International Journal of Forecasting 39 (2023) 541–555

Contents lists available at ScienceDirect

International Journal of Forecasting

journal homepage: www.elsevier.com/locate/ijforecast

How to "improve" prediction using behavior modification[☆]

Galit Shmueli[a,*], Ali Tafti[b]

[a] National Tsing-Hua University College of Technology Management, Hsinchu, Taiwan
[b] University of Illinois at Chicago, United States of America

ARTICLE INFO

Keywords:
Behavior modification
Behavioral big data
Machine learning
Prediction error
Causal intervention
Internet platforms
Reinforcement learning

ABSTRACT

Many internet platforms that collect behavioral big data use it to predict user behavior for internal purposes and for their business customers (e.g., advertisers, insurers, security forces, governments, political consulting firms) who utilize the predictions for personalization, targeting, and other decision-making. Improving predictive accuracy is therefore extremely valuable. Data science researchers design algorithms, models, and approaches to improve prediction. Prediction is also improved with larger and richer data. Beyond improving algorithms and data, platforms can stealthily achieve better prediction accuracy by pushing users' behaviors towards their predicted values, using behavior modification techniques, thereby demonstrating more certain predictions. Such apparent "improved" prediction can result from employing reinforcement learning algorithms that combine prediction and behavior modification. This strategy is absent from the machine learning and statistics literature. Investigating its properties requires integrating causal with predictive notation. To this end, we incorporate Pearl's causal *do*(.) operator into the predictive vocabulary. We then decompose the expected prediction error given behavior modification and identify the components impacting predictive power. Our derivation elucidates implications of such behavior modification to data scientists, platforms, their customers, and the humans whose behavior is manipulated. Behavior modification can make users' behavior more predictable and even more homogeneous; yet this apparent predictability might not generalize when business customers use predictions in practice. Outcomes pushed towards their predictions can be at odds with customers' intentions, and harmful to manipulated users.

1. Introduction: Prediction, prediction products, and behavior modification

Recent years have seen an incredible growth in predictive modeling of user behavior using behavioral big data in both industry and in academia. Behavioral big data are large and highly detailed datasets on human and social actions and interactions (Shmueli, 2017). Predictions based on such data now shape almost every aspect of modern life (Agrawal, Gans, & Goldfarb, 2018). Internet platforms, such as Google and Facebook, predict user behavior for internal purposes and for their customers who utilize the predictions for personalization, targeting, and other decision-making. Predicted behaviors of interest to platforms and their customers include the probability of purchase, churn, engagement, and even behaviors such as voting intentions and life events, such as pregnancy[1]. Henceforth in this paper, we use the term *customers* to refer to the platform's business customers,

[☆] This is an invited paper.
[*] Corresponding author.
E-mail address: galit.shmueli@iss.nthu.edu.tw (G. Shmueli).

[1] www.nytimes.com/2012/02/19/magazine/shopping-habits.html

https://doi.org/10.1016/j.ijforecast.2022.07.008

which we distinguish from the platform's *users*, whose behavior is subject to prediction and modification.

1.1. Prediction products

In *The Age of Surveillance Capitalism*, Zuboff (2019) describes the processes used by several large internet platforms that collect behavioral big data to package the raw material of users' actively shared data and passively generated data (e.g., location data, friendship ties, device information) into *prediction products*, which are then sold to business customers such as insurance companies, marketers, advertisers, security forces, governments, and political consulting firms. Prediction products "anticipate what you will do now, soon, and later" (Zuboff, 2019, p. 8). Such predictions are typically used by platform customers to modify and shape users' behavior toward desired commercial ends or other outcomes.[2] [3]

One example of a prediction product is the recently launched Google Analytics *predictive metrics* service, which "automatically enriches your data by bringing Google machine-learning expertise to bear on your dataset to predict the *future behavior* of your users".[4] Another example is Facebook's *loyalty prediction* service, which offers advertisers the ability to target users based on how they *will* behave, what they *will* buy, and what they *will* think[5].

Zuboff's (2019) book, which sounds an alarm about the increasingly intrusive and exploitative practices of digital platform firms, has been received with both acclaim and criticism. Her work takes a condemnatory stance toward the major digital platforms, arguing that their practices are trending not just toward influence but also toward pervasive control of human behavior, and thus they pose a great danger to the autonomous and lived experience of humanity. We take an analytical approach by considering the potential scope and effects of behavioral modification should digital platforms choose to engage in such strategies. Regardless of one's disposition toward Zuboff (2019)'s arguments, we need a better understanding and careful analysis of how the scenarios depicted in her book might play out in practice. Hence, we focus on prediction products and the strategies that platforms might implement to capture their place in this market. Essentially, the more accurate these prediction products, the higher value they provide to the platforms' customers, and in turn the higher the revenues for the platforms. Platforms might also compete for customers of their prediction products. Hence, platforms have a strong incentive to improve the accuracy of predictions.

1.2. Modifying user behavior

Digital platforms now routinely use *behavior modification* (BMOD) techniques to change the behaviors of their users both online and offline. These behaviors can include clicking an ad, purchasing an item, posting sensitive information, visiting a doctor, and voting. BMOD techniques are classically defined as "an observable, replicable and irreducible component of an intervention designed to alter or redirect causal processes that regulate behavior" (Michie, et al., 2013). BMOD techniques derive from principles of behaviorist psychology and include nudging, herding, and operant conditioning, among others. The most popular technique is the *nudge*, defined as "any aspect of the choice architecture that alters people's behavior in a predictable way without forbidding any options or significantly changing their economic incentives" (Thaler & Sunstein, 2009, p. 6). Designers of choice environments take advantage of human cognitive limitations to manipulate the choice environment to subtly guide behavior by gently nudging them toward certain choices (Schneider, Weinmann, & Vom Brocke, 2018). Zuboff (2019) identified two more types of behavior modification: *herding*, which is controlling key elements in a person's immediate context in order to guide their behavior towards a predictable one; and *operant conditioning*, a term coined by the famous behavioral psychologist BF Skinner, which uses positive and negative reinforcement to encourage certain behaviors and extinguish others. When implemented on platforms, BMOD techniques are known as *persuasive technology* (Fogg, 2002) and can be used to adaptively and automatically tailor behavioral interventions to exploit unique psychological characteristics and motivations. Platform BMOD is implemented via various machine learning algorithms that operate in a data-driven, autonomous, interactive, and sequentially-adaptive manner (Greene, Martens, & Shmueli, 2022). Behavioral interventions range in their transparency: some are visible to users, such as chatbots, suggestions by recommender systems, and app notifications, while others are less so, such as A/B testing,[6] feed filtering, comment moderation on social networks, and deceptive interface design choices (Mathur, et al., 2019). Platforms employ BMOD to provide personalized services, increase user engagement, "hook" users by habit formation (Eyal, 2014), and generate further behavioral big data, among other purposes. Stanford university's Behavior Design Lab director BJ Fogg lists seven types of *persuasive technology* tools (Fogg, 2002). While the field of marketing has used

[2] www.theguardian.com/technology/2019/jan/20/shoshana-zuboff-age-of-surveillance-capitalism-google-facebook

[3] Zuboff (2019, p.15) explains how these prediction products are traded in a new kind of marketplace for behavioral predictions that she calls *behavioral futures markets*. As Andrew and Baker (2021, p. 566) elaborate, the valuable combination of "data, predictive algorithms, and behavioral modification techniques" creates such an emerging behavioral futures market. For example, Google's "clickthrough rate" was the first globally successful prediction product, and its ad markets were the first to trade in human futures (https://www.project-syndicate.org/onpoint/surveillance-capitalism-exploiting-behavioral-data-by-shoshana-zuboff-2020-01).

[4] "Purchase Probability, which predicts the likelihood that users who have visited your app or site will purchase in the next seven days…Churn Probability, predicts how likely it is that recently active users will not visit your app or site in the next seven days." https://blog.google/products/marketingplatform/analytics/new-predictive-capabilities-google-analytics/ archived at https://archive.ph/Zms2O

[5] https://theintercept.com/2018/04/13/facebook-advertising-data-artificial-intelligence-ai/

[6] Randomized experiments, including the kinds of A/B tests done by digital platforms, impose interventions on a random subset of users in order to evaluate the effect of those interventions.

behavior modification even prior to the advent of the internet (Nord & Peter, 1980), today's technologies and big data enable more covert, pervasive, and powerful manipulation due to their networked, continuously updated, dynamic and pervasive nature (Yeung, 2017). Zuboff (2019) explains,

> "These interventions are designed to enhance certainty by doing things: they nudge, tune, herd, manipulate, and modify behavior in specific directions by executing actions as subtle as inserting a specific phrase into your Facebook news feed, timing the appearance of a BUY button on your phone, or shutting down your car engine when an insurance payment is late." (p. 200)

While these examples do not necessarily involve prediction, prediction-based BMOD is common in recommendation systems, targeted advertising, precision marketing, and other personalized interventions intended to cause human users to change their behavior in a specific direction that is beneficial to the intervention initiator, such as toward longer online engagement, higher purchase propensity, or increased information sharing.

1.3. Combining prediction and behavior modification

We focus on the new capability obtained by combining prediction and BMOD strategies that are now available to platforms. Zuboff (2019) described how a platform that uses prediction and behavior modification for its own commercial gain can be in conflict with the well-being and agency of users. Engineering decisions made to maximize profitability can lead to changes in social systems that are drastic, opaque, effectively unregulated, and massive in scale (Bak-Coleman, et al., 2021). The historian Yuval Noah Harari suggested that such a combination of powers provides the capability to "hack" human beings[7]:

> "The ability to hack human-beings means the ability to understand humans better than they understand themselves. Which means being able to predict their choices…to manipulate their emotions, to make decisions for them".

Scholars from computer science, behavioral science, media studies, and law have pointed out the strong incentives for platforms to "make predictions true". Thus, we are interested in studying the possibilities of "improving" prediction by using BMOD in the sense of "making predictions come true".

