eight Treatment Combinations

Treatment combination $\boldsymbol{t}'$	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

Notes. To protect sensitive data, ATE is proportionally rescaled to relative effect size. The optimal treatment combination (i.e., best treatment) is $\boldsymbol{t}^* = (1,0,1)'$.

*$p < 0.05$; **$p < 0.01$; ***$p < 0.001$; ****$p < 0.0001$.

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 of as the maximum possible video-watching time of any user.

Figure A6 in Online Appendix D.5 illustrates our DNN architecture. 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, the ReLU function (i.e., $\mathrm{ReLU}(x) = \max\{0,x\}$) is used as the activation function. We then concatenate the last layers of two DNNs, take the linear combination $(u = \theta_0(\boldsymbol{x}) + \theta_1(\boldsymbol{x})t_1 + \theta_2(\boldsymbol{x})t_2 + \theta_3(\boldsymbol{x})t_3)$ as the input of a sigmoid function layer, and add another linear layer (no intercept; the slope approximates θ_4^*) to output y. We implement our DNN in TensorFlow and train the DNN by Adam algorithm (Kingma and Ba 2014) under the mean squared loss. We emphasize that the details of crossfitting and DNN implementation are critical for empirical success. Accordingly, we defer this discussion and dedicate Section 5.2 specifically to it.

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 influence function $\psi(\boldsymbol{z}, \hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\Lambda}}; \boldsymbol{t}, \boldsymbol{t}_0)$ defined by (5). Specifically, we apply the estimators $\hat{\mu}_{\mathsf{DeDL}}(\cdot)$ and $\widehat{\mathcal{CI}}_{\mathsf{DeDL}}(\cdot;\mu)$ defined by Equations (33) and (35) in Online Appendix A.7 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 computed once $\hat{\boldsymbol{\theta}}(\cdot)$ is obtained because the distribution of the treatment combination $\boldsymbol{T}$ is known. See Section 5.2 for details.

Finally, we apply our DeDL framework to identify the treatment combination with the highest ATE, $\boldsymbol{t}^* := \arg\max_{t\in\mathcal{T}} \mu(\boldsymbol{t})$. Specifically, we identify the "best treatment" as $\hat{\boldsymbol{t}}^*_{\mathsf{DeDL}} := \arg\max_{t\in\mathcal{T}} \hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$. Define $\hat{\mu}^*_{\mathsf{DeDL}} := \max_{t\in\mathcal{T}} \hat{\mu}_{\mathsf{DeDL}}(\boldsymbol{t})$ as the ATE of the best treatment identified by our DeDL framework. We construct the estimators of ATE increment from treatment combination $\boldsymbol{t}$ to "best treatment" as $\hat{\tau}_{\mathsf{DeDL}}(\cdot) := \hat{\mu}^*_{\mathsf{DeDL}} - \hat{\mu}_{\mathsf{DeDL}}(\cdot)$ and $\widehat{\mathcal{CI}}_{\mathsf{DeDL}}(\cdot;\tau)$ defined by Equations (36) and (40) in Online Appendix A.8 to each treatment combination $\boldsymbol{t} \in \mathcal{T}$ and select the treatment combination(s) $\boldsymbol{t}$ with $\hat{\tau}_{\mathsf{DeDL}}(\boldsymbol{t})$ insignificant from zero as best treatment(s).

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 five commonly used approaches as benchmarks: (a) the LA approach, (b) the LR approach, (c) the pure deep learning (PDL) approach, and (d) the structured deep learning (SDL) approach. For implementation details of all four benchmarks, see Online 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, and thus, it is widely adopted by most 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 treatment 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 $\boldsymbol{t}$ are 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 treatment by selecting the treatment combination with the highest ATE based on the *predicted* outcomes of all treatment combinations. We remark that 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 for a 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 treatment 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 functions with debiasing.

The SDL approach is exactly the same as the DeDL approach with only one distinction. Unlike DeDL, which uses the influence function to debias the estimation from DL, the SDL approach simply uses the prediction from the DL to construct plug-in estimator $\hat{\mu}(\boldsymbol{t}) = \frac{1}{n}\sum_{i=1}^{n} H(\boldsymbol{x}_i, \hat{\boldsymbol{\theta}}(\boldsymbol{x}_i); \boldsymbol{t}, \boldsymbol{t}_0)$. Similar to the LR and PDL estimators, 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 $\hat{\mu}_\pi(\boldsymbol{t})$ ($\widehat{\mathcal{CI}}_\pi(\boldsymbol{t};\mu)$) to denote the ATE estimator (the confidence interval of $\hat{\mu}_\pi(\boldsymbol{t})$) generated by π for the treatment combination $\boldsymbol{t} \neq \boldsymbol{t}_0$. Likewise, we use $\hat{\tau}_\pi(\boldsymbol{t})$ ($\widehat{\mathcal{CI}}_\pi(\boldsymbol{t};\tau)$) to denote the estimator for the ATE difference between the optimal treatment $\boldsymbol{t}^*$ and the experiment combination $\boldsymbol{t} \neq \boldsymbol{t}_0$ (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 of 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 (CDR). 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 $\mu(\boldsymbol{t})$ has 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. Define the correct direction ratio for unobservable treatment combinations (CDRu) of π as $\mathcal{CDR}_u(\pi) = \frac{|\mathcal{T}_u^{cd}(\pi)|}{|\mathcal{T}_u|} \times 100\%$.
- Mean absolute percentage error (MAPE). 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 mean absolute percentage error 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 (MSE). The mean squared error 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 (MAE). The mean absolute error of π for unobservable treatment combinations (MAEu) is defined analogously as $\frac{1}{|\mathcal{T}_u|}\sum_{\boldsymbol{t}\in\mathcal{T}_u}|\mu(\boldsymbol{t}) - \hat{\mu}_\pi(\boldsymbol{t})|$.