Commenting on Facebook's loyalty prediction service,[8] law professor Frank Pasquale said that he worried how the company could turn algorithmic predictions into "self-fulfilling prophecies", since "once they've made this prediction, they have a financial interest in making it true".[9] Computer scientist and AI expert Stuart Russell described how platforms improve prediction by changing users' preferences so they become more predictable (Russell, 2019).[10] Media theorist Douglas Rushkoff explained that the better the platform is able to make users conform to their algorithmically determined destiny, the more it can "boast both its predictive accuracy and its ability to induce behavior change" (Rushkoff, 2019, p. 69).

We examine the combined prediction and BMOD capabilities from a data science point of view to study how such a combined strategy optimized to minimize prediction error can affect the different stakeholders: the platform, its customers purchasing prediction products, and the platform users. By describing the scenario in statistical terms, we show that aiming to minimize prediction error can result in misleading platform customers and manipulating humans in possibly dangerous directions, particularly when predicting *unwanted or risky behaviors* in the form of risk scores. An extreme example is predicting mental health risk for a healthcare stress reduction app. While the app maker aims to lower stress of high risk users, the platform can achieve low prediction error by turning high risk predictions into high risk realities.

The goal of this work is not to offer new evidence nor to characterize all the ways that digital platforms engage in behavioral modification strategies. Given that some firms have already demonstrated both the capabilities and incentives to implement various forms of BMOD, our goal is to introduce a technical vocabulary and notation that enables investigation of such strategies. Technical terminology and notation are needed in order to identify the properties and implications of the BMOD approach for the resulting predictive power. The thought process gives rise to various questions that we shall consider here and that future research can investigate in even more detail, including: can BMOD mask poor predictive capabilities? Can one infer the counterfactual of non-manipulated predictive power from the manipulated predictive power? Can customers who run routine A/B tests on the platform detect this scheme? What are the roles of personalized predictions and of personalized behavior modification within the error minimization strategy?

Our goal is to make transparent the effects of BMOD on predictive power, thereby facilitating the study of its impact on business, social, and humanistic aspects, as well as potential implications. In order to enable the analysis and evaluation of the effect of BMOD on predictive power, we use the *do*(.) operator proposed by Pearl (2009). This operator is useful for our approach because it expresses

[7] The TED Interview: "Yuval Noah Harari reveals the real dangers ahead" https://www.ted.com/talks/the_ted_interview_yuval_noah_harari_reveals_the_real_dangers_ahead, minute 34:15

[8] https://theintercept.com/2018/04/13/facebook-advertising-data-artificial-intelligence-ai/

[9] In addition, Frank Pasquale stated: "That is, once Facebook tells an advertising partner you're going to do some thing or other next month, the onus is on Facebook to either make that event come to pass, or show that they were able to help effectively prevent it (how Facebook can verify to a marketer that it was indeed able to change the future is unclear)".

[10] "Content selection algorithms on social media…are designed to maximize click-through. The solution is simply to present items that the user likes to click on, right? Wrong. The solution is to change the user's preferences so that they become more predictable". (Russell, 2019).

the manipulation of a variable within a system of causal relations. While *do* calculus is well developed for causal effects identification (Pearl, 2009), the challenge here lies in incorporating the causal *do*(.) operator into the existing correlation-based predictive framework.

The remainder of this paper is organized as follows. In Section 2 we describe the traditional statistical and machine learning approach for reducing prediction error. Section 3 introduces the new approach for reducing prediction error that relies on BMOD. In Section 4, we formalize the approach using statistical language and notation and analyze its implications. In Section 5 we discuss technical and business implications as well as humanistic and societal implications. Section 6 provides our conclusions.

2. Reducing prediction error by improving prediction

The fields of machine learning and statistics have been introducing new and improved models, algorithms, approaches, and even data, aimed at improving predictive power. Approaches such as regularization, boosting, and ensembles have proven highly useful in generating more precise predictions. From transparent regression models and tree-based algorithms, to more blackbox support vector machines, k-nearest neighbors, neural nets and especially deep learning algorithms, their justification and adoption lies in their ability to capture intricate signals linking inputs and a to-be-predicted output. Predictive performance is typically measured by out-of-sample prediction errors, which compare predicted values with actual values for new observations. More formally, the prediction error e_i for record i is defined as the difference between the actual outcome value y_i and its prediction $\hat{y}_i$, that is, $e_i = y_i - \hat{y}_i$. For a sample of n records, we have a set of actual outcome values $\boldsymbol{y} = [y_1, y_2, \ldots, y_n]$, a set of predicted outcome values $\hat{\boldsymbol{y}} = \left[\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n\right]$, and a set of prediction errors $\boldsymbol{e} = [e_1, e_2, \ldots, e_n]$. For each record i, we also have predictor information in the form of p measurements $\boldsymbol{x}_i = \left[x_{i,1}, \ldots, x_{i,p}\right]$. The predictor information for n records is contained in the matrix $\boldsymbol{X}$. Predicted values are obtained from $\hat{f}$, the model trained on a dataset of inputs $\boldsymbol{X}$ and actual outcomes $\boldsymbol{y}$ such that $\hat{\boldsymbol{y}} = \hat{f}(\boldsymbol{X})$.[11]

2.1. Targets of statistical and machine learning efforts

Machine learning algorithms and predictive statistical models[12] are designed and tuned to minimize an aggregation of the error values ($\boldsymbol{e}$) by operating on the predicted values ($\hat{\boldsymbol{y}}$). Improving predicted values is typically achieved by improving the following three components:

1. The structure of $\hat{f}$ that relates the predictor information $\boldsymbol{X}$ to the outcome;
2. The estimation/computation of $\hat{f}$ (e.g., through better algorithms); and
3. The quality and quantity of data ($\boldsymbol{X}$ and $\boldsymbol{y}$). Larger, richer behavioral datasets have been shown to improve predictive accuracy (Martens, Provost, Clark, & Junqué de Fortuny, 2016).

Towards this end, companies such as Google, Facebook, Uber, Netflix, and Amazon have been investing in improving predictions through collecting, buying, storing and processing unprecedented amounts and types of data. They have also hired top data science talent, acquired AI companies, and developed in-house predictive algorithms, platforms, and computational infrastructure.

In all these approaches, the actual outcome values $\boldsymbol{y}$ are considered fixed as if they represent unmanipulated outcomes. The top panel of Fig. 1 illustrates the statistical and machine learning approach for improving the above three components in order to minimize prediction error.

2.2. Components that affect predictive power: dissecting the expected predicted error (EPE)

When predicting an outcome y that is not expected to be manipulated between training and deployment, we anticipate a prediction error due to the inability of the model $\hat{f}$: (1) to correctly capture the underlying f even with unlimited training data (bias), (2) to correctly estimate f due to insufficient data (variance), and (3) to capture the errors for individual observations $\boldsymbol{\epsilon}$ (noise). For predicting a numerical outcome or probability for a new observation, these three sources are formalized through a bias–variance decomposition of the expected prediction error (EPE) using the squared-error loss[13] (Geman, Bienenstock, & Doursat, 1992):

$$\begin{aligned} EPE(\boldsymbol{x}) &= E\left[\left((y|\boldsymbol{x}) - \hat{f}(\boldsymbol{x})\right)^2\right] \\ &= E\left[\epsilon^2\right] + \left(f(\boldsymbol{x}) - E[\hat{f}(\boldsymbol{x})]\right)^2 + E\left[\left(\hat{f}(\boldsymbol{x}) - E[\hat{f}(\boldsymbol{x})]\right)^2\right] \\ &= \sigma^2 + \left[Bias(\hat{f}(\boldsymbol{x}))\right]^2 + Var(\hat{f}(\boldsymbol{x})). \end{aligned} \tag{1}$$

In statistics and machine learning, prediction is based on an assumption of continuity, where the predicted observations come from the same underlying processes and environment as the data used for training the predictive model and testing its predictive performance. The deterministic underlying function f and the random noise distribution are both assumed to remain unchanged between the time of model training and evaluation and the time

[11] We assume that a prediction is made at time t for an outcome y that will occur at a future time $t+k$ ($k > 0$). For record i, this can be written as $\hat{y}_i(t+k)|\boldsymbol{x}_i(t)$. However, we omit the time indexes for the sake of simplicity.

[12] The term *model* has different meanings in statistics and machine learning. We use the term *model* (or *predictive model*) in the statistical sense to represent the relationship between an outcome and predictors, denoted by f, independent of a particular dataset. We use *estimated model* or *trained model*, denoted by $\hat{f}$, to refer to the model obtained after being trained on data. We use the term *algorithm* for the procedure applied to the data that yields $\hat{f}$. For example, $y = \beta_0 + \beta_1 X + \epsilon$ is a model that can be trained on data using a maximum likelihood algorithm, resulting in the estimated model $\hat{y} = 5 + 3X$. In machine learning, *model* refers to the result of training an algorithm on data (equivalent to the *estimated model* when a statistical model exists). For example, a "classification tree model" in machine learning is the result of applying some tree algorithm to data.

[13] Assuming the underlying model $E[y|\boldsymbol{x}] = f(\boldsymbol{x}) + \epsilon$, where ϵ has zero mean and variance σ^2.