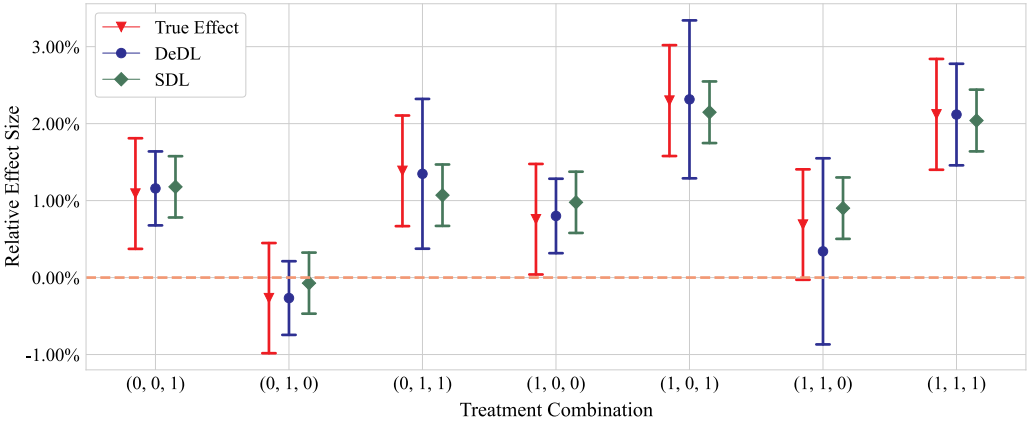
To give a clear picture of the comparisons among *all* eight (observable and unobservable) treatment combinations and identify the best treatment of them (see Section 4.5), we present the estimated treatment effects of all treatment combinations in Figure 5. In Table 3, we also calculate the above four metrics evaluated on both *unobserved* treatment combinations and *all* treatment combinations.

Table 3 documents the comparison of our DeDL approach against the LA, LR, SL, PDL, and SDL benchmarks with respect to the four performance metrics described above. The punchline message of the empirical analysis is that our DeDL estimator substantially outperforms all five benchmarks with any of these performance metrics, providing evidence that the proposed framework accurately estimates and infers 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 nonlinearity 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 mentioned, 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 5, 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 5 shows that the bias correction can help causal inference in two significant ways. First, by correcting the bias because of 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, whereas 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

Figure 5. (Color online) Detailed Comparisons Between the SDL and DeDL Estimators



Note. Bars represent confidence intervals.

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 specifying a concrete link function, we indeed sacrifice some prediction accuracy for the ability to derive bias correction terms and allow economic interpretation. However, because 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 applying the corresponding DML bias correction will outweigh the cost of a less flexible relationship between the treatment conditions and the outcome.

Last, we have also tested the standard logit model proposed by Farrell et al. (2020) such that $G(\boldsymbol{\theta}(\boldsymbol{x}), \boldsymbol{t}) = \frac{1}{1+\exp(-(\theta_0(\boldsymbol{x})+\theta_1(\boldsymbol{x})\sum_{i=1}^{m} t_i))}$. Despite the first-stage crossvalidation losses being similar for both the standard logit model and our DeDL, with the scaled MSE values of 0.0087 and 0.0086, respectively, we observe that the latter significantly outperforms the former in ATE estimation. The limited flexibility of the standard logit function weakens its effectiveness in accurately estimating ATE.

Table 3. Comparison of Different Estimators of ATE

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

Notes. The calculations of MSE, MSEu, MAE, and MAEu are based on the scaled outcome variable (see column (1) in Table 2). MSE and MSEu are scaled by multiplying a constant. MAE and MAEu are scaled by multiplying another constant.

4.5. Results on Best Treatment Identification

This subsection applies our DeDL framework to identify the optimal treatment combination with the highest ATE. By the ground-truth ATE estimates (column (1) in Table 2), the "true" optimal treatment combination is $\boldsymbol{t}^* = (1,0,1)'$. We compare different estimators for best treatment identification in Table 4, and we report the same set of performance metrics defined in Section 4.4. In particular, we focus on the comparison between $\tau(\boldsymbol{t})$ and $\hat{\tau}(\boldsymbol{t})$, the ground-truth treatment effect increment and its estimator. Among eight treatment combinations, the treatment effect increments of the "true" optimal treatment combination over both $(1,0,1)$ and $(1,1,1)$ insignificantly differ from zero. Column (1) in

Table 4. Comparison of Different Estimators of Best Treatment 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

Notes. The calculations of MSE and MAE are based on the scaled outcome variable (see column (1) in Table 2). MSE is scaled by multiplying a constant. MAE is scaled by multiplying another constant.

Table 4 indicates that only SDL and DeDL correctly identify both sign and statistical significance of all treatment effect increments. Table 4 shows that DeDL estimators significantly outperform the LA, LR, PDL, and SDL benchmarks with respect to all performance metrics in identifying the best treatment combination.

4.6. Results on Conditional Average Treatment Effects

To deepen our empirical analysis, we further validate the conditional average treatment effect (CATE) estimation across different demographic subgroups. Specifically, we classify users into four distinct groups based on two key demographic covariates: residential location (rural versus nonrural) and phone price tier (medium priced versus nonmedium priced[10]). This classification yields four distinct user groups: (1) nonrural area with a medium-priced phone (NM), denoted as $\mathcal{X}_{NM}$; (2) nonrural area with a nonmedium-priced phone (NN), denoted as $\mathcal{X}_{NN}$; (3) rural area with a medium-priced phone (RM), denoted as $\mathcal{X}_{RM}$; and (4) rural area with a nonmedium-priced phone (RN), denoted as $\mathcal{X}_{RN}$. This classification enables us to better understand the performance of DeDL across different demographic segments.

The implementation of CATE estimation follows the same fourfold crossfitting described in Section 5.2. For each fold s, we construct the CATE estimator $\hat{\mu}_s(\boldsymbol{t}|\mathcal{X}_j)$ for treatment combination $\boldsymbol{t}$ conditional on demographic segments $\boldsymbol{X} \in \mathcal{X}_j$ $(j = NM, NN, RM, RN)$. This estimator is computed by averaging $\psi(\boldsymbol{z}, \hat{\boldsymbol{\theta}}_s, \hat{\boldsymbol{\Lambda}}_s; \boldsymbol{t}, \boldsymbol{t}_0)$ across all observations where $\boldsymbol{x} \in \mathcal{X}_j$. The final CATE estimator $\mu(\boldsymbol{t}|\mathcal{X}_j) := \mathbb{E}[Y|\boldsymbol{T} = \boldsymbol{t}, X \in \mathcal{X}_j] - \mathbb{E}[Y|\boldsymbol{T} = \boldsymbol{t}_0, X \in \mathcal{X}_j]$ is obtained by taking the average of $\hat{\mu}_s(\boldsymbol{t}|\mathcal{X}_j)$ across four folds. The empirical results for CATE estimation are presented in Tables 5 and 6. Table 5 presents the group-size-weighted average metrics for CATE estimation on both unobserved treatment combinations and all treatment combinations across four distinct groups. Table 6 provides a detailed comparison of MAPE across different estimators for the CATE estimation with respect to four groups. Consistent with our main results on ATE estimation in Table 3, our method outperforms all other benchmarks in CATE estimation as well.

Our comprehensive empirical analyses, presented in Tables 3–6, demonstrate the superior performance of the DeDL estimator in three critical aspects: estimating ATEs of unobserved conditions, identifying optimal treatment combinations, and calculating CATEs across demographic segments. We highlight that this paper is among the first works to empirically investigate the accuracy of the DML framework to recover the ground-truth treatment effects through large-scale field experiments. Our empirical evidence also sheds light on how crucial bias correction is in estimating the causal effects, and it 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. Specifically, our carefully designed model layer within the DeDL framework, tailored for multiple-experiment settings, substantially outperforms standard unstructured deep learning approaches. We note that achieving these improvements requires extensive effort in model fine-

Table 5. Comparison of Different Estimators of CATE

	Unobserved treatment combinations		
Estimator	MAPEu, % (1)	MSEu (2)	MAEu (3)
LA	51.34	11.197	29.959
LR	26.11	5.898	18.129
PDL	17.03	4.252	12.662
SDL	26.47	5.425	18.191
DeDL	8.31	1.528	6.614
	All treatment combinations		
Estimator	MAPE, % (4)	MSE (5)	MAE (6)
LA	36.64	8.895	20.807
LR	29.04	4.500	17.732
PDL	17.34	2.593	11.142
SDL	22.77	3.292	14.177
DeDL	10.32	1.062	6.486

Notes. The calculations of MSE, MSEu, MAE, and MAEu are group size-weighted average. MSE and MSEu are scaled by multiplying a constant. MAE and MAEu are scaled by multiplying another constant.