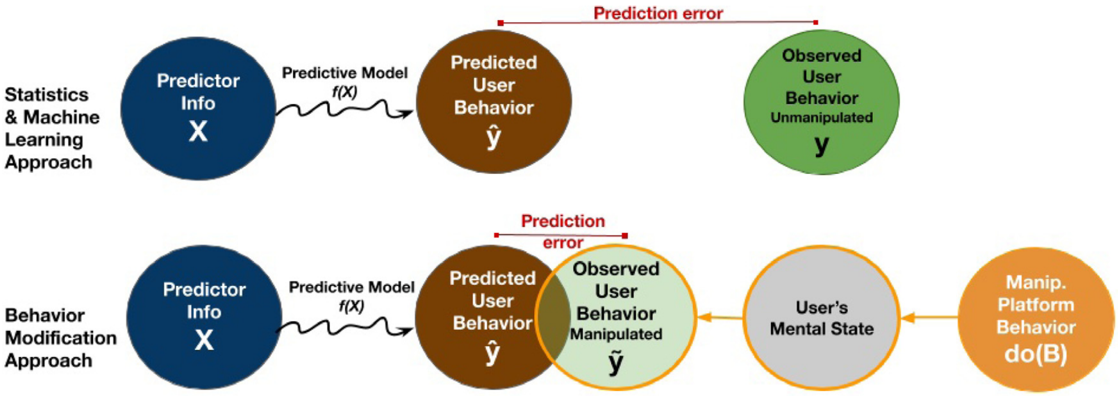

Prediction error
Statistics & Machine Learning Approach
Predictor Info X
Predictive Model f(X)
Predicted User Behavior ŷ
Observed User Behavior Unmanipulated y
Prediction error
Behavior Modification Approach
Predictor Info X
Predictive Model f(X)
Predicted User Behavior ŷ
Observed User Behavior Manipulated ỹ
User's Mental State
Manip. Platform Behavior do(B)

Fig. 1. Prediction error with no behavior modification (top) vs. with behavior modification (bottom). Manipulating platform behavior *do*(*B*) pushes the observed user behavior toward its predicted value. Note that only orange arrows denote causal effects; squiggly black arrows denote correlation-based predictive relationships.

of deployment. This assumption underlies the practice of randomly partitioning the data into separate training and test sets (or into multiple *folds* in cross validation), where the model is trained on the training data and evaluated on the separate test data. Of course, the continuity assumption is often violated to some degree depending on the distance (temporal, geographical, etc.) between the training/test data and the to-be-predicted data and how fast or abruptly the environment changes between these two contexts. These *dataset shift* challenges (e.g., see Moreno-Torres, Raeder, Alaiz-RodríGuez, Chawla, & Herrera, 2012) can increase prediction errors beyond the disparity observed between training and test prediction errors. Hence, determining the predictive power based on the test data might provide an overly optimistic estimate of the actual performance at deployment.

We note that the EPE is an expected value, and thus a population quantity. In practice, the EPE is typically estimated using the mean of the squared prediction errors (MSE) in the test dataset.

3. Reducing prediction error by modifying user behavior (“improving" prediction)

Platforms now have the incentive and technology[14] to minimize prediction errors in a direction that is absent from academic prediction research: by modifying *actual* behavior ($\boldsymbol{y}$).

BMOD techniques can be used to minimize the prediction error by pushing the behaviors of users toward their predicted values, thereby “improving" the apparent prediction capability. This can be achieved via a predict-then-modify process, as follows:

Predict: At time t, predict a user’s future behavior at time $t+k$ ($k>0$).

Modify: During period $(t, t+k)$, modify the user’s behavior toward the predicted value while continually monitoring the user’s behavior.

This sequence is illustrated in Fig. 2 for a sample of users (*showcase sample*). The outcome is observed at time $t+k$, thereby allowing the platform and customer to evaluate the predictive accuracy and subsequently make purchase decisions.

The two mechanisms that enable such a process are: (1) platforms with a plethora of powerful and tested BMOD tools, and (2) BMOD techniques designed to modify behavior *in a predictable manner*–here pushing outcome values toward their predicted values–in order “to shape individual, group, and population behavior in ways that continuously improve their approximation to guaranteed outcomes" (Zuboff, 2019, p. 339). Next, we describe reinforcement learning, a powerful machine learning method that can be used to implement this strategy.

3.1. Behavior modification via reinforcement learning

The financial incentives and technical capabilities of internet platforms might entice platform data science teams under pressure to showcase predictive performance to engage in this prediction “improvement" strategy either knowingly or myopically. A technology that makes this especially suited toward this end is *reinforcement learning* (RL), which is now often used for implementing BMOD by platforms (den Hengst, Grua, el Hassouni, & Hoogendoorn, 2020).

Different from supervised learning algorithms that learn from a pre-existing set of labeled data (*offline learning*), RL takes an *active learning* approach where an *agent* actively collects data by interacting with an unknown dynamic environment. In the case of RL on platforms, the unknown dynamic environment is the user. RL formalizes human-machine interaction histories as sequences of {state, action, reward} trajectories, which are generated while interacting in real-time with users as well as from previously collected user interactions (Sutton & Barto, 2018). The RL agent’s goal is to intervene in its

[14] For example, Facebook’s *AI backbone* FBLearner Flow combines machine learning and experimentation capabilities that can be applied to the entire Facebook userbase https://engineering.fb.com/core-data/introducing-fblearner-flow-facebook-s-ai-backbone/ archived at https://archive.ph/P4kkE

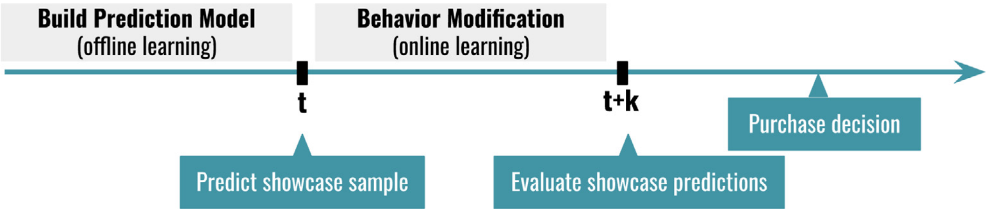


Fig. 2. Sequence of events: prediction at time t is followed by BMOD during $(t, t+k)$. After evaluating the predictive accuracy at time $t+k$, the customer decides whether to purchase the platform's prediction product.

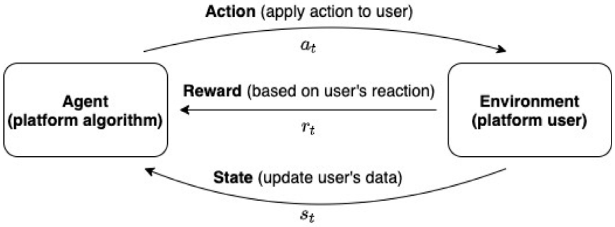

Action (apply action to user)
a_t
Agent
(platform algorithm)
Reward (based on user's reaction)
r_t
Environment
(platform user)
State (update user's data)
s_t

Fig. 3. Schematic showing the application of reinforcement learning to the users of an internet platform.

environment of human users to learn an optimal *policy* maximizing the accumulation of designer-specified rewards (e.g., clicks). Fig. 3 illustrates the operation of RL in a platform context.

RL is considered a machine learning approach because the agent takes a series of decisions to maximize the cumulative reward for a predefined task without being explicitly programmed to achieve the task. Thus, the goal of the agent is to learn the optimal behavior through repeated trial-and-error interactions with the environment and without human involvement (MathWorks, 2021). RL is more suitable than human-designed randomized experiments (e.g., A/B testing) when the space of possible interventions is huge, and there is no known functional form relating the outcome to the inputs. The platform environment belongs to the latter type, where the number and type of potential BMOD interventions is extremely high because a wide range of different content can be displayed[15] (e.g., ads, news items, or friend suggestions), with various ways to serve the content (e.g., format, timing, or device), and various positive and negative reinforcement types (e.g., positive reinforcement with rewards, recognition, or praise; or negative reinforcement with time pressure or social pressure). These can be further personalized by utilizing the traits of users combined with their implicit feedback (den Hengst et al., 2020). For example, Kosinski, Stillwell, and Graepel (2013) showed how Facebook users' Likes can predict their psychological attributes, ranging from sexual orientation to intelligence, and suggested that including such attributes can improve personalized interventions.[16] Due to the large range of interventions (with some being potentially continuous), platforms would be unwilling or unable to learn offline models for the entire space. Indeed, RL algorithms govern personalized interventions on many platforms. For example, recommender systems used by popular commercial platforms such as TikTok, Pandora, Instagram, and YouTube use interactive user data to select an optimal recommendation policy for a given user (Chen, et al., 2019; Zhou, et al., 2020). Facebook uses RL to modify a user's likelihood of clicking a push notification (Gauci, et al., 2018). LinkedIn uses multi-armed bandits, a simplified type of RL, for automating ad placement decisions (Tang, Rosales, Singh, & Agarwal, 2013), and Yahoo! uses contextual multi-armed bandits for personalized news recommendations (Li, Chu, Langford, & Schapire, 2010).

In summary, RL has three features that make it a useful approach for "improving" prediction: it employs interventions, it learns online (as opposed to modeling only pre-existing data), and it is based on delayed reward (allowing it to sequentially improve). The "improve" prediction strategy we've described will occur if during the period $(t, t+k)$, RL is deployed online to the showcase sample, with an objective function set to minimizing prediction error where the predictions were previously generated at time t (prior to the BMOD).[17] In other words, at time t, predictions are generated from a model that

[15] "People who do a lot of research on products may see an ad that features positive product reviews, whereas those who have signed up for regular deliveries of other products in the past might see an ad offering a discount for those who "Subscribe & Save". www.nytimes.com/2019/01/20/technology/amazon-ads-advertising.html

[16] "Online insurance advertisements might emphasize security when facing emotionally unstable (neurotic) users but stress potential threats when dealing with emotionally stable ones". (Kosinski et al., 2013).

[17] While the RL could potentially use offline data prior to time t to learn the policy for users in general, such offline learning would likely have a different objective function–such as maximizing user engagement–and would require a large set of users beyond the limited showcase sample. Moreover, it would be unable to exploit the feedback of individual users' idiosyncratic responses to online interaction with the RL agent.

learned offline prior to time t. The RL's objective function is then set based on these predictions, thereby allowing the RL agent to (sequentially) modify behavior toward those predictions in the period $(t, t+k)$. At time $t+k$, the resulting users' behaviors will be closer to their predictions, resulting in apparent "improved" predictive performance. As mentioned earlier, setting the RL's objective function in this manner can be done maliciously, myopically, or even erroneously.

3.2. A predict-then-modify scenario

The following (fictitious) scenario describes the different steps in a platform's engagement in the predict-then-modify strategy with the aim of selling a prediction product. Consider a ride-sharing or social media platform that wants to sell predicted risk scores for drivers to an insurance company. The insurance company plans to buy such predictions on an ongoing basis. However, they first require information about the expected prediction accuracy of the predicted driver risk scores to compare across platforms or to determine whether the accuracy would be sufficient for their purposes. To that end, the platform's data science team operationalizes risk in a manner that is measurable and acceptable to the insurance company. For example, they use the "app usage rate while driving" (y) as the behavior of interest (assuming that a high rate is indicative of higher risk). They also agree on the forecast horizon k of interest, such as "predicted risk in $k = 7$ days". Importantly, we assume that the platform's BMOD is capable of increasing/decreasing the app usage rate, such as by controlling the level of driver engagement with the app by varying the volume, type, or delivery mode of notifications. The platform's proof-of-concept process would continue as follows.

Step 1: Obtain a prediction model (*Train*). The data science team identifies a pre-trained prediction model with horizon k, $\hat{y}_{\tau+k} = \hat{f}(\boldsymbol{x}_\tau)$, which was trained previously using data from their platform (or that can be quickly retrained on an appropriate subset of driver data relevant to the insurance company, such as in a restricted geographical area).

Step 2: Select a showcase sample and generate predictions (*Predict*). At time t, the platform selects a sample of n drivers as the *showcase sample*, for whom they have data $\boldsymbol{x_t}$. For each driver i, the platform predicts their app usage rate at time $t+k$ ($\hat{y}_{i,t+k}$). These predictions can be shared immediately with the insurance company.

Step 3: Deploy BMOD (*Modify*). During the period $(t, t+k)$, the platform deploys BMOD sequentially to push the behavior of the showcase drivers toward $\hat{y}_{t+k}$. At any point in time $t' \in (t, t+k)$, the platform (i.e., RL agent) has information about $\hat{y}_{t+k}, \boldsymbol{x}_t, y_t$ and any further information that becomes available until t'. This step can be implemented via RL with an objective function of minimizing the deviation between the observed app usage rate and its corresponding (fixed) prediction $\hat{y}_{t+k}$. The RL agent utilizes the feedback afforded by the real-time stream of prediction error data given as the differences between $\hat{y}_{i,t+k}$ and $y_{i,t'}$ in order to continually nudge the user by varying the volume, type, or delivery mode of notifications.

Step 4: Evaluate the predictive accuracy (*Evaluate*). At time $t+k$, the behavior outcome of interest, app usage rate, is recorded for the showcase sample. These values are then compared with their predictions and performance is summarized using metrics such as the mean squared error (MSE). The performance metrics are shared with the insurance company.

At this point, the insurance company evaluates the predictive performance and decides whether to purchase the prediction product. A purchase decision will result in the platform selling predicted scores for other drivers beyond the showcase sample, possibly on an ongoing basis.

The effect of this predict-then-modify strategy on drivers is harmful because it turns high-risk predictions into high-risk realities. We can envision other prediction products involving risk prediction that lead to further dangerous human and social outcomes. One example is a security or social services agency interested in purchasing predicted criminal behavior scores. The platform's proof-of-concept process could nudge high-scoring platform users toward criminal behavior. Another example is a government interested in predicting citizens likely to become severely ill with COVID-19, for public health decision-making purposes. A platform showcasing their "COVID-19 susceptibility" scores product would push users toward or away from infection by encouraging or discouraging physical proximity with others and travelling to crowded locations. It should be noted that in these three scenarios, the gap between the platform's goal (showcasing accurate predictions) and the customer's objective (avoiding high risk drivers, decreasing criminal behavior, and reducing COVID-19 harm) will result in the behaviors of high-risk users being pushed in a direction that is not only ethically dubious but also at odds with the customer's interests.

3.3. Dataset shift that reduces prediction error

When prediction is not followed by behavior modification, differences between the training and deployment environments introduce uncertainty, typically by increasing bias and/or changing the noise distribution. As described earlier, such dataset shift challenges can cause larger prediction errors at deployment, and are therefore a major challenge in adversarial attacks and gaming scenarios, which involve mischief during training or prediction (Huang, Joseph, Nelson, Rubinstein, & Tygar, 2011). By contrast, in BMOD, training and deployment are made *closer* by design. While dataset shifts arising from uncontrollable and unforeseeable conditions increase uncertainty, BMOD intends to shift actual outcome values y *closer* to $\hat{y}$ thereby reducing uncertainty. For example, when predicting that a user is likely to become depressed, displaying depressing news, friends' posts, and depression-related ads will increase that user's chance of depression. Facebook's emotional contagion experiment by Kramer, Guillory, and Hancock (2014) demonstrated this capability. When predicting the arrival time for a delivery, incentivizing faster or slower driving can increase the accuracy of the predicted arrival time. Displaying donation amounts by friends with amounts similar to the user's predicted amount can increase the chance the user donates the predicted amount (e.g., Nook, Ong, Morelli, Mitchell, & Zaki, 2016).

3.4. "Improve" prediction vs. adversarial attacks and gaming

We note that the predict-then-modify strategy, which combines prediction and BMOD, differs from *adversarial attacks*, where an attacker interferes in the process of training or prediction by manipulating the training data, prediction model, or predicted values (Huang, et al., 2011). It also differs from gaming by strategic users, who manipulate their own behavior–the input data–to obtain a favorable prediction (e.g., Frankel & Kartik, 2019; Hardt, Megiddo, Papadimitriou, & Wootters, 2016; Munro, 2020). The first key difference is that adversarial attacks and gaming aim to modify *predictions*, whereas the strategy that we describe modifies the *actual behavior*. In the scenario that we describe, instead of directly manipulating data, the platforms influence the behavioral processes that generate the data. The second key difference is that, in adversarial attacks and gaming, the intervention is performed by an attacker or a user, whereas in our case both prediction and BMOD are performed by the platform.

3.5. "Improve" prediction vs. causal classification

The predict-then-modify strategy differs from individual treatment assignment that occurs in causal classification, that is, identifying individuals whose outcome would be positively changed by a treatment, such as in uplift modeling (Fernández-Loría, Provost, Anderton, Carterette, & Chandar, 2022; Olaya, Coussement, & Verbeke, 2020). A key difference is the sequence of actions: in causal classification prediction follows an intervention, whereas in the predict-then-modify scenario prediction precedes behavior modification. The business goals are vastly different: causal classification aims at effective precision targeting, whereas predict-then-modify uses causal manipulation to make the results of a prediction appear more accurate than it otherwise would have been.

4. Formalization and analysis

To study prediction error under the predict-then-modify strategy, it is useful to decompose the new form of EPE (under BMOD) into separate meaningful sources. This helps identify components such as bias, variance, and noise. However, we need a technical vocabulary that can encode both correlation-based prediction and causal-based actions.

The challenge is that the standard notation and terminology used in statistics and machine learning for predictive modeling are insufficient for formalizing the problem of minimizing prediction error by intentionally manipulating the actual outcome values using BMOD. The bottom panel in Fig. 1 illustrates this new scenario. Specifically, predictive terminology conveys correlation-based relationships, but is not suitable for denoting intentional manipulations. Fig. 1, which includes both causal arrows (orange) and a correlational connector (depicted as a squiggly black arrow, but with no causal interpretation[18]) is incoherent in the world of causal diagrams as well as in the world of prediction. While the combination of prediction and causal intervention is present in the machine learning areas of causal classification (e.g., uplift modeling) and recommendations, until recently the literature in those areas either did not use causal language at all or else it used causal language in an informal way (Zhang, Li, & Liu, 2021). Only recently have papers started to appear that use formal causal notation along with predictive notation in a unified way, such as Fernández-Loría and Provost (2022a), Fernández-Loría, et al. (2022), Gutierrez and Gérardy (2017), Olaya et al. (2020), Verbeke, Olaya, Guerry, and Van Belle (2022), Zhang et al. (2021). With the exception of Olaya et al. (2020), all these papers use the potential outcomes framework by Rubin (1974). Using a unified, formal notation for the prediction and causal components to represent scenarios that combine both actions is important for developing theory, gaining deeper insights, and generalizing methodology to more complex scenarios.[19]

In line with this effort to provide a unified notation that combines prediction and intervention, we propose formalizing the predict-then-modify scenario by integrating causal notation into existing predictive terminology in a parsimonious manner. We do this by adopting the $do(.)$ operator proposed by Pearl (2009), where $do(B)$ denotes that the variable B is not simply observed but rather is intentionally modified.[20] This allows us to incorporate intentional behavioral modification into the predictive modeling context. We then use this notation to decompose the EPE in order to identify the different components that affect predictive power.

4.1. Notation

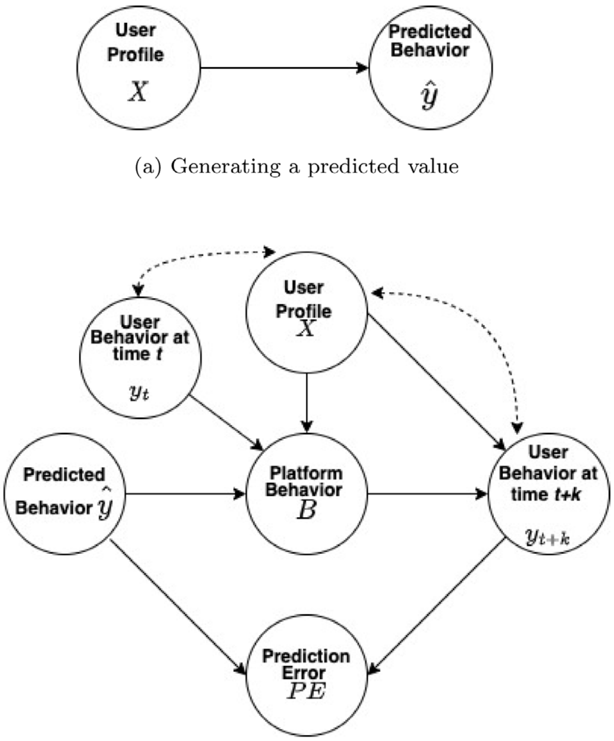
In the following, we omit the time subscripts used in Section 3.2 for a more parsimonious exposition. We also use a continuous y.

For behavior prediction (steps 1–2), we use the ordinary notation $\hat{y}$ to denote the predicted user behavior

[18] We use a single-headed squiggly arrow rather than a bi-directional straight arrow for the correlation-based predictive relationship to convey the asymmetric input-output roles of X and y. In standard causal diagrams, bi-directional arrows convey an unobservable variable affecting the two variables at the arrowheads, and there is no way to represent an asymmetric correlation-based predictive relationship.

[19] We thank Wouter Verbeke for this point.

[20] "The $do(x)$ operator is a mathematical device that helps us specify explicitly and formally what is held constant, and what is free to vary". (Pearl, 2009, p. 358).