Table 6. MAPE Comparison of Different Estimators of CATE for Four Groups

	Unobserved treatment combinations			
Estimator	NM, % (774,326)	NN, % (460,330)	RM, % (451,494)	RN, % (380,465)
LA	48.39	101.53	15.46	39.18
LR	18.46	18.90	42.69	30.74
PDL	0.12	12.52	33.94	36.8
SDL	23.20	5.66	44.46	36.96
DeDL	0.04	0.66	15.23	26.22
	All treatment combinations			
Estimator	NM, % (774,326)	NN, % (460,330)	RM, % (451,494)	RN, % (380,465)
LA	24.19	101.53	6.18	19.58
LR	28.40	18.90	36.63	33.46
PDL	7.60	12.51	27.82	30.57
SDL	21.63	5.66	38.23	27.43
DeDL	7.17	0.66	18.71	18.46

Note. The sizes of the groups are indicated in parentheses.

tuning, consistent with well-documented challenges in deep learning implementation.

5. Key Checkpoints in the Empirical Implementation

Whereas Section 3 outlines the theoretical framework for applying DML-based causal inference with DNNs, our empirical success, as discussed in Section 4, hinges on several key steps of practical implementation. In this section, drawing from our own experiences in deploying the DeDL framework in the multiple-experiment setting, we provide a detailed discussion of these crucial checkpoints, aiming to guide similar applications in other empirical contexts.

5.1. Covariate Balancing with Stratified Sampling

We observe from the DGP (1) that unconfoundedness is typically required when applying DML. However, in practice, the covariates may be often high dimensional. Even if the treatment assignment is well controlled and genuinely unconfounded as in our setting (see the definition of $\nu(\cdot|\cdot)$ in Section 3.1), the empirical correlation between covariates and treatment assignment could deviate significantly from the intended design. This can obstruct effective empirical implementation.

Thus, for a fair comparison of ATEs over different treatment combinations and keeping the user covariates similarly distributed under different treatment combinations in $\mathcal{T}$, in our empirical implementation, we adopt stratified sampling to randomly select 2,066,606 users from those who are targeted by all three experiments.[11] We emphasize that although such a stratified sampling is not part of the requirement in DML, we still find it critical for the practical effectiveness.

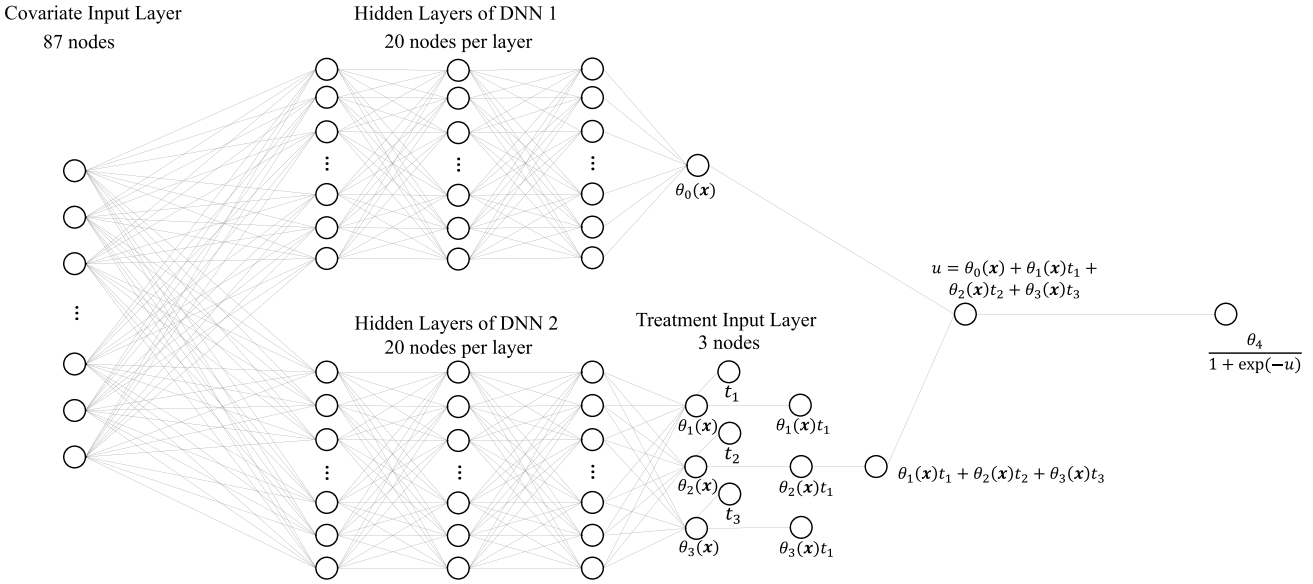
Specifically, based on the moments and quantiles of the covariate distribution, we partition the covariate space into 69,111 strata, and then, we randomly sample the same number of users whose covariates lie within the stratum for each treatment combination. After the stratified sampling, we construct a new data set that has about 258,325 users under each treatment combination (see column (3) in Table 2), and hence, there are 2,066,606 users in total. Hereafter, we call the data sample after stratified sampling the *stratified sample*, and all of the empirical analyses from now on are performed on the stratified sample. For the stratified sample, it can be seen that $\mathbb{P}[T_k = 1] = \mathbb{P}[T_k = 0] = 0.5 \quad (k = 1, 2, 3)$, independently distributed for different A/B tests. We detail the exact procedure of our stratified sampling in Online Appendix B.2. To confirm the success of randomization among our stratified sample of users, we compare users under different treatment combinations of 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. The complete randomization check results are listed in Table A2 in Online Appendix B.3. Table A2 in Online Appendix B.3 shows that 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 the 10 days before 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 are >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 A2 in Online Appendix B.3 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.