(a) Generating a predicted value

(b) Modifying behavior towards the predicted value

Fig. 4. Causal diagrams representing the separate steps comprising (a) generating a predicted value at time t and (b) modifying behavior toward the predicted value during $(t, t+k)$. The doubled-headed dotted arrows represent correlations between X and y because they are influenced by a common set of unobserved or latent variables.

given the user's profile $\boldsymbol{x}$ (which can also include contextual information such as the location and time of day). The predictions are generated by a predictive model that aims to capture the predictive (correlation-based) relationship between the outcome and the predictors from a training dataset:

$$y|\boldsymbol{x} = f(\boldsymbol{x}) + \epsilon. \tag{2}$$

Using the trained predictive model, $\hat{f}$, trained offline using preexisting data $(\boldsymbol{x}, \boldsymbol{y})$, a single-valued k-step ahead point prediction can be generated at time t for each observation i in the showcase sample:

$$\hat{y}_{i,t+k}|\boldsymbol{x}_{i,t} = \hat{f}(\boldsymbol{x}_{i,t}). \tag{3}$$

BMOD (step 3) requires the introduction of interventional notation. We use $do(B)$ to denote the intentionally-modified platform behavior. In particular, we use Pearl's "intervention as variable" formulation (Pearl, 2009, Section 3.2.2), which conceptualizes the intervention as an external force that alters the *function* between B and X. An important advantage of this formulation (compared with the simpler conceptualization of forcing B to take on a fixed value) is that "it contains information about richer types of interventions and is applicable to any change in the function relationship f_i and not merely to the replacement of f_i by a constant" (p. 71).

Fig. 4 shows causal diagrams that illustrate the two separate steps of first generating a prediction at time t and then modifying the behavior toward the prediction during $(t, t+k)$. Nodes represent the different variables. The arrows mean that the node with an incoming arrow(s) is a function of the nodes that point to it. Note that in Fig. 4(b), there is no arrow directed at $\hat{y}$ because it is assumed to be fixed at this stage (during $(t, t+k)$).

Next, we denote the manipulated behavior as $\tilde{y}$. We note that it is incorrect to write $\tilde{y} \doteq do(y)$ because the user's outcome y is not directly manipulated. Instead, the modified behavior is fully mediated. The platform tailors its behavior B ($do(B)$ or personalized $do(B_i)$) to manipulate the user's mental state (emotion, feeling, mood, thought, etc.), which then leads to the modified behavior $\tilde{y}_i$. This modification is specifically aimed at pushing the outcome toward its prediction, and thus $do(B)$ inherently uses $\hat{y}$. Using $\sim$ on top of the terms affected by $do(B)$, we therefore write[21]

$$\tilde{y}_{i,t+k} \doteq y_{i,t+k}|do(B_i), \boldsymbol{x}_{i,t}, \tag{4}$$

where $B_i = h(\hat{y}_{i,t+k}, \boldsymbol{x}_{i,t}, y_{i,t})$.[22] Including $\boldsymbol{x}_{i,t}$ and $y_{i,t}$ in $h(\cdot)$ represents personalized behavioral modification

[21] Expressions using the $do(.)$ operator are typically written within causal queries expressed in the form of conditional expectations or probabilities, such as $E[y_i|do(B_i), x_i]$. We take some license to show it outside of that usual casing because our focus here is on representing the manipulated outcome rather than estimating a causal effect.

[22] For simplicity, we assume a deterministic intervention B_i. However, our results in Section 4.2 are valid also for a stochastic

Table 1
Short and full notations.

Short notation	Full notation/definition	Description
y_i	$y_i\|\boldsymbol{x}_i = f(\boldsymbol{x}_i) + \epsilon_i$	Outcome under no manipulation
f	$f(\boldsymbol{x}) = E[y\|\boldsymbol{x}]$	True function under no manipulation
σ^2	$Var(\epsilon) = E[\epsilon^2]$	Noise variance under no manipulation
$\hat{y}_i$	$\hat{y}_i\|\boldsymbol{x}_i = \hat{f}(\boldsymbol{x}_i)$	Predicted outcome under no manipulation
$\hat{y}_{i,t+k}$	$\hat{y}_{i,t+k}\|\boldsymbol{x}_{i,t} = \hat{f}(\boldsymbol{x}_{i,t})$	Predicted k-step ahead outcome under no manipulation
B_i	$h(\hat{y}_{i,t+k}, \boldsymbol{x}_{i,t}, y_{i,t})$	Personalized intervention
f_{do}	$g(do(B), \boldsymbol{x}) = E[y\|do(B), \boldsymbol{x}]$	True function under $do(B)$
$\tilde{y}_{i,t+k}$	$y_{i,t+k}\|do(B_i), \boldsymbol{x}_{i,t} = g(do(B_i), \boldsymbol{x}_{i,t}) + \tilde{\epsilon}_i$	Manipulated outcome at time $t+k$
$\tilde{\sigma}^2$	$Var(\tilde{\epsilon}) = Var(\tilde{y}) = E[\tilde{\epsilon}^2]$	Noise variance under $do(B)$

based on the user's specific predictor information $\boldsymbol{x}_i$ (e.g., user i's browsing history, demographics, and location) and their outcome at time t. It should be noted that $\boldsymbol{x}_{i,t}$ and $y_{i,t}$ can impact the modified behavior both directly and indirectly via the platform's personalized BMOD. In fact, $\boldsymbol{X}_i$ is a collection of variables, where some might be used to personalize B_i and others (or even the same ones) could moderate the effect of B_i on $\tilde{y}_{i,t+k}$. These causal relationships are graphically depicted in the bottom diagram in Fig. 4.

For the modified outcome, we use f_{do} to denote the underlying function, which can be a completely different function from f:

$$\tilde{y}_{i,t+k} = f_{do}(do(B_i), \boldsymbol{x}_{i,t}) + \tilde{\epsilon}_i = g(do(B_i), \boldsymbol{x}_{i,t}) + \tilde{\epsilon}_i, \tag{5}$$

where we assume that $\tilde{\epsilon}_i$ has mean 0 and variance $\tilde{\sigma}^2$.

We note that the quantity $E[\tilde{y}_i|\boldsymbol{x}_i] - E[y_i|\boldsymbol{x}_i] = E[\tilde{y}_i - y_i|\boldsymbol{x}_i]$, is called the (population) *conditional average treatment effect* (CATE) (Athey & Imbens, 2016; Imbens & Rubin, 2015) or *individual treatment effect* (Shalit, Johansson, & Sontag, 2017), which is of key interest in treatment effect estimation and inference.[23]

The prediction errors at time $t+k$ are obtained by comparing the predicted values $\hat{y}_{t+k}$ to the manipulated outcomes $\tilde{y}_{t+k}$. Table 1 provides the short notation, full notation, and description for each of the terms mentioned above.[24] For clarity, we sometimes omit the subscripts i or t and $t+k$ in the following exposition. Together with Eqs. (4)–(5), we now have a sufficient vocabulary for examining the prediction error under BMOD.

4.2. EPE of modified outcomes ($\widetilde{EPE}$)

When outcome values are intentionally pushed *toward* their predictive values, we intuitively expect that the resulting EPE will be lower than the no-manipulation outcome values.[25] We can now formalize the following questions: given $\hat{f}$, a specific predictive model trained on data with no BMOD $(X, \boldsymbol{y})$, when will the EPE for a manipulated user with predictors $\boldsymbol{x}$ and manipulation $do(B_i) = b$ be lower than the EPE if the user was not manipulated? That is, for the loss function L, when will we get

$$E[L(\tilde{y}, \hat{f}(b, \boldsymbol{x}))] < E[L(y, \hat{f}(\boldsymbol{x}))]? \tag{6}$$

When might the manipulation lead to worse predictive power?

To answer these questions, we break down the EPE into several non-overlapping components. Using the standard L_2 loss function, we can obtain the EPE under BMOD as follows (the full derivation is given in the Appendix):

$$\begin{aligned}\widetilde{EPE}(\boldsymbol{x}) &= E\left[\left(y|do(B), \boldsymbol{x} - \hat{f}(\boldsymbol{x})\right)^2\right] \\ &= \tilde{\sigma}^2 + \left[CATE(\boldsymbol{x}) + Bias(\hat{f}(\boldsymbol{x}))\right]^2 + Var(\hat{f}(\boldsymbol{x})).\end{aligned} \tag{7}$$

Each of the terms in Eq. (7) has an interesting meaning and different implications for the effect of BMOD on EPE. The additive nature of this formulation provides insights into the roles of the data size, predictive model properties, and BMOD qualities. By comparing $\widetilde{EPE}(\boldsymbol{x})$ to $EPE(\boldsymbol{x})$ (the manipulated and non-manipulated scenarios), we can observe the following.

- *Data size:* Irrespective of whether manipulation is applied, the data size affects $\widetilde{EPE}$ via the variance of

modification of the form $B_i = h(\hat{y}_i, \boldsymbol{x}_i) + \upsilon_i$ where υ_i is an error term. Stochastic interventions could reflect effects that are beyond the platform's control, such that the treatment a user receives may differ slightly from the platform's intended treatment. For example, when the intervention is a displayed news item or ad, its choice might be affected by the posting activity of a user's friends, unexpected events in the news, the stock market, the weather, or the time(s) of day that the user may choose to log into the app. All of these could affect the news or ads viewed by the user, such that the firm may have intended a slightly different level in quantity or quality of treatment than what the user actually received. Moreover, the firm could decide to intentionally build in some noise into personalized treatments.

[23] Note that $y_i|\boldsymbol{x}_i$ assumes no manipulation. Pearl (2009, p. 70–72) offers an alternative formulation to encode manipulation vs. no manipulation by adding a binary intervention indicator I_B that obtains values in $\{do(B_i), idle\}$. In our case $I_B = idle$ for $y_i|\boldsymbol{x}_i$.