5.2. Building and Training the DNN with Care

In this section, we present the implementation details of using DeDL to estimate ATE. We emphasize that from our experience, it takes a substantial amount of effort to build and train the DNN in the DeDL framework. Failure to train a DNN with low crossvalidation errors is likely to invalidate estimation and inference in the second stage as detailed in Section 5.3.

We use the fourfold crossfitting proposed in Chernozhukov et al. (2018) to obtain the 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 observations. 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 a structured DNN (see Figure 6) and get an estimator of the unknown nuisance parameters $\hat{\boldsymbol{\theta}}_s(\cdot)$. To ascertain the architecture of DNN, we have explored one DNN for both $\theta_0^*(\cdot)$ and $\theta_k^*(\cdot)$ $(k \in \{1, 2, 3\})$ and two separate DNNs for $\theta_0^*(\cdot)$ and $\theta_k^*(\cdot)$ $(k \in \{1, 2, 3\})$. We also experimented with the number of layers equal to 3, 5, and 10 and the number of nodes equal to 10, 20, 50, and 100 per layer. We eventually adopted the DNN with a structure of combined two 3-layer DNNs with 20 nodes per layer that achieves the minimal mean squared loss on the validation data. To fit the structured DNN, we tested a range of learning rates from 0.00001 to 0.1 and applied

Figure 6. Structure of the Deep Neural Network Used in the Empirical Analysis



batch sizes of 32, 100, 200, and 500. By conducting a grid search over the combinations of hyperparameters, we set *learning rate* = 0.0001 and *batch size* = 100, which achieve the lowest crossvalidation loss. For each treatment combination $\boldsymbol{t}$, we use fold I_s to construct the ATE estimator $\hat{\mu}_s(\boldsymbol{t})$ following Equation (33) in Online Appendix A.7.

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 $\psi(\boldsymbol{z}, \hat{\boldsymbol{\theta}}_s, \hat{\boldsymbol{\Lambda}}_s; \boldsymbol{t}, \boldsymbol{t}_0)$ for each data point $\boldsymbol{z}$ in fold I_s. To be specific, after obtaining $\hat{\boldsymbol{\theta}}_s(\cdot)$, we calculate $\hat{\boldsymbol{\Lambda}}_s(\boldsymbol{x}) = \frac{2}{5}\sum_{\boldsymbol{t}\in\mathcal{T}_o} G_{\boldsymbol{\theta}}(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}), \boldsymbol{t}) G_{\boldsymbol{\theta}}(\hat{\boldsymbol{\theta}}_s(\boldsymbol{x}), \boldsymbol{t})'$ because the distribution of the observed treatment combination $\boldsymbol{t} \in \mathcal{T}_o$ is known in the stratified sample with $\mathbb{P}[\boldsymbol{T} = \boldsymbol{t} | \boldsymbol{X} = \boldsymbol{x}] = 0.2$. We remark that $\lambda(\boldsymbol{x})$ may be noninvertible for certain values of $\boldsymbol{x}$ because of numerical precision issues in empirical implementation. We can add an identity matrix multiplied by a small regularization parameter (e.g., 0.001, 0.01, and 1) to address this. The regularization parameter can be fine-tuned using grid search to improve performance and ensure invertibility in cases where direct inversion is affected by precision limitations. By averaging $\psi(\boldsymbol{z}, \hat{\boldsymbol{\theta}}_s, \hat{\boldsymbol{\Lambda}}_s; \boldsymbol{t}, \boldsymbol{t}_0)$, we obtain the estimated ATE of treatment combination $\boldsymbol{t}$ in fold index s as $\hat{\mu}_s(\boldsymbol{t})$.

The ATE estimate and the 95% confidence interval for each treatment combination in each fold are presented in the first four rows in each section in Table 7. After fourfold crossfitting, we aggregate $\hat{\mu}_s(\hat{\boldsymbol{t}})$ by taking their average value as the final estimator $\hat{\mu}(\boldsymbol{t})$ for each treatment combination $\boldsymbol{t}$. We report the results of the final estimator in the last row of each section in Table 7.

The estimators for best treatment identification are obtained through similar implementation procedures. Each fold will generate its estimators for the advantage of the best treatment over any treatment combination $\boldsymbol{t}$, $\tau(\boldsymbol{t})$, for $\boldsymbol{t} \neq \boldsymbol{t}^*$. Aggregating the estimators from four folds generates the DeDL estimator for the best treatment identification.

5.3. Using Training Error as the Compass Toward Success

Analyzing the theory behind our DeDL framework, it is clear that the convergence of nuisance parameter estimation, emphasized in Proposition 1(b), underpins the asymptotic normality established in Theorem 2 and subsequent valid inference. One may also refer to Online Appendix D.3 for further discussions regarding the impact of biased nuisance parameter estimation for DML. However, in practice, the true values of the nuisance parameters $\theta^*(\boldsymbol{x})$ are unobservable, making it impossible to directly assess the quality of this estimation. This raises a key practical question. How can we be confident that we are on the right track? In this context, the readily accessible crossvalidation error, which we refer to as the training error for simplicity, of the DNN can serve as a crucial proxy.

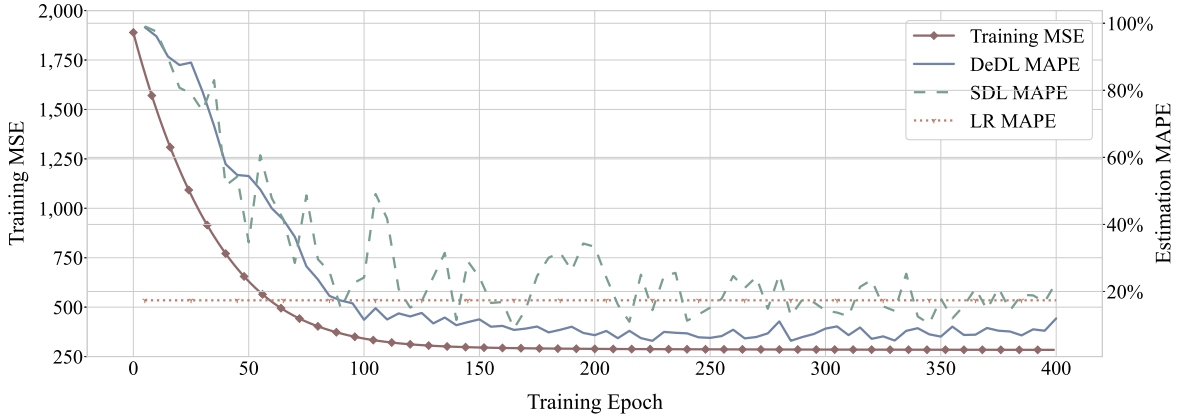
Specifically, we compare the estimation MAPE for all treatment combinations of the DeDL estimator with SDL and LR estimators under increasing DNN training epochs in Figure 7. As demonstrated in Figure 7, both DeDL and SDL estimators yield smaller MAPE as DNN training mean squared loss is smaller. We highlight that when DNN is poorly trained (i.e., within 100 epochs), the DeDL and SDL estimators have similar estimation MAPEs, which are dominated by the LR estimator. In this case, debias does not benefit causal effect estimation. However, as the DNN converges, DeDL

Table 7. Detailed Results of Fourfold 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]	04.22	0.212	0.461
		Mean	−1.158	[−0.678, −1.639]	06.14	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

Notes. The calculations of APE, SE, and AE are based on the scaled outcome variable (see column (1)). SE is scaled by multiplying a constant. AE is scaled by multiplying another constant. The significance levels are indicated with asterisks. NA, not applicable.
*$p < 0.05$; **$p < 0.01$; ***$p < 0.001$; ****$p < 0.0001$.

Figure 7. (Color online) MAPE Comparison with the DNN Training Epoch



starts to show significant advantages in estimation accuracy compared with both SDL and LR estimators. This phenomenon is well connected to the classical literature on semiparametric statistics, which requires that the convergence rate of the estimated parameter functions $\hat{\theta}(\cdot)$ be 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 debiasing via DML. We remark that there is no single recipe for judging whether a training error is sufficiently small for good estimation and inference, which requires context-specific judgmental calls based on domain knowledge. Additional discussions of the training error are provided in Online Appendix D.4.