[24] It is possible to use Rubin's potential outcomes notation intended for estimating treatment effects (e.g. Imbens & Rubin, 2015, p. 33). This requires defining $B = \{0, 1, 2, \ldots\}$ as the intervention assignment and denoting the outcome by $y_i(B)$, where $y_i(0)$ is the un-manipulated outcome. The quantity $y_i|do(B), \boldsymbol{x}$ is written as $y_i(B)|B, \boldsymbol{x}$. We prefer the $do(.)$ operator because it conveys the causal nature of the manipulation B and clearly differentiates it from the correlation-based prediction components $\boldsymbol{x}$. Another possibility is using the counterfactual notation given by Pearl, Glymour, and Jewell (2016, chapter 4) and recently used in algorithmic fairness research, e.g., by Kusner, Loftus, Russell, and Silva (2017), Zhang and Bareinboim (2018). While the counterfactual notation is more general than the $do(.)$ operator, since we focus on interventions that do not require counterfactual notions, the $do(.)$ operator provides a more parsimonious and elegant choice.

[25] In the prediction minimization process, *all* subjects are initially not B-manipulated and a later sample is B-manipulated using personalized modifications.

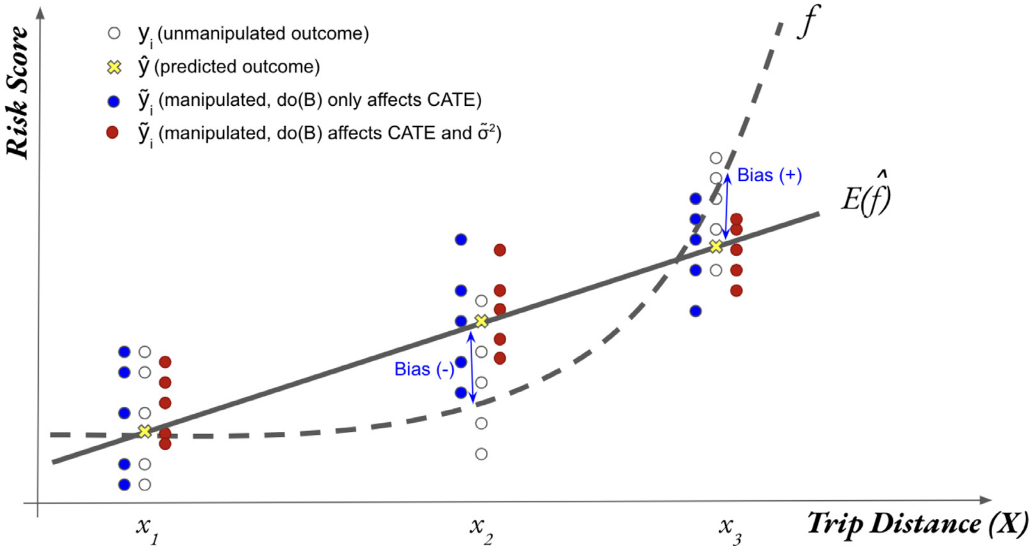


Fig. 5. Hypothetical prediction of risky driving behaviors given the distance by a ride-sharing platform. The plot illustrates the effect of BMOD on shifting the average outcome by $CATE = -Bias$ (blue circles) or on both shifting the average outcome and shrinking the variance $\tilde{\sigma}^2$ (red circles). Yellow cross symbols represent the predicted values $\hat{f}(x_i)$. (The schematic assumes a very large training sample, and thus $\hat{f} \approx E[\hat{f}]$.)

$\hat{f}$,[26] thereby indicating that larger training samples can improve not only the predictions but also the average manipulated prediction error. Pushing the outcome toward a more stable prediction leads to smaller errors.

- *Magnitude of BMOD effect:* The second term shows the role of the average BMOD magnitude (CATE) in countering the bias of $\hat{f}$. This term is minimized when $CATE = -Bias(\hat{f})$, that is, when on average, $do(B)$ pushes the user's behavior in a direction and magnitude that exactly counters the bias of $\hat{f}$. Thus, an effective BMOD can improve predictive power by combating the bias in $\hat{f}$ provided that $0 < CATE < -2Bias$ or $-2Bias < CATE < 0$.
- *Noise (homogeneity of prediction errors):* Compared with σ^2 in the no-manipulation *EPE*, the first term in $\widetilde{EPE}$ is $\tilde{\sigma}^2$, that is, the noise variance *under BMOD*. This means that BMOD can also affect the *variability* of the prediction errors across different users.

4.3. Modification strategies

We now examine the trade-offs and implications of the four $\widetilde{EPE}$ sources ($\tilde{\sigma}$, $CATE$, $Bias(\hat{f})$, and $Var(\hat{f})$) for the EPE. The insights obtained help us to determine the ideal modification under different scenarios in terms of algorithm bias and variance.

To this end, we again consider the ride-sharing example in Section 3.2, where a ride-sharing or social media platform showcases its risk prediction capability to an insurance company. The platform can modify the behaviors of drivers by manipulating their engagement with their app while driving. Fig. 5 shows a scatter plot of driver risk scores (y-axis) vs. daily distance (x-axis) for a fictitious showcase sample of drivers (hollow circles). Suppose that the predictions are the drivers' predicted rates of app usage while driving (*risk scores*) and the goal is to minimize the (squared) differences between the predicted and actual values. Suppose that the risk is a quadratic function of the driving distance but that the predictive model estimates a linear relationship. The linear model's predicted risk scores are shown as yellow crosses in Fig. 5. We consider three specific distances: x_1, x_2, and x_3. While $\hat{f}$ is biased, there is no bias for x_1, the bias is negative for x_2, and the bias is positive for x_3 (for simplicity, the schematic assumes a very large training sample, and thus $\hat{f} \approx E[\hat{f}]$). Next, the blue and red circles represent two types of BMOD, where the first BMOD only affects *CATE*, whereas the second BMOD affects both *CATE* and $\tilde{\sigma}$.

Scenario 1: Low-bias $\hat{f}$ trained on a very large sample.

This scenario is akin to applying deep learning algorithms to massive amounts of training data. The very large sample size means that $Var(\hat{f}) \approx 0$ and $Bias(\hat{f})$ is very small. The strategy of setting $CATE = -Bias(\hat{f})$ is optimal if the BMOD also does not increase the error heterogeneity such that $\tilde{\sigma}^2 \leq \sigma^2$. Because the bias is low, the optimal BMOD should have a small effect. In Fig. 5, $\hat{f}(x_1)$ has no bias, and thus applying BMOD to drivers with distance x_1 will introduce bias, which is only useful if it can sufficiently shrink the variability of the resulting risky behaviors (red circles).

Scenario 2: High-bias $\hat{f}$ trained on a very large sample.

Algorithms that lead to $\hat{f}$ with high bias include naive Bayes, linear regression, shallow trees, and k-nearest neighbors with large values for k. As in scenario 1, here

[26] The machine learning *bias* is asymptotic in sample size because an algorithm is biased "if no matter how much training data we give it, the learning curve will never reach perfect accuracy". (Provost & Fawcett, 2013).

too $Var(\hat{f}) \approx 0$. Fig. 5 shows that the strategy of setting $CATE = -Bias(\hat{f})$ increases the average risky behavior for x_2 by $|Bias(\hat{f}(x_2))|$ and decreases it for x_3 by $|Bias(\hat{f}(x_3))|$. This strategy is optimal if the BMOD also decreases (or at least does not increase) the error heterogeneity such that $\tilde{\sigma}^2 \leq \sigma^2$. While a small modification effect in the right direction can help counter bias, the ideal modification effect must be as large as the bias. It should be noted that EPE is computed for a specific $\boldsymbol{x}$, and thus generalizing the rule to any $\boldsymbol{x}$ requires either assuming homoskedastic errors $\tilde{\epsilon}$ or that the inequality holds for all $\boldsymbol{x}$ ($\forall \boldsymbol{x}\, \tilde{\sigma}^2_{\boldsymbol{x}} \leq \sigma^2_{\boldsymbol{x}}$).

Scenario 3: High-variance $\hat{f}$.

If the predictive model f is trained on a relatively small sample and the algorithm employed leads to high variance in $\hat{f}$, then the potential minimization of $\tilde{\sigma}^2$ and/or $\left[CATE + Bias(\hat{f})\right]^2$ using BMOD might be negligible relative to $Var(\hat{f})$. Because BMOD is based on nudging behavior toward $\hat{f}(\boldsymbol{x}_i)$, a highly volatile $\hat{f}$ might result in erratic $do(B_i)$ modifications in terms of the magnitude or even the direction. A sequential BMOD, such as using an RL agent, would take longer to learn.

5. Discussion

We describe a new strategy that platforms can use to reduce prediction error that is completely different from approaches considered in the fields of statistics and machine learning. We focus on interventions implemented after exhausting the possibilities of prediction using offline data with the goal of making these predictions appear more accurate before they are used by a customer. This strategy combines prediction with BMOD, and thus formalizing it in technical language requires supplementing predictive notation with causal terminology. Using the $do(.)$ operator, we can describe the entire system including the training data, the predictive model, and the BMOD.

While our $\widetilde{EPE}$ formula is also applicable to BMOD for commercial benefit (e.g., advertising), we focus on the more general case, which can lead to dangerous and perhaps unintentional outcomes that not only harm users but also platform customers. This outcome can be achieved intentionally or myopically by data science teams when the platform uses automated personalization algorithms, such as RL with an objective function set to minimize some function of $\hat{y} - \tilde{y}$, that is, the difference between predicted and manipulated outcomes. In these cases, the platform and customer goals can be misaligned, such as in risk prediction applications where the customer aims to reduce risk while the platform pushes risky users toward risky actions.

5.1. Technical and business implications

The contrast between the bias-variance decomposition in the manipulated and non-manipulated scenarios highlights two key sources of manipulated prediction error: the CATE–bias relationship and its tradeoff with the manipulated noise variance. We now use these insights to address the questions that we posed earlier.

1. *Can BMOD mask poor predictive performance?* Behavioral big data are noisy, sparse, and high-dimensional (De Cnudde, Martens, Evgeniou, & Provost, 2020). BMOD can improve $\widetilde{EPE}$ by countering the bias in $\hat{f}$ as well as by reducing the noise variance. This means that poor performance of a predictive model due to high bias and/or variance, and/or due to the data noisiness, can be masked by $do(B)$. Therefore, platform customers who want to achieve the (manipulated) prediction accuracy level demonstrated by the platform must acquire both the predictions *and* the ability to apply BMOD similar to that performed by the platform. Purchasing the predictions alone might lead to much weaker predictive performance when deployed to non-manipulated users (or by applying a less effective BMOD).
2. *Can one infer the counterfactual EPE from the manipulated $\widetilde{EPE}$?* The difference between the two quantities of no-manipulation EPE and behavior-modified $\widetilde{EPE}$ involves $CATE$, $bias(\hat{f})$, σ, and $\tilde{\sigma}$.[27] Some of these quantities can be estimated by the platform (e.g., CATE) but others are more difficult to estimate, or even impossible. Hence, it is unlikely that the no-manipulation predictive power can be ascertained from the manipulated $\widetilde{EPE}$. Thus, platform customers who want to evaluate the no-manipulation predictive power will need to acquire information about the estimated EPE at the time of model testing.
3. *Can customers detect the manipulation via A/B tests?* This would require that the customer knows which of its users were in the platform's showcase sample. Otherwise, A/B tests are unlikely to detect the error minimization strategy because of the random allocation of users in an A/B test. This randomization spreads BMOD-affected users across the A and B conditions and thus the difference between the group averages will cancel out the BMOD effect. The A/B test statistic and its statistical significance are therefore not impacted by BMOD.
4. *What are the roles of personalized prediction and personalized BMOD in error minimization?* Personalized prediction plays an important role because the more accurate the prediction, the less BMOD is needed to reach good predictive accuracy. In particular, low-bias algorithms trained on very large samples (to shrink $Var(\hat{f})$) are advantageous in terms of requiring a smaller BMOD effect to minimize EPE. Hence, platform efforts and investments in improving personalized predictions are warranted. In addition, in the “improve" prediction strategy, personalizing the BMOD plays a complementary role because an ideal BMOD reduces not only the

[27] $\widetilde{EPE} - EPE = \tilde{\sigma}^2 - \sigma^2 + CATE^2 + 2 \times CATE \times Bias(\hat{f})$.

average magnitude of the errors but also their variability, so that errors are more consistent across users. This highlights the role of *personalized BMOD* in which companies or platforms may invest: utilizing a user's personal $\boldsymbol{x}_i$ data to select the best modification, $do(B_i) = h(\hat{y}_i, \boldsymbol{x}_i)$, among the very large space of potential B interventions. Personalized BMOD has the potential to minimize $\widetilde{EPE}$ more equally both within a certain user profile $\boldsymbol{x}$ and across different user profiles by lowering the conditional bias via manipulating CATE and by shrinking the variance of the (manipulated) outcomes.

5.2. Humanistic and societal implications

Behavior modification, now pervasively applied by platforms to their "data subjects", is geared toward optimizing the platform's commercial interest, often at the cost of the well-being and agency of users. Russell (2019) highlighted the advantage of making users' behaviors more predictable: "A more predictable user can be fed items that they are likely to click on, thereby generating more revenue. People with more extreme political views tend to be more predictable in which items they will click on".

Persuasive technology, a design philosophy now implemented on platforms ranging from e-commerce sites and social networks to smartphones and fitness wristbands, aims at generating *behavioral change* and *habit formation*, most often without the knowledge or consent of users (Rushkoff, 2019). This application of BMOD to platform *users* is more extreme than an organization applying BMOD to its *employees* to increase the organization's productivity. Clearly, this type of use diverges from the original intention of BMOD procedures "to change socially significant behaviors, with the goal of improving some aspect of a person's life" (Miltenberger, 2015, p. 5).

Given the often conflicting goals of data subjects and the platforms that collect and use their data, as well as manipulate their behavior, it is important to introduce causal notation into the predictive environment, thereby allowing statistics, machine learning, and computational social science researchers to study their technical properties and implications. By introducing and integrating causal notation into the predictive terminology we can start studying how BMOD might appear to produce "better" predictions, thereby allowing us to examine the effects of different BMOD types, magnitudes, variations, and directions on the anticipated outcomes.

6. Conclusion and future directions

BMOD can make the behavior of users not only more predictable but also more homogeneous. However, the apparent "predictability" at the time of predictive evaluation $(t+k)$ can be deceptive for the platform customers to the extent that predictive performance based on manipulated behavior does not generalize outside the platform environment. The apparent predictability at time $t+k$ also does not generalize within the platform if the exact same BMOD is no longer used. This is because in a prediction product sales scenario, the pre-sale BMOD is a temporary action applied by the platform to a limited showcase sample and tuned to minimize a function of $\hat{y} - \tilde{y}$ for this sample. Post-sale, it is reasonable that the exact same BMOD will no longer be used. Even if the platform's BMOD has long-term effects beyond time $t + k$, it would only affect the showcase sample but not new users scored for the prediction product. The latter are no longer subject to the predict-then-modify strategy. Hence, the generalizability is still compromised. Finally, the scenario where the platform continues to apply the same BMOD to users beyond the showcase sample, thereby affecting users in the short or long term, is in itself alarming because it raises concerns about whether the business customer entity is aware it is receiving an artificially altered prediction product and is complicit in this, or if it is being deceived.

Importantly, outcomes pushed toward their predictions may be at odds with the interests of the platform customers and harmful to manipulated users. While platforms have the incentive and capabilities to minimize the prediction errors, the predict-then-modify strategy is likely to occur given the growing use of automated personalization techniques, such as RL, that interact with users, apply sequential interventions, and combine prediction and BMOD. Therefore, it is essential to have a useful technical vocabulary that integrates intentional BMOD into the correlation-based predictive framework to facilitate studies of such contemporary strategies. We demonstrated how this notation can be useful for exploring issues such as how BMOD can mask poor predictive power, whether one can infer the counterfactual of the non-manipulated predictive power from the manipulated predictive power, whether customers who run routine A/B tests on a platform can detect such BMOD schemes, and the possible roles of personalized BMOD schemes.

Future research can adopt our proposed technical vocabulary and notation to further examine the potential scope and effects of BMOD by digital platforms. One possible direction is the consideration of dichotomous outcomes. Our $\widetilde{EPE}$ derivation is for a continuous outcome. Deriving $\widetilde{EPE}$ for a binary outcome would be useful in applications where the behavior of interest is dichotomous, such as voting/not-voting, responding/not-responding, or passing/failing some criterion. In these cases, the predict-then-modify scenario would push the user's behavior toward the outcome with the higher predicted probability. There are different ways to formulate the EPE for binary predictions, as well as other ways to measure predictive accuracy. The results obtained for the ordinary unmanipulated scenario show that the effects of bias and variance on EPE are multiplicative rather than additive, and the literature reports conflicting results regarding their roles (e.g., Domingos, 2000; Friedman, 1997). Formulating the effect of BMOD on the resultant prediction error in the binary outcome case can further highlight the distinction between estimating causal effects and selecting appropriate interventions (Fernández-Loría & Provost, 2022b). We leave the derivations of such formulas for future research.

The scenario we described is for a prediction product with a single prediction horizon k. An interesting future direction may involve considering a platform that wants to showcase predictions for multiple horizons simultaneously (e.g., "risk on the next day" and "risk in the next week"). In this case, an additional layer of complexity is added because trying to apply BMOD for one horizon might affect the other horizon.

Another further direction could involve using the notation together with causal diagrams to formalize and study *non-platform* predict-then-modify scenarios where the organization only generates predictions but the prediction itself leads to behavior modification. For example, a stock recommendation can affect the stock price in financial markets, and in healthcare, a patient's predicted health outcome affects the behavior of those with knowledge of the prediction (patient, doctor, etc.) in a manner that can eventually modify that outcome.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We thank Foster Provost, Rob Hyndman, Pierre Pinson, Soumya Ray, Sam Ransbothan, Carlos Cinelli, Wouter Verbeke, Travis Greene, Boaz Shmueli, Patricia Kuo, Aurélie Lemmens, Rodrigo Belo, David Budescu, Raquelle Azran, Bill Meeker, and participants of 2020 SCECR symposium, 2020 ENBIS conference, 2021 IIF Symposium, and seminars at the University of Iowa, Purdue, Erasmus University, Nova SBE, and University of Antwerp for invaluable feedback. We thank Noa Shmueli for Figure 1 artwork. G. Shmueli's research is partially supported by Ministry of Science and Technology, Taiwan [Grant 108-2410-H-007-091-MY3].

Appendix. Derivation of the $\widetilde{EPE}$ bias–variance decomposition (Eq. (7))

The derivation of Eq. (7) (bias–variance decomposition under BMOD) is given as follows. For convenience, we use f and $\hat{f}$ to denote the no-manipulation true function and its estimated model, respectively. For the manipulated scenario, we use f_{do} to denote the true function under BMOD.

For a new observation with inputs $\boldsymbol{x}$ and manipulated outcome $y|do(B)$, $\boldsymbol{x}$, we can decompose the EPE as follows (for convenience, we omit the subscripts i, t, and $t+k$):

$$\begin{aligned}\widetilde{EPE}(\boldsymbol{x}) &= E\left[\left(y|do(B), \boldsymbol{x}-\hat{f}(\boldsymbol{x})\right)^2\right]\\ &= E\left[(\tilde{y}-\hat{f})^2\right]\\ &= E\left[(\tilde{y}-f_{do}+f_{do}-\hat{f})^2\right]\\ &= E\left[(\tilde{y}-f_{do})^2\right]+E\left[(f_{do}-\hat{f})^2\right]\\ &\quad +2E\left[(\tilde{y}-f_{do})(f_{do}-\hat{f})\right].\end{aligned} \tag{A.1}$$

These three terms can be further simplified. The first term can be simplified by noting that $\tilde{y}=f_{do}+\tilde{\epsilon}$:

$$E\left[(\tilde{y}-f_{do})^2\right]=E[\tilde{\epsilon}^2]=\tilde{\sigma}^2. \tag{A.2}$$

The second term can be written as:

$$\begin{aligned}E\left[(f_{do}-\hat{f})^2\right] &= E\left[\left(f_{do}-E(\hat{f})+E(\hat{f})-\hat{f}\right)^2\right]\\ &= \left(f_{do}-E[\hat{f}]\right)^2+E\left[\left(\hat{f}-E[\hat{f}]\right)^2\right]\\ &= \left(f_{do}-E[\hat{f}]\right)^2+Var(\hat{f})\end{aligned} \tag{A.3}$$

because the cross product is zero:

$$2E\left[f_{do}-E[\hat{f})\right]\left(E[\hat{f}]-\hat{f}\right)=2\left(f_{do}-E[\hat{f}]\right)\left(E[\hat{f}]-E[\hat{f}]\right)=0. \tag{A.4}$$