6. Synthetic Experiments

Using synthetic experiments, we gauge the robustness of our approach under different scenarios. Because of the space limit, we defer the details of these discussions to the Online Appendix and only provide a high-level summary here. We first validate our theory by varying the number of experiments with $m \in \{4, 6, 8, 10\}$ in Online Appendix D.2. This experiment shows that DeDL consistently outperforms across all m values. As m increases, simpler models, like LA and LR, experience rapid performance degradation because of their oversimplified linear extrapolation, which fails to capture complex treatment effects. In contrast, the performances of SDL and DeDL remain relatively stable. In Online Appendix D.3, we test the performance of our DeDL estimators with a potentially large bias to estimate $\hat{\boldsymbol{\theta}}(\cdot)$; we find that DeDL is fairly robust, with moderate biases. However, when this bias becomes excessively large, the effectiveness of DeDL diminishes. This highlights the importance of exercising careful DNN training during the first stage of the process. We also systematically assess the performance of DeDL under model misspecification and shed light on how to test and select the link function in practice (Online Appendix D.4). Resonating with our discussions in Section 5.3, this analysis highlights the crucial role of training or in-sample validation error in DML with DNNs. Furthermore, in Online Appendix D.5, we investigate a practical setting where the observed $\boldsymbol{X}$ distribution deviates from the population, and we discuss how to use the rebalancing method to get trustworthy estimates.

7. Conclusion

In this paper, we leverage the DeDL framework to infer treatment effects for concurrent experiments and to identify the best treatment combination. We show the superior performance of our method using data from three A/B tests on a large-scale online 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 by discussing two future directions. First, for multilevel discrete and continuous treatments, although our link functions can still be applied, researchers could design more flexible link functions that better fit the richer treatment assignment mechanisms for the identification and convergence of parameter functions. Second, researchers could combine the current framework with classical causal inference methods, such as instrumental variables and difference in differences, for more general applications.

Acknowledgments

The authors thank Department Editor Vivek Farias, the anonymous associate editor, and three referees for their very helpful and constructive comments, which have led to significant improvements in both the content and exposition of this study. The authors also thank the industry partner for their support in sharing the data, implementing the algorithm, and conducting the experiment.

Endnotes

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

[2] See https://www.mckinsey.com/capabilities/growth-marketing-and-sales/our-insights/prime-day-and-the-broad-reach-of-amazons-ecosystem.

[3] See https://github.com/zikunye2/deep_learning_based_causal_inference_for_combinatorial_experiments.git.

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

[5] Information on notations is as follows. 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 are represented by lowercase 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 $\boldsymbol{M} > 0$ to denote that matrix $\boldsymbol{M}$ is positive definite.

[6] 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 $\boldsymbol{Y} \in \mathbb{R}^{d_Y}$ $(d_Y > 1)$ is straightforward.

[7] Throughout this paper, we make a regularity assumption commonly used in the DNN-estimation literature (e.g., Yarotsky 2017) (i.e., Assumption 3 in Online Appendix A.2), which requires that true parameter $\boldsymbol{\theta}^*(\cdot)$ is uniformly bounded and sufficiently smooth.

[8] The notation "$\lesssim$" means less than or equal to up to a constant independent of K.

[9] For the sake of simplicity, we refer to the changes of several parameters in the recommendation algorithm as a 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.

[10] Medium priced is defined as RMB 1,000–2,000.

[11] 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 eight conditions. This leads to inconsistency to (3) and the theoretical discussion (Proposition 2) if one directly uses sample means as the ground truth to validate the treatment effects. One solution would be to redefine the treatment effect and rederive a rather complex new influence function that admits different covariate distributions, but we opt for a simpler approach: the stratified sampling.

References

Adcock B, Dexter N (2021) The gap between theory and practice in function approximation with deep neural networks. *SIAM J. Math. Data Sci.* 3(2):624–655.

Angrist JD, Pischke JS (2009) *Mostly Harmless Econometrics: An Empiricist's Companion* (Princeton University Press, Princeton, NJ).

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

Athey S, Imbens G (2016) Recursive partitioning for heterogeneous causal effects. *Proc. Natl. Acad. Sci. USA* 113(27):7353–7360.

Athey S, Imbens GW, Wager S (2018) Approximate residual balancing: Debiased inference of average treatment effects in high dimensions. *J. Roy. Statist. Soc. Ser. B Statist. Methodology* 80(4): 597–623.

Bertsimas D, Imai K, Li ML (2022) Distributionally robust causal inference with observational data. Preprint, submitted October 15, https://arxiv.org/abs/2210.08326.

Bojinov I, Simchi-Levi D, Zhao J (2023) Design and analysis of switchback experiments. *Management Sci.* 69(7):3759–3777.

Box GEP, Hunter WG, Hunter JS (1978) *Statistics for Experimenters: An Introduction to Design, Data Analysis, and Model Building* (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 Sci.* 61(5):949–962.

Candogan O, Chen C, Niazadeh R (2024) Correlated cluster-based randomized experiments: Robust variance minimization. *Management Sci.* 70(6):4069–4086.

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

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

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

Chiang HD, Kato K, Ma Y, Sasaki Y (2022) Multiway cluster robust double/debiased machine learning. *J. Bus. Econom. Statist.* 40(3): 1046–1056.

Dasgupta T, Pillai NS, Rubin DB (2015) Causal inference from 2^K factorial designs by using potential outcomes. *J. Roy. Statist. Soc. Ser. B Statist. Methodology* 77(4):727–753.