We can further write Eq. (A.3) as a function of the bias and variance of $\hat{f}$:

$$\begin{aligned}\left[f_{do}-E[\hat{f}]\right]^2+Var(\hat{f}) &= E\left[f_{do}-f+f-E[\hat{f}]\right]^2+Var(\hat{f})\\ &= \left[CATE+Bias(\hat{f})\right]^2+Var(\hat{f}).\end{aligned} \tag{A.5}$$

Finally, using the independence of the new observation's prediction error $\tilde{\epsilon}$ from the prediction $\hat{f}$ based on the training data ($E[\tilde{\epsilon}\hat{f}]=0$), we see that the third term is zero:

$$2E\left[(\tilde{y}-f_{do})(f_{do}-\hat{f})\right]=2E[\tilde{\epsilon}](f_{do}-\hat{f})=2f_{do}E(\tilde{\epsilon})-2E[\tilde{\epsilon}\hat{f}]=0. \tag{A.6}$$

Therefore, we can write $\widetilde{EPE}$ from Eq. (A.1) as:

$$\begin{aligned}\widetilde{EPE}(\boldsymbol{x}) &= E\left[\left(y|\boldsymbol{x}, do(B)-\hat{f}(\boldsymbol{x})\right)^2\right]\\ &= \tilde{\sigma}^2+\left[CATE+Bias(\hat{f})\right]^2+Var(\hat{f}).\end{aligned} \tag{A.7}$$

References

Agrawal, A., Gans, J., & Goldfarb, A. (2018). *Prediction machines: the simple economics of artificial intelligence*. Harvard Business Press.

Andrew, J., & Baker, M. (2021). The general data protection regulation in the age of surveillance capitalism. *Journal of Business Ethics*, *168*(3), 565–578.

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

Bak-Coleman, J. B., Alfano, M., Barfuss, W., Bergstrom, C. T., Centeno, M. A., Couzin, I. D., et al. (2021). Stewardship of global collective behavior. *Proceedings of the National Academy of Sciences*, *118*(27).

Chen, M., Beutel, A., Covington, P., Jain, S., Belletti, F., & Chi, E. H. (2019). Top-k off-policy correction for a REINFORCE recommender system. In *Proceedings of the twelfth ACM international conference on web search and data mining* (pp. 456–464).

De Cnudde, S., Martens, D., Evgeniou, T., & Provost, F. (2020). A benchmarking study of classification techniques for behavioral data. *International Journal of Data Science and Analytics*, *9*(2), 131–173.

den Hengst, F., Grua, E. M., el Hassouni, A., & Hoogendoorn, M. (2020). Reinforcement learning for personalization: A systematic literature review. *Data Science*, *3*(2), 107–147.

Domingos, P. (2000). A unified bias-variance decomposition. In *Proceedings of 17th international conference on machine learning* (pp. 231–238).

Eyal, N. (2014). *Hooked: how to build habit-forming products*. Penguin.

Fernández-Loría, C., & Provost, F. (2022a). Causal classification: Treatment effect estimation vs. outcome prediction. *Journal of Machine Learning Research*, *23*(59), 1–35.

Fernández-Loría, C., & Provost, F. (2022b). Causal decision making and causal effect estimation are not the same…and why it matters. *INFORMS Journal on Data Science*.

Fernández-Loría, C., Provost, F., Anderton, J., Carterette, B., & Chandar, P. (2022). A comparison of methods for treatment assignment with an application to playlist generation. *Information System Research*.

Fogg, B. J. (2002). *Persuasive technology: using computers to change what we think and do*. Morgan Kaufmann.

Frankel, A., & Kartik, N. (2019). Improving information from manipulable data. arXiv preprint arXiv:1908.10330.

Friedman, J. H. (1997). On bias, variance, 0/1—loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, *1*(1), 55–77.

Gauci, J., Conti, E., Liang, Y., Virochsiri, K., He, Y., Kaden, Z., et al. (2018). Horizon: Facebook's open source applied reinforcement learning platform. arXiv preprint arXiv:1811.00260.

Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, *4*(1), 1–58.

Greene, T., Martens, D., & Shmueli, G. (2022). Barriers to academic data science research in the new realm of algorithmic behaviour modification by digital platforms. *Nature Machine Intelligence*, *4*(4), 323–330.

Gutierrez, P., & Gérardy, J.-Y. (2017). Causal inference and uplift modelling: A review of the literature. In *International conference on predictive applications and APIs* (pp. 1–13). PMLR.

Hardt, M., Megiddo, N., Papadimitriou, C., & Wootters, M. (2016). Strategic classification. In *Proceedings of the 2016 ACM conference on innovations in theoretical computer science* (pp. 111–122).

Huang, L., Joseph, A. D., Nelson, B., Rubinstein, B. I., & Tygar, J. D. (2011). Adversarial machine learning. In *Proceedings of the 4th ACM workshop on security and artificial intelligence* (pp. 43–58).

Imbens, G. W., & Rubin, D. B. (2015). *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press.

Kosinski, M., Stillwell, D., & Graepel, T. (2013). Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the National Academy of Sciences*, *110*(15), 5802–5805.

Kramer, A. D., Guillory, J. E., & Hancock, J. T. (2014). Experimental evidence of massive-scale emotional contagion through social networks. *Proceedings of the National Academy of Sciences*, *111*(24), 8788–8790.

Kusner, M., Loftus, J., Russell, C., & Silva, R. (2017). Counterfactual fairness. In *Proceedings of the 31st international conference on neural information processing systems* (pp. 4069–4079).

Li, L., Chu, W., Langford, J., & Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on world wide web* (pp. 661–670).

Martens, D., Provost, F., Clark, J., & Junqué de Fortuny, E. (2016). Mining massive fine-grained behavior data to improve predictive analytics.. *MIS Quarterly*, *40*(4), 869–888.

Mathur, A., Acar, G., Friedman, M. J., Lucherini, E., Mayer, J., Chetty, M., et al. (2019). Dark patterns at scale: Findings from a crawl of 11K shopping websites. *Proceedings of the ACM on Human-Computer Interaction*, *3*(CSCW), 1–32.

MathWorks (2021). What is reinforcement learning? MathWorks documentation.

Michie, S., Richardson, M., Johnston, M., Abraham, C., Francis, J., Hardeman, W., et al. (2013). The behavior change technique taxonomy (v1) of 93 hierarchically clustered techniques: building an international consensus for the reporting of behavior change interventions. *Annals of Behavioral Medicine*, *46*(1), 81–95.

Miltenberger, R. G. (2015). *Behavior modification: principles and procedures* (6th ed.). Cengage Learning.

Moreno-Torres, J. G., Raeder, T., Alaiz-RodríGuez, R., Chawla, N. V., & Herrera, F. (2012). A unifying view on dataset shift in classification. *Pattern Recognition*, *45*(1), 521–530.

Munro, E. (2020). Learning to personalize treatments when agents are strategic. arXiv preprint arXiv:2011.06528.

Nook, E. C., Ong, D. C., Morelli, S. A., Mitchell, J. P., & Zaki, J. (2016). Prosocial conformity: Prosocial norms generalize across behavior and empathy. *Personality and Social Psychology Bulletin*, *42*(8), 1045–1062.

Nord, W. R., & Peter, J. P. (1980). A behavior modification perspective on marketing. *Journal of Marketing*, *44*(2), 36–47.

Olaya, D., Coussement, K., & Verbeke, W. (2020). A survey and benchmarking study of multitreatment uplift modeling. *Data Mining and Knowledge Discovery*, *34*(2), 273–308.

Pearl, J. (2009). *Causality* (2nd ed.). Cambridge University Press.

Pearl, J., Glymour, M., & Jewell, N. P. (2016). *Causal inference in statistics: a primer*. John Wiley & Sons.

Provost, F., & Fawcett, T. (2013). *Data science for business: what you need to know about data mining and data-analytic thinking*. O'Reilly Media, Inc.

Rubin, D. B. (1974). Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology*, *66*(5), 688.

Rushkoff, D. (2019). *Team human*. WW Norton & Company.

Russell, S. (2019). *Human compatible: artificial intelligence and the problem of control*. Penguin.

Schneider, C., Weinmann, M., & Vom Brocke, J. (2018). Digital nudging: guiding online user choices through interface design. *Communications of the ACM*, *61*(7), 67–73.

Shalit, U., Johansson, F. D., & Sontag, D. (2017). Estimating individual treatment effect: generalization bounds and algorithms. In *Proceedings of the 34th international conference on machine learning-volume 70* (pp. 3076–3085). JMLR.

Shmueli, G. (2017). Research dilemmas with behavioral big data. *Big Data*, *5*(2), 98–119.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: an introduction*. MIT Press.

Tang, L., Rosales, R., Singh, A., & Agarwal, D. (2013). Automatic ad format selection via contextual bandits. In *Proceedings of the 22nd ACM international conference on information & knowledge management* (pp. 1587–1594).

Thaler, R. H., & Sunstein, C. R. (2009). *Nudge: improving decisions about health, wealth, and happiness*. Penguin.

Verbeke, W., Olaya, D., Guerry, M.-A., & Van Belle, J. (2022). To do or not to do? Cost-sensitive causal classification with individual treatment effect estimates. *European Journal of Operational Research*.

Yeung, K. (2017). 'Hypernudge': Big data as a mode of regulation by design. *Information, Communication & Society*, *20*(1), 118–136.

Zhang, J., & Bareinboim, E. (2018). Fairness in decision-making—the causal explanation formula. In *Thirty-second AAAI conference on artificial intelligence*.

Zhang, W., Li, J., & Liu, L. (2021). A unified survey of treatment effect heterogeneity modelling and uplift modelling. *ACM Computing Surveys*, *54*(8), 1–36.

Zhou, S., Dai, X., Chen, H., Zhang, W., Ren, K., Tang, R., et al. (2020). Interactive recommender system via knowledge graph-enhanced reinforcement learning. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval* (pp. 179–188).

Zuboff, S. (2019). *The age of surveillance capitalism: the fight for a human future at the new frontier of power*. Profile Books.