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

Edelman B, Luca M, Svirsky D (2017) Racial discrimination in the sharing economy: Evidence from a field experiment. *Amer. Econom. J. Appl. Econom.* 9(2):1–22.

Fan Q, Hsu YC, Lieli RP, Zhang Y (2022) Estimation of conditional average treatment effects with high-dimensional data. *J. Bus. Econom. Statist.* 40(1):313–327.

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

Farias V, Li A, Peng T (2021) Learning treatment effects in panels with general intervention patterns. *Adv. Neural Inform. Processing Systems* 34:14001–14013.

Farrell MH, Liang T, Misra S (2020) Deep learning for individual heterogeneity: An automatic inference framework. Preprint, submitted October 28, https://arxiv.org/abs/2010.14694.

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

Goli A, Lambrecht A, Yoganarasimhan H (2024) A bias correction approach for interference in ranking experiments. *Marketing Sci.* 43(3):590–614.

Gordon BR, Moakler R, Zettelmeyer F (2023) Close enough? A large-scale exploration of non-experimental approaches to advertising measurement. *Marketing Sci.* 42(4):768–793.

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

Hornik K, Stinchcombe M, White H (1989) Multilayer feedforward networks are universal approximators. *Neural Networks* 2(5):359–366.

Imbens GW, Rubin DB (2015) *Causal Inference in Statistics, Social, and Biomedical Sciences* (Cambridge University Press, Cambridge, UK).

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

Kallus N, Mao X, Udell M (2018) Causal inference with noisy and missing covariates via matrix factorization. *Adv. Neural Inform. Processing Systems* 31:6921–6932.

Kingma DP (2014) Adam: A method for stochastic optimization. Preprint, submitted December 22, https://arxiv.org/abs/1412.6980.

Knaus MC (2022) Double machine learning-based programme evaluation under unconfoundedness. *Econometrics J.* 25(3):602–627.

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

Kohavi R, Tang D, Xu Y (2020) *Trustworthy Online Controlled Experiments: A Practical Guide to A/B Testing* (Cambridge University Press, Cambridge, UK).

Lee MR, Shen M (2018) Winner's curse: Bias estimation for total effects of features in online controlled experiments. *Proc. 24th ACM SIGKDD Internat. Conf. Knowledge Discovery Data Mining* (ACM, New York), 491–499.

Lim AE, Shanthikumar JG, Shen ZM (2006) Model uncertainty, robust optimization, and learning. *Models, Methods, and Applications for Innovative Decision Making* (INFORMS, Cantonsville, MD), 66–94.

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

Newey WK (1994) The asymptotic variance of semiparametric estimators. *Econometrica* 62(6):1349–1382.

Pashley NE, Bind MAC (2023) Causal inference for multiple treatments using fractional factorial designs. *Canadian J. Statist.* 51(2):444–468.

Song Y, Sun T (2024) Ensemble experiments to optimize interventions along the customer journey: A reinforcement learning approach. *Management Sci.* 70(8):5115–5130.

Tang D, Agarwal A, O'Brien D, Meyer M (2010) Overlapping experiment infrastructure: More, better, faster experimentation. *Proc. 16th ACM SIGKDD Internat. Conf. Knowledge Discovery Data Mining* (ACM, New York), 17–26.

Tang J, Qi Z, Fang E, Shi C (2025) Offline feature-based pricing under censored demand: A causal inference approach. *Manufacturing Service Oper. Management* 27(2):535–553.

Wager S, Athey S (2018) Estimation and inference of heterogeneous treatment effects using random forests. *J. Amer. Statist. Assoc.* 113(523):1228–1242.

Wooldridge JM (2010) *Econometric Analysis of Cross Section and Panel Data* (MIT Press, Cambridge, MA).

Wu CFJ, Hamada MS (2011) *Experiments: Planning, Analysis, and Optimization* (John Wiley & Sons, Hoboken, NJ).

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

Xiong T, Wang Y, Zheng S (2020) Orthogonal traffic assignment in online overlapping A/B tests. EasyChair technical report, Tencent Inc., Shenzhen, China.

Xiong R, Chin A, Taylor S (2023) Bias-variance tradeoffs for designing simultaneous temporal experiments. *The KDD'23 Workshop Causal Discovery, Prediction Decision* (PMLR, New York), 115–131.

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 (2023) Cold start to improve market thickness on online advertising platforms: Data-driven algorithms and field experiments. *Management Sci.* 69(7):3838–3860.

Zeng Z, Dai H, Zhang DJ, Zhang H, Zhang RP, Xu Z, Shen Z-JM (2023) The impact of social nudges on user-generated content for social network platforms. *Management Sci.* 69(9): 5189–5208.

Zhang Y, Politis DN (2022) Ridge regression revisited: Debiasing, thresholding and bootstrap. *Ann. Statist.* 50(3):1401–1422.

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 Sci.* 66(6): 2589–2609.